Sistemas de Autenticación

Métodos de sistemas de Autenticación

Los métodos de autenticación están en función de lo que utilizan para la verificación y estos se dividen en tres categorías:

- Sistemas basados en algo conocido. Ejemplo, un *password* (Unix) o *passphrase* (PGP).
- Sistemas basados en algo poseído. Ejemplo, una tarjeta de identidad, una tarjeta inteligente(*smartcard*), dispositivo usb tipo epass token, smartcard o dongle criptográfico.
- Sistemas basados en una característica física del usuario o un acto involuntario del mismo: Ejemplo, verificación de voz, de escritura, de huellas, de patrones oculares.

Sistemas de autenticación biométrica

A pesar de la importancia de la criptología en cualquiera de los sistemas de identificación de usuarios vistos, existen otra clase de sistemas en los que no se aplica esta ciencia, o al menos su aplicación es secundaria. Es más, parece que en un futuro no muy lejano estos serán los sistemas que se van a imponer en la mayoría de situaciones en las que se haga necesario autenticar un usuario: son más amigables para el usuario (no va a necesitar recordar *passwords* o números de identificación complejos, y, como se suele decir, el usuario puede olvidar una tarjeta de identificación en casa, pero nunca se olvidará de su mano o su ojo) y son mucho más difíciles de falsificar que una simple contraseña o una tarjeta magnética; las principales razones por la que no se han impuesto ya en nuestros dias es su elevado precio, fuera del alcance de muchas organizaciones, y su dificultad de mantenimiento ([GKK97]).

Estos sistemas son los denominados **biométricos**, basados en características físicas del usuario a identificar. El reconocimiento de formas, la inteligencia artificial y el aprendizaje son las ramas de la informática que desempeñan el papel más importante en los sistemas de identificación biométricos; la criptología se limita aquí a un uso secundario, como el cifrado de una base de datos de patrones retinales, o la transmisión de una huella dactilar entre un dispositivo analizador y una base de datos. La autenticación basada en características físicas existe desde que existe el hombre y, sin darnos cuenta, es la que más utiliza cualquiera de nosotros en su vida cotidiana: a diario identificamos a personas por los rasgos de su cara o por su voz. Obviamente aquí el agente reconocedor lo tiene fácil porque es una persona, pero en el modelo aplicable a redes o sistemas Unix el agente ha de ser un dispositivo que, basándose en características del sujeto a identificar, le permita o deniegue acceso a un determinado recurso.

Tabla 8.1: Comparación de métodos biométricos.

	Ojo - Iris	Ojo - Retina	Huellas dactilares	Geometría de la mano	Escritura - Firma	Voz
Fiabilidad	Muy alta	Muy alta	Alta	Alta	Alta	Alta
Facilidad de uso	Media	Baja	Alta	Alta	Alta	Alta
Prevención de ataques	Muy Alta	Muy alta	Alta	Alta	Media	Media
Aceptación	Media	Media	Media	Alta	Muy alta	Alta
Estabilidad	Alta	Alta	Alta	Media	Media	Media
Identificación y autenticación	Ambas	Ambas	Ambas	Autenticación	Ambas	Autenticación
Estándars	-	-	ANSI/NIST, FBI	-	-	SVAPI
Interferencias	Gafas	Irritaciones	Suciedad, heridas, asperezas ...	Artritis, reumatismo ...	Firmas fáciles o cambiantes	Ruido, resfriados ...
Utilización	Instalaciones nucleares, servicios médicos, centros penitenciarios	Instalaciones nucleares, servicios médicos, centros penitenciarios	Policía, industrial	General	Industrial	Accesos remotos en bancos o bases de datos
Precio por nodo en 1997 (USD)	5000	5000	1200	2100	1000	1200

Aunque la autenticación de usuarios mediante métodos biométricos es posible utilizando cualquier característica única y mesurable del individuo (esto incluye desde la forma de teclear ante un ordenador hasta los patrones de ciertas venas, pasando por el olor corporal), tradicionalmente ha estado basada en cinco grandes grupos ([Eve92]). En la tabla 8.1 ([Huo98], [Phi97]) se muestra una comparativa de sus rasgos más generales, que vamos a ver con más detalle en los puntos siguientes.

Los dispositivos biométricos tienen tres partes principales; por un lado, disponen de un mecanismo automático que lee y captura una imagen digital o analógica de la característica a analizar. Además disponen de una entidad para manejar aspectos como la compresión, almacenamiento o comparación de los datos capturados con los guardados en una base de datos (que son considerados válidos), y también ofrecen una interfaz para las aplicaciones que los utilizan. El proceso general de autenticación sigue unos pasos comunes a todos los modelos de autenticación biométrica: **captura** o lectura de los datos que el usuario a validar presenta, **extracción** de ciertas características de la muestra (por ejemplo, las minucias de una huella dactilar), **comparación** de tales características con las guardadas en una base de datos, y **decisión** de si el usuario es válido o no. Es en esta decisión donde principalmente entran en juego las dos características básicas de la fiabilidad de todo sistema biométrico (en general, de todo sistema de autenticación): las tasas de falso rechazo y de falsa aceptación. Por tasa de **falso rechazo** (*False Rejection Rate*, FRR) se entiende la probabilidad de que el sistema de autenticación rechaze a un usuario legítimo porque no es capaz de identificarlo correctamente, y por tasa de **falsa aceptación** (*False Acceptance Rate*, FAR) la probabilidad de que el sistema autentique correctamente a un usuario ilegítimo; evidentemente, una FRR alta provoca descontento entre los usuarios del sistema, pero una FAR elevada genera un grave problema de seguridad: estamos proporcionando acceso a un recurso a personal no autorizado a acceder a él.

