Ciberseguridad

Grado en Ingeniería Informática

M2 - Criptografía, Identificación y Control de Accesos

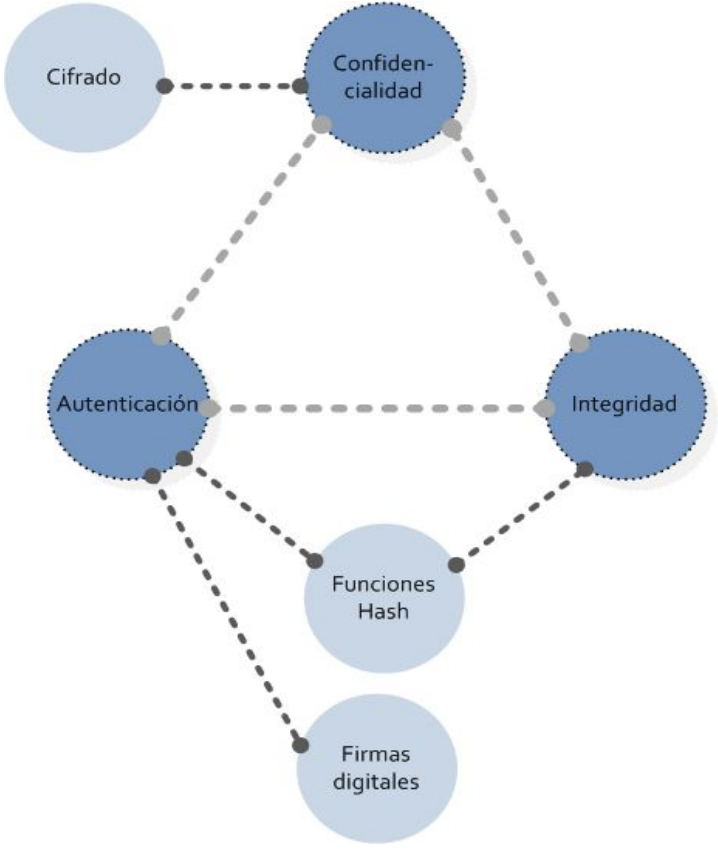
Oscar Delgado
oscar.delgado@uam.es

Álvaro Ortigosa (Coord.)
alvaro.ortigosa@uam.es

UAM Universidad Autónoma de Madrid

Autenticación

“Verificación de la identidad de otra parte”



Autenticación

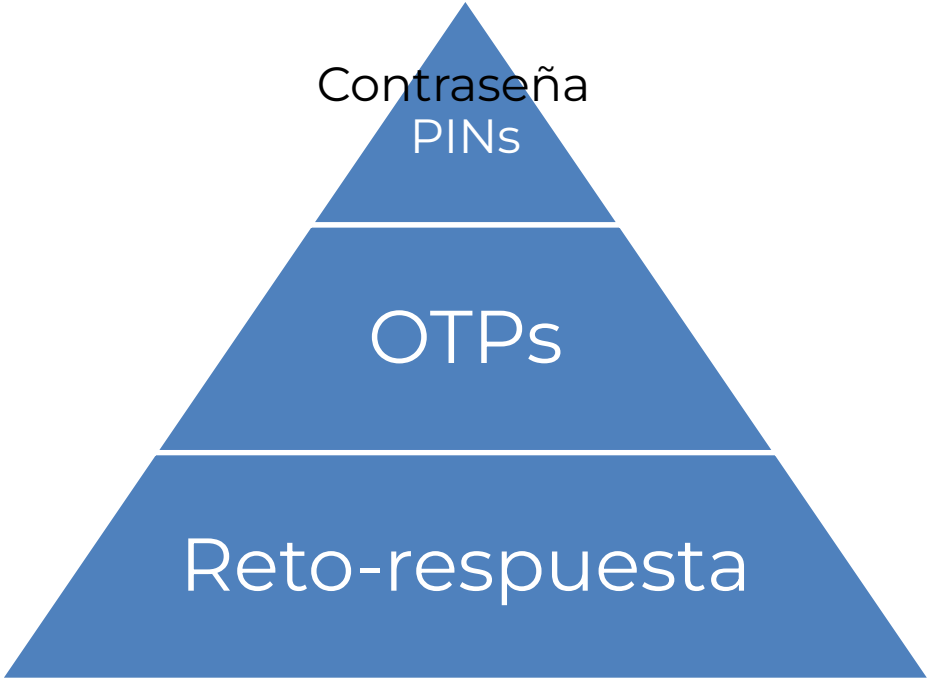
- La “otra parte” puede ser una persona, aplicación, máquina, etc...

- **Identificación**: autenticación específica de personas
- **Autorización**: gestión de los permisos de una parte ya autenticada

Autenticación

Algo que sabemos	Algo que tenemos	Algo que somos
• Contraseñas • PINs • Claves criptográficas	• DNIe • Tarjetas de coordenadas • Tokens OTPs • Teléfonos móviles	• Firma manuscrita • Huellas dactilares • Voz • Retina

Autenticación: fortaleza



1 factor (1F) = secreto

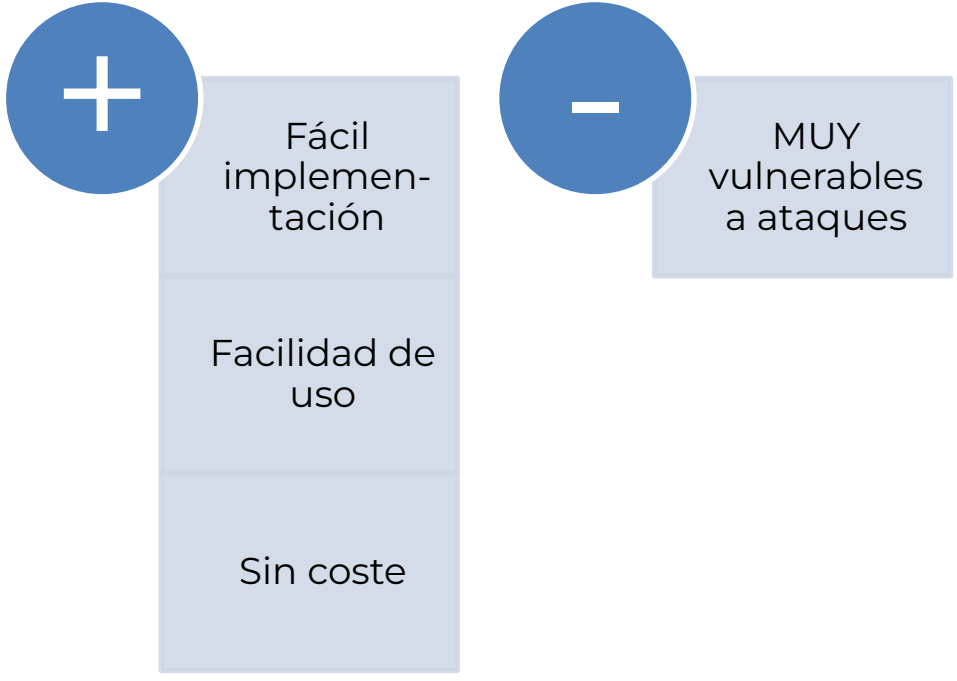
2F = 1F + token
(Doble factor)

