VÍDEO intypedia009es

LECCIÓN 9: INTRODUCCIÓN AL PROTOCOLO SSL

AUTOR: Dr. Alfonso Muñoz Muñoz

Dr. Ingeniero de Telecomunicación. Universidad Politécnica de Madrid
R&D Security Researcher. T>SIC Group – UPM

ALICIA

Hola, bienvenidos a intypedia. Hoy vamos a explicar los fundamentos del protocolo criptográfico SSL/TLS. Un protocolo de vital importancia en el comercio electrónico y en las redes privadas virtuales seguras. Un tema muy interesante… ¡Acompáñanos!

1. ORÍGENES DE SSL. ATAQUES A LA IDENTIDAD E INTEGRIDAD DE LOS DATOS

BERNARDO

La empresa Netscape Communication, famosa por la creación del navegador Web Netscape Navigator, creó y desarrolló en la década de los 90 el protocolo estándar SSL (Secure Sockets Layer) un procedimiento para proporcionar comunicaciones seguras en una red.

ALICIA

Este protocolo será de gran utilidad en Internet ya que cuando navegamos se pueden producir múltiples ataques a la información intercambiada (integridad y confidencialidad) así como ataques sobre la identidad de las personas o los servicios (autenticación) a los que nos conectamos, como ya vimos en la lección 4.

BERNARDO

Así es. En estos escenarios es donde es interesante utilizar el protocolo SSL. Por desgracia su implementación no siempre tiene lugar y su mal uso por parte de los usuarios puede facilitar ataques sin necesidad de invertir las medidas de seguridad definidas en el mismo.

En las redes de telecomunicaciones, clásicamente en Internet, estos ataques se suelen materializar mediante ataques de hombre en el medio, Man-in-the-middle MITM, ya que al no habilitar procedimientos de integridad y autenticidad es posible alterar o modificar la información en tránsito, así como suplantar la identidad de los extremos de la comunicación.

Hoy día este tipo de ataques son muy sencillos mediante el uso de herramientas libres disponibles en Internet. Un caso de Man-in-the-middle recientemente famoso por la publicación de la herramienta Firesheep, consiste en robar la cookie de sesión de un usuario (la que le autentica) cuando accede a servicios de una página web como Facebook, Twitter o una cuenta en Google.

ALICIA

Vaya… ¿y cómo es esto posible?

BERNARDO

Esto es posible si el atacante tiene acceso al tráfico intercambiado (por ejemplo, porque está en la misma red wifi) y ese tráfico puede leerse sin problemas. Una vez se dispone de la cookie de sesión que autentica al usuario, es posible suplantarle y acceder a las páginas que da acceso esa validación. El ataque basado en robar una cookie de sesión (HTTP session hijacking attacks) es conocido desde hace mucho tiempo pero la herramienta Firesheep (una extensión para el navegador Firefox) desarrollada por Eric Butler en octubre de 2010 demuestra que estos ataques a día de hoy son triviales. Curiosamente, este incidente hizo que ciertas compañías habilitaran el protocolo SSL para acceder a sus servicios de una forma segura, un caso significativo fue Facebook. Quizás antes no lo consideraron por la relación entre complejidad, coste y ralentización de las comunicaciones.

ALICIA

Muy interesante. ¿Cómo funciona este protocolo?

BERNARDO

Esa curiosidad es buena Alicia… vamos a verlo en detalle.

2. FUNCIONAMIENTO DE SSL. SSL HANDSHAKE PROTOCOL

BERNARDO

SSL (Secure Sockets Layer) es un protocolo criptográfico que proporciona autenticación, integridad y confidencialidad de la información en una comunicación cliente/servidor a través de una red, como lo sería Internet. Se ejecuta en una capa entre los protocolos de aplicación como HTTP, SMTP o NNTP y el protocolo de transporte TCP. Hoy día también es posible implementarlo sobre UDP.

Habitualmente, se configura para que el servidor de la comunicación sea autenticado mientras que el cliente se mantenga sin autenticar (esto es típico por ejemplo cuando nos conectamos a la página web de nuestro banco online en Internet mediante https, donde la s de seguro significa que se está utilizando el protocolo SSL). La autenticación mutua, de cliente y servidor, también es posible mediante este protocolo.

ALICIA

Una duda…. He oído que un protocolo denominado TLS sirve también para estas cuestiones. ¿Es lo mismo SSL y TLS?

BERNARDO

Buena pregunta Alicia porque mucha gente no lo tiene claro. El protocolo SSL ha servido de base para desarrollar el protocolo TLS (Transport Layer Security), actualmente en su versión 1.2 (o también conocido como SSL 3.3) recogido en la norma RFC 5246. Conceptualmente SSL y TLS son parecidos, únicamente TLS mejora el SSL "clásico" en algunos aspectos como puede ser la protección frente a nuevos ataques, proporcionar nuevos algoritmos criptográficos, evitar que se pueda forzar a utilizar versiones del protocolo más vulnerables, etc.

