FACULTAD DE INGENIERÍA, ARQUITECTURA Y URBANISMO

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

TESIS

Evaluación de Algoritmos Asimétricos para mejorar la Seguridad de los Segmentos en la Capa de Transporte

PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS

Autor(es)

Bach. Campos Davila, Marcos Eduardo
ORCID: https://orcid.org/0000-0002-3747-0853
Bach. Castro Quesquen, Jaime Elton
ORCID: https://orcid.org/0000-0002-3507-8612

Asesor

Mg. Bravo Ruiz Jaime Arturo
ORCID: https://orcid.org/0000-0003-1929-3969

Línea de Investigación

Ciencias de la información como herramientas multidisciplinares y estrategias en el contesto industrial y de organizaciones

Sublínea de Investigación

Nuevas tendencias digitales orientadas al análisis y uso estratégico de la información

Pimentel – Perú

2024

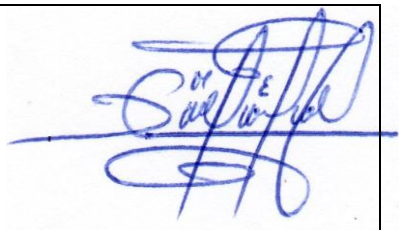
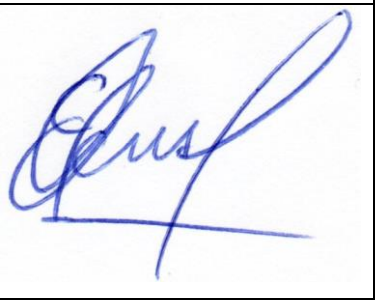
ANEXO 01: DECLARACIÓN JURADA DE ORIGINALIDAD

Quien(es) suscribe(n) la DECLARACIÓN JURADA, soy(somos) **Campos Dávila Marcos Eduardo, Castro Quesquén Jaime Elton, egresados.** Del Programa de Estudios de **Ingeniería de Sistemas** de la Universidad Señor de Sipán S.A.C, declaro (amos) bajo juramento que soy (somos) autor(es) del trabajo titulado:

EVALUACIÓN DE ALGORITMOS ASIMÉTRICOS PARA MEJORAR LA SEGURIDAD DE LOS SEGMENTOS EN LA CAPA DE TRANSPORTE

El texto de mi trabajo de investigación responde y respeta lo indicado en el Código de Ética del Comité Institucional de Ética en Investigación de la Universidad Señor de Sipán, conforme a los principios y lineamientos detallados en dicho documento, en relación con las citas y referencias bibliográficas, respetando el derecho de propiedad intelectual, por lo cual informo que la investigación cumple con ser inédito, original y autentico.

En virtud de lo antes mencionado, firman:

Campos Davila Marcos Eduardo	DNI: 47498197	
Castro Quesquen Jaime Elton	DNI: 45602729	

Pimentel, 19 de diciembre del 2023.

Reporte de similitud

NOMBRE DEL TRABAJO

Contenido_TESIS_MarcosCampos_Elton Castro_FINAL turnitin.docx

RECUENTO DE PALABRAS

8491 Words

RECUENTO DE CARACTERES

48111 Characters

RECUENTO DE PÁGINAS

35 Pages

TAMAÑO DEL ARCHIVO

1.4MB

FECHA DE ENTREGA

Sep 26, 2024 12:55 PM GMT-5

FECHA DEL INFORME

Sep 26, 2024 12:56 PM GMT-5

10% de similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos.

- 8% Base de datos de Internet
- Base de datos de Crossref
- 5% Base de datos de trabajos entregados
- 0% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

Excluir del Reporte de Similitud

- Material bibliográfico
- Coincidencia baja (menos de 8 palabras)
- Material citado

EVALUACIÓN DE ALGORITMOS ASIMÉTRICOS PARA MEJORAR LA SEGURIDAD DE LOS SEGMENTOS EN LA CAPA DE TRANSPORTE

Aprobación del jurado

MG. BANCES SAAVEDRA, DAVID ENRIQUE

Presidente del jurado de tesis

MG. GUEVARA ALBURQUEQUE, LAURITA BELÉN

Secretario del jurado de tesis

Mg. BRAVO RUIZ, JAIME ARTURO

Vocal del jurado de tesis

Dedicatoria

A Dios por la salud, por la vida, su orientación y vitalidad que me llevó a poder alcanzar uno de mis objetivos más deseados, a mi prometida Brengie e hijo Kaleb, a mis padres Marcos Campos y Marilú Davila por su amor incondicional, sus sacrificios y por enseñarme el valor de la educación y la perseverancia, de igual modo a mis hermanos Karin y Matius, por haberme apoyado continuamente y ser uno de mis motivos a para poder culminar este proyecto, a los docentes, por su dedicación y enseñanzas que han sido cruciales para mi desarrollo profesional y personal a ustedes les dedico mis logros y éxitos, ustedes son impulso para poder lograr mis metas.