3F = 2F + rasgo biométrico
(Diffie-Hellman (TLS))

Contraseñas

- Ambas partes, y sólo ellas, conocen un secreto compartido
- Primer método, y aún el más utilizado de largo

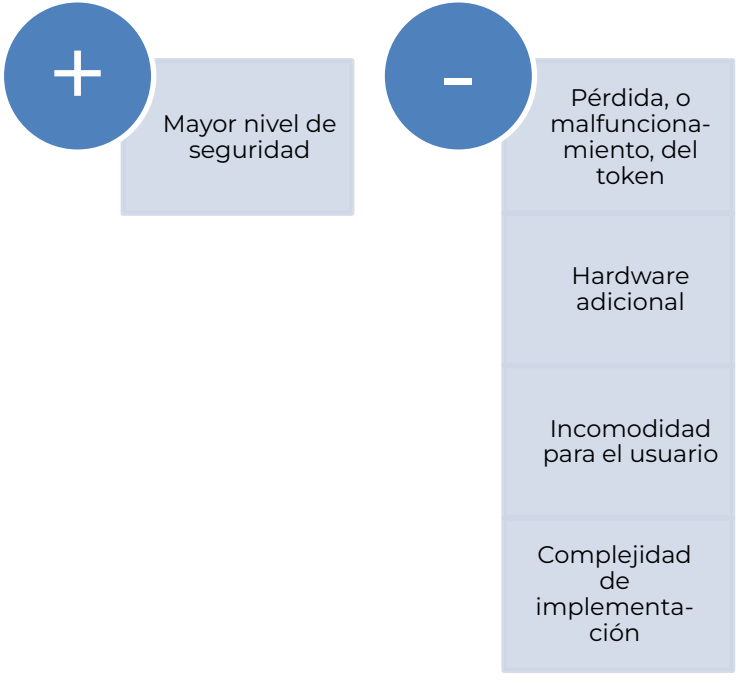
Contraseñas



Sistemas 2F

- El usuario debe demostrar, además del conocimiento de un secreto, la posesión de un **token**

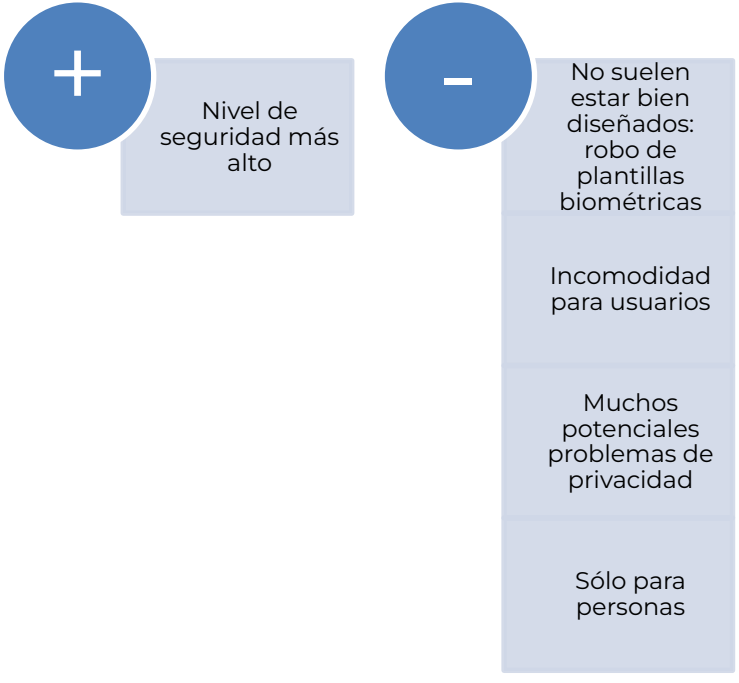
Sistemas 2F


+
Mayor nivel de seguridad
-
Pérdida, o malfunciona-miento, del token
Hardware adicional
Incomodidad para el usuario
Complejidad de implementa-ción

Sistemas biométricos

- Utilizan un rasgo biométrico para la identificación del usuario

Sistemas biométricos



Sistemas de doble factor

Autenticación

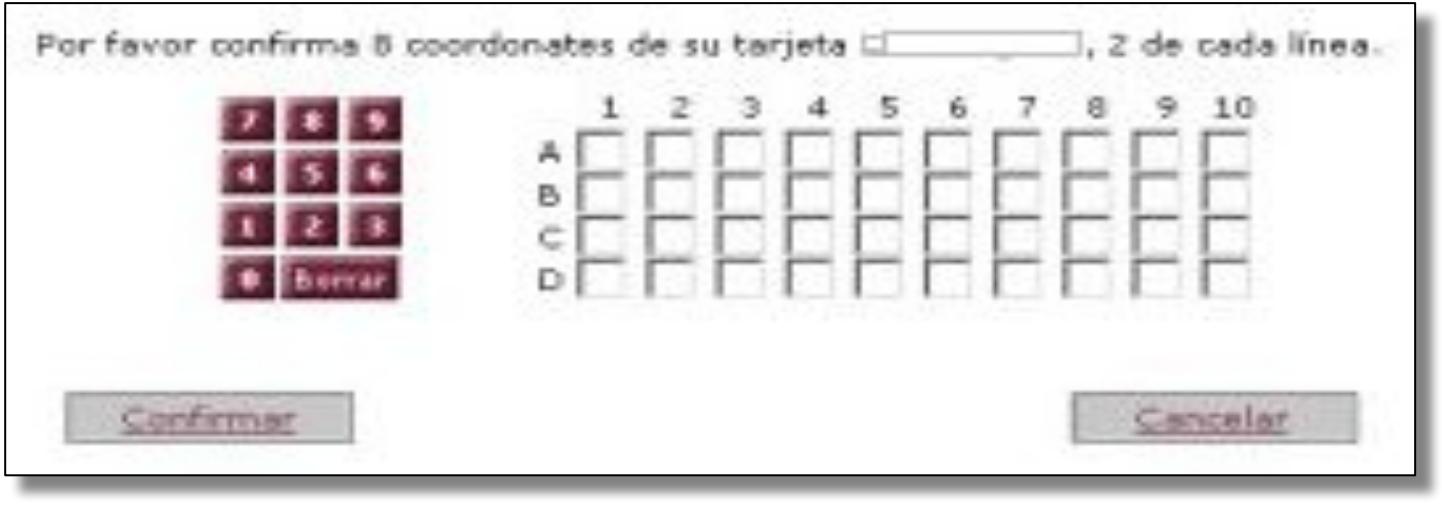
Algo que sabemos	Algo que tenemos	Algo que somos
• Contraseñas • PINs • Claves criptográficas	• DNIe • Tarjetas de coordenadas • Tokens OTPs • Teléfonos móviles	• Firma manuscrita • Huellas dactilares • Voz • Retina

Doble factor

- Pretenden superar los problemas de las contraseñas
- Se introduce un nuevo “secreto” fuera de banda (del mundo digital), típicamente un objeto físico

Tarjetas de coordenadas

Los delincuentes se adaptan a todo: piden una serie de coordenadas futuras

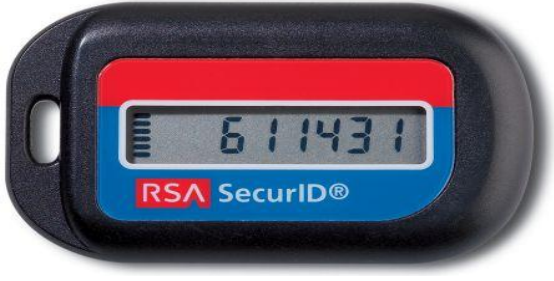


Tokens OTP

✔ Buen nivel de seguridad

- Incómodo (pérdidas)
- Coste elevado

¿Y el DNIe?



