REDES DE INGENIERÍA
http://revistas.udistrital.edu.co/ojs/index.php/REDES/index

Análisis de herramientas de interceptación para el control de ataques reales de suplantación con certificados SSL

Analysis of interception tools against real spoofing attacks using SSL certificates

Tipo de artículo: Artículo Reporte de Caso

Fecha de recepción:

Fecha de aprobación:

Resumen

En este artículo se analiza y evalúa la respuesta de dos herramientas de seguridad para la interceptación y análisis de tráfico SSL (Sophos UTM 9 y Checkpoint Firewall) ante distintos escenarios reales de suplantación de certificados SSL. Se encontró una respuesta similar en efectividad y rendimiento en las dos herramientas, aunque ninguna de las dos fue suficiente para controlar algunas de las amenazas actuales a pesar de estar clasificadas entre las soluciones MITM más robustas del mercado.

Palabras clave

HTTPS, Interceptación, MITM, SSL,, TLS, Suplantación.

Abstract

In this article we analyze and evaluate the response of two different security tools for SSL traffic interception and (Sophos UTM 9 and Checkpoint Firewall) when facing real SSL visual spoofing attacks. A similar response in effectiveness and performance was found for both tools, although neither of them was sufficient to control some of the current threats , in spite of being classified as some of the strongest tools in the market.

Key words:

HTTPS, Interception, MITM, SSL,, TLS, spoofing.

Introducción

El uso de certificados digitales SSL se ha extendido como indicador de una comunicación segura entre servidores y clientes web. Sin embargo, a pesar de implementar el protocolos SSL/ TLS, con

el cual se provee autenticación, privacidad e integridad en el intercambio de información a través de la web, la seguridad en el tráfico HTTPS puede ser vulnerada.

La criptografía propia de los protocolos SSL/TLS no es un fácilmente vulnerada, por lo cual las técnicas de Criptoanálisis no son usadas comúnmente para ejecutar ataques. Por otro lado, son comunes ataques de interceptación en el canal y la suplantación de certificados y autoridades certificadoras.

Los ataques de suplantación por agentes externos a las páginas o servicios web se han convertido en un desafío para las entidades certificadoras; la autenticación debe ser verificada en los distintos niveles de la cadena de certificación con el fin de evitar la suplantación de identidad o espionaje de datos. Las Entidades de Certificación Digital utilizan lo que se conoce como la Infraestructura de Clave Pública (PKI, por sus siglas en inglés), que es el conjunto de elementos tecnológicos que, mediante la utilización de un par de claves criptográficas, una privada que solo posee el suscriptor del servicio y una pública que se incluye en el certificado digital, logran:

- Identificar a quien se envía una comunicación.
- Impedir que terceras personas puedan observar los mensajes que se envían a través de medios electrónicos.
- Impedir que un tercero pueda alterar la información que es enviada a través de medios electrónicos.
- Evitar que el suscriptor del servicio de certificación digital que envió un mensaje electrónico pueda después negar dicho envío.

Los ataques de suplantación son generalmente perpetrados junto con ataques de interceptación como el Man in the middle(MITM), los cuales permiten a un atacante re-enrutar la comunicación entre los dos extremos cliente-servidor sin que ninguna de las partes que establecen la comunicación tenga conocimiento. Cada extremo envía el tráfico al atacante y lo recibe del mismo, sin darse cuenta, asumiendo que se está comunicando con el usuario de destino de una manera segura. Estas interceptaciones son logradas generalmente por vulnerabilidades en el lado del cliente web, bien sea por errores en la configuración o en la operación de la interfaz de usuario o como consecuencia de una infección de malware

Para mitigar estos ataques desde el lado del cliente, se deben establecer políticas de seguridad como la actualización regular de software, implementación de infraestructura de seguridad perimetral y el análisis periódico de posibles fallas de seguridad. Esto es especialmente crítico dentro del sector financiero, en el cual información altamente sensible y confidencial es transmitida a través de la web, y para cuyos sitios y servicios web los protocolos SSL/TLS son implementados prácticamente por regla general. En el mercado se ofrecen distintas soluciones de software que ofrecen monitorear y alertar posibles ataques de suplantación de certificados SSL. En este artículo se busca evaluar el alcance y desempeño de distintas herramientas ante escenarios comunes y reales de suplantación de certificados SSL.

Para lograr dicho objetivo, dentro la segunda parte de este artículo se revisará generalidades del protocolo SSL y los certificados digitales. En la tercera parte se procederá a estudiar y reproducir los principales vectores de ataques a través de los cuales es vulnerada la seguridad del protocolo

SSL. En la cuarta parte se revisarán algunas de las contramedidas y herramientas propuestas por distintas partes como respuesta a las amenazas previamente descritas. En la quinta parte se plantean los distintos escenarios de prueba y en la sexta se evalúa la respuesta de las herramientas seleccionadas ante diferentes escenarios de ataque: Suplantación visual usando certificados inválidos, redirección HTTP-HTTPS y suplantación visuales con dominios similares y certificado válidos Finalmente se comparan los resultados obtenidos.

Protocolo SSL/TLS

El principal objetivo del protocolo SSL es proveer privacidad y confiabilidad entre dos aplicaciones que se comunican. El protocolo está compuesto por dos capas: El protocolo de registro SSL y el protocolo de Handshake SSL. El protocolo de registro SSL se encarga de encapsular otros protocolos a más alto nivel. Por otro lado, el protocolo SSL handshake permite tanto al cliente como al servidor autenticarse e intercambiar un algoritmo de encripción y llaves criptográficas antes de que el protocolo de aplicación reciba o transmita cualquier bit de datos [1]. Las principales etapas de un protocolo de handshake típico son descritas a continuación [2]: Primero, el cliente envía un mensaje Client Hello solicitando al servidor el inicio de la negociación SSL. En este mensaje se especifican la lista de cifrados soportados y un número aleatorio generado por el cliente. En segundo lugar, el servidor responde con un mensaje ServerHello, con el cual el servidor selecciona el cifrado que será usado durante la comunicación y un número aleatorio generado por el servidor. Tercero, el servidor envía un mensaje Certificate con el propósito de autenticar su identidad y el cual contiene la llave pública del servidor y el nombre del host, los cuales son firmados digitalmente por una autoridad certificadora y cuya verificación debe ser realizada por el cliente. Luego de esto, el servidor envía un mensaje ServerHelloDone para confirmar al cliente que ha finalizado su parte de la negociación y procede a esperar la respuesta del cliente. A continuación, el cliente envía un mensaje ClientKeyExchange que contiene las llaves simétricas que ambas partes usarán. Finalmente, el cliente y el servidor intercambian un mensaje ChangeCipherSec para notificar mutuamente que los datos de la aplicación en uso dentro de la sesión iniciada, serán encriptados usando la llave de sesión derivada.

