Clase de Seguridad

Capa de (?)

DC - FCEyN - UBA

1 de Noviembre de 2017

Menú del día

Referencias

- Computer Networks: A Systems Approach, Fifth Edition, Peterson.
- Introduction to Computer Security, Matt Bishop, Addison-Wesley, 2004.
- Seguridad de la Información (http://dc.uba.ar/materias/seginf).

Definición

Seguridad en algún punto implica ausencia de **riesgo**. Esto puede depender de nociones abstractas y sociales como la **confianza**.

Problema

En las redes, y por sobre todo en la Internet, es muy complejo estar exento de posibles **vulnerabilidades**. Lo ideal es identificar los riesgos más peligrosos respecto a la información que se desea resguardar y tomar las **medidas** pertinentes y factibles según su costo y beneficio.
Es importante identificar los posibles atacantes y los recursos de los que disponen.

Principales aspectos que atañen a la seguridad

- Confidencialidad.
- Integridad.
- Disponibilidad.

Las leyes, coyuntura, individuos y organizaciones definen los diversos grados de **riesgo** que se puede **mitigar o soportar** respecto a estos conceptos bajo ciertos entornos.

Confidencialidad

Es el **ocultamiento o encubrimiento** de información y/o recursos. Los mecanismos de **control de acceso** son los encargados de preservar este rasgo de la información. Si son vulnerados o **comprometidos**, se pierde la propiedad y la confidencialidad es corrompida. A veces es tan o más importante proteger el mero conocimiento de la existencia de la información que el contenido de la misma.

Integridad

Es la prevención para que no sucedan **modificaciones impropias o sin autorización** en la información. Existe tanto la integridad de los **datos** como la integridad de **origen**. Mecanismos de autenticación y control de acceso sirven para atenuar el riesgo de comprometer este rasgo de la información. Suele ser más dificultosa de proveer que la confidencialidad.

Disponibilidad

Concierne la **posibilidad** de **utilizar** el recurso o la información necesaria. Se suelen utilizar mecanismos estadísticos sobre patrones de utilización de datos para asegurar la disponibilidad. Los famosos ataques de **denegación de servicio** apuntan a vulnerar este concepto.

Amenaza

Es una **potencial** violación de algún rasgo concerniente a la seguridad. No necesariamente tiene que realizarse. Las acciones que permiten que se concrete son denominadas **ataques**.

Políticas vs. Mecanismos

- Una política o norma de seguridad describe qué está permitido y qué no.
- Un mecanismo de seguridad es un método, herramienta o procedimiento que fuerza el cumplimiento de la política.

Resumiendo...

Dada una especificación (formal o informal) de una política de seguridad describiendo qué es seguro o permisible y qué no, se deben implementar los mecanismos de seguridad adecuados para **prevenir** los posibles ataques, o en su defecto **detectarlos** y poder recuperarse de ellos. Es importante notar que a veces alcanza con simplemente **identificar** y actuar.

Definición

La **criptografía** consiste en construir y analizar protocólos para prevenir que terceros no deseados tengan acceso a mensajes privados. Esto permite garantizar una o varias de las siguientes propiedades: confidencialidad, integridad, autenticación y no repudio.

Un **sistema criptográfico** es una tupla (E,D,M,K,C) tal que:

- M es el conjunto de los **textos válidos**.
- K el conjunto de **claves**.
- C es el conjunto de los textos cifrados.
- E: M x K $\rightarrow$ C.
- D: C x K $\rightarrow$ M.

Dos operaciones básicas

Sustitución

Consiste en **modificar caracteres** del texto plano bajo un modelo uniforme. El único ataque posible es mediante la estadística, pero es un ataque muy certero. Ejemplos son: Caesar, Vigenère y One-time pad.