SMS

- ✔ Canal fuera de banda

- ☐ Incómodo (retrasos)
- ☐ Coste
- ☐ Vulnerable a ataques
- ☐

Tarjetas criptográficas

- Vulnerables a ataques de MITM en tiempo real

  - Un troyano puede capturar el PIN de acceso a la clave privada o cambiar el texto a firmar
  - Ataques de análisis de potencia (PAA)
    - ¡Muy comunes!
    - Trabajos de Ross Anderson
  - Ataques lógicos a la API de acceso

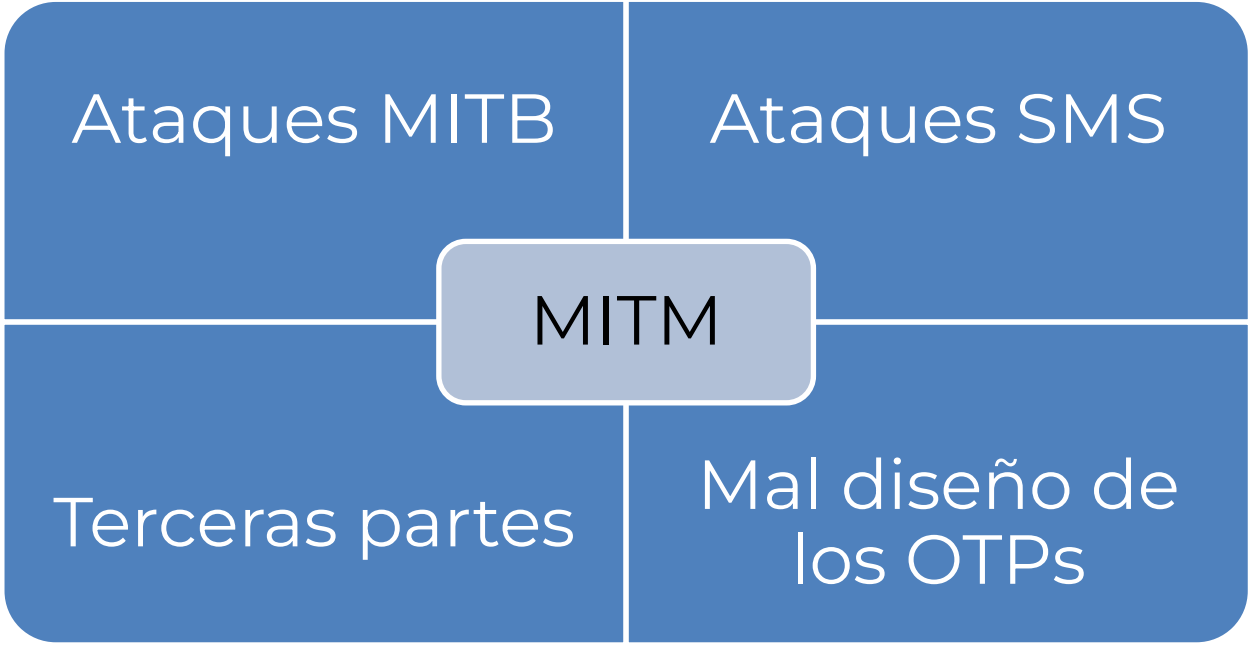
Doble factor en banca online

Esquema	Técnica	Ventajas	Inconvenientes
Algo que sabes	Contraseñas	Fáciles de utilizar	Pueden ser capturadas
Algo que posees	Tarjeta de coordenadas	Barata y relativamente sencilla de utilizar	• Puede perderse u olvidarse • Difícil de compartir
	Token hardware	Difícil manipulación	• Caro • Puede perderse u olvidarse • Difícil de compartir
	Teléfono móvil	No requiere hardware adicional	• Puede agotarse la batería o no tener cobertura • Coste de la comunicación para el cliente y el banco
Algo que eres	Ritmo de tecleo	• No requiere hardware adicional • Alta aceptabilidad	• No es fiable • Tiempo de registro elevado

Doble factor

Ataques

Ataques al doble factor



Terceras partes

- Los tokens OTPs dependen principalmente de terceras partes

- En 2011 RSA sufrió una gravísima vulnerabilidad en su SecurID:
  - Accedieron a la clave maestra que permitía generar nuevos tokens

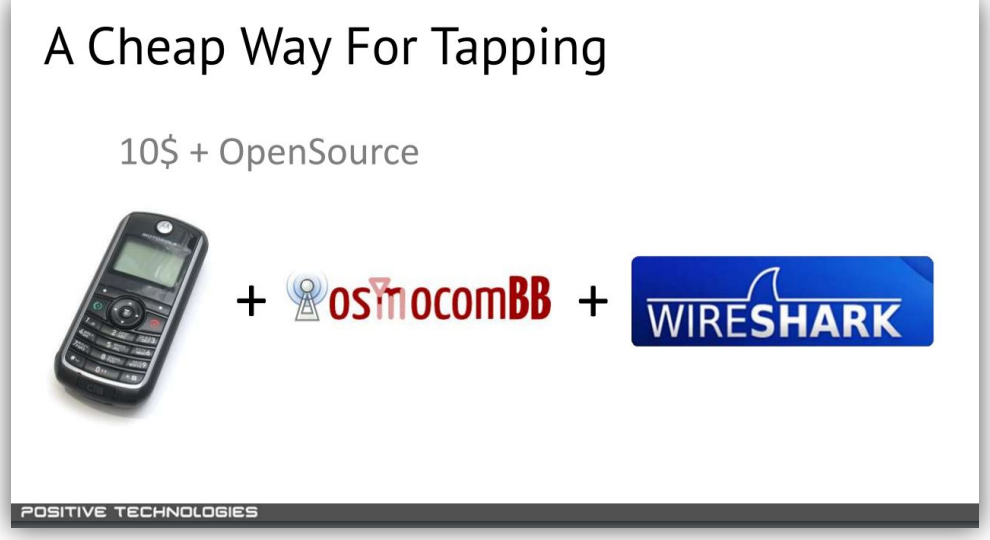
Ataques al doble factor

Ataques SMS

Ataques SMS

- **Spolier**

Es “sencillo” interceptar llamadas y SMS



Ataque al sistema de señalización SS7

Diseño de los OTPs

- En ocasiones, los OTPs son:
  - Predecibles
  - Reset de la pérdida de un token con un mecanismo mucho más débil: correo
  - Fallos de implementación: permiten reusar el token de un usuario con otro