SSL es independiente del protocolo de aplicación, y por tanto cualquier protocolo de más alto nivel puede ser usado sobre el protocolo SSL transparentemente.

- Certificados Digitales y estándar X.509

La validación de la identidad de las distintas partes cliente-servidor realizada por el protocolo SSL/TLS está basada en el estándar X.509, el cual especifica el formato de los certificados digitales de identidad, así como las consideraciones necesarias para validar la integridad y aplicabilidad de estos de acuerdo a su propósito [3]. En la práctica, los certificados comerciales validan solo la identidad del servidor, y son a menudo firmados por una autoridad certificadora intermedia (autoridad delegada), en vez de ser firmados por una autoridad certificadora raíz segura, las cuales generalmente no cuentan con acceso público a través Internet para reducir riegos de ataques. De esta manera, el servidor envía una cadena de certificados al cliente, compuesta por

el certificado propio del servidor y los certificados de las autoridades certificadoras intermedias. Cada certificado es criptográficamente firmado por la entidad del siguiente certificado en la cadena. Un certificado válido debe llevar a través de la correspondiente cadena a una CA segura y de confianza para el cliente.

El proceso de la validación realizada por X.509 está definido en el RFC5280[4] y consiste en los siguientes puntos principales:

- Validar la firma digital de un certificado.
- Seguir la cadena de certificado para determinar las autoridades certificadoras (CA) relevantes.
- Validar los certificados intermedios encontrados.
- Determinar si la autoridad certificadora raíz es segura/de confianza.
- Determinar el asunto para el cual aplica el certificado, para comparar con las expectativas de los protocolos de más alto nivel.
- Determinar si el certificado ha sido revocado.

Vectores de Ataque

A partir de los ataques cibernéticos identificados más comúnmente del lado del cliente [5] y con los cuales se han logrado vulnerar la seguridad de canales que usan el protocolo SSL/TLS, se han encontrado [6] las siguientes categorías principales de ataques:

- Fallos criptográficos y problemas de implementación:

El protocolo SSL puede hacer uso de distintos algoritmos criptográficos para proveer servicios de seguridad. Esto hace que el protocolo se tan seguro como el algoritmo criptográfico que utiliza (cifrado y hash). Otro posible fallo de seguridad se encuentra en la implementación de estos algoritmos. Por ejemplo, si el tamaño de la llave no es suficiente, o si el sistema RNG (Random Number Generator) no funciona apropiadamente, el protocolo SSL no será seguro[7]. Mientras algunos autores consideran a SSL/TLS lo suficientemente seguro criptográficamente [5], otros han expuesto y estudiado distintas debilidades en el protocolo. En el trabajo Sirohi [8] se listan en orden cronólogico (1992-2016) aproximadamente veintidós ataques que han vulnerado el protocolo SSL, explotando el protocolo de handshake, el protocolo de registro, el protocolo de datos de aplicación y la infraestructura PKI, entre otros. Entre los ataques más recientes se destacan: Ataque Triple handshake, ataque FREAK y ataque Logjam.

Sin embargo, a pesar de las distintas vulnerabilidades detectadas a lo largo de la historia del protocolo, de acuerdo a diferentes estudios de ataques ejecutados en la práctica, especialmente en sitios financieros y de comercio electrónico [7], no se ha encontrado que este sea un vector de ataque muy común, aunque estas vulnerabilidades y ataques representan un alto riesgo.

- Vulnerabilidades de las interfaces de usuario

Los usuarios de Internet, así como los dispositivos que usan para conectarse a la red, están naturalmente expuestos a vulnerabilidades de manera constante. Si es posible ganar control sobre el dispositivo del usuario o modificar su configuración, bien sea a través de la instalación de malware u otras técnicas, la seguridad del usuario estará sin duda comprometida. Adicional a esto, existen otras técnicas enfocadas en atacar al usuario, aprovechando la manera en que este interactúa con la interfaz de usuario o la manera en que dicha interfaz se configura. La interfaz de usuario para los casos objetivo de este trabajo será en general el navegador web. A continuación se listan algunas de los vectores de ataque con los que más comúnmente se explotan las interfaces de usuario:

MITM: En este tipo te ataques se intercepta la comunicación entre pares. El atacante actúa como proxy entre el cliente y el servidor, manteniendo una sesión SSL/TLS diferente con cada uno y enviando mensajes entre ellos, sin que ninguna de las partes lo note. De esta manera el atacante puede registrar todos los mensajes transmitidos por el canal e incluso puedo modificarlos. Actualmente estos ataques pueden ser fácilmente ejecutados, a través de distintas técnicas cómo: (a) usar un certificado inválido y hacer que el cliente/usuario lo acepte en su navegador, (b) usar un certificado válido en un dominio registrado por el atacante y el cual pueda ser confundido por el cliente con un dominio legítimo y (c) no usar certificado y ejecutar el ataque en tiempo real.

La figura 1 ilustra un ataque MITM – SSL, usando un certificado fraudulento entre el navegador y el servidor HTTPS.

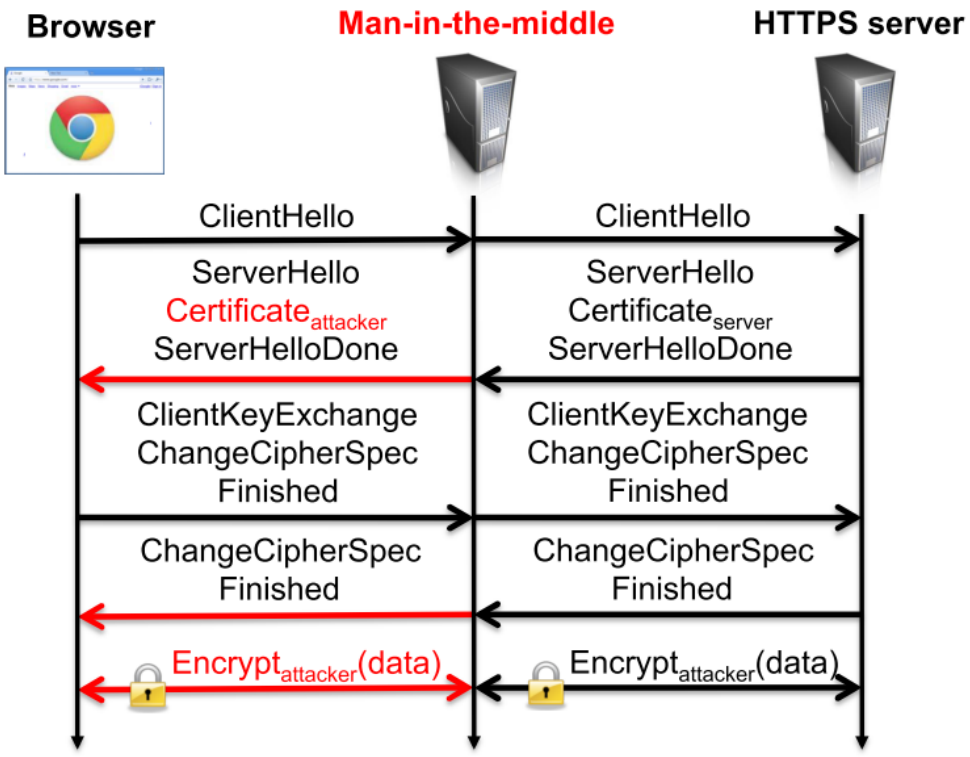


Fig. 1. Ataque MITM SSL entre navegador y servidor, usando un certificado SSL falso para impersonar el servidor al conectarse con el cliente. [2]

