Evaluación de la Seguridad de Certificados Digitales en las Plataformas Financieras de Ecuador

Assessment of the Security of Digital Certificates in the Financial Platforms in Ecuador

David Peñarrieta
Gobierno Autónomo Descentralizado Municipal de Junín, Ecuador
d.penarrieta@junin.gob.ec

Marlon Navia
Universidad Técnica de Manabí, Ecuador
marlon.navia@utm.edu.ec

Eliana García
Universidad Técnica de Manabí, Ecuador
eliana.garcia@utm.edu.ec

Dannyll Zambrano
Universidad Técnica de Manabí, Ecuador
michellc.zambrano@utm.edu.ec

Revista Tecnológica ESPOL - RTE
vol. 36, no. 2, p. 174 - 189, 2024
Escuela Superior Politécnica del Litoral, Ecuador
ISSN: 0257-1749
ISSN-E: 1390-3659
rte@espol.edu.ec

Received: 24 July 2024
Accepted: 24 December 2024

DOI: https://doi.org/10.37815/rte.v36n2.1222

Resumen: Este artículo muestra un diagnóstico de la aplicación de los certificados digitales en los servicios de banca virtual del Ecuador. La importancia de esta temática se fundamenta en los crecientes ataques a los servicios electrónicos de plataformas financieras en la región y el mundo, a causa de la explotación de vulnerabilidades descubiertas por ciberdelincuentes, en la frágil aplicación de los conjunto de cifrados. El objetivo de la investigación es mostrar el nivel de seguridad que poseen estos portales de banca online (personas), en la aplicabilidad de los protocolos SSL/TLS, con sus respectivos suites de cifrados del lado del servidor. Para ello, se analizaron 18 entidades financieras, mediante la herramienta en línea SSL server Test de Qualys SSLLabs. Se encontró que el 20% de las entidades bancarias analizadas presentan debilidad en la aplicabilidad de los certificados digitales, lo que podría ocasionar ataques informáticos a dichas plataformas virtuales, durante el proceso de comunicación entre cliente/servidor a través del servicio de internet. La confidencialidad, integridad y disponibilidad de los datos son características indispensables de la seguridad de la información que un usuario debe recibir en el servicio de banca virtual personas. Como aporte adicional de este trabajo, se realiza una revisión de las recomendaciones de uso de certificados digitales de acuerdo con las regulaciones que emite la IETF a través de los respectivos RFC.

Palabras clave: Ataques, banca electrónica, conjunto de cifrado, SSL, TLS, vulnerabilidades.

Abstract: This article presents a diagnosis of the application of digital certificates in the virtual banking services of Ecuador. The importance of this topic is based on the increasing attacks on electronic services of financial platforms in the region and the world, due to the exploitation of vulnerabilities discovered by cybercriminals in the weak application of cipher suites. The objective of the research is to show the level of security of these online banking portals (individuals), in the applicability of

SSL/TLS protocols, with their respective cipher suites on the server side. Eighteen financial entities were analyzed using the online tool SSL Server Test by Qualys SSL Labs. It was found that 20% of the analyzed banking entities show weaknesses in the applicability of digital certificates, which could lead to cyberattacks on these virtual platforms during the client/server communication process over the internet. Confidentiality, integrity, and availability of data are indispensable characteristics of information security that a user should receive in the virtual banking service. Additionally, this work reviews the recommendations for the use of digital certificates according to the regulations issued by the IETF through the respective RFCs.

Keywords: Attacks, Electronic Banking, Cipher Suite, SSL, TLS, Vulnerabilities.

Introducción

Ingresar credenciales de acceso en un formulario de un sitio web es un requisito necesario para utilizar los servicios de una plataforma electrónica, ello implica introducir datos de carácter personal, por lo tanto, esta información se supone que es confidencial, ajena a cualquier agente externo, por lo que, no debería ser conocido por nada o nadie. Los usuarios no sofisticados se preocupan poco por la seguridad de los sitios electrónicos.

Las aplicaciones web implementadas por las instituciones financieras, ya sean estas públicas o privadas, han marcado un antes y un después en la manera tradicional de hacer banca, cada día acrecienta la cantidad de clientes que acceden a este medio electrónico; pero este proceso de modernización genera ciertas interrogantes a los clientes de los servicios de banca online, ¿es confiable este servicio? ¿es segura esta aplicación web?; es aquí cuando aparecen términos conocidos, concernientes a la seguridad como: vulnerabilidades, ataques cibernéticos, hackers, ciberdelincuencia, todos ellos correlacionados de alguna manera.

Los profesionales informáticos y directivos de las entidades financieras constantemente pretenden disminuir las debilidades en el servicio de banca virtual, por ejemplo, se destaca la implementación de varios factores de autenticación, aplicación de seguridad en la capa de transporte (encriptación de la información que transita en el internet, desde el cliente hacia el servidor y viceversa), entre otras. La confidencialidad, integridad y disponibilidad de los datos son características indispensables de la seguridad de la información que un usuario debe recibir en el servicio de estos canales virtuales, de allí que se hace imprescindible el uso de los protocolos de seguridad para acceder a estos portales web.

La Figura 1 muestra un ejemplo de una aplicación de banca virtual, cuya URL utiliza el protocolo HTTPS (Hypertext Transfer Protocol Secure) que es la versión segura del protocolo HTTP, ampliamente utilizado para la transmisión de datos en la web. En este ejemplo, el navegador Mozilla reconoce que esta página web utiliza un certificado digital seguro, la información que transita desde y hacia el servidor viaja cifrada, gracias al uso del protocolos de seguridad TLS (Transport Layer Security), la aplicación de certificados de seguridad, técnicas de encriptación, uso de llaves simétricas y asimétricas, la combinación de suites de cifrados convierten al protocolo HTTP en HTTPS (S de security), evitando que personas mal intencionadas intercepten y utilicen los datos para usos indebidos.

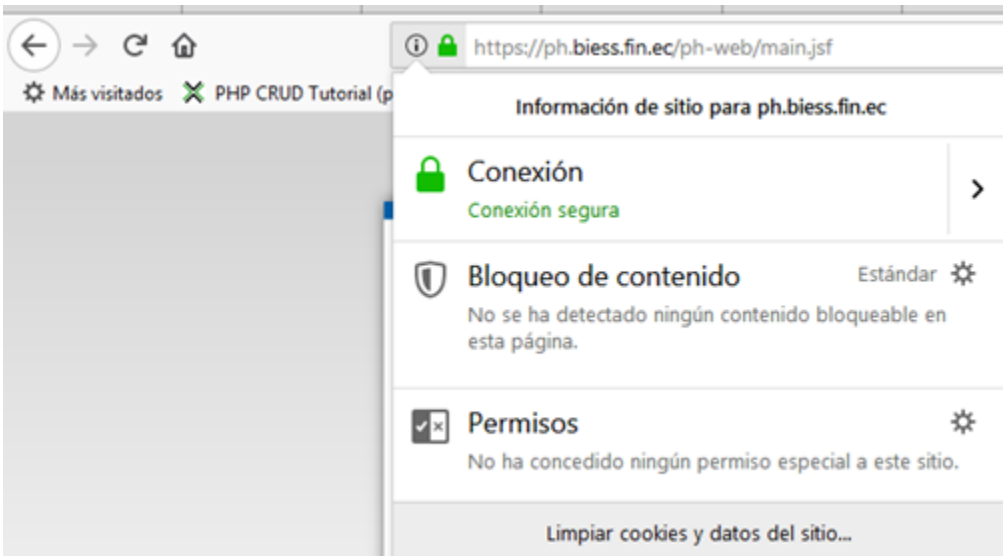

Figura 1

Conexión segura de un sitio web, a través de un dominio con https

Ante este contexto, el propósito de este trabajo de investigación es evaluar el nivel de seguridad de los certificados digitales en el servicio de banca electrónica, en lo referente al protocolo de seguridad TLS y sus versiones correspondientes, el beneficio que este conlleva, así como, las consecuencias que produciría el no aplicar una configuración adecuada del lado del servidor. También se busca promover a los usuarios no sofisticados la masificación de esta temática, que cada día cobra más valía en el servicio de banca en línea.