MITB: seguridad del navegador

Ataques

Malware

- El perfil de ataque ha cambiado radicalmente:
  - Más del 80% de las intrusiones con éxito son vía Web
  - Se escribe malware específico y con un objetivo claro: robar credenciales
- La seguridad de una aplicación Web NO se puede separar de la del navegador del cliente

Troyanos

- Los troyanos no son, desde luego, nada nuevo:
  - En los 70's aparecieron los primeros troyanos que imitaban la pantalla de login del SO

- La profesionalización del crimen online los ha hecho resurgir como herramienta de fraude

Troyanos

- Los métodos tradicionales de phishing (SPAM, similitud y ofuscación de dominios) han ido dejando de funcionar paulatinamente

- **Consecuencia**: los phishers han buscado nuevas técnicas más sofisticadas: los troyanos especializados

Troyanos bancarios

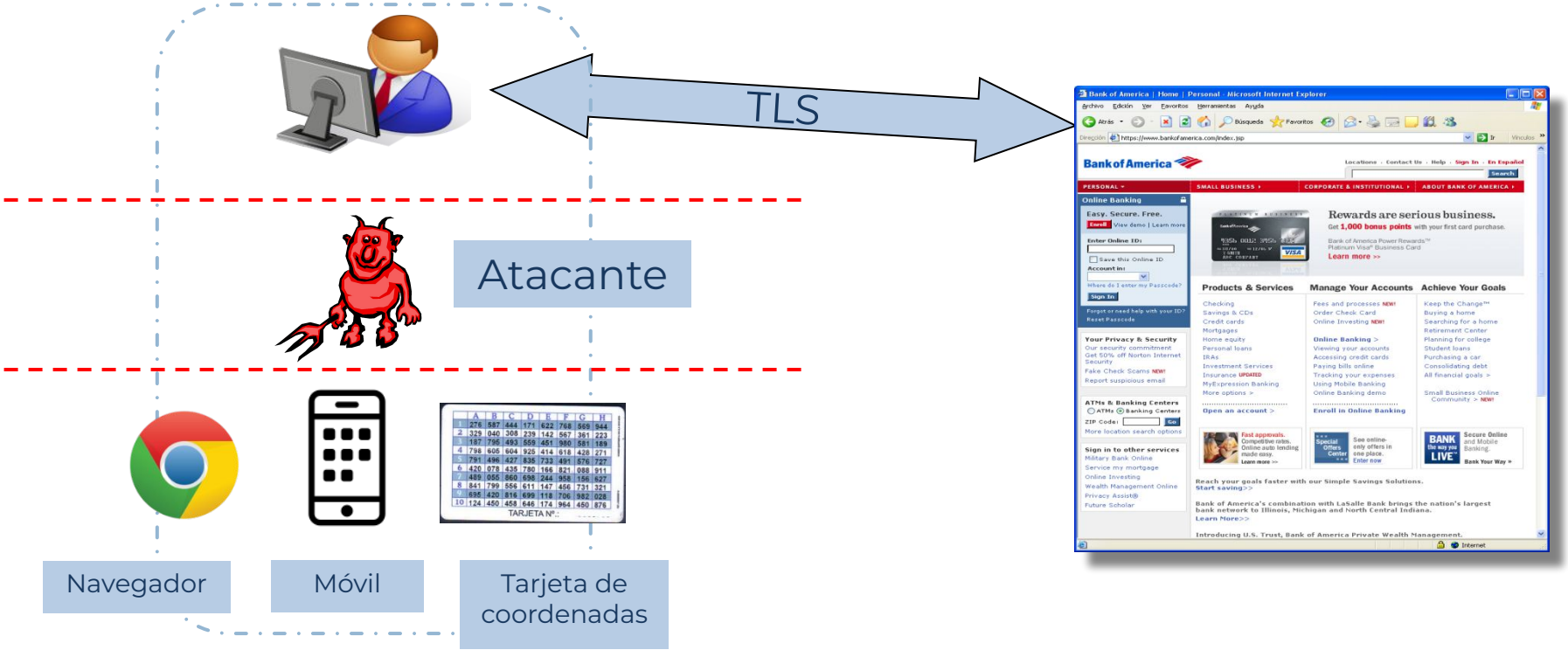
- Un troyano especializado en banca electrónica captura las credenciales y las envía a un "repositorio" central controlado por el atacante

- Las credenciales capturadas incluyen el usuario, la contraseña y la 'firma electrónica', necesaria para realizar transferencias y otras operaciones importantes

Ataques Man-in-the-Browser

- Sin duda, la GRAN amenaza para la seguridad Web
- Malware que realiza un ataque MITM al navegador:
  - Tiene acceso al DOM de la página en claro (incluso con SSL)
  - Puede realizar cualquier modificación ANTES del envío al servidor

Ataque genérico al doble factor



Atacando un banco

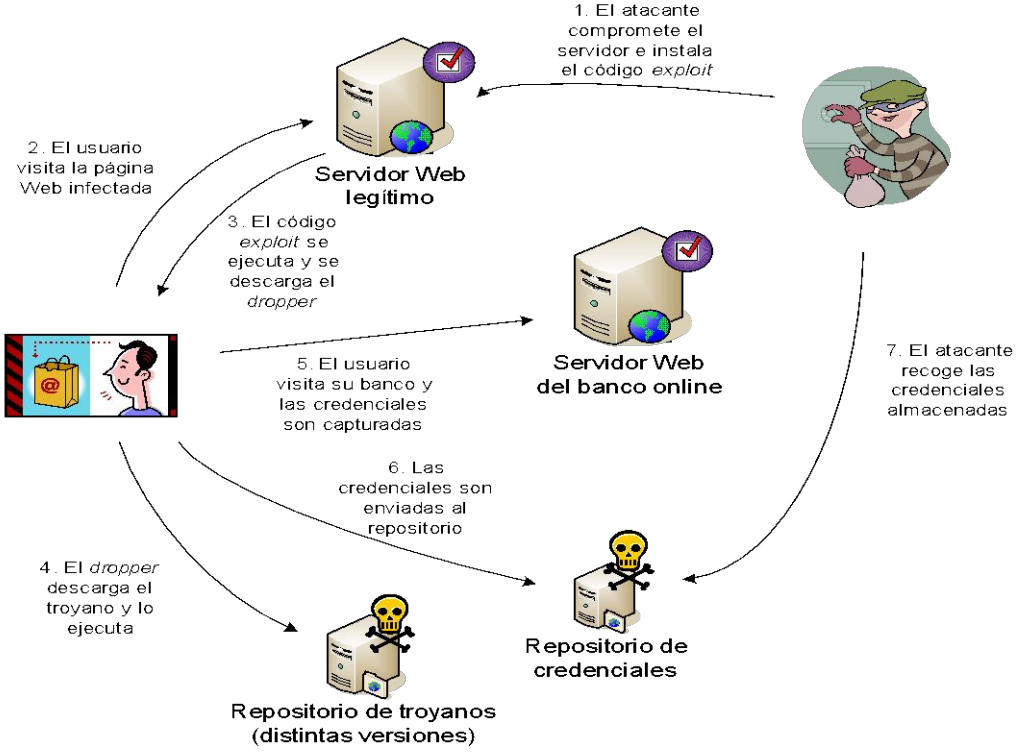
Vectores de infección

Web	Unidades USB y discos duros	Correo electrónico	Redes P2P
• Explotando vulnerabilidades en los navegadores • Vía más común	• Unidades de red • Dropbox	• PDFs, .zip	• Más del 80% de los ejecutables contienen algún tipo de malware

Infección vía Web

1. Comprometen sitios Web e instalan en ellos el código exploit del navegador
2. Utilizan vulnerabilidades de los navegadores más utilizados
3. El usuario, al visitar la página, provoca la ejecución del *dropper*
4. Éste descarga el código del troyano en sí y lo ejecuta.