Por último, y antes de entrar más a fondo con los esquemas de autenticación biométrica clásicos, quizás es conveniente desmentir uno de los grandes mitos de estos modelos: la vulnerabilidad a ataques de simulación. En cualquier película o libro de espías que se precie, siempre se consigue `engañar' a autenticadores biométricos para conseguir acceso a determinadas instalaciones mediante estos ataques: se simula la parte del cuerpo a analizar mediante un modelo o incluso utilizando órganos amputados a un cadáver o al propio usuario vivo (crudamente, se le corta una mano o un dedo, se le saca un ojo...para conseguir que el sistema permita la entrada). Evidentemente, esto sólo sucede en la ficción: hoy en día cualquier sistema biométrico - con excepción, quizás, de algunos modelos basados en voz de los que hablaremos luego - son altamente inmunes a estos ataques. Los analizadores de retina, de iris, de huellas o de la geometría de la mano son capaces, aparte de decidir si el miembro pertenece al usuario legítimo, de determinar si éste está vivo o se trata de un cadáver.

Verificación de voz

En los sistemas de reconocimiento de voz no se intenta, como mucha gente piensa, reconocer lo que el usuario dice, sino identificar una serie de sonidos y sus características para decidir si el usuario es quien dice ser. Para autenticar a un usuario utilizando un reconocedor de voz se debe disponer de ciertas condiciones para el correcto registro de los datos, como ausencia de ruidos, reverberaciones o ecos; idealmente, estas condiciones han de ser las mismas siempre que se necesite la autenticación.

Cuando un usuario desea acceder al sistema pronunciará unas frases en las cuales reside gran parte de la seguridad del protocolo; en algunos modelos, los denominados de texto dependiente, el sistema tiene almacenadas un conjunto muy limitado de frases que es capaz de reconocer: por ejemplo, imaginemos que el usuario se limita a pronunciar su nombre, de forma que el reconocedor lo entienda y lo autentique. Como veremos a continuación, estos modelos proporcionan poca seguridad en comparación con los de texto independiente, donde el sistema va `proponiendo' a la persona la pronunciación de ciertas palabras extraídas de un conjunto bastante grande. De cualquier forma, sea cual sea el modelo, lo habitual es que las frases o palabras sean características para maximizar la cantidad de datos que se pueden analizar (por ejemplo, frases con una cierta entonación, pronunciación de los diptongos, palabras con muchas vocales...). Conforme va hablando el usuario, el sistema registra toda la información que le es útil; cuando termina la frase, ya ha de estar en disposición de facilitar o denegar el acceso, en función de la información analizada y contrastada con la de la base de datos.

El principal problema del reconocimiento de voz es la inmunidad frente a *replay attacks*, un modelo de ataques de simulación en los que un atacante reproduce (por ejemplo, por medio de un magnetófono) las frases o palabras que el usuario legítimo pronuncia para acceder al sistema. Este problema es especialmente grave en los sistemas que se basan en textos preestablecidos: volviendo al ejemplo anterior, el del nombre de cada usuario, un atacante no tendría más que grabar a una persona que pronuncia su nombre ante el autenticador y luego reproducir ese sonido para conseguir el acceso; casi la única solución consiste en utilizar otro sistema de autenticación junto al reconocimiento de voz. Por contra, en modelos de texto independiente, más interactivos, este ataque no es tan sencillo porque la autenticación se produce realmente por una especie de desafío-respuesta entre el usuario y la máquina, de forma que la cantidad de texto grabado habría de ser mucho mayor - y la velocidad para localizar la parte del texto que el sistema propone habría de ser elevada -. Otro grave problema de los sistemas basados en reconocimiento de voz es el tiempo que el usuario emplea

hablando delante del analizador, al que se añade el que éste necesita para extraer la información y contrastarla con la de su base de datos; aunque actualmente en la mayoría de sistemas basta con una sola frase, es habitual que el usuario se vea obligado a repetirla porque el sistema le deniega el acceso (una simple congestión hace variar el tono de voz, aunque sea levemente, y el sistema no es capaz de decidir si el acceso ha de ser autorizado o no; incluso el estado anímico de una persona varía su timbre...). A su favor, el reconocimiento de voz posee la cualidad de una excelente acogida entre los usuarios, siempre y cuando su funcionamiento sea correcto y éstos no se vean obligados a repetir lo mismo varias veces, o se les niegue un acceso porque no se les reconoce correctamente.