Marcos Campos.

Dedico este logro a las personas que han sido mi mayor apoyo y fuente de inspiración durante esta significativa etapa de mi vida, y especialmente a Dios, por su infinita bondad, sabiduría y amor. Gracias por darme la fuerza, la paciencia y la perseverancia para superar cada obstáculo y alcanzar mis metas. A mis padres, Carmen y Jaime, por su amor incondicional, su paciencia y su fe en mis capacidades. Gracias por inculcarme el valor del esfuerzo y por ser mis pilares en cada etapa de mi vida. A mi hermano Andy, por siempre confiar en mí y ser un constante apoyo. También se lo dedico a quienes han contribuido a mi formación y desarrollo. Esta tesis refleja el esfuerzo conjunto y las enseñanzas valiosas que he recibido de cada uno.

Con todo mi agradecimiento y cariño.

Elton Castro.

Agradecimiento

Quiero manifestar mi agradecimiento a mi familia extendida, amigos cercanos y seres queridos por su continuo apoyo y comprensión. Cada palabra de aliento ha sido un impulso para superar los momentos difíciles.

También deseo agradecer a los docentes que, de diversas formas, han influido en el desarrollo de esta tesis. Su guía y conocimientos han enriquecido significativamente mi experiencia, proporcionando perspectivas valiosas y contribuyendo al crecimiento académico de esta investigación.

Índice

Índice de Figuras

Índice de tablas.

Evaluación de Algoritmos Asimétricos para mejorar la Seguridad de los Segmentos en la Capa de Transporte

Resumen

La investigación se centra en la evaluación de algoritmos asimétricos para mejorar la seguridad de los segmentos en la capa de transporte. En este contexto, la adopción de protocolos como SSL/TLS y HTTPS se destaca como una recomendación clave. Estos protocolos, al cifrar la información y autenticar la conexión cliente-servidor, añaden una capa adicional de seguridad esencial para salvaguardar la integridad y confidencialidad de los datos. Para alcanzar este propósito, se han delineado objetivos específicos que abarcan el análisis de la situación actual de las páginas web más visitadas, la identificación de los algoritmos operativos en la capa de transporte de cada página web, la simulación de un entorno de servidor local, la aplicación de los algoritmos identificados y, finalmente, la medición del rendimiento obtenido por dichos algoritmos. Se simuló un entorno de servidor local para aplicar los algoritmos identificados y medir su rendimiento en términos de confidencialidad, integridad y disponibilidad de los segmentos en la capa de transporte. Los resultados obtenidos indican que la aplicación de algoritmos asimétricos mejora efectivamente la seguridad en la capa de transporte. En conclusión, la investigación proporciona una comprensión profunda de cómo los algoritmos asimétricos pueden optimizar la seguridad en la capa de transporte, contribuyendo así al conocimiento y mejora de la ciberseguridad en entornos digitales. Se formularon recomendaciones específicas para mejorar la seguridad en las páginas web más visitadas y se destacó la necesidad de implementar medidas de seguridad más robustas para proteger la información transmitida a través de la capa de transporte.

Palabras Clave: Algoritmos asimétricos, Entidad Certificadora, capa de transporte, SSL/TLS, HTTP/HTTPS.