Existe otra modalidad de este ataque en la cual las conexiones son establecidas de manera forzosa en protocolo HTTP, evitando que la información sea cifrada. En esta modalidad el atacante se interpone entre el servidor y cliente a través de distintas técnicas (modificación en el proxy del navegador, suplantación ARP o acceso abierto a Red Wi-Fi). En cuanto el atacante puede interceptar el tráfico entre ambos extremos, busca identificar cuando una conexión va a ser redireccionada de HTTP a HTTPS. Al detectar este comportamiento, el atacante envía falso tráfico a la víctima, evitando que la redirección sea completada y posteriormente establece una conexión HTTPS con el servidor de destino. De esta manera, la victima establece una conexión legítima con el sitio deseado y recibe el contenido esperado, así como el atacante también lo ha recibido. Sin embargo, la victima ha enviado su información a través del protocolo HTTP, sin ningún cifrado o certificado, con lu cual el atacante puede monitorear y visualizar la información de la víctima en texto plano. Algunos navegadores web no notan ningún comportamiento irregular, al tratarse de una conexión HTTP legítima y por lo tanto no alertan al cliente [9].

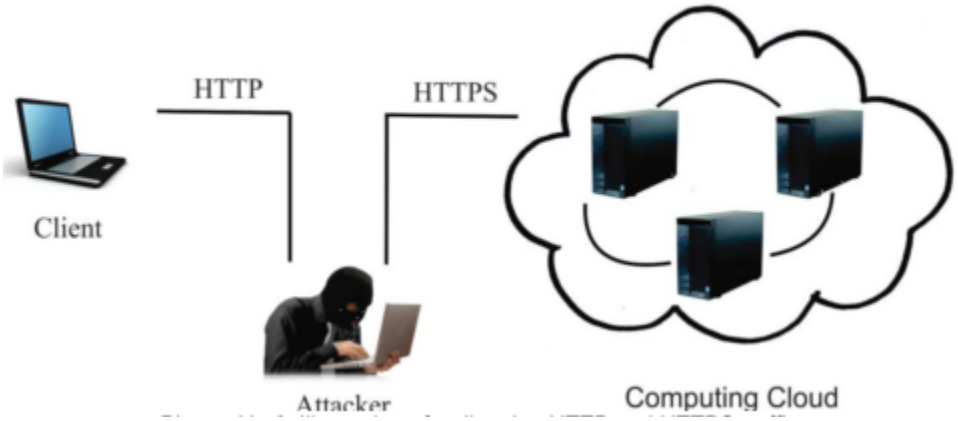


Fig. 2 Redirección de tráfico HTTP a tráfico HTTPS [9]

Validación incorrecta de certificados

- Certificados Autofirmados: Son certificados similares a los firmados por una autoridad raíz no fiable, pero en esta ocasión el certificado es su propia autoridad raíz. Cualquiera puede crear su propio certificado autofirmado y no hay verificación externa de esto, por lo cual estos certificados no deben aceptarse como válidos, por lo cual los navegadores web normalmente emiten un mensaje de advertencia solicitando la validación del usuario para proceder.

- Certificados Expirados: La validación de certificados realizada a través de X.509 verifica las fechas de expedición y expiración de los certificados, de manera que la fecha de revisión debe encontrarse dentro del rango de dichas fechas.

- Aceptación de CA desconocidas:

- Revisión de revocación de certificados: En cuanto se valida que la seguridad del certificado no es segura, bien sea porque la llave privada o la autoridad certificadora hayan sido comprometidas, o porque el dueño del certificado ya califica para este, los certificados deben ser revocados y reportados a lista de revocación de certificados (CRLs) y al protocolo de estatus de certificados en línea (OCSP).

- Basic Constraints: Existe una falla en la validación de certificados expuesta por Moxie Marlinspike en el 2002[10], en la cual se demuestra cómo se puede validar como seguras a CA intermedias que no son seguras, de acuerdo al valor del parámetro "Basic Constraints" propio del certificado. Si este campo se encuentra en valor "false", entonces se indica que la CA intermedia no tiene un certificado válido. Sin embargo es posible que algunos navegadores y proxies de interceptación no validen apropiadamente dicha propiedad, permitiendo que cualquier atacante con un certificado válido puede actuar como una CA intermedia y firmar certificados.

Suplantación de la cadena de certificación.

La seguridad de los certificados SSL depende en gran medida de la confiabilidad de las autoridades certificadoras intermedias que participan de la cadena de certificación y de sus certificados. Una cadena válida debe llevar a que la identidad del servidor sea autenticada por una autoridad certificadora raíz de confianza para el cliente. En años recientes se han identificado ataques en los cuales se han suplantado tanto las autoridades certificadoras intermedias como los certificados de estas (Ataque a CA Comodo-Diginotar[10], CA TurkTrust[11], CA MCS Holdings[12]), permitiendo que un certificado falso verificado por la cadena de certificación sea catalogado como válido y pueda ser usado para suplantar dominios.

Contramedidas

- Nuevos protocolos: Fuzzy Secure Sockets Layer (FSSL)[11]
- Validación de certificados en el navegador

Se han propuesto e implementado distintas contramedidas desde los navegadores web que permiten bien sea bloquear o advertir el acceso a sitios web con certificados no válidos o sospechosos:

HTTP Strict Transport Security (HSTS) [34]: Se trata de una especificación que permite hacer siempre obligatorio el uso del protocolo https aunque el usuario no lo escriba en la barra del navegador. La idea fundamental es que el servidor web mediante cabeceras http obligue al navegador web a conectarse directamente por https, evitando así ataques de redirección HTTP-HTTPS.