Transposición

Consiste simplemente en **reordenar** los caracteres. Basicamente son **anagramas**. Muy simples de vulnerar a partir de medidas **estadisticas** y de **frecuencia** de aparición de caracteres, duplas de caracteres, etc. Ejemplo: Rail fence

Taxonomía

- Criptografía Simétrica o de Clave Secreta
  - Cifrado de Flujo
  - Cifrado en Bloque
- Funciones de Hash
  - MDC - Modification Detection Code (sin clave)
  - MAC - Message Authentication Code (con clave)
- Criptografía Asimétrica o de Clave Pública

Criptografía Simétrica

- Existe una única clave K
- K se utiliza para encriptar texto plano
- K se utiliza para desencriptar texto encriptado (si y solo si este fue encriptado con K)
- La clave es el secreto que comparten ambos extremos de la comunicación
- El hecho de que la misma clave deba ser conocida por ambos extremos de la comunicación es el punto débil de este algoritmo comparado con uno de clave pública

Criptografía Simétrica

Block Cipher

- Orientado a bloques **bits** y no a caracteres.
- Usa tanto transposición como sustitución.
- Trabaja en **bloques** de una longitud fija en bits.

Ejemplos

- 3DES.
- AES es recomendado hoy en día. Claves más largas, es más rápido y resuelve las vulnerabilidades encontradas en DES.

Stream Cipher

- Orientado a flujos de **bits**.
- Utiliza xor
  Emisor. mensaje Xor k = c
  Receptor. c Xor k = mensaje

Criptografía Asimétrica

- **Diferentes** claves para cifrar y descifrar.
- A diferencia de los métodos clásicos no se debe compartir una clave secreta.
- Una de las claves es **pública** y la otra (**secreta**) es privada.

Se deben cumplir tres propiedades

- Debe ser fácil cifrar o descifrar dada la clave adecuada.
- Debe ser inviable computacionalmente derivar la clave privada a partir de la pública.
- Debe ser inviable computacionalmente derivar la clave privada a partir de un texto descifrado.

Criptografía Asimétrica

RSA

- Los conceptos teóricos que sustentan este algoritmo son ciertas propiedades **matemáticas** respecto a los **números primos** (factorizar números grandes es costoso), módulos y exponenciación. Un algoritmo puede encontrar un par de claves, que debido a estas propiedades **no son derivables** una de la otra, pero tienen la propiedad de ser la **inversa**.
- Es un algoritmo lento comparado con algoritmos que encriptan usando criptografía simétrica

Criptografía Asimétrica

Diversos Usos

- No repudio.
- Integridad.
- Confidencialidad.

Funciones de hash

Definicíon $h : A \to B$

- Dado $x \in A, h(x)$ es fácil de computar.
- $\forall y \in B$ es inviable encontrar $x \in A$ tq $h(x) = y$.
- Es inviable encontrar $x, \hat{x} \in A$ tq $x <> \hat{x}$ y $h(x) = h(\hat{x})$.
  Nota. inviable significa inviable computacionalmente

Vulneradas

- HAVAL
- MD5
- SHA-1

Seguras!

- SHA-256
- SHA-3
- BLAKE2

Lifetime de funciones de hash

Lifetimes of popular cryptographic hashes (the rainbow chart)

Function	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
Snefru																												
MD2 (128-bit)[1]																												
MD4																												
MD5															[2]													
RIPEMD															[2]													
HAVAL-128[1]															[2]													
SHA-0																												
SHA-1																												[3]
RIPEMD-160																												
SHA-2 family																		[4]										
SHA-3 (Keccak)																												

Key	Didn't exist/not public	Under peer review	Considered strong	Minor weakness	Weakened	Broken	Collision found

Fuente. `http://valerieaurora.org/hash.html`

Esquemas que utilizan hashes

Message digest

- Consiste en adosarle al mensaje a enviar un valor que luego puede ser recomputado y constatado para verificar la **integridad** (igual que en ip).
- Para proveer seguridad se usan funciones de hash con clave o se cifran los hash sin clave.