Este manuscrito presenta un diagnóstico del uso de las suites de cifrados de varias bancas virtuales del Ecuador, en ciertos casos denotan debilidad o fortaleza en la aplicación de los certificados digitales; además, se presenta el tipo de vulnerabilidades a las cuales son susceptibles o inmunes.

El documento está constituido en varias secciones. La sección I muestra la introducción sobre el entorno del problema, antecedentes y propósito relevante del artículo; además tópicos relacionados a la seguridad en la capa de transporte, las vulnerabilidades más conocidas, recomendaciones de la aplicabilidad de suites de cifrados, para entender con mayor objetividad el tema presentado.

La sección II describe las herramientas y los métodos utilizados para obtener los datos. En la sección III se muestran los resultados del diagnóstico realizado a las entidades financieras mediante el uso de test de certificados digitales en línea. Y, por último, la sección IV concluye este trabajo y discute las direcciones de trabajos futuros.

Secure Sockets Layer

SSL es un protocolo de encriptación que fue concebido para establecer seguridad en las comunicaciones por internet, fue desarrollado originalmente por Netscape, quienes lo introdujeron para su plataforma comercial Netscape Navigator (Ristić, 2014).

SSL v1.0 fue la primera versión del protocolo, pero no salió a producción porque la comunidad criptográfica realizó muchas observaciones, haciendo énfasis en la aplicación de algoritmos criptográficos débiles.

Transport Layers Security

En mayo de 1996, se conformó un grupo de trabajo para migrar SSL de Netscape a IETF (Ristić, 2014). El proceso fue lento debido a las luchas políticas entre Microsoft y Netscape, una consecuencia de la disputa más grande para dominar la Web (Ristić, 2014). TLS 1.0 fue finalmente lanzado en enero de 1999, IETF realizó la publicación formal del protocolo a través del RFC (Request for Comments) 2246 (Dierks & Allen, 1999). Aunque las diferencias con SSL 3 no eran grandes, el nombre fue cambiado para complacer a Microsoft (Ristić, 2014).

TLS v1.2 fue aprobado mediante RFC 5246 en agosto de 2008 (Dierks and Rescorla, n.d.), esta especificación del protocolo, proporciona privacidad e integridad de los datos entre dos aplicaciones que se comunican, el protocolo es compuesto por dos capas: el protocolo de registro TLS y el protocolo de enlace TLS.

El protocolo se define en varias normas por la IETF (The Internet Engineering Task Force (IETF), n.d.) , incluyendo entre otras RFCs 2246, 3546, 4346, 4366, 4680, 4492, 5246, 5288, 5746, 6176 y 6655.

TLS se basa en la tecnología Infraestructura de Clave Pública (PKI), la cual garantiza que el tráfico se envíe al destinatario correcto (Ristić, 2014). Es el protocolo de cifrado más utilizado en Internet, tiene como objetivo garantizar la integridad y confidencialidad del mensaje, el cual transita por la red en la capa de transporte. Permite la comunicación segura de extremo a extremo, evitando la intercepción y alteración de la información.

La Figura 2 muestra la negociación de estos parámetros en un caso genérico. Hace uso de los siguientes intercambios entre el cliente y el servidor.

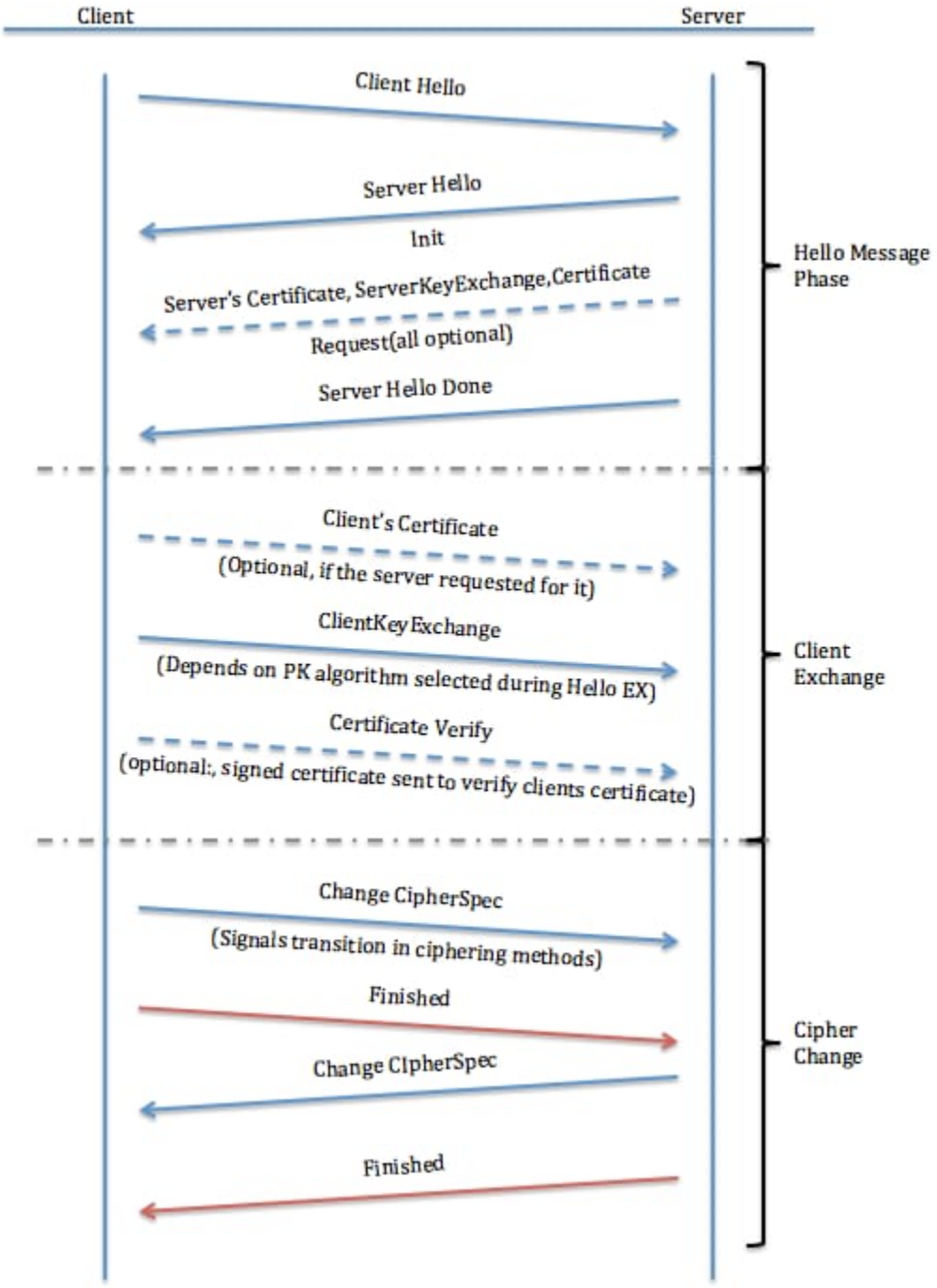

Figura 2

Transacción e intercambio de paquetes entre cliente/servidor, mediante SSL (Cisco, 2017)

1. El cliente inicia una consulta enviando un mensaje de tipo ClientHello, que contiene en particular los conjuntos de cifrado que admite.