The Public Key Pinning Extension for HTTP (HPKP): Esta propuesta permite a los sitios especificar sus propias llaves públicas con un header HTTP y hacer que los navegadores rechacen cualquier certificado con llaves públicas desconocidas. HPKP provee protección en contra de

ataques MITM SSL que usan certificado no autorizados, aunque estos sean válidos o de confianza. HSTS y HPKP requieren que los usuarios inicialmente accedan a los sitios legítimos de forma segura, antes de navegar en redes inseguras. Chrome e Internet explorer usan una lista pre-cargada de llaves públicas, aunque no rechazan certificados de firmantes localmente confiados (antivirus, vigilancia corporativa y malware) [2]

TLS Origin-Bound Certificate (TLS-OBC): Esta propuesta retoma la autenticación TLS del cliente, permitiendo que los navegadores generen certificados autofirmados para el cliente bajo demanda, sin necesitar ninguna configuración de usuario [12]. Esto bloquea gran parte de las técnicas de ataque MITM, ya que no es fácil impersonar al cliente, a menos que la llave privada autofirmada sea robada del navegador. Sin embargo esta propuesta requiere de cambios de código en la pila de red del servidor y conlleva un mayor costo computacional.

Validación de certificados con notarías: Se delega la confianza de si un certificado es válido en la decisión de distintos usuarios en red, comparando el certificado de servidor desde distintos puntos de red (notarías) para encontrar inconsistencias. Ya que las validaciones son realizadas desde distintas y diversas redes, un ataque de impersonación local podría ser fácilmente detectado. Hay ya distintas implementaciones relacionadas que en su mayoría funcionan como add-ons de distintos navegadores web (Perspectives[13], Convergence[14], DetectTor, Crossbear)

Certificate pinning: Con esta técnica se busca detectar cuando es modificada una cadena de confianza, asociando un certificado digital a un dominio concreto. Así, un dominio A como ww.example.com, se vincula con certificado o autoridad de certificación B específica. Si se detecta que una autoridad de certificación B' diferente (que depende de una autoridad de certificación raíz de la que se confía) intenta generar un certificado para al dominio A, se generará una alerta. Actualmente no ha sido posible implementar este concepto [6], sin embargo se han realizado trabajos con avances importantes hacia esta dirección.

Proxies de interceptación SSL/TLS: Una estrategia usada actualmente por distintas organizaciones es desplegar controles de seguridad que intercepten los canales extremos a extremo. El despliegue de proxies en medio de conexiones cifradas ha sido un tema polémico. Sin embargo, en una encuesta realizada a 1,261 usuarios en relación a los proxies SSL/TLS proxy se encontró que a gran parte de los usuarios estaban conforme con el uso de proxies MITM mientras fueran usado para efectos benévolos [15]. A pesar de esto, muchos también manifestaron su preocupación por la posible intrusión en su privacidad, por lo cuales opciones más transparentes y orientadas al usuario deberían ser empleadas [16].

- Monitoreo: Capturar todos los atributos del handshake SSL/TLS y del certificado de servidor para ser analizados y determinar si infringe un conjunto de políticas de seguridad previamente definidas. Se han utilizado técnicas a través del uso de plugins en los navegadores web [2] [17], aplicaciones flash embebidas con campañas de Google Adwords.

- Políticas a revisar [18]: Ubicación Geográfica, expiración del certificado, certificados autofirmado, tamaño de la cadena de certificación, suites de cifrado exportables, algoritmos de encripción débiles, sujeto de certificado con nombre de dominio inconsistente.

- Desventajas: (1) Aumento de la probabilidad de amenazas. El tráfico encriptado tiene un alto interés para los atacantes, y por lo tanto contar con un punto único donde todas las sesiones cifradas pueden ser vistas en texto plano, hacen del proxy de interceptación un objetivo de ataque de alto valor. (2) Confianza transitiva. Para el análisis de tráfico SSL los proxies de interceptación funcionan como una autoridad certificadora raíz para los usuarios finales. De esta manera, no es el usuario quien directamente valida la autenticidad del servidor sino el proxy por sí mismo. Si el servido es considerado seguro por el proxy, lo será entonces para el cliente, pero no es el cliente quien toma esta decisión. Esto convierte al proxy de interceptación en un blanco de ataque importante.

Escenarios de Prueba

Especificaciones del entorno de prueba: Sistema Operativo: Windows 10 Pro Versión_1703, Procesador Intel® Core™ i7-4500U CPU @1,80GHz, RAM: 8,00 GB. Herramientas de Seguridad MITM: Check Point Security Gateway(Servidor Virtualizado), Sophos UTM9 (Servidor Virtualizado). Herramientas de ataque: Kali Linux, ARP Spoofing, Sslstrip , Cpanel, Servidor de Hosting.

- **Escenario 1: Certificados no válidos**

Para este escenario se hacen las pruebas correspondiente dentro de la topología de red indicada en la figura 3, donde el atacante fuera de la red interna de la victima publica un sitio web con certificados inseguros desde cualquier lugar de la red pública de Internet .

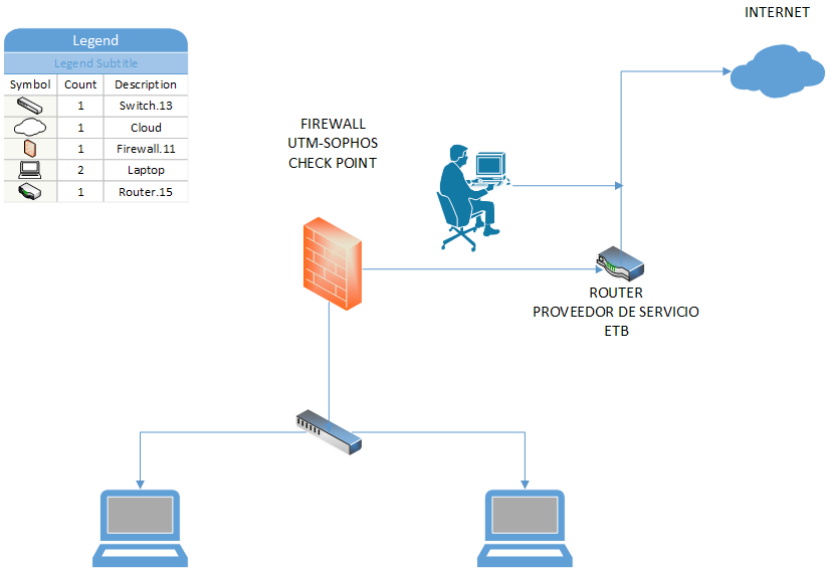


Fig. 3. Topología Escenario de Prueba 1. [Fuente: Autor]

Se prueba el acceso a distintos subdominios provistos por el equipo de Chromium en su página de pruebas (badssl.com), cada uno de los cuales tiene un certificado inválido debido a distintos parámetros y razones (Ver figura 4). Se observan los resultados para evaluar la respuesta de las dos herramientas.