Esquemas que utilizan hashes

HMAC (Hash-based Message Authentication Code)

- Consiste en Hashear el mensaje mezclado con la clave.

  $$HMAC(K, m) = H((K + opad)||H((K + ipad)||m))$$

  donde K es la clave simétrica, $+$ representa XOR y $||$ representa concatenación.
- Características
  - Clave simétrica
  - Brinda **integridad y autenticación**.
  - Está demostrado que la seguridad de HMAC depende de la función de hash que se utilice (RFC 2104).

Esquemas que utilizan hashes

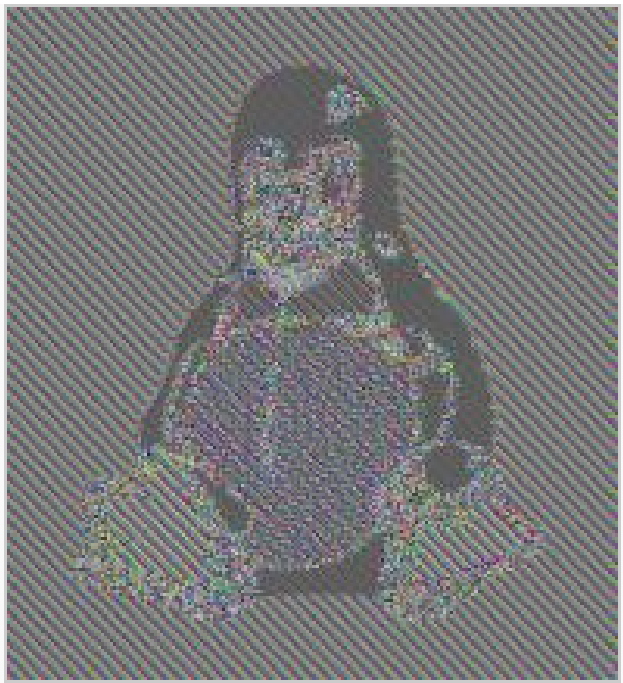
Firma Digital

- Consiste en hashear el mensaje y encriptar el resultado con la clave privada del emisor.
- Necesita los siguientes elementos
  - Algoritmo para generar el par de claves (privada y pública)
  - Algoritmo firmante que dado un mensaje y una clave genere una firma
  - Algoritmo para verificar la firma que dado un mensaje, una clave pública y una firma puede aceptar o rechazar la misma.
- Características
  - Clave asimétrica
  - Brinda **autenticación, no repudio e integridad**.

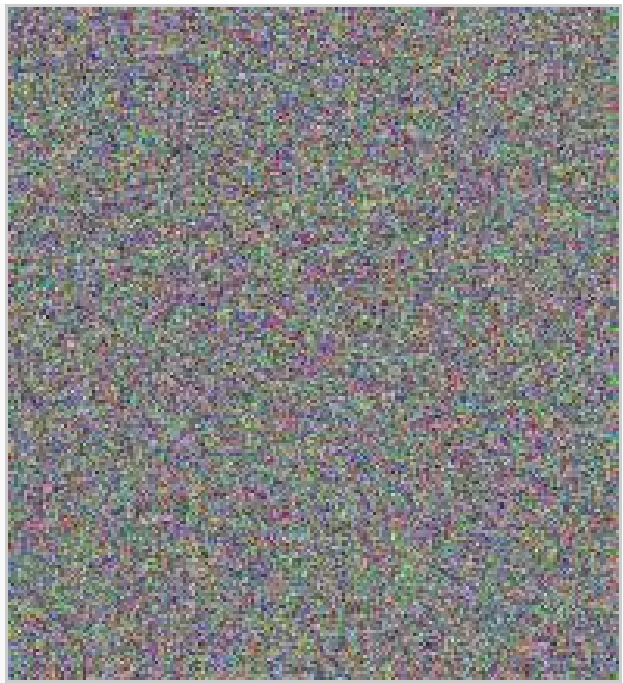
Ataque Estadístico

- Existe una relación matemática entre el texto encriptado y el texto plano.
- Esa relación puede deducirse si se tiene suficiente información, logrando desencriptar el texto o conocer la key usada para encriptar (mejor caso para el atacante)

Original image



Encrypted using ECB mode



Modes other than ECB result in pseudo-randomness

Tipos de Claves

- **Intercambio.** Asociadas a un ente (máquina / persona).
- **Sesión.** Asociadas a una comunicación.
  Se usan claves de sesión nuevas por cada comunicación permitiendo disminuir las posibilidades de ataques estadísticos.