Infección vía Web

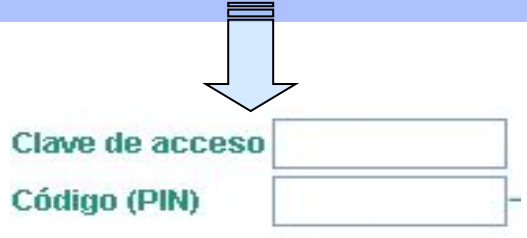


Inyección de código HTML

- Una de las técnicas más comunes de los troyanos MITB
- Consiste en manipular el código HTML de la página Web del banco antes de que ésta sea presentada al usuario

Inyección de código HTML

```
<td class="clave_acceso_pin" width="95">Clave de acceso</td>
<td width="60"><input NAME="USER" TYPE="text" SIZE="10" MAXLENGTH="10" value="" tabindex="1"></td>
<td class="clave_acceso_pin">Código (PIN)</td>
<td><input NAME="PIN" TYPE="password" SIZE="10" value="" tabindex="2" onfocus="javascript:foco_pin('PIN','LOGIN')"></td>
<td><img src="../3099/img/flecha.png" alt="flecha" title="usa el teclado virtual"></td>
<td rowspan="2" align="left">
<SCRIPT LANGUAGE="JavaScript" type="text/javascript"> teclado("PIN",'LOGIN'); </SCRIPT></td>
```



Inyección de código HTML

```
<td class="clave_acceso_pin" width="95">Clave de acceso</td>
<td width="60"><input NAME="USER" TYPE="text" SIZE="10" MAXLENGTH="10" value="" tabindex="1"></td>
<td class="clave_acceso_pin">Código (PIN)</td>
<td><input NAME="PIN" TYPE="password" SIZE="10" value="" tabindex="2"
onfocus="javascript:foco_pin('PIN','LOGIN')"></td>
<td><input NAME="Firma" TYPE="password" SIZE="10" value="" tabindex="2" onfocus=""></td>
<td><img src="../3099/img/flecha.png" alt="flecha" title="usa el teclado virtual"></td>
<td rowspan="2" align="left">
<SCRIPT LANGUAGE="JavaScript" type="text/javascript"> teclado("PIN",'LOGIN'); </SCRIPT></td>
```

Inyección de código HTML

- ¡Indetectable para un usuario no experto!

- El ‘candadito’ está perfectamente cerrado:
  - El certificado presentado por el banco es legítimo
  - La URL del servidor es correcta
  - El origen de los datos no ha sido manipulado, sólo su presentación al usuario

Inyección de código HTML

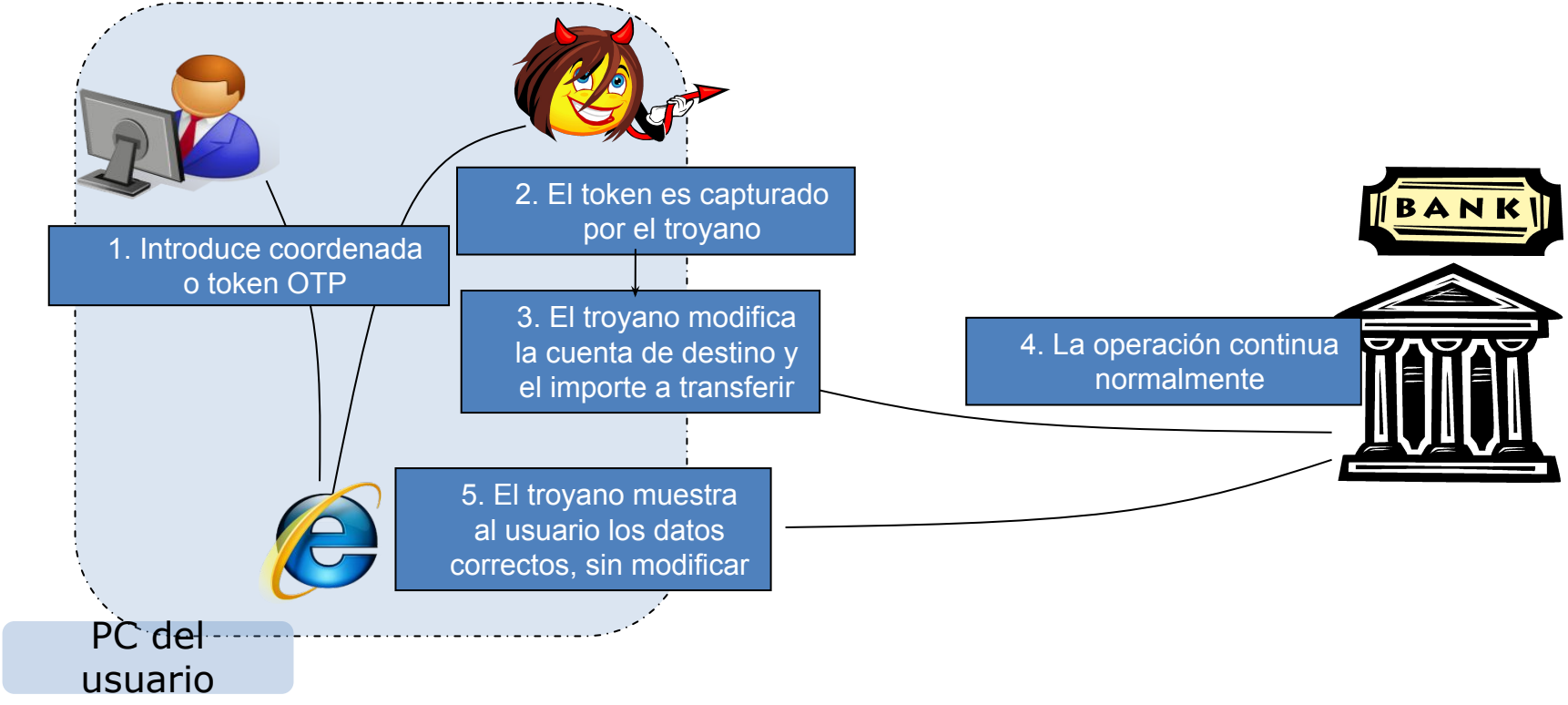
- ¿Dónde reside el problema?
  - Los navegadores no son confiables
  - Los datos se manipulan en capa de aplicación: una sesión SSL no es garantía de seguridad

Contramedidas actuales

- Los consejos habituales a los usuarios ya no son garantía de seguridad:
  - “Compruebe el candado del navegador”
  - “No pinche en ningún enlace, escriba directamente la URL en el navegador”
  - “Mantenga su antivirus actualizado”

- Aunque, por supuesto, no deben dejar de darse ni de seguirse

Ataques al doble factor: versión 2.0



Troyanos vs tarjetas criptográficas

- Un troyano puede capturar el PIN de la tarjeta antes de que se realice la correspondiente llamada al API

- Si queremos firmar un texto (¿declaración de la renta?)
  - ¿Cómo sabemos que el texto que se nos presenta para firmar es el que realmente se entrega a la tarjeta para su firma?