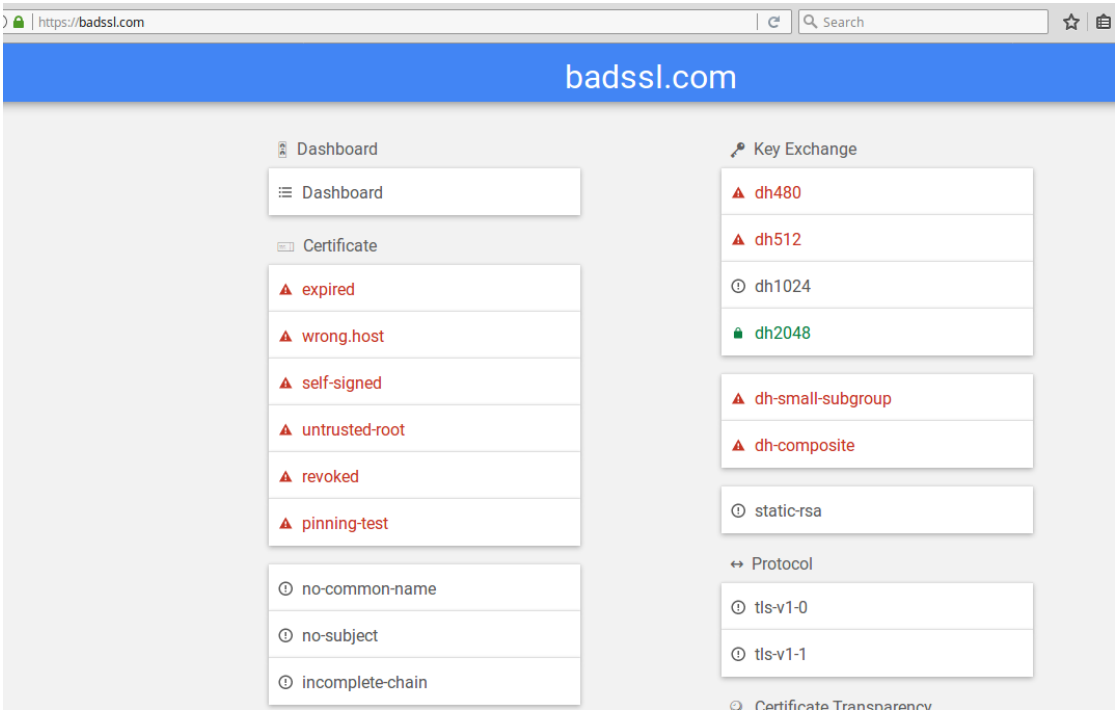


Fig.4.Página de pruebas para certificados SSL inválidos desarrollada por el equipo de Chromium[19] .

Escenario 2: Redirección HTTP –HTTPS

En este escenario se realiza un ataque dentro de la topología de red mostrada en la figura 5. Allí se plantea que el atacante ha perpetrado la red interna de la víctima y realiza un ataque ARP spoofing pasivo para suplantar la puerta de enlace predeterminada de la red y "escuchar" el tráfico. Luego de esto se ejecuta la redirección e interceptación de tráfico desde una máquina atacante con sistema operativo Kali Linux, haciendo uso de la herramienta SSLstrip. SSLstrip es una herramienta que de manera transparente intercepta el tráfico HTTP en un red, revisa los enlaces y redirecciones HTTPS , y mapea estos enlaces hacia páginas visualmente similares con protocolo HTTP, permitiendo que se capture la información enviada por el cliente en texto plano (Ver figura 6) . Adicionalmente, esta herramienta también permite hacer visualizar íconos en la barra de direcciones que se asemejan a los íconos de candado generados por los navegadores para sitios conexiones con protocolo HTTPS.

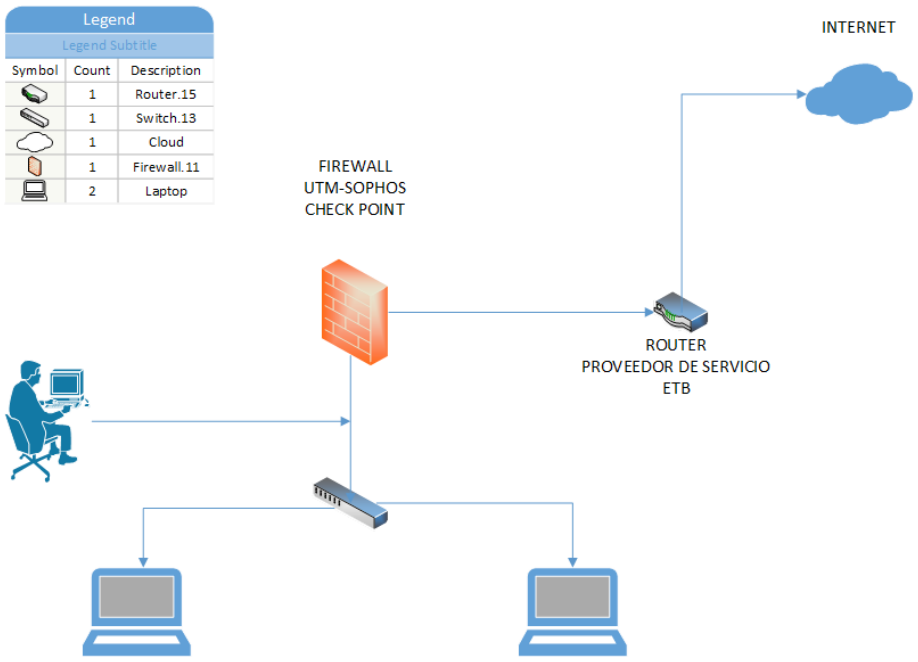


Fig.5 .Escenario 2. Ataque de redirección HTTP a HTTPS con sslstrip y ARP spoofing. [Fuente: Autor]