Generar clave de sesión

Confiando en un tercero (Cathy)

- Se lo conoce como Trusted Third Party (TTP)
- Algoritmo
  1. Alice $\rightarrow$ Cathy: $\{$ Request de clave de Sesión con Bob $\}E_{Alice}$
  2. Cathy $\rightarrow$ Alice: $\{$ K_{ses} $\}E_{Alice}$
  3. Cathy $\rightarrow$ Alice: $\{$ K_{ses} $\}E_{Bob}$
  4. Alice $\rightarrow$ Bob: $\{$ K_{ses} $\}E_{Bob}$

     Nota. Las claves usadas son de tipo simétricas. También es posible hacer este esquema si la clave de 1 es privada y las de 2,3 y 4 son públicas
- Kerberos. Es un servicio que provee claves de sesión basado en el algoritmo anterior. En la realidad es bastante más complejo ya que evita el **man-in-the-middle** o otras posibles vulnerabilidades.

Generar clave de sesión

Utilizando criptografía de clave pública

- Asumiendo que
  - Alice tiene la clave pública de Bob
  - Bob tiene la clave pública de Alice
- Protocolo
  - Alice $\rightarrow$ Bob: $\{ \{ K_{ses} \} E^{-}_{Alice} \} E^{+}_{Bob}$
  - Luego Bob accede a la clave usando el siguiente algoritmo

    $\{ \{ K_{ses} \} D^{+}_{Alice} \} D^{-}_{Bob}$

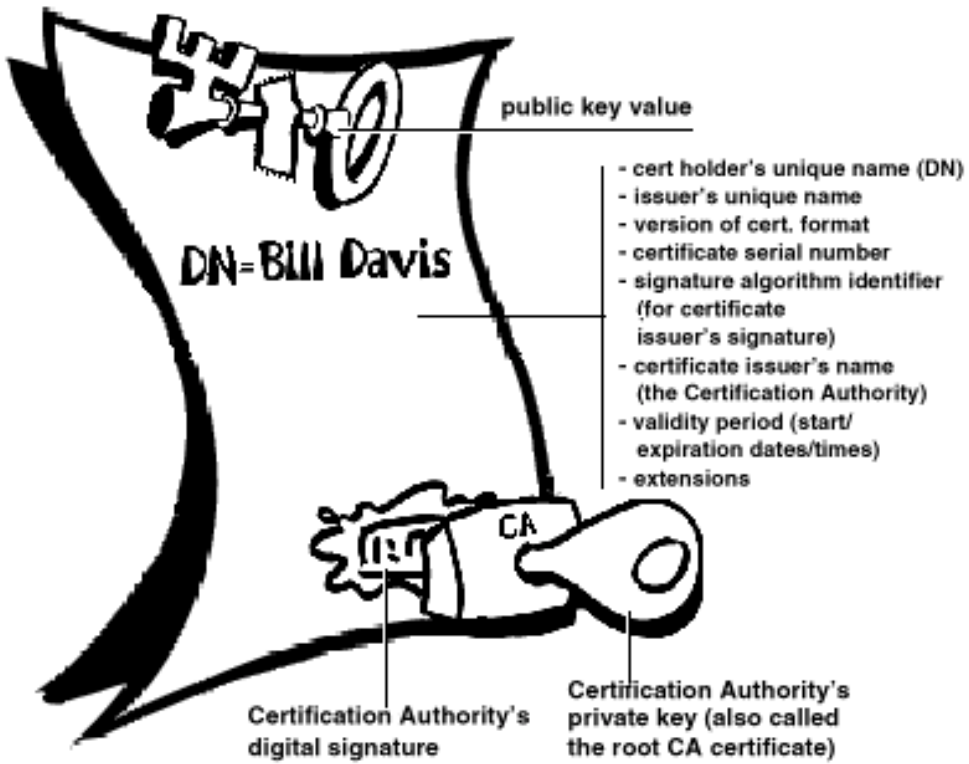
Desafíos

La cuestión más importante es la validación y manejo de las claves públicas y privadas.