2. El servidor responde con un ServerHello que contiene la suite retenida.

3. El servidor envía un mensaje de Certificado que, en particular, contiene su clave pública dentro de un certificado digital.

4. El servidor transmite en un ServerKeyExchange un valor efímero que firma usando la clave privada asociada con la clave pública anterior.

5. El servidor manifiesta que está en espera con un ServerHelloDone.

6. Después de la verificación del certificado y la autenticación del valor anterior, el cliente, a su vez, elige un valor efímero que cifra mediante la clave pública del certificado y luego lo transmite en un ClientKeyExchange;

7. El servidor señala que activa la suite completamente con un ChangeCipherSpec.

8. El cliente envía un mensaje (Finished) finalizado, el primer mensaje está protegido de acuerdo con el conjunto de cifrado y los secretos proceden del intercambio de claves efímeras.

9. El servidor señala la activación de la misma suite con un ChangeCipherSpec.

10. El servidor envía a su vez un mensaje (Finished) finalizado, su primer mensaje seguro.

En agosto de 2018 se aprobó definitivamente TLS v1.3 (Rescorla, 2018), luego de 28 revisiones realizadas por el grupo de trabajo de la IETF.

En TLS, toda la seguridad comienza con la identidad de cifrado del servidor. Se necesita una clave privada fuerte para evitar que los ciberdelincuentes lleven a cabo ataques de suplantación. Es primordial tener un certificado válido y fuerte, que otorgue a la clave privada el derecho a representar un nombre de un host en particular. Sin estos dos elementos fundamentales, ninguna otra cosa puede estar segura.

De acuerdo a la información mostrada en el portal SSL Pulse (2019), de 135.254 sitios encuestados en julio de 2023, el 61,9% son denominados sitios seguros, de estas plataformas electrónicas el 63.5%

admiten el protocolo TLS v1.3 (+0,5% en comparación con junio de 2023), el 99,9% del total soportan TLS v1.2. En ese mismo contexto, el 38, 1% restante tienen una seguridad inadecuada, del lado del servidor permite el uso de versiones declaradas obsoletas (Dell & Farrell, 2021) TLS v1.1, TLS v1.0 y de esta porción un grupo minoritario 2,1% aún permiten el uso de SSL v3.0 y SSL v2.0 (SSL Pulse, 2019).

Conjunto de cifrados

Es un grupo de algoritmos y reglas utilizadas para cifrar y descifrar información. Están diseñados para garantizar la seguridad y confidencialidad de los datos mediante la transformación de la información en un formato legible con la clave correspondiente.

Con la idea de lograr mejor seguridad a través de la capa de transporte en el modelo TCP IP, estos protocolos ofrecen servicios de autenticación, encriptación, intercambio de claves de cifrado e integridad. Para gestionar cada una de estas características, se puede parametrizar un algoritmo específico en función de las necesidades a nivel de servidor.

La Figura 3 modela un ejemplo de cómo se conforma un Conjunto de Cifrado en el protocolo TLS 1.2.

Figura 3

Componentes de un Conjunto de Cifrados TLS 1.2

La Tabla 1 muestra los componentes combinables de los conjuntos de suites de cifrados considerados por IANA (Internet Assigned Numbers Authority) (Rescorla, 2008), exclusivos para el protocolo TLS en su versión 1.2.

Tabla 1
Posibles valores combinables en los Conjunto de Cifrados TLS 1.2 (Rescorla, 2008)

Componente	Valores posibles
Protocolo	TLS
Intercambio de claves	ECDH, ECDHE
Autenticación	ECDSA, RSA
Algoritmo de cifrado	AES
Longitud de clave	128, 256
Modo de cifrado	CBC, GCM
Algoritmo de Hash	SHA256, SHA384

En la versión de TLS 1.3 existen diferencias en la estructura del conjunto de cifrado, son más cortos que los respectivos conjuntos TLS 1.2, no hacen referencia al tipo de certificado, y al mecanismo de intercambio de claves, por lo que el número de negociaciones necesarias para determinar los parámetros de cifrado ha disminuido. Los conjuntos de cifrado en TLS 1.3 tienen el siguiente aspecto, según se observa en la Figura 4.

Figura 4
Componentes de un conjunto de cifrados TLS v1.3

Los esquemas de cifrados con autenticación AE (Auntenthicated Encryption) utilizan un cifrador simétrico (de bloque o de flujo) y un mecanismo MAC. AEAD (Authentication Encryption with Associated Data) que ofrece la capacidad de aplicar el cifrado solo a la autentificación (Centro Criptológico Nacional, 2017).

Los esquemas AE/AEAD autorizados para uso recomendado son según el Centro Criptológico Nacional (2017):

- Cifrador de bloque AES: (EAX, GCM y CCM).
- Cifrador de flujo CHACHA20 :(POLY1305).

En TLS v1.3, los conjuntos basados en cifrado de bloque y cifrado de flujo incluyen: TLS_AEAD_SHA256 y TLS_AEAD_SHA384 (Rescorla, 2018).

La Tabla 2 muestra los conjuntos de suites de cifrados exclusivo para el protocolo TLS en su versión 1.3, considerados por IANA en el RFC8446 apéndice B.4 (Rescorla, 2018).

Tabla 2

Conjuntos de cifrados para TLS 1.3 - Consideraciones IANA (Rescorla, 2018)

Conjunto de Cifrados TLS 1.3	
Descripción	**Valor**
TLS_AES_128_GCM_SHA256	{0x13,0x01}
TLS_AES_256_GCM_SHA384	{0x13,0x02}
TLS_CHACHA20_POLY1305_SHA256	{0x13,0x03}
TLS_AES_128_CCM_SHA256	{0x13,0x04}
TLS_AES_128_CCM_8_SHA256	{0x13,0x05}

Debilidades y Amenazas

Las organizaciones y empresas ven como su superficie de ataque aumenta en un panorama de intimidaciones cada vez más hostil. Existen varias amenazas descubiertas desde la invención del protocolo SSL ahora TLS, las que han evolucionado en el tiempo, muchas han sido resueltas con las versiones actualizadas del protocolo. Los ciberdelincuentes actúan aprovechando debilidades técnicas descubiertas.

Las versiones anteriores de TLS 1.2 (incluido las versiones SSL) contienen vulnerabilidades.

Algunas configuraciones del servidor son expuestas a *exploits*. Ataques como Beast, Crime, Breach, Heartbleed, Poodle, Freak, Logjam son susceptibles a las debilidades presentadas al aplicar configuraciones básicas en los certificados digitales de los servidores web. A continuación se abordaran algunos tipos de posibles ataques a los que son expuestos, sobre todo, las detectadas en el proceso de investigación.