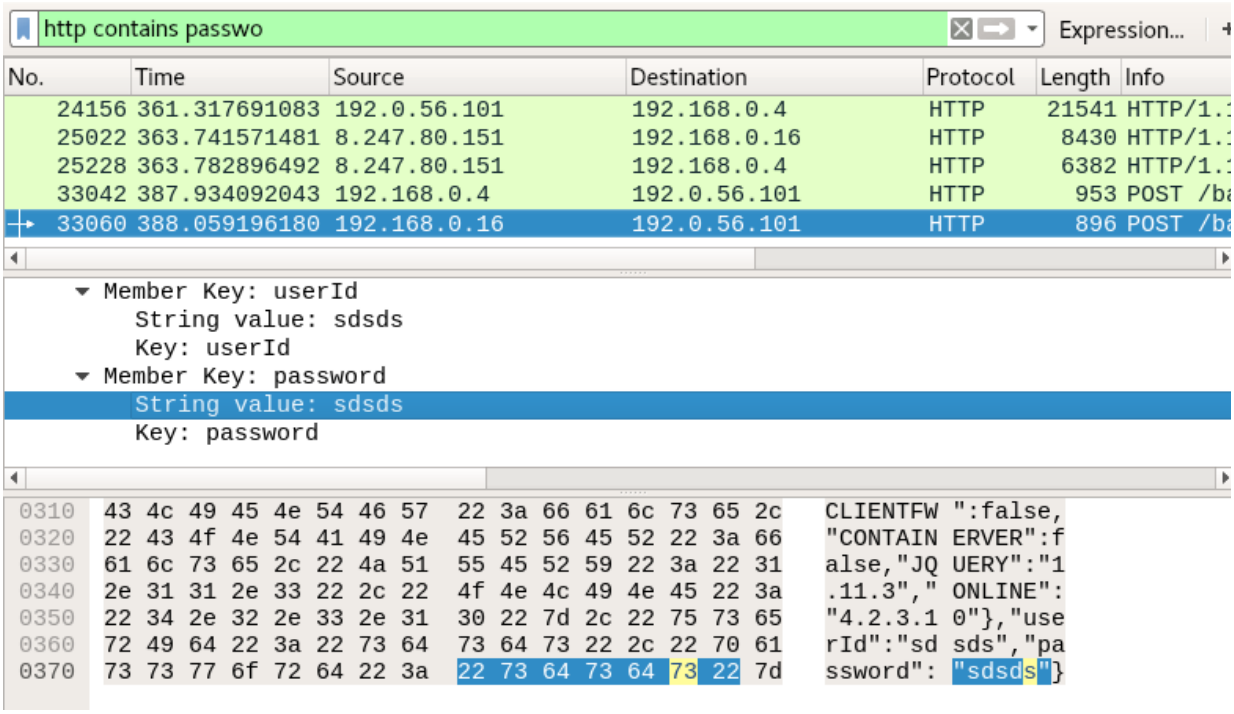


Fig. 6. Escenario 2. Captura de datos en texto plana con ataque de redirección. [Fuente: Autor]

Escenario 3: Ataque de phishing real con certificado válido

Para este escenario se usa la misma topología del primer escenario (Figura 3), donde el atacante se encuentra fuera de la red interna de la victima. Se analiza un ataque real, reportado como Phishing por usuarios web en la plataforma Phishtank (phishtank,com), en el cual un subdominio similar al dominio legítimo de un banco es usado para confundir con mayor facilidad a los usuarios. Dicho subdominio cuenta con un certificado válido, expedido por cPanel (CA de confianza), como se muestra en la Figura 7.

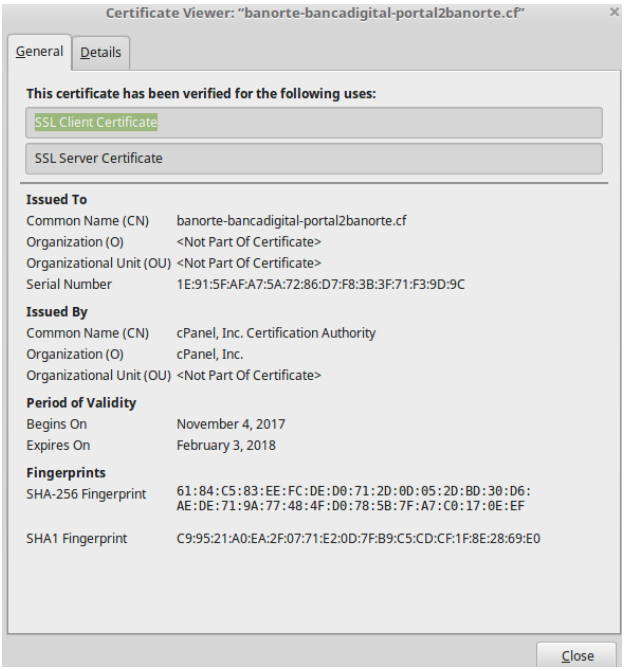


Fig.7. Escenario 3. Detalles de certificado. [Fuente: Autor]

Escenario 4: Ataque de prueba con certificado válido

Nuevamente se usa la topología del primer escenario (Figura 3), con el atacante en la red pública. Se emula un ataque real, haciendo registro de un dominio que usa un nombre asociado a la marca de un banco conocido (bancolmena.xyz). Se publica el sitio en la web (figura 8)y se instala un certificado SSL legitimo de manera gratuita emitido por cPanel (Comodo), el cual cumple con todas las validaciones (figura 9).

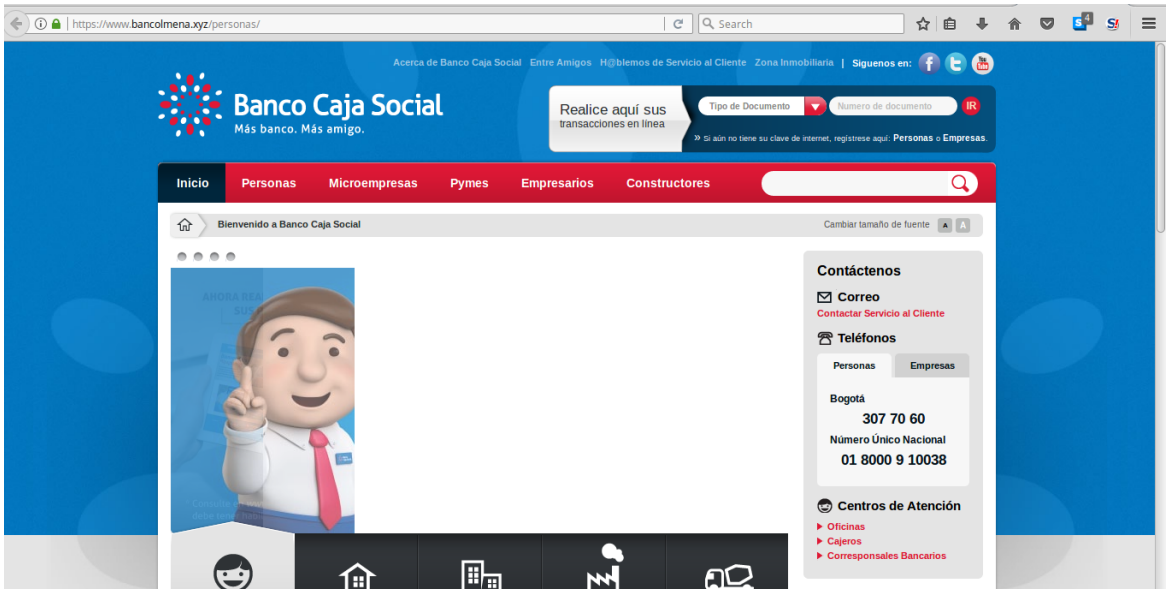
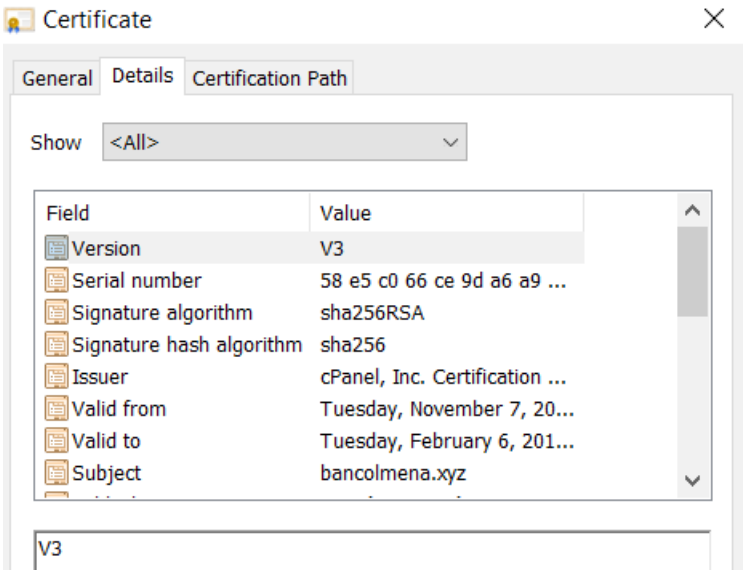