Abstract

The research focuses on the evaluation of asymmetric algorithms to improve segment security at the transport layer. In this context, the adoption of protocols such as SSL/TLS and HTTPS stands out as a key recommendation. These protocols, by encrypting information and authenticating the client-server connection, add an additional layer of security essential to safeguard data integrity and confidentiality. To achieve this purpose, specific objectives have been outlined, covering the analysis of the current situation of the most visited web pages, the identification of the algorithms operating at the transport layer of each web page, the simulation of a local server environment, the application of the identified algorithms and, finally, the measurement of the performance obtained by these algorithms. A local server environment was simulated to apply the identified algorithms and measure their performance in terms of confidentiality, integrity and availability of segments at the transport layer. The results obtained indicate that the application of asymmetric algorithms effectively improves security at the transport layer. In conclusion, the research provides an in-depth understanding of how asymmetric algorithms can optimize security at the transport layer, thus contributing to the knowledge and improvement of cybersecurity in digital environments. Specific recommendations were made to improve security in the most visited web pages and highlighted the need to implement more robust security measures to protect information transmitted through the transport layer.

Keywords: Asymmetric algorithms, Certificate Authority, transport layer, SSL/TLS, HTTP/HTTPS.

I. INTRODUCCIÓN

La protección de la información actualmente se ve amenazada por la constante evolución de riesgos cibernéticos, tales como malwares, virus y diversas tácticas maliciosas empleadas por individuos con intenciones perjudiciales [1]. Estas amenazas comprometen la seguridad de los datos al atentar contra su confidencialidad, integridad y autenticidad [2]. Frente a estos desafíos, se han desarrollado diversos enfoques destinados a contrarrestar las vulnerabilidades existentes y las amenazas potenciales que podrían acarrear consecuencias significativas. Estos métodos implican la utilización, creación y aplicación de algoritmos de cifrado con el propósito de asegurar la información y prevenir su manipulación por parte de individuos maliciosos. La evaluación de algoritmos asimétricos representa un elemento crucial en la búsqueda constante por fortalecer la parte de la seguridad de los segmentos en la capa transporte. En un mundo cada vez más digital y propenso a amenazas, la criptografía y la elección cuidadosa de algoritmos asimétricos son esenciales para proteger la integridad y confidencialidad de los datos transmitidos, destacando su papel crucial en la ciberseguridad [3]. Este proceso evaluativo no solo responde a las crecientes sofisticaciones en ciberseguridad, sino que también anticipa y contrarresta vulnerabilidades en la infraestructura de comunicación, asegurando una sólida defensa en la transmisión de datos. La criptografía, a través de protocolos de cifrado como SSL y TLS, protege diversos tipos de información digital, desde contraseñas, correos electrónicos hasta transacciones bancarias, páginas web y de comercio electrónico [3].

Los sitios web tienen varios tipos de vulnerabilidades, como la verificación incorrecta del usuario ingresado por el usuario, el script sin seguridad y el error en la configuración de la aplicación web [4]. Las vulnerabilidades de seguridad más frecuentes son la inyección de SQL y el script cruzado (cross-site scripting)[1]. Según la evaluación de seguridad realizada por el Centro de Defensa de la Aplicación en más de 250 aplicaciones de comercio electrónico, banca en línea y sitios corporativos, es probable que estos ataques se vean

afectados más del 85 % de las aplicaciones web [5].

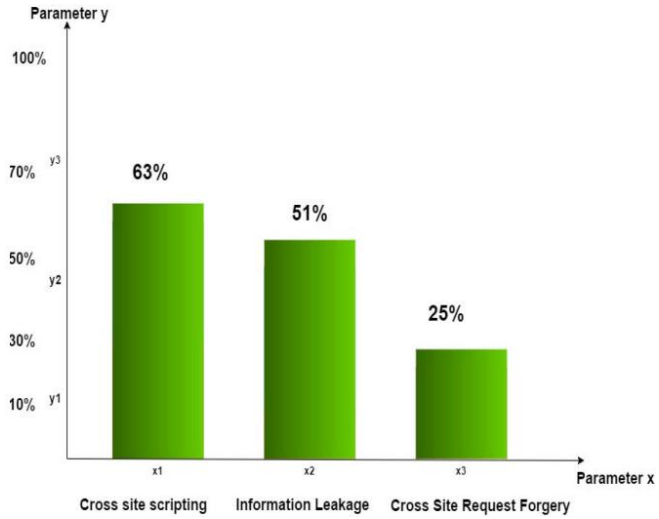


Fig. 1. Vulnerabilidades más desarrolladas en los sitios web.

Nota. Porcentaje de las vulnerabilidades más desarrolladas en las páginas web donde se tiene a cross site scripting con un porcentaje de 63%, information leakage con 51% y por último a cross site request forgery con un porcentaje de 25 % [5].