Poodle SSL v3

Poodle posee la capacidad de que un atacante (MITM) pueda bajar la versión actualizada de los navegadores modernos y forzar el uso de SSL3, alterando su versión a una desactualizada en sus conexiones en todo el trayecto hasta que puede ser vulnerada. Hay una solución a este problema, a través del indicador de TLS_FALLBACK_SCSV (Advisory, 2014), este es un mecanismo que resuelve los inconvenientes causados por reintentar conexiones fallidas y, por lo tanto, evita que los ciberdelincuentes induzcan a los navegadores usar SSL 3.0; debe ser implementada en los clientes y del lado del servidor con el fin de ser eficaz.

En un funcionamiento normal, SSL3 no debe ser permitido su uso del lado del servidor. Aunque existe un grupo de clientes que usan Sistemas Operativos desactualizados, ello conlleva a utilizar versiones de navegadores obsoletos, es el caso de Internet Explorer 6 en Windows XP.

La Figura 5 (Statcounter, 2023) muestra que en Ecuador hay un grupo minoritario aproximadamente un 6 % de ordenadores que aún utilizan versiones de Sistemas Operativos Windows (XP, 7, 8, 8.1), respectivamente. A corto plazo, es posible mitigar varias técnicas de ataques en este caso Poodle, evitando el uso de suites con modos de cifrados CBC.

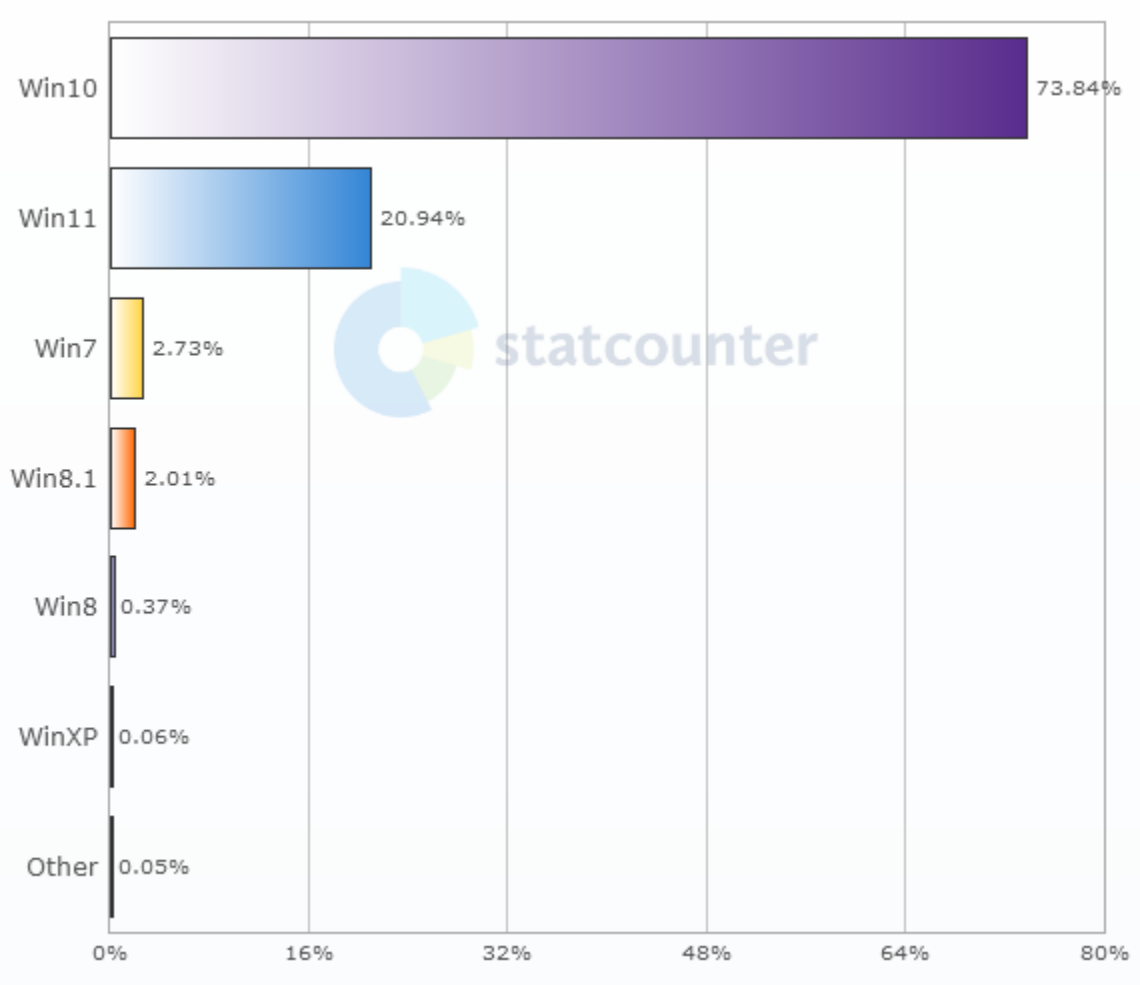

Figura 5

Uso de SO. Microsoft Desktop en Ecuador, período (septiembre 2022- septiembre 2023)

Ataque Freak

La vulnerabilidad Freak (Factoring RSA Export Keys) es posible si el servidor implementa alguno de los cifrados de exportación de claves (Ristić, 2014), un atacante podrá recuperar la clave RSA para luego

interceptar la comunicación entre cliente y servidor usando técnicas de tipo MITM (Man in the Middle), y dado que, el atacante es capaz de cifrar y descifrar toda la conversación, el cliente no será advertido en ningún momento, perdiendo la integridad y la confidencialidad de la información.

Para mitigar esta vulnerabilidad, es altamente recomendable Factoring & Export (1990):

- Del lado del servidor: Desactivar el soporte para versiones TLS del tipo EXPORT, desactivar SSL y versiones TLS inferiores a la versión 1.2.

- Los clientes: mantener actualizado los navegadores con la versión más reciente.

Ataque Drown

Básicamente el atacante se aprovecha de que un servidor permita conexiones SSLv2, para capturar las sesiones TLS y descifrar las claves RSA. Se necesitan capturar aproximadamente 1.000 sesiones TLS utilizando intercambios de claves RSA, después hacer unas 40.000 conexiones SSLv2 y realizar 2^50 operaciones de cifrado simétrico, para descifrar 1 sesión TLS (Aviram et al., 2016). Este ataque para ser efectivo requiere de un poco de paciencia, de lograr el objetivo, un atacante puede obtener información valiosa.

Ataque Beast

Las versiones del protocolo criptográfico TLS (v1.1 y v1.2) son inmunes al exploit (Rizzo & Duong, 2011). El ataque extrae pequeñas partes de datos encriptados, trabajo basado en la debilidad previamente conocida en la construcción predecible del vector de inicialización (IV) como se usa en TLS 1.0 (Ristić, 2014). Beast es el primer ataque que realmente desencripta las comunicaciones HTTPS, dado que el resto de amenazas SSL se han basado, por lo general, en el robo de identidad de sitios certificados como seguros.