Verificación de escritura

Aunque la escritura (generalmente la firma) no es una característica estrictamente biométrica, como hemos comentado en la introducción se suele agrupar dentro de esta categoría; de la misma forma que sucedía en la verificación de la voz, el objetivo aquí no es interpretar o entender lo que el usuario escribe en el lector, sino autenticarlo basándose en ciertos rasgos tanto de la firma como de su rúbrica.

La verificación en base a firmas es algo que todos utilizamos y aceptamos día a día en documentos o cheques; no obstante, existe una diferencia fundamental entre el uso de las firmas que hacemos en nuestra vida cotidiana y los sistemas biométricos; mientras que habitualmente la verificación de la firma consiste en un simple análisis visual sobre una impresión en papel, estática, en los sistemas automáticos no es posible autenticar usuarios en base a la representación de los trazos de su firma. En los modelos biométricos se utiliza además la forma de firmar, las características dinámicas (por eso se les suele denominar *Dynamic Signature Verification*, DSV): el tiempo utilizado para rubricar, las veces que se separa el bolígrafo del papel, el ángulo con que se realiza cada trazo...

Para utilizar un sistema de autenticación basado en firmas se solicita en primer lugar a los futuros usuarios un número determinado de firmas ejemplo, de las cuales el sistema extrae y almacena ciertas características; esta etapa se denomina de *aprendizaje*, y el principal obstáculo a su correcta ejecución son los usuarios que no suelen firmar uniformemente. Contra este problema la única solución (aparte de una concienciación de tales usuarios) es relajar las restricciones del sistema a la hora de *aprender* firmas, con lo que se decrementa su seguridad.

Una vez que el sistema conoce las firmas de sus usuarios, cuando estos desean acceder a él se les solicita tal firma, con un número limitado de intentos (generalmente más que los sistemas que autentican mediante contraseñas, ya que la firma puede variar en un individuo por múltiples factores). La firma

introducida es capturada por un lápiz óptico o por una lectora sensible (o por ambos), y el acceso al sistema se produce una vez que el usuario ha introducido una firma que el verificador es capaz de distinguir como auténtica.

Verificación de huellas

Típicamente la huella dactilar de un individuo ha sido un patrón bastante bueno para determinar su identidad de forma inequívoca, ya que está aceptado que dos dedos nunca poseen huellas similares, ni siquiera entre gemelos o entre dedos de la misma persona. Por tanto, parece obvio que las huellas se convertirían antes o después en un modelo de autenticación biométrico: desde el siglo pasado hasta nuestros días se vienen realizando con éxito clasificaciones sistemáticas de huellas dactilares en entornos policiales, y el uso de estos patrones fué uno de los primeros en establecerse como modelo de autenticación biométrica.

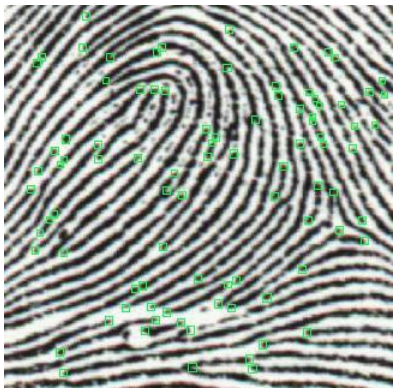


Figura 8.2: Huella dactilar con sus minucias extraídas. ©1998 Idex AS, http://www.idex.no/.

Cuando un usuario desea autenticarse ante el sistema situa su dedo en un área determinada (área de lectura, no se necesita en ningún momento una impresión en tinta). Aquí se toma una imagen que posteriormente se normaliza mediante un sistema de finos espejos[9.2] para corregir ángulos, y es de esta imagen normalizada de la que el sistema extrae las minucias (ciertos arcos, bucles o remolinos de la huella) que va a comparar contra las que tiene en su base de datos; es importante resaltar que lo que el sistema es capaz de analizar no es la huella en sí sino que son estas minucias, concretamente la posición relativa de cada una de ellas. Está demostrado que dos dedos nunca pueden poseer más de ocho minucias comunes, y cada uno tiene al menos 30 o 40 de éstas (en la figura 8.2 podemos ver una imagen de una huella digitalizada con sus minucias). Si la comparación de las posiciones relativas de las minucias leídas con las almacenadas en la base de datos es correcta, se permite el acceso al usuario, denegándosele obviamente en caso contrario.

Los sistemas basados en reconocimiento de huellas son relativamente baratos (en comparación con otros biométricos, como los basados en patrones retinales); sin embargo, tienen en su contra la incapacidad temporal de autenticar usuarios que se hayan podido herir en el dedo a reconocer (un pequeño corte o una quemadura que afecte a varias minucias pueden hacer inútil al sistema). También elementos como la suciedad del dedo, la presión ejercida sobre el lector o el estado de la piel pueden ocasionar lecturas erróneas. Otro factor a tener muy en cuenta contra estos sistemas es psicológico, no técnico: hemos dicho en la introducción que un sistema de autenticación de usuarios ha de ser aceptable por los mismos, y generalmente el reconocimiento de huellas se asocia a los criminales, por lo que muchos usuarios recelan del reconocedor y de su uso ([vKPG97]).