La vulnerabilidad se relaciona con fallas y debilidades de seguridad que representan una amenaza para los sistemas que pueden causar daños. Cada día, se encuentran numerosas vulnerabilidades, según una encuesta que revela el valor promedio mínimo de 30 a 40 vulnerabilidades diarias. Según los datos recopilados con información detallada de CVE (Common Vulnerabilities and Exposures), en 2017, se registraron 14,600 vulnerabilidades en comparación con 6.447 en 2016. Tan pronto como el atacante determina la amenaza o brecha potencial y determina su acceso, puede obtener acceso no autorizado al sistema. Los atacantes usan herramientas y tecnologías específicas para identificar y analizar estas debilidades de seguridad. La organización Open Web Application Security Project (OWASP) ofrece la vulnerabilidad más reciente y más importante relacionada con las aplicaciones web. La Tabla 1 muestra 10 vulnerabilidades básicas informadas por OWASP en 2017, junto con las correspondientes debilidades comunes de las debilidades (CWE) [6].

Vulnerability	CWE	Rank
Injection	CWE-1027	A1
Broken Authentication & Session Management	CWE-1028	A2
Sensitive Data Exposure	CWE-1029	A3
XML External Entities	CWE-1030	A4
Broken Access Control	CWE-1031	A5
Security Misconfiguration	CWE-1032	A6
Cross-Site Scripting (XSS)	CWE-1033	A7
Insecure Deserialization	CWE-1034	A8
Using Components with known vulnerabilities	CWE-1035	A9
Insufficient Logging & Monitoring	CWE-1036	A10

Fig. 2. Top 10 de vulnerabilidades del 2017

Nota. Como se describe en OWASP, donde muestra las 10 principales vulnerabilidades con su enumeración común de debilidades (CWE) [6].

En el Caribe y América Latina han visto una gran cantidad de ataques cibernéticos, según varios informes de seguridad cibernética. Según las estadísticas, cada segundo ocurre más de 1.600 ciberataques en la región. En la primera mitad de 2022, hubo 384 000 ataques de ransomware en todo el mundo, y esta región representó el 14 % del total. Existe una clara correlación entre el tamaño económico y la digitalización y el número de ciberataques. Por ejemplo, Brasil fue el país que experimentó un poco más de la mitad de todos los ciberataques, después esta México (23 %), seguido Colombia (8 %) y finalmente Perú (6 %) [7]. En los últimos años, los sitios web han sufrido cada vez más ataques. El Informe de amenazas a la seguridad de Internet de 2016 encontró que, en 2015, el 78% de los sitios web tenían vulnerabilidades que abarcaban errores, problemas del navegador y plugins defectuosos, que pueden ser aprovechados por personas malintencionadas para realizar ataques web [8].

La seguridad que es un criterio fundamental para los sitios web, ya que el riesgo de ataques aumenta cuando el sistema no está actualizado y la comunicación no se realiza a través de canales cifrados. Los sitios web que son vulnerables a ataques como la inyección

SQL y el cross-site scripting pueden experimentar interrupciones en el servicio, pérdida de datos e incluso volverse completamente inoperativos [9].

Para asegurar la confiabilidad y seguridad de la información, es fundamental emplear algoritmos de cifrado. Además, es necesario salvaguardar la privacidad e integridad de la información compartida entre los usuarios de la red mediante diversas formas de codificación, autenticación y encriptación. En este contexto, la ciberseguridad desempeña un papel crucial al iniciar cualquier comunicación digital. Es por ello que se debe velar por la protección de la identidad y los datos. Se han creado diversos tipos de algoritmos de cifrado, todos orientados a proteger una o varias facetas de la seguridad de la información, tales como integridad, confidencialidad, autenticación y no repudio. Evaluar la efectividad de cada algoritmo resulta complicado, ya que pueden presentar vulnerabilidades frente a ataques cibernéticos. La mayoría de estos algoritmos utilizan claves de diferentes longitudes. Por ejemplo, algunos recurren a tácticas engañosas, como el envío de imágenes falsas para hacerlas pasar como reales, un tipo de engaño conocido como malware. Existe un riesgo significativo de que las personas confíen en estos archivos y los descarguen en sus dispositivos móviles o computadoras, facilitando así la entrada del malware [10].