Ataque Crime

Es otro tipo de ataque a SSL/TLS, a través del cual se podría obtener cookies de sesiones pese a estar realizándose vía HTTPS, mediante el uso de malware JavaScript para extraer las cookies del cliente en un ataque MITM activo (Ristić, 2014). La debilidad que permite descifrar tal información, se da por la compresión, la cual funciona localizando secuencias que se repiten, acortándolas en forma de token. Cuando una cadena de texto contiene otra subcadena que se repite en otro punto del texto, ambas serán sustituidas por el token.

En el caso de enviar una petición por HTTPS, con la compresión DEFLATE activada, todo lo enviado será comprimido y cifrado, tanto las cabeceras como el cuerpo. El problema de la compresión ya

es comentado en el RFC 3749 (Transport Layer Security Protocol Compression Methods) (The Internet Engineering Task Force (IETF), 2004), donde en el apartado 'Security Considerations' explican que la compresión puede llegar a ser utilizada para deducir el mensaje que había sido cifrado.

Openssl Padding Oracle (CVE-2016-2107)

Es un ataque que utiliza la validación de relleno de un mensaje criptográfico para descifrar el texto cifrado.

En esta vulnerabilidad se determinó que OpenSSL filtraba la información de temporización al descifrar los registros encriptados del protocolo TLS/SSL, cuando la conexión utiliza el conjunto de cifrado AES_CBC y el servidor compatible con AES-NI (OpenSSL, 2016). Un atacante MITM podría usar esta debilidad para recuperar texto sin formato de paquetes cifrados mediante el uso de un servidor TLS/SSL (OpenSSL, 2016).

Paddding Oracle se introdujo como parte de la solución incorrecta del ataque Lucky 13 (CVE-2013-0169) (*Blog de Internet Security Auditors: Seguridad SSL/TLS: LUCKY 13*, n.d.). La revisión de relleno se reescribió para que se realice en tiempo constante, asegurándose de que siempre se lean los mismos bytes y se comparen con los bytes MAC o los de relleno.

Return Of Bleichenbacher's Oracle Threat (Robot)

Esta vulnerabilidad permite realizar operaciones de descifrado en firma RSA con la clave privada de un servidor TLS. En 1998, Daniel Bleichenbacher (Bleichenbacher, 1998) descubrió que los mensajes de fallos dados en los servidores SSL por errores en el relleno PKCS # 1 v1.5 permitían un ataque de texto cifrado elegido de forma adaptativa; este ataque rompe por completo la confidencialidad de TLS cuando se utiliza con encriptación RSA. Los hosts HTTPS que admiten intercambios de clave de cifrado RSA son vulnerables; un atacante puede registrar pasivamente el tráfico y luego descífralo. Este ataque afecta los modos de cifrado TLS que usan cifrado RSA (Böck et al., 2017). La gran mayoría de las conexiones TLS actuales usan el intercambio de claves Elifftic Curve Diffie Hellman y necesitan RSA solo para firmar. Para las suites de cifrados que no utilizan ECDHE es recomendable mejor deshabilitar los modos de encriptación RSA. Al deshabilitar el cifrado RSA se refiere a todos los cifrados que comienzan con TLS_RSA; se excluye los cifrados que usan TLS_ECDHE_RSA ellos no se ven afectados por Bleichenbacher's (Böck et al., 2017).

Zombie POODLE y GOLDENDOODLE

Es un exploit de pilas TLS modernas mediante la técnica clásica del oráculo de relleno CBC, descrita por Serge Vaudenay (Vaudenay, 2002); aplicables solo si el servidor usa la versión del protocolo TLS

1.2, TLS 1.1 y TLS 1.0 cuyos conjuntos de cifrados usan el modo de encadenamiento de bloques de cifrado CBC (Sannegowda, 2019).

Estas vulnerabilidades afectan a los servidores que usan el protocolo HTTPS, con modos criptográficos que deberían haber quedado obsoletos, sin embargo, inexplicablemente aún son compatibles con TLSv1.2 (Young, 2019a).

Las debilidades son expuestas por CVE - MITRE (2014). Un servidor configurado con un perfil cliente SSL/TLS puede ser vulnerable a un ataque de texto cifrado elegido contra conjuntos CBC (Cipher Block Chaining). Cuando se explota, puede resultar en la recuperación de texto sin formato de mensajes cifrados a través de un ataque MITM, a pesar de que el atacante no haya obtenido acceso a la clave privada del servidor.

GOLDENDOODLE se puede usar para secuestrar sesiones TLS autenticadas si el servidor revela la validez del relleno de los registros de datos de la aplicación, de tal manera que un atacante MITM pueda reconocer el relleno bien formado independientemente de un Código de Autenticación de Mensaje (MAC) válido (Young, 2019b).

La diferencia entre GOLDENDOODLE y Zombie POODLE o POODLE TLS es el rendimiento (Young, 2019a).

Recomendaciones de aplicación de suites de cifrados

Como se especifica en el RFC 8446 (Rescorla, 2018), se recomienda aplicar y configurar del lado del servidor TLS 1.3 y TLS 1.2. Es suficiente si se requiere la compatibilidad con clientes, cumpliendo con las directivas del servidor, configurando correctamente suites de cifrado sólidas, es importante realizar constantemente auditorías internas mediante sistemas de monitoreo de amenazas. Adicional a ello, se deben deshabilitar las versiones anteriores de TLS 1.2.

Hay varios parámetros obsoletos que deben ser evitados (Ristic, 2017):

- Los conjuntos Diffie-Hellman Anónymos (ADH) no proporcionan autenticación.
- Los conjuntos de cifrado NULL no proporcionan cifrado.
- Evitar el modo de cifrado TLS que usa cifrado RSA (TLS_RSA).
- Exportar conjuntos de cifrado es inseguro cuando se negocian en una conexión.

- Suites débiles (típicamente de 40 y 56 bits) usan cifrado que puede romperse fácilmente.
- RC4 es inseguro.
- 3DES es lento y débil.
- Evitar el uso de OpenSSL en versiones desactualizados.

Desactivar los suites de cifrado que aplican modo CBC en los certificados de seguridad de los servidores, empresas como IBM han encontrado ciertas vulnerabilidades (Merget et al., 2019).

Para la mayoría de los sitios web, la seguridad proporcionada por las claves RSA de 2048 bits es suficiente. En 2048 bits, estas claves proporcionan alrededor de 112 bits de seguridad (Centro Criptográfico Nacional, 2023). Si se desea más seguridad, hay que considerar que las claves RSA no se adaptan muy bien. Para obtener 128 bits de seguridad, es necesario claves RSA 3072 bits, provocando un proceso de cifrado de información más lento.

ECDSA proporcionan una alternativa que ofrece una mayor seguridad y un mejor rendimiento. Las claves ECDSA, en 256 bits, proveen 128 bits de seguridad (Centro Criptográfico Nacional, 2023). Un pequeño número de clientes más antiguos no son compatibles con ECDSA, a diferencia de los clientes modernos. Es posible conseguir lo mejor de ambos mundos y desplegar claves simultáneamente RSA y ECDSA, obviando la sobrecarga de administración de tal configuración.

Otro aspecto fundamental es confiar principalmente en las suites AEAD (en TLS 1.2 son los que usan los algoritmos AES-GCM y ChaCha20-Poly1305), las cuales proporcionan una autenticación fuerte, un intercambio de claves, un secreto avanzado y un cifrado de al menos 128 bits (Centro Criptográfico Nacional, 2023).