Verificación de patrones oculares

Los modelos de autenticación biométrica basados en patrones oculares se dividen en dos tecnologías diferentes: o bien analizan patrones retinales, o bien analizan el iris. Estos métodos se suelen considerar los más efectivos: para una población de 200 millones de potenciales usuarios la probabilidad de coincidencia es casi 0, y además una vez muerto el individuo los tejidos oculares degeneran rápidamente, lo que dificulta la falsa aceptación de atacantes que puedan robar este órgano de un cadáver.

La principal desventaja de los métodos basados en el análisis de patrones oculares es su escasa aceptación; el hecho de mirar a través de un binocular (o monocular), necesario en ambos modelos, no es cómodo para los usuarios, ni aceptable para muchos de ellos: por un lado, los usuarios *no se fían* de un haz de rayos analizando su ojo[9.3], y por otro un examen de este órgano puede revelar enfermedades o características médicas que a muchas personas les puede interesar mantener en secreto, como el consumo de alcohol o de ciertas drogas. Aunque los fabricantes de dispositivos lectores aseguran que sólo se analiza el ojo para obtener patrones relacionados con la autenticación, y en ningún caso se viola la privacidad de los usuarios, mucha gente no cree esta postura oficial (aparte del hecho de que la información es procesada vía *software*, lo que facilita introducir modificaciones sobre lo que nos han vendido para que un lector realice otras tareas de forma enmascarada). Por si esto fuera poco, se trata de sistemas demasiado caros para la mayoría de organizaciones, y el proceso de autenticación no es todo lo rápido que debiera en poblaciones de usuarios elevadas. De esta forma, su uso se ve reducido casi sólo a la identificación en sistemas de alta seguridad, como el control de acceso a instalaciones militares.

Retina

La vasculatura retinal (forma de los vasos sanguíneos de la retina humana) es un elemento característico de cada individuo, por lo que numerosos estudios en el campo de la autenticación de usuarios se basan en el reconocimiento de esta vasculatura.

En los sistemas de autenticación basados en patrones retinales el usuario a identificar ha de mirar a través de unos binoculares, ajustar la distancia interocular y el movimiento de la cabeza, mirar a un punto determinado y por último pulsar un botón para indicar al dispositivo que se encuentra listo para el análisis. En ese momento se escanea la retina con una radiación infrarroja de baja intensidad en forma de espiral, detectando los nodos y ramas del área retinal para compararlos con los almacenados en una base de datos; si la muestra coincide con la almacenada para el usuario que el individuo dice ser, se permite el acceso.

Uso de Contraseñas

Una **contraseña** o **clave** (en inglés ***password***) es una forma de autentificación que utiliza información secreta para controlar el acceso hacia algún recurso. La contraseña normalmente debe mantenerse en secreto ante aquellos a quien no se les permite el acceso. Aquellos que desean acceder a la información se les solicita una clave; si conocen o no conocen la contraseña, se concede o se niega el acceso a la información según sea el caso.

El uso de contraseñas se remonta a la antigüedad: los centinelas que vigilaban una posición solicitaban el «santo y seña» al que quisiera pasar. Solamente le permiten el acceso a aquella persona que conoce la seña. En la era tecnológica, las contraseñas son usadas comúnmente para controlar el acceso a sistemas operativos de computadoras protegidas, teléfonos celulares, decodificadores de TV por cable, cajeros automáticos de efectivo, etc. Un típico ordenador puede hacer uso de contraseñas para diferentes propósitos, incluyendo conexiones a cuentas de usuario, accediendo al correo electrónico (e-mail) de los servidores, accediendo a bases de datos, redes, y páginas Web, e incluso para leer noticias en los periódicos (diarios) electrónicos.

En la lengua inglesa se tienen dos denominaciones distintivas para las contraseñas: **password** (palabra de acceso) y **pass code** (código de acceso), donde la primera no implica necesariamente usar alguna palabra existente (sin embargo, es normal emplear alguna palabra familiar o de fácil memorización por parte del usuario), la primera suele asociarse también al uso de códigos alfanuméricos (también llamado **PIT** - *Personal Identification Text*), mientras que la segunda frecuentemente se liga a la utilización de algún código numérico (asimismo llamado **PIN** - *Personal Identification Number*). Esto ocurre igualmente

en el habla española, ya que en ocasiones clave y contraseña se usan indistintamente.

Posibilidad de que algún atacante pueda adivinar o inventar la contraseña

La posibilidad de que algún atacante pueda proporcionar una contraseña que adivinó es un factor clave al determinar la seguridad de un sistema. Algunos sistemas imponen un límite de tiempo después de que sucede un pequeño número de intentos fallidos de proporcionar la clave. Al no tener otras vulnerabilidades, estos sistemas pueden estar relativamente seguros con simples contraseñas, mientras estas no sean fácilmente deducibles, al no asignar datos fácilmente conocidos como nombres de familiares o de mascotas, el número de matrícula del automóvil o contraseñas sencillas como "*administrador*" o "*1234*".