El estudio de la criptografía, que se basa en las matemáticas, este se encarga de asegurar la privacidad, seguridad de la información y evitar que un extraño o adversario conozca el contenido secreto de la información [11]. El papel crucial de los algoritmos criptográficos radica en salvaguardar la información y datos del usuario. Se han empleado muchos algoritmos de cifrado, ya sea simétricos o también asimétricos, con el fin de salvaguardar los datos. No obstante, estos algoritmos demandan una considerable cantidad de recursos informáticos, como tiempo de procesamiento, consumo de batería y uso de memoria [12].

Cuando se realiza cualquier tipo de transacción o se envía información por la WWW, la principal preocupación es la seguridad de la información que se está compartiendo. La idea de que alguien pueda interceptar los datos y utilizarlos de manera malintencionada es una preocupación constante. Debido a esto, la criptografía ha surgido como una herramienta fundamental para proteger la información en línea. Aunque la criptografía es una solución efectiva, su implementación requiere de muchos recursos técnicos, humanos y económicos para garantizar la seguridad de la información en una organización [2].

En el ámbito actual de la comunicación en la capa de transporte, surge una inquietud fundamental relacionada con la eficiencia de los algoritmos asimétricos destinados a salvaguardar la seguridad de los segmentos de datos. El Protocolo de Seguridad de la Capa de Transporte (TLS) representa la principal solución para establecer canales seguros en la World Wide Web. Con más de dos décadas de desarrollo, TLS ha experimentado un notable crecimiento, incorporando una amplia variedad de métodos de cifrado [13]. En el contexto de estos requisitos específicos, los protocolos de seguridad TLS (Transport Layer Security) y SSL (Secure Socket Layer) se presentan como sistemas criptográficos que funcionan en una capa subyacente a la de la aplicación. Estos protocolos se encargan de implementar un cifrado integral de extremo a extremo, contribuyendo así a fortalecer la seguridad de diversos protocolos, entre los que se destaca HTTPS [14].

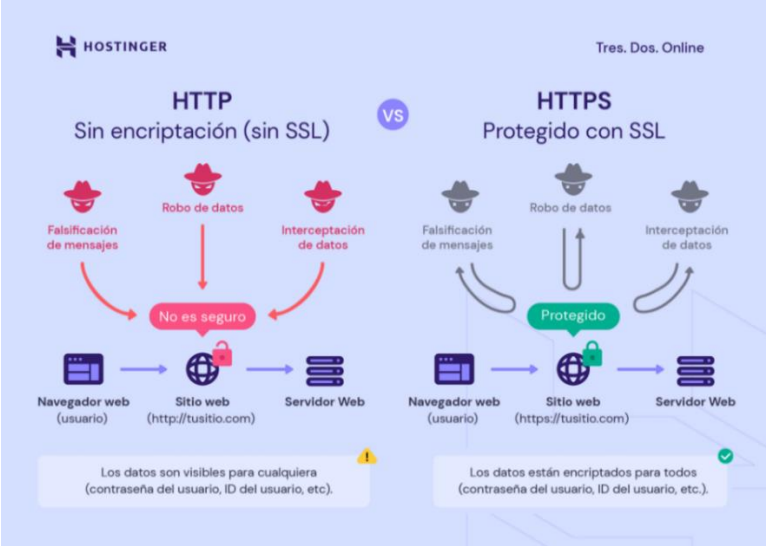


Fig. 3. Diferencia entre TLS y HTTPS

Nota. HTTP se utiliza para comunicar y transferir datos entre aplicaciones. HTTPS es su versión segura con cifrado, por lo que la mayoría de los navegadores modernos marcan ahora los sitios HTTP como no seguros. [15].

Se examina la seguridad en la transmisión de datos en servidores de aplicaciones, centrándose en certificados digitales SSL/TLS gratuitos. Su estudio destaca el creciente uso de Internet y la necesidad de evaluar la efectividad de los certificados digitales en entornos web. Utilizando Kali Linux v2.0, identifican vulnerabilidades en servidores de la Universidad Nacional de Loja, enfocándose en amenazas como ataques de intermediarios y denegación de servicio (DoS). Proponen Let's Encrypt como la Autoridad Certificadora (AC) ideal para certificados gratuitos, detallando su implementación en servidores Apache y Nginx. Se enfatiza la importancia de características clave como cifrado robusto y actualizaciones periódicas. Aunque reconocen las limitaciones en ataques DoS y fuerza bruta, subrayan la eficacia contra amenazas como el phishing [1]. Este estudio proporciona una guía valiosa para implementaciones seguras en contextos específicos, destacando la idoneidad de certificados gratuitos en ciertos escenarios. Además, resalta que el 70% de las amenazas identificadas fueron mitigadas mediante la implementación de certificados digitales SSL/TLS gratuitos [14].