Estas especificaciones de cifrado admiten el intercambio de claves de curva elíptica efímera, el cifrado en modo AES-GCM y los algoritmos de integridad de mensajes basados en SHA-256 y SHA-384 (IBM, 2021).

Con base a estas recomendaciones, IETF exhorta a aplicar el uso de los siguientes conjuntos de cifrados del protocolo TLS 1.2 en los servidores como primera opción, (Tabla 3).

Tabla 3

Conjunto de Cifrados TLS 1.2 (Recomendación) (Sheffer et al., 2022)

Recomendación - Conjunto de Cifrados TLS 1.2	
Descripción	**Valor**
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	={} 0xC0,0x2B;
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	={} 0xC0,0x2C;
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	={} 0xC0,0x2F;
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	={} 0xC0,0x30;

Otra recomendación importante para los servidores que utilizan TLS para proteger el tráfico HTTP (es decir, HTTPS) esincluir el uso HSTS (HTTP Strict Transport Security), como se especifica en RFC 6797 (Hodges et al., 2012). Se debe aplicar esta especificación para mitigar muchas vulnerabilidades, de esta forma, se mejora notablemente la seguridad.

Con HSTS habilitado, los navegadores ya no permiten hacer clic en los errores de advertencia del certificado, que normalmente son triviales de explotar. Además, ya no enviarán solicitudes inseguras (texto sin formato), incluso si se les solicita.

HSTS define un mecanismo que permite a los sitios electrónicos declarar accesibles solo a través de conexiones seguras, impide que un atacante convierta una conexión HTTPS en HTTP. El mecanismo instruye al UA (user agent) finalizar la conexión con el servidor de destino, si ocurre algún error mientras intenta establecer comunicación mediante el protocolo (TLS) (Hodges et al., 2012).

La política HSTS define una nueva cabecera HTTP llamada 'Strict-Transport-Security' que puede ser enviada por un servidor web a sus clientes, con el fin de especificar una política con respecto a cómo el navegador va a manejar las conexiones futuras. Además, el host de un recurso web puede declarar que su política se aplica a todo el subárbol de nombres de dominio enraizado en su nombre de host. Esto permite que (HSTS) proteja las llamadas "cookies de dominio", que se aplican a todos los subdominios del nombre de host de un recurso web determinado.

Materiales y Métodos

La investigación adoptó un enfoque cuantitativo y descriptivo, ya que se analizó métricas de seguridad de los certificados digitales en los

dominios de las 18 entidades financieras que ofrecen servicios de banca electrónica personas, mediante la herramienta SSL Server Test, obteniendo datos esenciales y categóricos. Se aplicó el enfoque evaluativo, dado que los resultados fueron interpretados para determinar el nivel de seguridad de los sitios electrónicos.

Se evaluó un total de 15 bancos del sector privado y 3 del sector público. La selección de estas entidades se basó en su relevancia y representatividad en el mercado financiero actual.

SSL Server Test

Es una herramienta de test en línea desarrollada por Qualys SSL Labs (Qualys, 2015), un esfuerzo de investigación no comercial, su creador y director es Ivan Ristic. Este instrumento expone la funcionalidad completa del servidor sobre el uso del protocolo SSL/ TLS de forma inmediata, permitiendo la evaluación de la configuración del servidor a los administradores de los sitios web, de esta forma se pueden tomar los correctivos necesarios para mejorar la seguridad en la capa de transporte. Incluye un simulador de handshake, el cual realiza simulaciones entre los suites de cifrados soportados, de acuerdo a los protocolos permitidos por la configuración del lado del servidor, corroborando la compatibilidad de los diferentes navegadores con sus respectivas versiones del lado del cliente.

La elección de SSL Server Test se basó en la amplia aceptación de la comunidad de seguridad informática, su capacidad para realizar evaluaciones exhaustivas y detalladas, y su continua actualización de mejores prácticas. Los informes generados por SSLLabs proporcionaron una visión completa de la configuración SSL/TLS de cada sitio, permitiendo identificar fortalezas y debilidades.

Procesamiento de Datos

Los datos recolectados mediante la herramienta SSL Server Test fueron registrados en una hoja de cálculo y procesados utilizando tablas dinámicas, las cuales posibilitaron agrupar y filtrar la información de manera flexible, identificar patrones y tendencias en la configuración de los certificados digitales. Los datos útiles organizados permitieron modelar los resultados de manera gráfica.

Resultados y Discusión

En este trabajo se obtuvieron resultados importantes sobre la aplicación de los certificados digitales en los sitios web de banca electrónica (personas) del país, ello fue posible gracias al uso de la herramienta en línea SSL Server Test.

Para efecto de esta investigación, los bancos ecuatorianos objeto de este estudio se los clasificó de acuerdo a su calificación de riesgo de instituciones financieras, obtenidas en el sitio web de la Superintendencia de Bancos, con corte de septiembre del año 2022 (Superintendencia de Bancos, 2023). Siendo la mejor ponderación "AAA" y la menor "E".

SSL Server Test de SSLLabs también posee valoraciones de calificación, donde "A+" es la mayor y "F" es la menor para este caso de estudio.

Tabla 4

Calificación SSLLabs obtenida en el estudio realizado a los servicios de banca online

Clasificación de Riesgo Financiero	Calificación SSLLabs					Total
	F	C	B	A	A+	
-A		1			1	2
AA				1		1
AA+			1		3	4
AAA-	1		1	4	3	9
AAA				1	1	2
Total	1	1	2	6	8	18

La Tabla 4 muestra las calificaciones obtenidas por los bancos públicos y privadas del sistema financiero del Ecuador, en pocos casos se denotó cierta debilidad en la configuración de los conjuntos de cifrados de sus respectivas plataformas electrónicas.

El 80 % de las bancas virtuales en línea analizadas propiciaron una buena aplicación del uso del certificado de seguridad. Existieron dos entidades bancarias con la mejor calificación de riesgo financiero "AAA", una de ellas obtuvo calificación "A+" en el test de SSLLabs, como se manifestó anteriormente. Eso significa que posee una buena parametrización en la aplicación del conjunto de cifrados del protocolo TLS del lado del servidor y el restante se adjudicó "A". Tres entidades crediticias pertenecientes al grupo de riesgo financiero "AAA-" obtuvieron la mejor calificación SSLLabs "A+", determinando seguridad en su proceso de comunicación cliente/ servidor a través de la capa de transporte.

El 20% del total de las plataformas bancarias virtuales que se analizaron obtuviero un grado considerable de vulnerabilidad, esto se debe a la permisibilidad del uso de protocolos TLS/SSL considerados obsoletos, el intercambio de llaves, algoritmos y modos de encriptación débiles.