Otros sistemas almacenan o transmiten una pista a modo de sugerencia de recordatorio de la contraseña, de manera que la propia pista puede ser fundamental para el acceso de algún atacante. Cuando esto ocurre, (y suele ser común), el atacante intentará suministrar contraseñas frecuentemente en una alta proporción, quizás utilizando listas extensamente conocidas de contraseñas comunes. También están sujetas a un alto grado de vulnerabilidad aquellas contraseñas que se usan para generar claves criptográficas, por ejemplo, cifrado de discos, o seguridad wi-fi, por lo tanto son necesarias contraseñas más inaccesibles en estos casos.

Formas de almacenar contraseñas

Algunos sistemas almacenan contraseñas como archivos de texto. Si algún atacante gana acceso al archivo que contienen las contraseñas, entonces todas éstas se encontrarán comprometidas. Si algunos usuarios emplean la misma contraseña para diferentes cuentas, éstas estarán comprometidas de igual manera. Los mejores sistemas almacenan las contraseñas en una forma de protección criptográfica, así, el acceso a la contraseña será más difícil para algún espía que haya ganado el acceso interno al sistema, aunque la validación todavía sigue siendo posible.

Un esquema criptográfico común almacena solamente el texto de la contraseña codificado, conocido como*hash*. Cuando un usuario teclea la contraseña en este tipo de sistema, se genera a partir de la contraseña y mediante un algoritmo el código hash equivalente para esa contraseña, y si el resultante (hash) coincide con el valor almacenado, se permite el acceso al usuario.

El texto codificado de la contraseña se crea al aplicar una función criptográfica usando la contraseña y normalmente, otro valor conocido como *salt* en inglés. El *salt* previene que los atacantes construyan una lista de valores para contraseñas comunes. Las funciones criptográficas más comunes son la MD5 y SHA1. Una versión modificada de DES fue utilizada en los primeros sistemas Unix.

Procedimientos para cambiar las contraseñas

Usualmente, un sistema debe proveer una manera de cambiar una contraseña, ya sea porque el usuario sospeche que la contraseña actual ha (o ha sido) descubierto, o como medida de precaución. Si la nueva contraseña es introducida en el sistema de una manera no cifrada, la seguridad puede haberse perdido incluso antes de que la nueva contraseña haya sido instalada en la base de datos. Si la nueva contraseña fue revelada a un empleado de confianza, se gana poco. Algunos web sites incluyen la opción de recordar la contraseña de un usuario de una manera no cifrada al mandárselo por e-mail.

Los **Sistemas de Administración de Identidad**, se utilizan cada vez más para automatizar la emisión de reemplazos para contraseñas perdidas. La identidad del usuario se verifica al realizar algunas preguntas y compararlas con las que se tienen almacenadas. Preguntas típicas incluyen las siguientes: "¿Dónde naciste?", "¿Cuál es tu película favorita?", "¿Cuál es el nombre de tu mascota?" En muchos casos las respuestas a estas preguntas pueden ser adivinadas, determinadas con un poco de investigación, u obtenidas a través de estafa con ingeniería social. Mientras que muchos usuarios han sido advertidos para que nunca revelen su contraseña, muy pocos consideran el nombre de su película favorita para requerir este tipo de seguridad.

Probabilidad que una contraseña pueda ser descubierta

Estudios en la producción de sistemas informáticos han indicado por décadas constantemente que cerca de 40% de las contraseñas elegidas por usuarios se conjeturan fácilmente.

- Muchos de los usuarios no cambian la contraseña que viene predeterminada en muchos de los sistemas de seguridad. Las listas de estas contraseñas están disponibles en el Internet.

- Una contraseña puede ser determinada si un usuario elige como contraseña un dato personal que sea fácil de descubrir (por ejemplo: el número de ID o el número de cuenta de un estudiante, el nombre del novio/a, la fecha de cumpleaños, el número telefónico, etc.). Los datos personales sobre individuos están ahora disponibles en diferentes fuentes, muchas de ellas están en línea, y pueden obtenerse frecuentemente por alguien que use técnicas de ingeniería social, como actuar como un trabajador social que realiza encuestas.

- Una contraseña es vulnerable si puede encontrarse en una lista. Los diccionarios (frecuentemente de forma electrónica) están disponibles en muchos lenguajes, y existen listas de contraseñas comunes.