Independientemente de la versión del protocolo, para poder establecer una comunicación con requisitos de integridad, confidencialidad y autenticidad, es necesario acordar previamente unos parámetros de seguridad que permitirán establecer a continuación una comunicación segura. Estas fases son realizadas mediante el conocido como SSL/TLS Handshake Protocol. No obstante, no debe olvidarse también de otro protocolo importante como es el SSL/TLS Record Protocol que especifica la forma de encapsular los datos transmitidos y recibidos, incluso los de negociación.

ALICIA

Veo que has estudiado el tema. ¿Puedes explicarme cómo funciona el SSL Handshake protocol con más detalle?

BERNARDO

Este protocolo se centra en:

Primero, negociar entre cliente y servidor los algoritmos que se utilizarán en la comunicación. Por ejemplo, algunos de los algoritmos criptográficos que se utilizan para cifrar, intercambiar claves y firmar son: 3DES, IDEA, AES, RSA, Diffie-Hellman, DSA, SHA-2, etc.

Segundo, realizar el intercambio de claves y la autenticación basada en certificados digitales, utilizando una validación mediante una infraestructura de clave pública PKI cuando es necesario.

Y tercero, el cifrado del tráfico basado en criptografía simétrica. Se genera una clave de sesión para la comunicación en función de los parámetros negociados. Esta clave facilitará el cifrado de los datos. Como hemos analizado en lecciones anteriores, la criptografía simétrica es más rápida y requiere menos recursos hardware, lo cual es ideal para su ejecución en el cliente,

habitualmente con menos recursos que el servidor. La criptografía asimétrica sólo se utiliza en el intercambio de claves y en el firmado. Por tanto, este protocolo es un buen ejemplo de la utilidad de la cifra híbrida.

Una vez se concluye la negociación de estos parámetros, comienza la conexión segura. Si alguna fase de la negociación falla, entonces la conexión no se establece. A modo de ejemplo, veamos los mensajes más importantes que se intercambian en la fase de negociación hasta establecer la comunicación mediante un "simple TLS handshake" en el que se autentica sólo al servidor mediante su certificado.

1. El cliente envía un mensaje "ClientHello" especificando la versión más alta del protocolo TLS soportada, un número aleatorio, y una lista de algoritmos de autenticación, cifrado y MAC (Message Authentication Code), así como algoritmos de compresión.

2. El servidor responde con un mensaje "ServerHello", indicando la versión del protocolo seleccionado (la más alta que soporten cliente y servidor), un número aleatorio, los algoritmos que selecciona de los enviados por el cliente y su certificado digital (mediante un mensaje Certificate) para autenticarle.

3. El cliente verifica el certificado del servidor, típicamente mediante una autoridad de confianza o PKI. A continuación el cliente responde con un mensaje ***ClientKeyExchange***, el cual contiene una *PreMasterSecret (un número secreto)*, con información para generar la clave de sesión. Si se utiliza el algoritmo RSA este mensaje irá cifrado con la clave pública del servidor y este número aleatorio generado por el cliente será de 48 bytes.

4. El cliente y el servidor usan los números aleatorios intercambiados y la PreMasterSecret (el servidor necesita utilizar su clave privada para recuperarla). Con estos datos calculan un secreto común, denominado "master secret". Todas las subclaves de la conexión serán derivadas de ésta mediante la función pseudoaleatoria establecida.

5. El cliente ahora envía un registro **ChangeCipherSpec** e indica al servidor que a partir de ese momento toda la información intercambiada será autenticada y, si así lo estableció el servidor, cifrada.

6. Finalmente el cliente envía un mensaje **Finished** firmado y cifrado, conteniendo un hash y MAC de los mensajes negociados anteriormente.

7. El servidor intentará descifrar el mensaje Finished enviado por el cliente y verifica el hash y el MAC. Si el descifrado o la verificación falla, la conexión no tiene lugar.

8. Si todo va bien, el servidor envía un ChangeCipherSpec indicando al cliente que a partir de ese momento todo lo que envíe estará firmado y, si fue negociado, también cifrado. El servidor envía su mensaje Finished firmado y cifrado, validándolo el cliente, conteniendo un hash y MAC de los mensajes negociados anteriormente.

9. Finaliza la fase de negociación pudiendo intercambiar mensajes cliente y servidor autenticados y cifrados (si así fue establecido).

ALICIA

¡Qué curioso! Y lo interesante es que todo esto se hace de forma transparente al usuario.

3. APLICACIONES DEL PROTOCOLO SSL. COMERCIO ELECTRÓNICO Y VPNS

BERNARDO

Sin duda el éxito el protocolo SSL/TLS es debido a la expansión del comercio electrónico en Internet. La mayoría de las instituciones financieras defiende su uso: Visa, MasterCard, American Express, etc. Hoy día es difícil pensar en comunicaciones seguras en Internet sin el uso del protocolo SSL, los certificados digitales y las infraestructuras de clave pública. Esto es manifiesto en la securización del tráfico web, https en lugar de http.