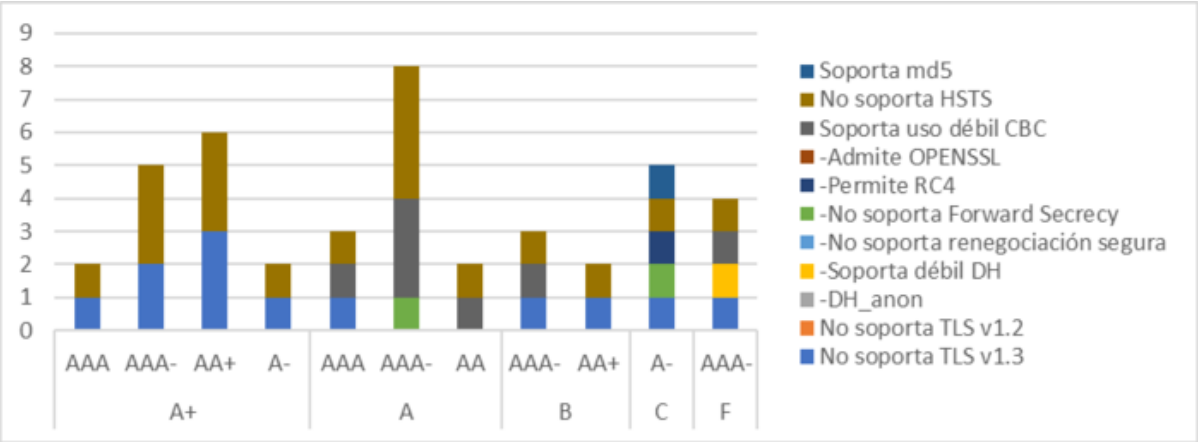

Figura 6

Debilidades en la aplicación de suite de cifrados de las diferentes bancas en línea

La Figura 6 muestra las debilidades que presentaron las entidades financieras en la aplicación de suites de cifrados en las respectivas configuraciones de los servidores. En algunos casos no soportaron el protocolo TLS en sus versiones 1.3 (actual) y 1.2 (versión aún. segura), muchos no aplican la política de HSTS, varios no soportan Forward Secrecy, otros usan conjuntos de cifrados con el modo CBC, Renegociación Segura, otras develaciones importantes, varias entidades permitieron el uso de OpenSSL, Diffie Helman y DH_anon, además soportaron el cifrado RC4.

La Figura 7 devela los ataques a los que fueron susceptibles ciertas entidades financieras ecuatorianas, que aplican débiles configuraciones en las suites de cifrados del lado del servidor. Ello es mucho más permisible cuando se utilizan sistemas operativos o navegadores web obsoletos, los que se muestran vulnerables ante cualquier amenaza de ataque, orquestado por algún ciberdelincuente u organización delictiva.

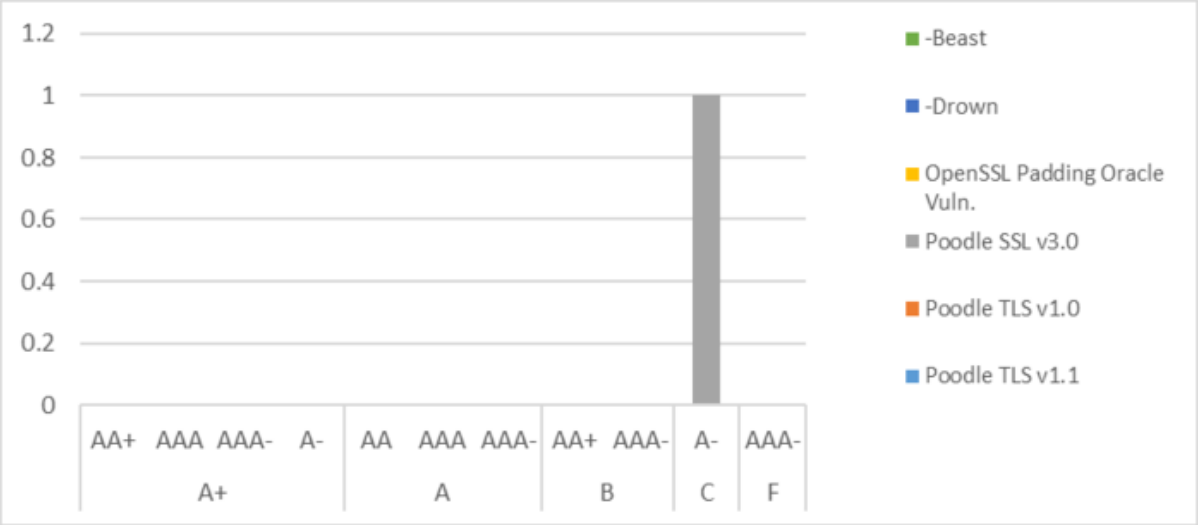

Figura 7

Ataque a los que están expuestos las plataformas financieras virtuales ecuatorianos

Algo interesante es que no necesariamente la calificación financiera del banco tiende a estar relacionada con la calificación de la evaluación SSL/TLS. Es así que, de las entidades con calificación AAA-, la segunda mejor del grupo analizado, se encontraron casos con la mejor y la peor evaluación de seguridad.

Conclusiones

Se concluye que el 80% de las entidades financieras objetos de esta investigación se muestran seguras en sus respectivas plataformas de banca electrónica personas, gracias a las configuraciones de fuertes conjuntos de cifrados del lado del servidor.

De acuerdo a la información obtenida en esta investigación, se determinó que existe cierta debilidad en la aplicabilidad de los certificados digitales del lado del servidor en varias de las entidades financieras del Ecuador. El 20% de las plataformas de banca virtual (personas), posee vulnerabilidades, es decir, que son propensos a sufrir ataques cibernéticos, a causa de las configuraciones débiles existentes en las suites de cifrados implementados.

El análisis de los datos obtenidos afirma que existen un grupo minúsculo de personas en el Ecuador que hacen uso de sistemas operativos y navegadores obsoletos, a ello, se suma que ciertas entidades financieras permiten el acceso a las plataformas de banca personas con estas herramientas desactualizadas por la aplicación de suites de cifrados débiles, lo cual, se convierte en una amenaza por parte de los ciberdelincuentes.

En el contexto de los servicios de banca electrónica, la seguridad en la capa de transporte es un pilar fundamental para proteger la confidencialidad, integridad y disponibilidad de las transacciones financieras. Por ello, es necesario la implementación de protocolos criptográficos robustos, mecanismos de autenticación multifactor y sistemas de detección de intrusos. La combinación de estos mecanismos de seguridad informática es esencial para mitigar las amenazas cibernéticas y garantizar la confianza de los clientes. Además, es crucial realizar evaluaciones de vulnerabilidad periódicas y mantener los sistemas actualizados para responder de manera proactiva a la evolución del panorama de amenazas en la red informática más insegura conocida como internet.

Sería importante complementar este estudio con un trabajo futuro, para analizar la seguridad en el proceso de autenticación, en las aplicaciones en los canales electrónicos de banca personas de las entidades financieras del país. Otra investigación podría ir enfocada a las políticas de seguridad que aplican estas entidades.

Reconocimientos

Los autores declaran la contribución y participación equitativa de roles de autoría para esta publicación.

Referencias

Advisory, S. (2014). This POODLE Bites : Exploiting The. *Google Security Blog*. https://security.googleblog.com/2014/10/this-poodle-bites-exploiting-ssl-30.html

Aviram, N., Schinzel, S., Somorovsky, J., Heninger, N., Dankel, M., Steube, J., Valenta, L., Adrian, D., Halderman, J. A., Dukhovni, V., Käsper, E., Cohney, S., Engels, S., Paar, C., & Shavitt, Y. (2016). Drown: Breaking TLS using SSLv2. *Proceedings of the 25th USENIX Security Symposium*, 689–706. https://drownattack.com/drown-attack-paper.pdf

Bleichenbacher, D. (1998). A Chosen Ciphertext Attack against Protocols based on the RSA Encryption Standard PKCS \textsf\symbol351. *Proc. of Crypto '98, 1462*, 1–12.