- En pruebas sobre sistemas en vivo, los ataques de diccionarios son rutinariamente acertados, por lo que el software implementado en este tipo de ataques ya se encuentra disponible para muchos sistemas. Una contraseña muy corta, quizás elegida por conveniencia, es más vulnerable si un hacker puede obtener la versión criptográfica de la contraseña. Las computadoras son en la actualidad lo suficientemente rápidas para intentar todas las contraseñas en orden alfabético que tengan menos de 7 caracteres, por ejemplo:

Una **contraseña débil** sería una que fuese muy corta o que fuese la predeterminada, o una que pudiera adivinarse rápidamente al buscar una serie de palabras que es posible encontrar en diccionarios, nombres propios, palabras basadas en variaciones del nombre del usuario. Una contraseña fuerte debe ser suficientemente larga, al azar, o producirse sólo por el usuario que la eligió, de modo tal que el 'adivinarla' requiera un largo tiempo. Ese tiempo 'demasiado largo' variará de acuerdo al atacante, sus recursos, la facilidad con la que la contraseña se pueda descubrir, y la importancia de ésta para el atacante. Por lo tanto, una contraseña de un estudiante quizás no valga la pena para invertir más de algunos segundos en la computadora, mientras que la contraseña para acceder al control de una transferencia de dinero del sistema de un banco puede valer varias semanas de trabajo en una computadora.

'Fuerte' y 'débil' tienen significado solamente con respecto a tentativas de descubrir la contraseña de un usuario, ya sea por una persona que conoce al usuario, o una computadora que trate de usar millones de combinaciones. En este contexto, los términos pueden tener una precisión considerable. Pero nótese que una contraseña 'fuerte' en este sentido puede ser robada, truqueada o extraída del usuario ya sea mediante la extracción del historial de un teclado, grabada mediante aparatos de comunicación o copiada de notas dejadas por olvido.

Protocolos utilizados

Los protocolos de autenticación se negocian inmediatamente después de determinar la calidad del vínculo y antes de negociar el nivel de red.

Algunos protocolos de autentificación son:

* PAP: Protocolo de autentificación de contraseña
* CHAP: Protocolo de autentificación por desafío mutuo
* SPAP: Protocolo de autentificación de contraseña de Shiva
* MS-CHAP y MS-CHAP v2: Protocolo de autentificación por desafío mutuo de

Microsoft (variantes de CHAP)

* EAP: Protocolo de autentificación extensible
* Diameter
* Kerberos
* NTLM (también conocido como NT LAN Manager)
* PEAP:P rotocolo de autenticación extensible protegido
* RADIUS
* TACACS y TACACS+

CHAP es un protocolo de autenticación por desafío mutuo (CHAP, en inglés: *Challenge Handshake Authentication Protocol*).

Es un método de autentificación remota o inalámbrica. Diversos proveedores de servicios emplean CHAP. Por ejemplo, para autentificar a un usuario frente a un ISP.

La definición de CHAP está contenida en la RFC 1994.

CHAP es un método de autentificación usado por servidores accesibles vía PPP. CHAP verifica periódicamente la identidad del cliente remoto usando un intercambio de información de tres etapas. Esto ocurre cuando se establece el enlace inicial y puede pasar de nuevo en cualquier momento de la comunicación. La verificación se basa en un secreto compartido (como una contraseña).

1. Después del establecimiento del enlace, el agente autentificador manda un mensaje que "desafía" al usuario.

2. El usuario responde con un valor calculado usando una función hash de un sólo sentido, como la suma de comprobación MD5.

3. El autentificador verifica la respuesta con el resultado de su propio cálculo de la función hash. Si el valor coincide, el autentificador informa de la autentificación, de lo contrario terminaría la conexión.

4. A intervalos aleatorios el autentificador manda un nuevo "desafío" con lo que se repite el proceso.

CHAP protege contra los ataques de REPLAY mediante el uso de un identificador

que se va incrementando y un valor de desafío variable. CHAP requiere que el cliente mantenga el secreto disponible en texto plano.

DIAMETER

DIAMETER es un protocolo de red para la autenticación de los usuarios que se conectan remotamente a la Internet a través de la conexión por línea conmutada o TLC, también provee servicios de autorización y auditoría para aplicaciones tales como acceso de red o movilidad IP. El concepto básico del protocolo DIAMETER, cuyo desarrollo se ha basado en el protocolo RADIUS, es de proporcionar un protocolo base que pueda ser extendido para proporcionar servicios de autenticación, autorización y auditoría, (denominados por los expertos por sus siglas AAA) a nuevas tecnologías de acceso. DIAMETER está diseñado para trabajar tanto de una manera local como en un estado de alerta, sondeo y captura, que en inglés se le denomina *roaming* de AAA, que le permite ofrecer servicios sumamente móviles, dinámicos, flexibles y versátiles.