Fig. 8. Escenario 4. Ataque ejecutado haciendo uso de un certificado válido para un dominio intencionalmente registrado con un nombre asociado a una entidad legítima.



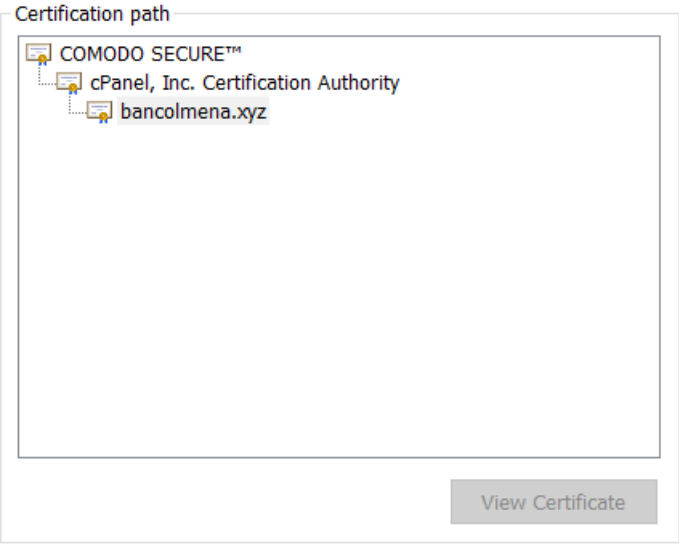


Fig. 9. Escenario 4. Detalles del certificado configurado. [Fuente: Autor]

Resultados

Escenario 1: Certificados Inválidos

Se prueba el acceso a los distintos subdominios del proyecto badsssl.com haciendo uso del navegador Chrome, obteniendo los siguientes resultados:

Tabla 1.Escenario 1: Efectividad

Error Certificado	SOPHOS UTM	Checkpoint
Expirado	Bloqueado por Firewall	Bloqueado por Firewall
Autofirmado	Bloqueado por Firewall	Bloqueado por Firewall
Error de coincidencia de nombre	Bloqueado por Navegador	Bloqueado por Navegador
Autoridad no confiable	Bloqueado por Firewall	Bloqueado por Firewall
Revocado	No Bloqueado	Bloqueado por Firewall
HPKP Erróneo	No bloqueado	No bloqueado
SHA1 Intermedio	No bloqueado	No bloqueado

Tabla 2. Escenario 1: Consumo CPU

Consumo Promedio	**SOPHOS UTM**	**Checkpoint**
40%-49%	43%-52%	40%-57%

Escenario 2: Redirección HTTP a HTTPS

Para este escenario aunque no se encontró algún bloqueo o alerta por parte de las herramientas de Firewall, se vieron distintas respuestas de acuerdo al navegador web empleado:

Tabla 3. Escenario 2 . Efectividad

Bloqueo de Firewall		**Bloqueo de Navegador**		
		Chrome v62.0.3202.94	Firefox 57.0	Internet Explorer 11.096
SOPHOS UTM	NO	SI	No, pero genera advertencias al ingresar los datos.	NO
Checkpoint	NO	NO	SI	NO

Tabla 4. Escenario 2: Consumo CPU

Consumo Promedio	**SOPHOS UTM**	**Checkpoint**
48%-54%	49%-57%	50%-56%

Escenario 3: Ataque real con certificado válido

En este escenario se valida la respuesta tanto de las herramientas firewall como de los navegadores ante un ataque real reportado en algunas listas negras:

Tabla 5 . Escenario 3. Efectividad

Bloqueo de Firewall		**Bloqueo de Navegador**		
SOPHOS UTM	Checkpoint	Chrome v62.0.3202.94	Firefox 57.0	Internet Explorer 11.096
NO	NO	NO	NO	NO

No se encontró acción preventiva o correctiva alguna por parte de las herramientas o navegadores.

Escenario 4: Ataque implementado con certificado válido

En este escenario también se valida la respuesta tanto de las herramientas firewall como de los navegadores, esta vez ante el ataque de suplantación implementado.

Tabla 6. Escenario 4. Efectividad

Bloqueo de Firewall		**Bloqueo de Navegador**		
SOPHOS UTM	Checkpoint	Chrome v62.0.3202.94	Firefox 57.0	Internet Explorer 11.096
NO	NO	NO	NO	NO

Para este ataque tampoco fue efectiva ninguna de las validaciones realizadas por las herramientas evaluadas. Sin embargo, a pesar de que este ataque fu implementado con fines puramente educativos y no fue expuesto a ningún usuario, fue detectado por la entidad bancaria correspondiente y suspendido por el proveedor de hosting 12 días después implementado. Este es un periodo de detección bastante extenso en el cual se hubieran podido afectar bastantes usuarios, aunque puede que en caso de haber pretendido realmente capturar información de usuarios la detección fuese más rápida.

Discusión

En el escenario 1 se encuentra respuesta y bloqueo efectivo para los distintos casos evaluados, dado que se validan vulnerabilidades ampliamente investigadas y evaluadas. Cómo única diferencia en el primer escenario se encontró que Sophos no toma ninguna acción cuando se accede a un sitio con certificado web revocado, tanto por CRL o por OCSP.

En el escenario 2 se encontró que el ataque de redirección no es controlado por ninguna de las herramientas de Firewall. Aunque con las políticas de seguridad adecuadas era posible alertar y evitar el ataque de ARP spoofing, se debe tener en cuenta que existen distinta técnicas de ataque para la interceptación que pueden ser usadas. Sin embargo, cabe resaltar que pesar de la omisión de los firewall, se encontró que el ataque fue validado y bloqueado de manera efectiva por algunos navegadores web que implementan la política HSTS, luego de haber visitado por primera vez el sitio de manera segura (antes del ataque). En este escenario fue posible evidenciar la importancia de los navegadores en la prevención de ataques.