SSL también puede ser usado para tunelizar una red completa y crear una red privada virtual (VPN). Esto se puede realizar por ejemplo con la herramienta de software libre OpenVPN, que ofrece conectividad punto-a-punto con validación jerárquica de usuarios y host conectados remotamente, entre otras bondades. También es común su uso para proporcionar autenticación y cifrado en tráfico de voz IP (VoIp), es decir, transmisión de voz en Internet mediante el protocolo de red IP.

ALICIA

Por lo que me has contado entiendo que al estar el protocolo SSL por encima de la capa de transporte y por debajo de la capa de aplicación, sería posible securizar las comunicaciones de cualquier software con este protocolo. Pero me surgen algunas dudas. ¿Si en mi empresa quisiéramos desarrollar una web de comercio electrónico, cómo podría establecer este protocolo en mi servidor? ¿Esto lo puedo hacer yo o se lo tengo que comprar a una empresa? ¿Cómo se generan las claves que se intercambian y el certificado digital? ¿Cómo lo instalo?

BERNARDO

Pues existen varias formas de realizarlo. Si nos centramos exclusivamente en tecnologías gratuitas, el uso de un servidor web Apache, el módulo mod_ssl y el software libre OpenSSL sería suficiente para tus fines, incluso si desearas autenticar a los clientes que se conectan mediante certificados digitales X.509. Una vez instalado y configurado todo el software, tanto el servidor como el navegador web del cliente se encargarán de todo el proceso criptográfico. Esto sería tema suficiente para otra lección específica pero en Internet existe mucha información al respecto por si deseas profundizar.

4. SEGURIDAD DEL PROTOCOLO SSL

ALICIA

Bernardo, ¿es posible anular la seguridad proporcionada por el protocolo SSL/TLS?

BERNARDO

En general podríamos decir que el protocolo SSL/TLS es un protocolo bastante seguro. Lógicamente es matizable en función de la versión del protocolo y la implementación del mismo que estemos hablando. En la actualidad, la versión más moderna del protocolo TLS con las extensiones recomendadas puede considerarse seguro, evitando ataques de inyección de peticiones que fuercen al protocolo a realizar acciones indebidas o la renegociación de los parámetros, vulnerando la seguridad final del canal seguro a establecer.

En la práctica los ataques actuales se centran en engañar al usuario, especialmente si su ordenador está controlado por un troyano, o aprovecharse de malas configuraciones del software. Habitualmente se intenta hacer pensar al usuario que está en una comunicación cifrada cuando no lo está, por ejemplo simulando ese candado que se observa en una página web "segura", hacerle aceptar certificados digitales que no son los válidos para el servidor al que se desea conectar, forzar a utilizar versiones del protocolo o algoritmos criptográficos con alguna debilidad descubierta (downgrade), etc. No te extrañe que durante años muchas personas hayan confiado y sigan hoy confiando en la seguridad de un sitio web simplemente por observar un candado amarillo en su navegador web.

En cualquier caso en la siguiente lección profundizaremos en los ataques más famosos para esquivar la seguridad del protocolo SSL.

ALICIA

Bernardo, no me gustaría esperar hasta la siguiente lección… ¿Qué puedo hacer de momento para conectarme a mi banco de una manera más segura?

BERNARDO

No es nada complejo. Como sabes, el uso de SSL es vital para el acceso seguro a servicios de la administración electrónica o a la banca online. Unas recomendaciones iniciales consistirían en escribir directamente la dirección URL (de tu banco) con el prefijo https en la barra de direcciones del navegador, y si el navegador web indica que el certificado digital de la entidad a la que deseamos conectarnos no es reconocido o válido, no aceptarlo bajo ningún concepto, y así la conexión no tendrá lugar. En el caso más extremo, si quieres con algo de paranoia, y como medida adicional, se podría verificar la firma del certificado digital del servidor al que nos conectamos, si la tenemos anotada previamente, mirando en el navegador Web. Si lo deseas existen herramientas para forzar siempre la conexión por https, por ejemplo la extensión del navegador Firefox HTTPS Everywhere. Además, deberías configurar tu navegador adecuadamente para que el protocolo OCSP (Online Certificate Status Protocol) denegara certificados digitales revocados. Por otro lado, si estás interesada en analizar la seguridad de la tecnología SSL empleada en un servidor dado puedes estudiar, para empezar, documentación gratuita publicada por la organización OWASP (The Open Web Application Security Project), por ejemplo el tutorial Testing for SSL-TLS (OWASP-CM-001).

En cualquier caso, en todo esto profundizaremos en la siguiente lección.

ALICIA

Esperaré un poco entonces. Creo que por hoy es suficiente, en la próxima lección seguiremos con este tema tan interesante. En intypedia tienes información adicional sobre esta lección. ¡Adiós!

BERNARDO

¡Hasta luego!

Guión adaptado al formato intypedia a partir del documento entregado por el Dr. Alfonso Muñoz de la Universidad Politécnica de Madrid, España.

Madrid, España, Julio de 2011

http://www.intypedia.com

http://twitter.com/intypedia