Ataques al doble factor

- Este ataque es perfectamente posible contra la práctica totalidad de la banca electrónica española.
- El problema radica en que el reto no es generado utilizando la información que se pretende autenticar:

  - Número de cuenta de destino
  - Importe de la transacción

Muy bien, ¿cómo lo arreglamos?

Método	Descripción	¿Efectivo contra MITB?	¿Por qué?
Usuario/ Contraseña	Incluso contraseñas fuertes, KDFs	No	Malware puede interceptar, modificar o esperar hasta que el usuario se haya autenticado correctamente
Tarjeta de coordenadas	Tanto en papel como electrónicas	No	
Token OTPs	Tanto hardware dedicado como SMS o apps en el móvil	No	

Muy bien, ¿cómo lo arreglamos?

Método	Descripción	¿Efectivo contra MITB?	¿Por qué?
EMV-CAP OTP	Tarjeta bancaria con chip	No	Malware puede interceptar, modificar o esperar hasta que el usuario se haya autenticado correctamente
Tarjeta inteligente (DNIe)	Certificado digital almacenado en un tarjeta criptográfica	No	
Anti-virus	-	A veces	La tasa de mutación de este malware es altísima: siempre hay una variante indetectable

Muy bien, ¿cómo lo arreglamos?

Método	Descripción	¿Efectivo contra MITB?	¿Por qué?
Ordenador aislado, usado únicamente para banca online	Teóricamente seguro, pero no realista en la mayoría de entornos	Sí, pero muy incómodo	
Navegador securizado que arranca desde un USB	Un navegador securizado, con la URL del banco hard-codeada	Sí, pero muy incómodo	Problemas en entornos corporativos que tienen USB y CD-ROM desactivados
EMV-CAP OTP	Con un lector especial, el cliente recibe los detalles de la transacción fuera de banda, en una pantalla	Sí, pero muy incómodo	Aún así, es fácil no ver que el último dígito de la cuenta es diferente, por ejemplo

Muy bien, ¿cómo lo arreglamos?

Método	Descripción	¿Efectivo contra MITB?	¿Por qué?
EMV-CAP OTP	Con un lector especial, el cliente recibe los detalles de la transacción fuera de banda, en una pantalla	Sí, pero muy incómodo	Aún así, es fácil no ver que el último dígito de la cuenta es diferente, por ejemplo

Doble factor

Resumen

Navegadores

- Tengamos en cuenta que trabajamos con hardware, SO y software (navegador) de propósito general

- Con esos requisitos, es **imposible** implementar un sistema **seguro**

- El mejor método posible: **OTP fuera de banda con vinculación de datos**

Resumen

- Las defensas exclusivamente preventivas no dan mucho más de sí (dado el entorno operativo de los usuarios)

- A largo plazo, la solución tendría que venir por aumentar la resistencia del sistema frente a su integridad:
  - Detección temprana del fraude, estudios de riesgo en tiempo real, retardar las transferencias a ciertos países)

Futuro del doble factor

- Generar modelos de comportamiento del usuario:
  - Los seres humanos somos bastante predecibles
  - Todo lo que se salga de lo “habitual” se confirma manualmente, en base a una puntuación de riesgo
- Utilizar técnicas de IA para autenticar al usuario, incluyendo datos del GPS, micrófono o acelerómetro:
  - El móvil del usuario está en Madrid, ¿por qué está intentando ordenar una transferencia desde Rusia?

Futuro del doble factor

- Los *wereables* jugarán también un papel importante:
  - Si un usuario no tiene cerca de su teléfono o PC una pulsera “autenticadora” la transacción no puede llevarse a cabo

API REST

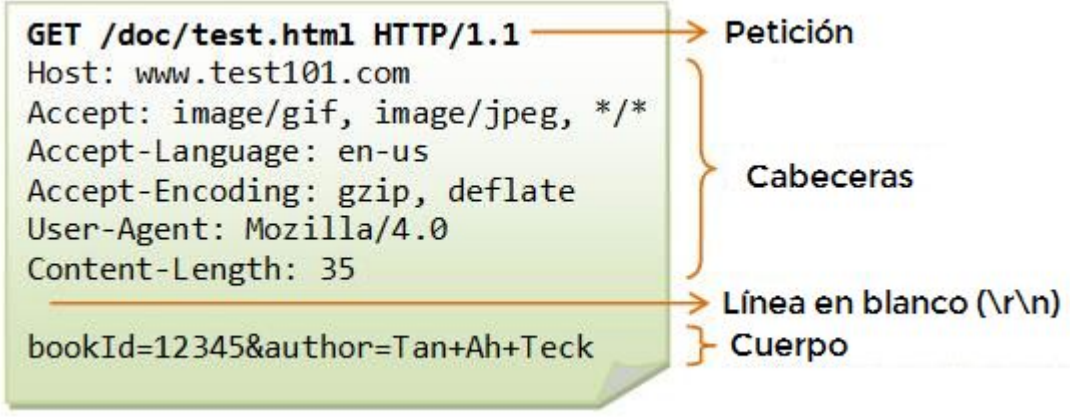
Introducción

Cambio de paradigma

- Las API REST no son ya soluciones de “startup”
- El futuro es interoperación, y esto solo es posible con estándares sencillos