Blog de Internet Security Auditors: Seguridad SSL/TLS: LUCKY 13. (n.d.). Retrieved January 7, 2024, from https://blog.isecauditors.com/2020/04/seguridad-ssl-tls-lucky13.html

Böck, H., Somorovsky, J., & Young, C. (2017). Return Of Bleichenbacher's Oracle Threat (ROBOT). *Cryptology EPrint Archive*.

Centro Criptográfico Nacional. (2023). *Guía de Seguridad de las TIC CCN-STIC 221 Guía de Mecanismos Criptográficos autorizados por el CCN*. https://www.ccn-cert.cni.es/es/guias-de-acceso-publico-ccn-stic/6954-ccn-stic-221-guia-de-mecanismos-criptograficos-autorizados-por-el-ccn-1/file.html

Centro Criptológico Nacional. (2017). *Guía de Seguridad de las TIC CCN-STIC 811. Interconexión en el ENS*. https://www.ccn-cert.cni.es/es/series-ccn-stic/800-guia-esquema-nacional-de-seguridad/521-ccn-stic-811-interconexion-en-el-ens/file?format=html

Cisco. (2017). *SSL Introduction with Sample Transaction and Packet Exchange - Cisco*. 1–8. https://www.cisco.com/c/en/us/support/docs/security-vpn/secure-socket-layer-ssl/116181-technote-product-00.html#anc6

CVE - MITRE. (2014). *CVE Record | CVE*. CVE - MITRE. https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-6593

Dell, K. M., & Farrell, E. S. (2021). RFC 8996 Deprecating TLS 1.0 and TLS 1.1. *Internet Engineering Task Force, 3329*, 8422. https://www.rfc-editor.org/info/rfc8996

Dierks, T., & Allen, C. (1999). *The TLS Protocol Version 1.0* (Issue 2246). https://www.ietf.org/rfc/rfc2246.txt

Dierks T. and Rescorla E. (n.d.). *The Transport Layer Security (TLS) Protocol Version 1.2 [RFC 5246]*. Retrieved May 10, 2022, from https://www.rfc-editor.org/rfc/pdfrfc/rfc5246.txt.pdf

Factoring, F., & Export, R. S. A. (1990). *Vulnerabilidad Freak de SSL*. https://www.ecucert.gob.ec/wp-content/uploads/2021/07/Ficha-Tecnica-Freak-SSL.pdf

Hodges, J., Jackson, C., & Barth, A. (2012). HTTP Strict Transport Security (HSTS). In *IETF - Internet Engineering Task Force*. https://doi.org/10.17487/rfc6797

IBM. (2021). *Cipher suite considerations when upgrading to TLS V1.2 - IBM Documentation*. https://www.ibm.com/docs/en/zos/2.4.0?topic=protocols-cipher-suite-considerations-when-upgrading-tls-v12

Merget, R., Somorovsky, J., Aviram, N., Young, C., Fliegenschmidt, J., Schwenk, J., & Shavitt, Y. (2019). Scalable Scanning and Automatic Classification of TLS Padding Oracle Vulnerabilities. *USENIX Security Symposium*.

OpenSSL. (2016). *Memory corruption in the ASN.1 encoder (CVE-2016-2108)*. https://www.openssl.org/news/secadv/20160503.txt

Qualys, I. (2015). *SSL Server Test*. Projects. https://www.ssllabs.com/ssltest/

Rescorla, E. (2008). [ECC] TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM) [RFC 5289]. *RFC 5289*, 1–6. https://datatracker.ietf.org/doc/html/rfc5289

Rescorla, E. (2018). *The Transport Layer Security (TLS) Protocol Version 1.3 [RFC 8446]*. https://doi.org/https://doi.org/10.17487/rfc8446

Ristic, I. (2017). SSL and TLS Deployment Best Practices. *Wiki, 4*(December), 1–14. https://github.com/ssllabs/research/wiki/SSL-and-TLS-Deployment-Best-Practices

Ristić, I. (2014). *Bulletproof SSL and TLS: Understanding and Deploying SSL/TLS and PKI to Secure Servers and Web Applications* (Feisty Duck Limited (Ed.); Vol. 2015, Issue build 592). https://www.feistyduck.com/books/bulletproof-tls-and-pki/bulletproof-tls-and-pki-2ed-sample.pdf

Rizzo, J., & Duong, T. (2011). *The Beauty (RC4) and The BEAST (TLS) – HACKMAGEDDON*. Hakmagenon. https://www.hackmageddon.com/2011/09/25/the-beauty-rc4-and-the-beast-tls/

Sannegowda, Y. (2019). *Zombie POODLE and GOLDENDOODLE Vulnerabilities | Qualys Security Blog*. https://blog.qualys.com/product-tech/2019/04/22/zombie-poodle-and-goldendoodle-vulnerabilities

Sheffer, Y., Saint-Andre, P., & Fossati, T. (2022, November). *Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)*. https://doi.org/10.17487/RFC9325

SSL Pulse. (2019). *Qualys SSL Labs - SSL Pulse*. SSL Pulse. https://www.ssllabs.com/ssl-pulse/

Statcounter. (2023). Desktop Windows Version Market Share Ecuador | Statcounter Global Stats. In *Statcounter Global Stats*. https://gs.statcounter.com/windows-version-market-share/desktop/ecuador/#monthly-202209-202309-bar

SUPERINTENDENCIA DE BANCOS. (2023). *Calificación de Riesgo Instituciones Financieras*. Web Page. https://www.superbancos.gob.ec/bancos/calificacion-de-riesgo-instituciones-financieras-2022/

The Internet Engineering Task Force (IETF). (n.d.). *Introduction to the IETF*. Retrieved January 5, 2024, from https://www.ietf.org/about/introduction/

The Internet Engineering Task Force (IETF). (2004). *RFC 3749 - TLS Compression Methods*. https://www.rfc-editor.org/rfc/pdfrfc/rfc3749.txt.pdf

Vaudenay, S. (2002). Security flaws induced by cbc padding – Applications to SSL, IPSEC, WTLS. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2332*, 534–545. https://doi.org/https://doi.org/10.1007/3-540-46035-7_35

Young, C. (2019a). *Introducing Zombie POODLE and GOLDENDOODLE*. Tripwire. https://www.tripwire.com/state-of-security/zombie-poodle-goldendoodle

Young, C. (2019b). *"TripWire Vert, What is GOLDENDOODLE Attack?"*https://www.tripwire.com/state-of-security/goldendoodle-attack

David Peñarrieta, Marlon Navia, Eliana García, Dannyll Zambrano

Evaluación de la Seguridad de Certificados Digitales en las Plataformas Financieras de Ecuador
Assessment of the Security of Digital Certificates in the Financial Platforms in Ecuador

Revista Tecnológica ESPOL - RTE
vol. 36, no. 2, p. 174 - 189, 2024
Escuela Superior Politécnica del Litoral, Ecuador
rte@espol.edu.ec

ISSN: 0257-1749
ISSN-E: 1390-3659

DOI: https://doi.org/10.37815/rte.v36n2.1222