Para los escenarios 3 y 4 ni los Firewall ni los navegadores alertaron o bloquearon de alguna manera los ataques, a pesar de ser este un vector de ataque común actualmente. Ante la reciente facilidad de obtención e instalación de certificados SSL, la cual se demostró en el montaje del escenario 4.

Conclusiones

Ambas herramientas evaluadas mostraron resultados similares en rendimiento y efectividad, con una mínima ventaja por parte de la herramienta Checkpoint en el primer escenario. Sin embargo, la eficiencia aunque equitativa fue deficiente para los escenarios 2, 3 y 4, demostrando que las

herramienta de seguridad MITM por si solas no son herramientas de prevención y mitigación suficientes ante ataques de suplantación web. Es necesario el uso conjunto de distintas herramientas, estrategias y políticas para minimizar de manera considerable los riesgos, sin transferir toda la responsabilidad de la prevención al usuario, quien regularmente es el más vulnerable.

A través de este trabajo fue posible revisar y evaluar la validez de ideas comunes en la seguridad informática corporativa, especialmente bancaria, en donde a los usuarios regularmente se les presentan los certificados SSL y las soluciones de Firewall como garantes de seguridad.

Trabajos Futuros

De acuerdo a las tendencias actuales de uso de tecnología y de ataques cibernéticos, se encuentra un oportunidad de investigación importante en los vectores de ataque y contramedidas en dispositivos móviles. Ya se han hecho algunas revisiones al respecto [20][21], mas sin embargo las técnicas de ataque continuan en constante evolución y sofisticación. Por otro lado, junto a los ataques evolucionan las propuestas de posibles contramedidas, resaltándose la idea de posibles modelos predictivos de riesgo, teniendo en cuenta los patrones encontrados al analizar distintos parámetros de los certificados SSL asociados a sitios web en los cuales se ha encontrado fraude.

Referencias

[1] A. Freier, P. Karlton, and P. Kocher, “The Secure Sockets Layer (SSL) Protocol Version 3.0,” Internet Eng. Task Force, vol. RFC 6101, pp. 1–67, 2011.

[2] L. S. Huang, A. Rice, E. Ellingsen, and C. Jackson, “Analyzing forged SSL certificates in the wild,” Proc. - IEEE Symp. Secur. Priv., pp. 83–97, 2014.

[3] J. Jarmoc, D. Unit, and T. Intelligence, “SSL/TLS interception proxies and transitive trust,” Black Hat Eur., pp. 1–21, 2012.

[4] D. Cooper and S. Farrell, “rfc5280 X.509 formats and semantics,” pp. 1–151, 2008.

[5] R. Oppliger, R. Rytz, and T. Holderegger, “Internet banking: Client-side attacks and protection mechanisms,” Computer (Long. Beach. Calif), vol. 42, no. 6, pp. 27–33, 2009.

[6] A. Mu and A. Guzm, “Contramedidas en la suplantaci´ on de autoridades de certificaci´ on. Certificate pinning,” pp. 2–5, 2014.

[7] E. S. Alashwali, “Cryptographic vulnerabilities in real-life web servers,” 2013 3rd Int. Conf. Commun. Inf. Technol. ICCIT 2013, pp. 6–11, 2013.

[8] P. Sirohi, “A comprehensive study on security attacks on SSL / TLS Protocol,” no. October, pp. 893–898, 2016.

[9] M. Saraca, T. Radojevic, N. Stanisic, S. Adamovic, and D. Radovanovic, “Safety of e-business Applications in Serbia: Applied Knowledge Based on SSL Traffic,” J. Internet Bank. Commer., vol. 19, no. 3, pp. 1–15, 2014.

[10] M. Marlinspike, “New Tricks For Defeating SSL In Practice,” 2009. [Online]. Available: https://www.blackhat.com/presentations/bh-dc-09/Marlinspike/BlackHat-DC-09-Marlinspike-Defeating-SSL.pdf.

[11] S. H. Mortazavi, M. S. Yazdani, F. Jalilzadeh, and P. S. Avadhani, “A novel secure protocol called FSSL using fuzzy controller for web security,” Conf. Proc. - IEEE Int. Conf. Syst. Man Cybern., vol. 2014–Janua, no. January, pp. 1192–1197, 2014.

[12] M. Dietz, A. Czeskis, and D. S. Wallach, “Origin-Bound Certificates : A Fresh Approach to Strong Client Authentication for the Web,” Proc. 21st USENIX Conf. Secur. Symp., pp. 16--16, 2012.

[13] D. Wendlandt, D. G. Andersen, and A. Perrig, “Perspectives: Improving SSH-style Host Authentication with Multi-Path Probing.,” USENIX Annu. Tech. Conf., pp. 321–334, 2008.

[14] M. Moxie, “SSL and the Future of Authenticity,” in RSA Conference, 2012.

[15] S. Ruoti, K. Seamons, and D. Zappala, “TLS Proxies: Friend or Foe?”

[16] M. T. ONeill, “The State of Man-in-the-Middle TLS Proxies: Prevalence and User Attitudes,” Brigham Young University, 2016.

[17] S. M. Jawi and F. H. M. Ali, “Non-intrusive SSL/TLS proxy implementation and issues,” 2015 IEEE Student Conf. Res. Dev. SCOReD 2015, pp. 684–689, 2016.

[18] S. M. Jawi and F. H. M. Ali, “Rules and results for SSL/TLS nonintrusive proxy based on JSON data,” 2016 6th Int. Conf. IT Converg. Secur. ICITCS 2016, 2016.

[19] A. King and L. Garron, “Site for testing clients against bad SSL configs,” 2015. [Online]. Available: https://badssl.com/.

[20] Y. Wang, C. Hahn, and K. Sutrave, “Mobile payment security, threats, and challenges,” Proc. 2016 2nd Conf. Mob. Secur. Serv. MOBISECSERV 2016, 2016.

[21] X. Wei and M. Wolf, “A Survey on HTTPS Implementation by Android Apps: Issues and Countermeasures,” Appl. Comput. Informatics, vol. 13, no. 2, pp. 101–117, 2017.

INFORMACIÓN DE LOS AUTORES

Primer Autor: Diana Carolina Angulo
Ingeniera Electrónica – Universidad Distrital Francisco José de Caldas – Colombia.
Líder técnica de SOC – Easy Solution SAS – Colombia – dcanguloc@correo.udsitrital.edu.co
Segundo Autor: Juan Felipe Henao
Ingeniero de Sistemas – Universidad de Nariño – Colombia.
Contratista – Instituto Distrital de la participación y acción comunal - País –
juan.felipe.h@hotmail.com