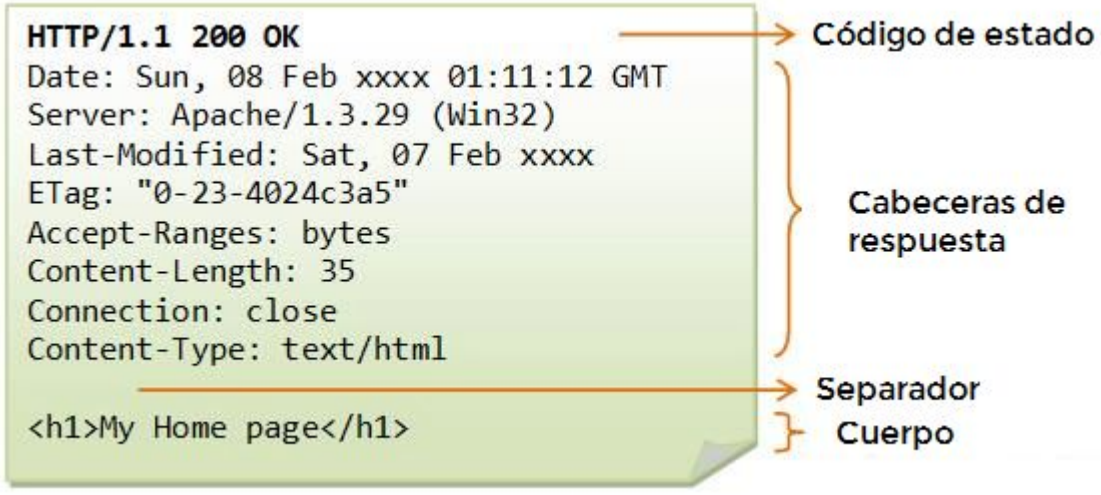
API REST

Basadas en HTTP



API REST

Basadas en HTTP



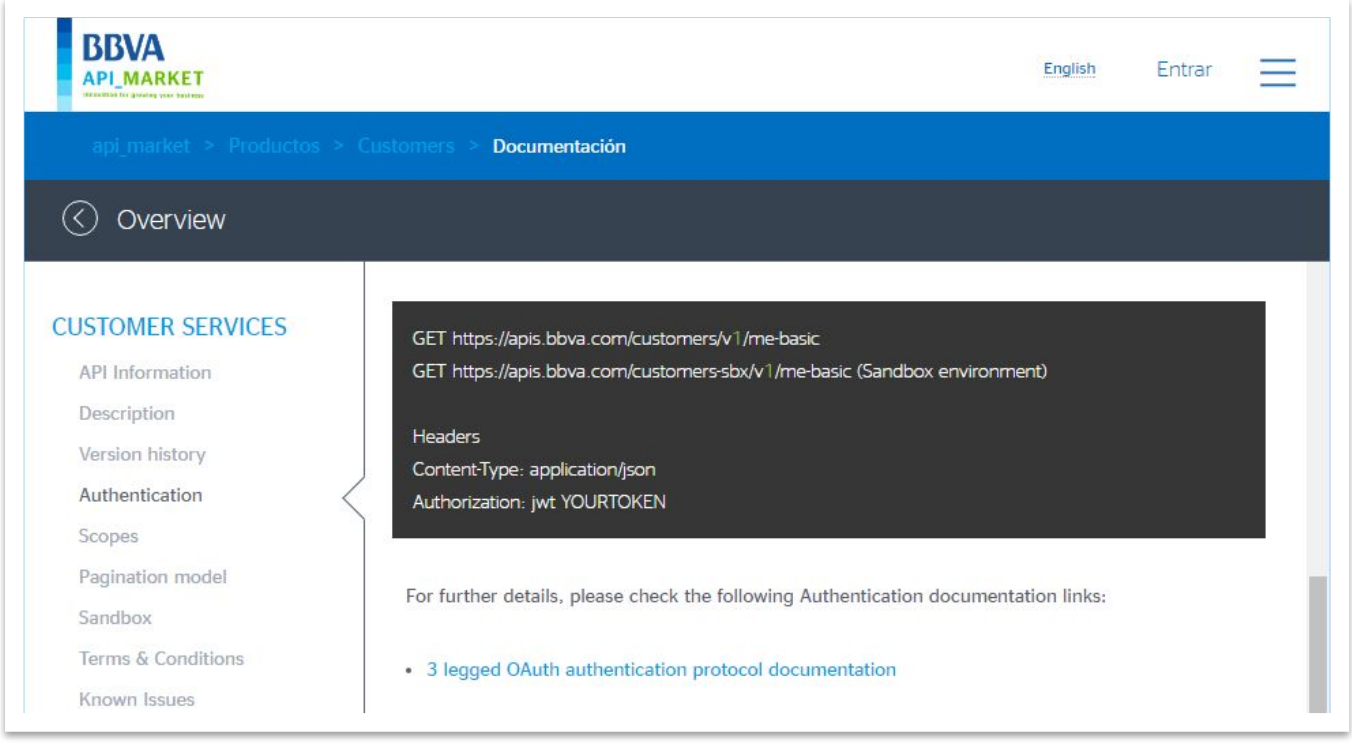
API REST

Características

Propiedad	Descripción
Sin estado	Cada petición contiene toda la información necesaria para su ejecución: • Algunas peticiones se pueden cachear • ¿Cómo mantenemos el estado? • ¿Y la autenticación y/o autorización?
Verbos estándar	POST (crear), GET (leer), PUT (editar) y DELETE (eliminar)
Separación cliente/servidor	Separación entre interfaz y almacenamiento de datos en el servidor: mejora portabilidad, aumenta escalabilidad y permite desarrollo en paralelo de diferentes componentes en diferentes lenguajes

API REST

Ejemplo: el futuro de la banca



API REST

Autenticación y autorización

Autenticación y autorización

- Un protocolo puro sin estado evita accesos a la BD: esencial en aplicaciones de alta carga

- Sin embargo, el estado es necesario en la mayoría de las situaciones
  - Realizar y mantener la autenticación
  - Otros aspectos: roles, datos internos de la app, etc.

Autenticación y autorización

	Método	
HTTP	HTTP Basic	Inseguro. Usuario + contraseñas codificados en B64
	HTTP Digest	Inseguro. Utiliza MD5
Cookies	Session IDs /Cookies	Stateful: el servidor necesita mantener sesiones activas en BD, y el cliente una cookie
Basados en tokens	Tokens “propios”	Probablemente inseguro. NO inventes tu propio esquema, confía en los estándares
	OAuth2	Sin estado. Complejo
	JWT	Sin estado. Sencillo,

Cookies vs Tokens

- Aplicaciones SPA, APIs REST, IoT no pueden utilizar el esquema tradicional de sesiones y cookies:
  - No es posible acceder a la BD en cada petición

IDs de sesión y cookies

1. Usuario introduce sus credenciales
2. El servidor las verifica y crea una sesión en la BD
3. El cliente guarda una cookie en su navegador con el ID de sesión
4. Esta cookie se envía en cada petición posterior y se verifica contra la BD
5. Cuando el usuario hace logout, se destruye la sesión en cliente y servidor