El nombre es un juego de palabras respecto al protocolo RADIUS, su predecesor (un diámetro es el doble del radio). Diameter no es directamente compatible hacia atrás, pero proporciona un método de actualización desde RADIUS. Las principales diferencias son:

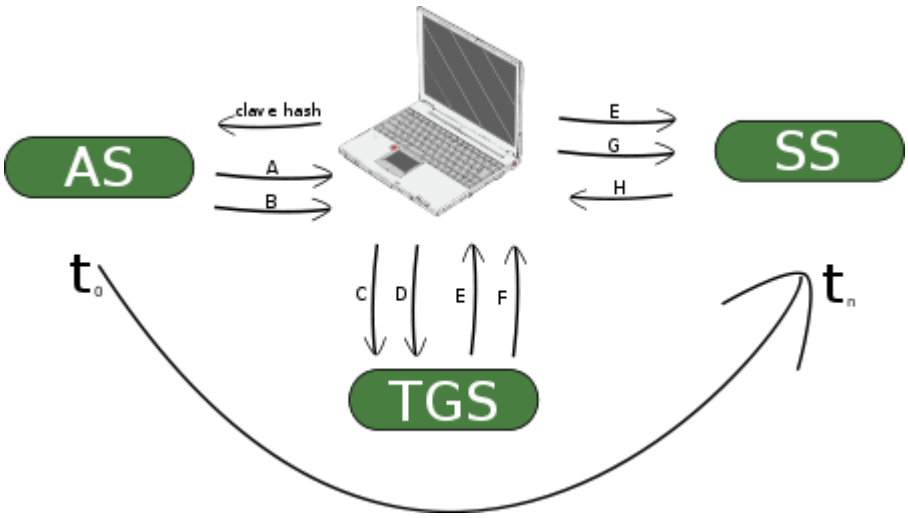
- usa protocolos de transportes fiables (TCP o SCTP, no UDP)
- usa seguridad a nivel de transporte (IPSEC o TLS)
- tiene compatibilidad transicional con RADIUS
- tiene un espacio de direcciones mayor para AVPs (*Attribute Value Pairs*, pares atributo-valor) e identificadores (32 bits en lugar de 8)
- es un protocolo peer-to-peer en lugar de cliente-servidor: admite mensajes iniciados por el servidor
- pueden usarse modelos con y sin estado
- tiene descubrimiento dinámico de peers (usando DNS SRV y NAPTR)
- tiene negociación de capacidades
- admite ACKs en el nivel de aplicación, definiendo métodos de fallo y máquinas de estado (RFC 3539)
- tiene notificación de errores
- tiene mejor compatibilidad con roaming
- es más fácil de extender, pudiendo definirse nuevos comandos y atributos
- incluye una implementación básica de sesiones y control de usuarios

Kerberos es un protocolo de autenticación de redes de ordenador que permite a dos computadores en una red insegura demostrar su identidad mutuamente de manera segura. Sus diseñadores se concentraron primeramente en un modelo de cliente-servidor, y brinda autenticación mutua: tanto cliente como servidor verifican

la identidad uno del otro. Los mensajes de autenticación están protegidos para evitar eavesdropping y ataques de Replay.

Kerberos se basa en criptografía de clave simétrica y requiere un tercero de confianza. Además, existen extensiones del protocolo para poder utilizar criptografía de clave asimétrica.

Cómo funciona



Funcionamiento de Kerberos.

A continuación se describe someramente el protocolo. Se usaran las siguientes abreviaturas:

- AS = Authentication Server
- TGS = Ticket Granting Server
- SS = Service Server.

En resumen el funcionamiento es el siguiente: el cliente se autentica a sí mismo contra el AS, así demuestra al TGS que está autorizado para recibir un ticket de servicio (y lo recibe) y ya puede demostrar al SS que ha sido aprobado para hacer uso del servicio kerberizado.

En más detalle:

1. Un usuario ingresa su nombre de usuario y password en el cliente
2. El cliente genera una clave hash a partir del password y la usará como la clave secreta del cliente.
3. El cliente envía un mensaje en texto plano al AS solicitando servicio en nombre del usuario.
4. El AS comprueba si el cliente está en su base de datos. Si es así, el AS envía dos mensajes al cliente:

    1. Mensaje A: Client/TGS session key cifrada usando la clave secreta del usuario
    2. Mensaje B: Ticket-Granting Ticket (que incluye el ID de cliente, la dirección de red del cliente, el período de validez y el Client/TGS session key) cifrado usando la clave secreta del TGS.
5. Una vez que el cliente ha recibido los mensajes, descifra el mensaje A para obtener el client/TGS session key. Esta session key se usa para las posteriores comunicaciones con el TGS. (El cliente no puede descifrar el mensaje B pues para cifrar éste se ha usado la clave del TGS). En este momento el cliente ya se puede autenticar contra el TGS.
6. Entonces el cliente envía los siguientes mensajes al TGS:
    1. Mensaje C: Compuesto del Ticket-Granting Ticket del mensaje B y el ID del servicio solicitado.
    2. Mensaje D: Autenticador (compuesto por el ID de cliente y una marca de tiempo), cifrado usando el client/TGS session key.
7. Cuando recibe los mensajes anteriores, el TGS descifra el mensaje D (autenticador) usando el client/TGS session key y envía los siguientes mensajes al cliente:
    1. Mensaje E: Client-to-server ticket (que incluye el ID de cliente, la dirección de red del cliente, el período de validez y una Client/Server session key) cifrado usando la clave secreta del servicio.
    2. Mensaje F: Client/server session key cifrada usando el client/TGS session key.
8. Cuando el cliente recibe los mensajes E y F, ya tiene suficiente información para autenticarse contra el SS. El cliente se conecta al SS y envía los siguientes mensajes:
    1. Mensaje E del paso anterior.
    2. Mensaje G: un nuevo Autenticador que incluye el ID de cliente, una marca de tiempo y que está cifrado usando el client/server session key.
9. El SS descifra el ticket usando su propia clave secreta y envía el siguiente mensaje al cliente para confirmar su identidad:
    1. Mensaje H: la marca de tiempo encontrada en el último Autenticador recibido del cliente más uno, cifrado el client/server session key.
10. El cliente descifra la confirmación usando el client/server session key y chequea si la marca de tiempo está correctamente actualizada. Si esto es así, el cliente confiará en el servidor y podrá comenzar a usar el servicio que este ofrece.
11. El servidor provee del servicio al cliente.