Certificados

Aparecen los **certificados** que básicamente son relaciones entre claves públicas y cierta entidad. Para darle validez a estos certificados se confía en **autoridades certificantes** o **cadenas de confianza**.
X.509v3 es el estándar de que define los formatos de certificados (RFC 6818 la última versión).

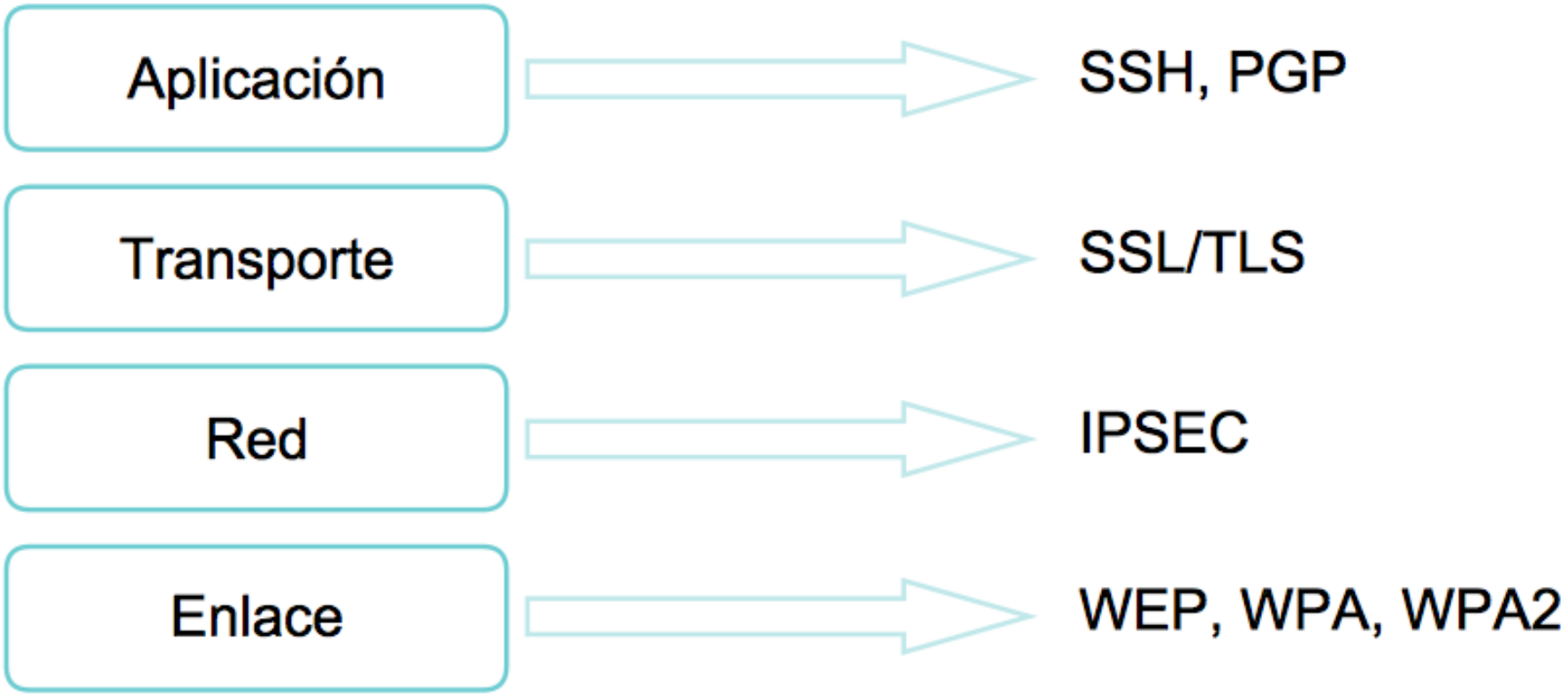
Certificado X.509



Demo Certificados

1. Browser.
2. Binary:
   1. openssl s_client -showcerts -connect host:443 </dev/null 2>/dev/null | openssl x509 -outform PEM > mycertfile.pem
   2. openssl x509 -in mycertfile.pem -text -noout

¿En qué nivel?


Aplicación
SSH, PGP
Transporte
SSL/TLS
Red
IPSEC
Enlace
WEP, WPA, WPA2

TLS

- Versión actualizada de SSL (Secure Sockets Layer).
  - La última versión de SSL (Netscape) fue 3.0
  - La última versión productiva de TLS es la v1.2 mientras que ya existe un draft para para la v1.3.
  - Especificado en RFC 2246 (1999). Muchas actualizaciones: RFC 6176 (2011).
- Protege una sesión entre cliente y servidor. El caso más conocido es HTTP (navegador y web server).
- Requiere protocolo de transporte confiable, por ejemplo TCP.

Servicios

- Autenticación.
  - Del servidor frente el cliente
  - Opcionalmente, del cliente frente al servidor.
  - Mediante **certificados** de clave pública.
- Integridad.
  - Mediante **MAC**.
- Confidencialidad.
  - Opcional.
  - Mediante **cifrado** con algoritmo simétrico.