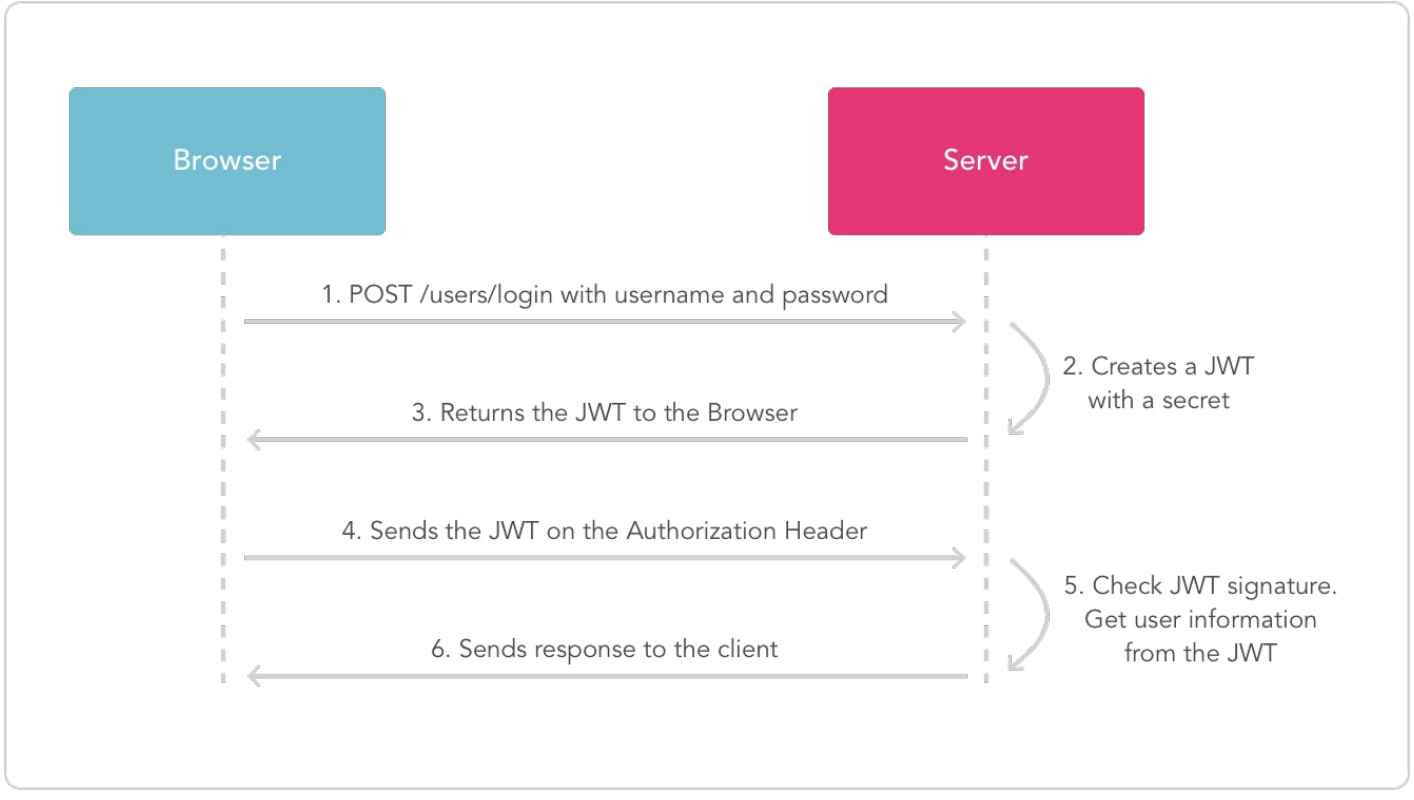
Tokens de autenticación

- Completamente sin estado: el servidor no mantiene registro de sesiones ni tokens emitidos
- Cada token está firmado, lo que permite comprobar su validez sin más información
- Se envía como una cabecera

```
Authorization: Bearer {Token}
```

Tokens

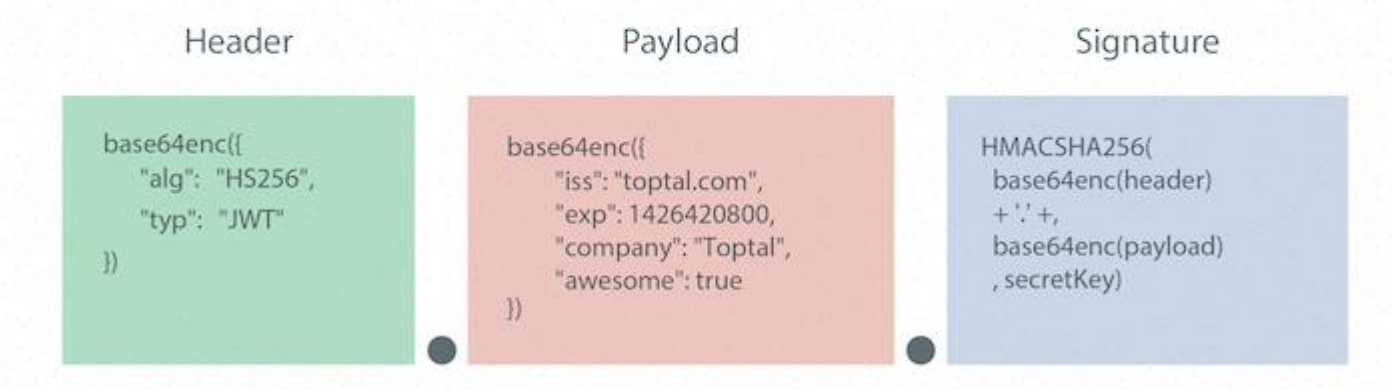
JSON Web Tokens (JWT)



Fuente: https://jwt.io/introduction/

Tokens

JSON Web Tokens (JWT)




```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJpc3MiOiJ0b3B0YWwuY29tIiwiZXhwIjoxNDI2...29tZSI6dHJ1ZX0.
yRQYnWzskCZUxPwaQupWkiUzKELZ49eM7oWxAQK_ZXw
```

JSON Web Tokens (JWT)

EJERCICIO

- Conéctate a www.jwt.io
  - Crea, modifica y verifica distintos tokens
  - ¿Cómo podrías codificar el hecho de que el usuario pertenece a los roles RRHH y ADMIN?
  - Conéctate ahora a www.samtool.io y compara un mismo token. ¿Qué diferencias encuentras?

Ventajas tokens JWT

Sin estado, escalables, desacoplados

- No hay accesos a la BD:
  - Se simplifica mucho el desarrollo de las apps

- Pueden utilizarse servicios de terceras partes:
  - Auth0

CORS: Cross-Origin Resource Sharing

- Por defecto, la política de “mismo origen” limita a los navegadores las peticiones a otros dominios

- JWT, a diferencia del basado en cookies tradicional, no tiene problemas con estas restricciones

Ventajas tokens JWT

Almacenamiento de datos

- Las cookies suelen almacenar el ID de sesión, fecha de expiración y poco más
- Los tokens pueden almacenar cualquier meta-dato en JSON

Ventajas tokens JWT

Rendimiento

- Los tokens son auto-contenidos:
  - Se cambia I/O (acceso a BD, lenta) por CPU (cálculo de un HMAC, rápido)

- Ejemplo: API con llamada a /api/orders

Tradicional	JWT

• Verificar que la sesión es válida
  • Obtener datos del usuario y verificar que es admin
  • Obtener datos finales
	• Verificación token (CPU)
	• Un único acceso a BD para recuperar datos finales

Aplicaciones móviles

- Las aplicaciones móviles nativas y las cookies tienen algunos problemas

- Es mucho más sencillo utilizar tokens en estas plataformas

Tamaño

- Dependiendo del uso, el tamaño de un token puede crecer rápidamente
- El token debe incluirse en CADA petición al servidor

Inconvenientes tokens JWT

¿Dónde almacenar los tokens?

Almacenamiento local del navegador	Cookies
☐ Sandboxed y no puede ser accedido por otros dominios ✔ No vulnerable a CSRF	✔ No tienen la limitación del mismo dominio ☐ Limitadas a 4Kb ☐ Vulnerables a CSRF

Revocación de tokens

- Un inconveniente MUY serio
- No hay forma de invalidar un token individual sin introducir esquemas con estado, como listas negras

Firmados, no cifrados

- Recuerda, ¡los tokens JWT están firmados, y no cifrados!
- No se puede almacenar ningún dato secreto en ellos
- Si es necesario, utilizar JSON Web Encryption (JWE)