Sistemas de Autenticación en Sistemas Operativos

Autenticación clásica

En un sistema Unix habitual cada usuario posee un nombre de entrada al sistema o login y una clave o password; ambos datos se almacenan generalmente en el fichero /etc/passwd. Este archivo contiene una línea por usuario donde se indica la información necesaria para que los usuarios puedan conectar al sistema y trabajar en él, separando los diferentes campos mediante `:'.

Al contrario de lo que mucha gente cree, Unix no es capaz de distinguir a sus usuarios por su nombre de entrada al sistema. Para el sistema operativo lo que realmente distingue a una persona de otra (o al menos a un usuario de otro) es el UID del usuario en cuestión; el login es algo que se utiliza principalmente para comodidad de las personas (obviamente es más fácil acordarse de un nombre de entrada como *toni* que de un UID como *2643*, sobre todo si se tienen cuentas en varias máquinas, cada una con un UID diferente).

Para cifrar las claves de acceso de sus usuarios, el sistema operativo Unix emplea un criptosistema irreversible que utiliza la función estándar de C crypt, basada en el algoritmo DES. Para una descripción exhaustiva del funcionamiento de crypt. Esta función toma como clave los ocho primeros caracteres de la contraseña elegida por el usuario (si la longitud de ésta es menor, se completa con ceros) para cifrar un bloque de texto en claro de 64 bits puestos a cero; para evitar que dos passwords iguales resulten en un mismo texto cifrado, se realiza una permutación durante el proceso de cifrado elegida de forma automática y aleatoria para cada usuario, basada en un campo formado por un número de 12 bits (con lo que conseguimos 4096 permutaciones diferentes) llamado salt. El cifrado resultante se vuelve a cifrar utilizando la contraseña del usuario de nuevo como clave, y permutando con el mismo salt, repitiéndose el proceso 25 veces. El bloque cifrado final, de 64 bits, se concatena con dos bits cero, obteniendo 66 bits que se hacen representables en 11 caracteres de 6 bits cada uno y que, junto con el salt, pasan a constituir el campo password del fichero de contraseñas, usualmente /etc/passwd. Así, los dos primeros caracteres de este campo estarán constituidos por el salt y los 11 restantes por la contraseña cifrada

Problemas del modelo clásico

Los ataques de texto cifrado escogido constituyen la principal amenaza al sistema de autenticación de Unix; a diferencia de lo que mucha gente cree, no es posible descifrar una contraseña, pero es muy fácil cifrar una palabra junto a un determinado salt, y comparar el resultado con la cadena almacenada en el fichero de claves. De esta forma, un atacante leerá el fichero /etc/passwd (este fichero ha de tener permiso de lectura para todos los usuarios si queremos que el sistema funcione correctamente), y mediante un programa adivinador (o crackeador) cifrará todas las palabras de un fichero denominado diccionario (un fichero ASCII con un gran número de palabras de cualquier idioma o campo de la sociedad: historia clásica, deporte, cantantes...), comparando el resultado obtenido en este

proceso con la clave cifrada del fichero de contraseñas; si ambos coinciden, ya ha obtenido una clave para acceder al sistema de forma no autorizada.

Shadow Password

Otro método cada día más utilizado para proteger las contraseñas de los usuarios el denominado Shadow Password u oscurecimiento de contraseñas. La idea básica de este mecanismo es impedir que los usuarios sin privilegios puedan leer el fichero donde se almacenan las claves cifradas.

Envejecimiento de contraseñas

En casi todas las implementaciones de Shadow Password actuales se suele incluir la implementación para otro mecanismo de protección de las claves denominado envejecimiento de contraseñas (Aging Password). La idea básica de este mecanismo es proteger los passwords de los usuarios dándoles un determinado periodo de vida: una contraseña sólo va a ser válida durante un cierto tiempo, pasado el cual expirará y el usuario deberá cambiarla.

Realmente, el envejecimiento previene más que problemas con las claves problemas con la transmisión de éstas por la red: cuando conectamos mediante mecanismos como telnet, ftp o rlogin a un sistema Unix, cualquier equipo entre el nuestro y el servidor puede leer los paquetes que enviamos por la red, incluyendo aquellos que contienen nuestro nombre de usuario y nuestra contraseña.