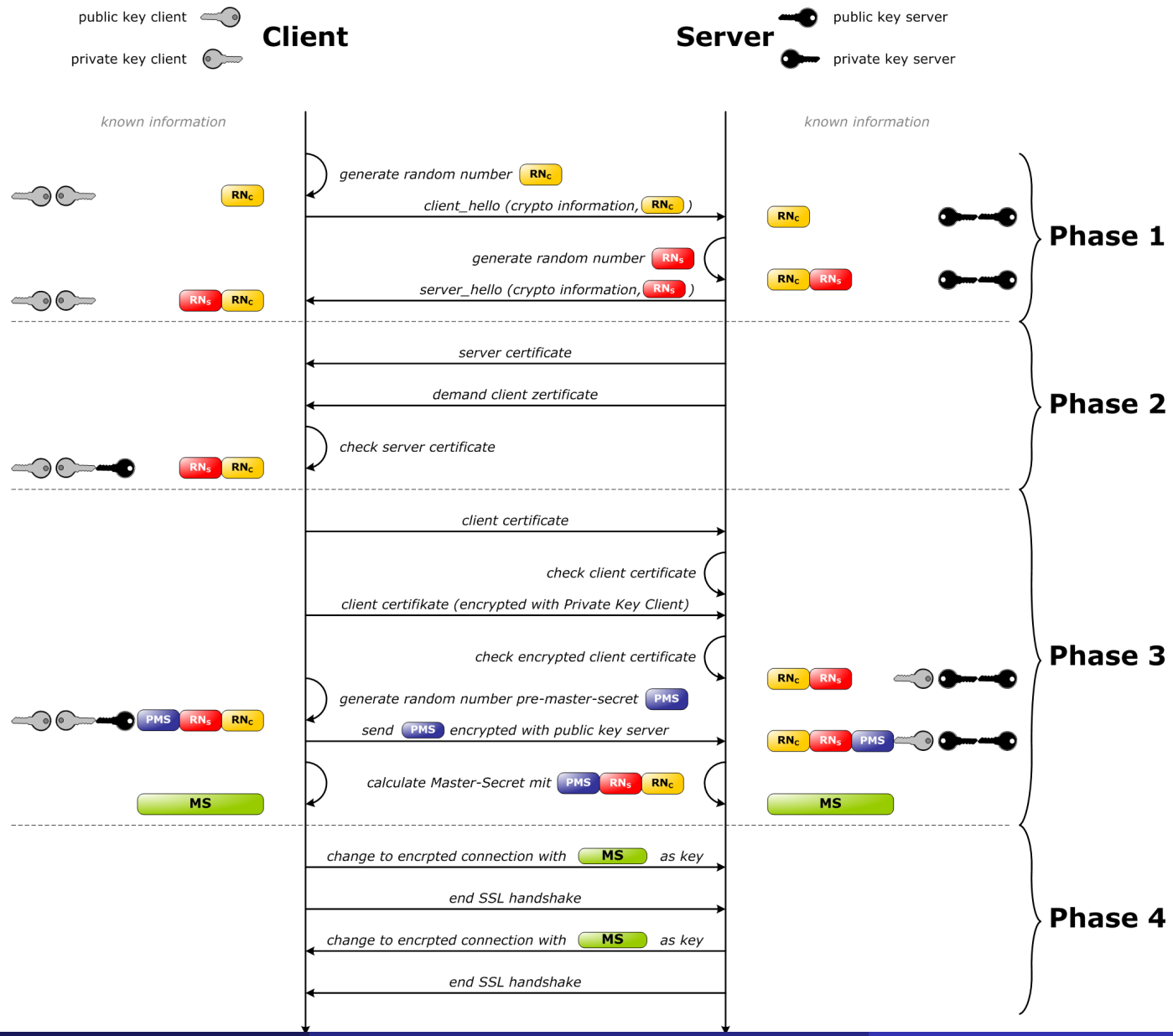
Handshake SSL

- Se **negocian** los **algoritmos** a utilizar durante la conexión.
- Se **autentica** al servidor o los dos entes.
- Se genera un canal seguro para definir una **master key**.
- Se derivan las **claves** necesarias a partir de la master.
- Se constata la **integridad** de todos los mensajes de intercambio de claves.

Ejercicio

- ¿Qué información tienen disponible ambos extremos antes de iniciar el handshake?
- Describa una posible implementación de un handshake para establecer una conexión segura.

A alto nivel


public key client
private key client
Client
Server
public key server
private key server
known information
known information
generate random number RNc
client_hello (crypto information, RNc)
generate random number RNs
server_hello (crypto information, RNs)
Phase 1
server certificate
demand client zertificate
check server certificate
Phase 2
client certificate
check client certificate
client certifikate (encrypted with Private Key Client)
check encrypted client certificate
generate random number pre-master-secret PMS
send PMS encrypted with public key server
calculate Master-Secret mit PMS RNs RNc
MS
MS
Phase 3
change to encrpted connection with MS as key
end SSL handshake
change to encrpted connection with MS as key
end SSL handshake
Phase 4

Pregunta integradora

Accederían a su home banking, asumiendo por supuesto que está implementado sobre la capa TLS (HTTPS), desde el wifi abierto de una plaza pública?

Respuesta integradora

TLS itself is no more vulnerable over a public wifi connection, than over “regular“ internet. It was designed to be used in open channels.
However, there a few other aspects to consider:

- Often users don’t browse directly to the HTTPS site, they start off at the HTTP site and redirect from there. E.g you browse to http://example.org/, and click the Email link, which redirects you to https://mail.example.org/. Since the original HTTP page is not encrypted, that malicious user can modify your traffic, causing the Email link to NOT redirect to HTTPS, but maybe somewhere else. For example, if you clicked the Email link on example.org’s homepage, would you notice that it took you to http://mail.exxxample.org/? (as an example). You might, someone else might not.
- If the user hijacks your connection, but provides his own bogus SSL certificate instead of example.org’s - your browser will show an error, that the cert is not valid. However, most users will just click through this, allowing the attacker to MITM to your secure site, over SSL.
- Don’t assume the public hotspot is securely configured. Pwned routers are all too common, and can cause way more damage irrelevant of your SSL.

¿Repasando...?

¿Dudas?,

¿Preguntas?