En el campo de la criptografía y la puesta en marcha de sistemas de protección a través de protocolos de seguridad como HTTPS (SSL/TLS), se han llevado a cabo numerosos estudios y publicaciones, entre los que sobresalen los siguientes:

La criptografía de curva elíptica (ECC) está emergiendo como una alternativa atractiva a los criptosistemas tradicionales de clave pública como RSA, DSA y DH. ECC proporciona el mismo nivel de seguridad utilizando claves más pequeñas, lo que significa un cálculo más rápido, un menor consumo de energía y ahorros en memoria y ancho de banda. Estas características hacen que ECC sea particularmente atractivo para dispositivos móviles y también pueden reducir la carga computacional en servidores web seguros. Este artículo examina el impacto en el rendimiento del uso de ECC con SSL, el principal protocolo de seguridad en Internet. Se desarrolló y utilizó una versión mejorada de OpenSSL compatible con ECC para evaluar el rendimiento del servidor web Apache. Los resultados obtenidos muestran que, en condiciones reales de funcionamiento, un servidor web Apache puede procesar entre un 13% y un 31% más de solicitudes HTTPS por segundo utilizando ECC-160 que RSA-1024, lo que refleja el nivel de seguridad a corto plazo. En los niveles de seguridad necesarios para proteger los datos posteriores a 2010, el uso de ECC-224 en lugar de RSA-2048 puede mejorar el rendimiento del servidor entre un 120% y un 279%. En resumen, combinar ECC con SSL puede mejorar significativamente el rendimiento de su servidor web. Estos resultados sugieren que ECC puede ser una opción atractiva para proporcionar comunicaciones de Internet sin comprometer el rendimiento del servidor [16].

Este estudio tiene como objetivo abordar la sobrecarga común en los servidores SSL/TLS, provocada por múltiples solicitudes simultáneas o posibles ataques de Denegación de Servicio (DoS), que resultan en una disminución del rendimiento. Ante la necesidad de utilizar más hardware para mantener tiempos de respuesta aceptables, se plantea la alternativa de buscar algoritmos más eficientes en lugar de expandir la infraestructura. El enfoque se centra en tres algoritmos del protocolo de saludo de SSL/TLS para mejorar el

rendimiento sin sacrificar la seguridad ni requerir inversiones significativas en hardware. Se realiza un exhaustivo análisis de estas soluciones, detallando su rendimiento y seguridad, con especial atención a la comparación de desventajas entre los algoritmos. Destaca la técnica RSA asistida por el cliente, que, a pesar de aumentar el ancho de banda y la carga de memoria para el cliente, ofrece el mejor rendimiento al trasladar parte de la computación criptográfica a este. En conclusión, este estudio propone una estrategia efectiva para optimizar el rendimiento de SSL/TLS, superando las limitaciones tradicionales sin comprometer la seguridad del sistema ni requerir costosos incrementos de hardware [17].

En este análisis, se investiga el protocolo handshake de la versión 1.3 del protocolo de seguridad Transport Layer Security (TLS). Se aborda exhaustivamente tanto el protocolo completo TLS 1.3, que incluye el modo de un viaje de ida y vuelta con firmas para autenticación e intercambio de claves (curva elíptica) Diffie-Hellman efímera ((EC)DHE)), como el modo abreviado de reanudación/"PSK", que utiliza una clave pre-compartida para autenticación (con intercambio opcional de claves (EC)DHE y establecimiento de clave con tiempo de ida y vuelta cero). Este análisis, enmarcado en la seguridad reduccionista, emplea un modelo de seguridad de intercambio de claves multietapa, etiquetando cada clave de sesión derivada con propiedades como autenticación unilateral o mutua, seguridad hacia adelante y protección contra ataques de repetición. Los resultados demuestran que los modos de protocolo TLS 1.3 establecen claves de sesión con las propiedades de seguridad deseadas bajo supuestos criptográficos estándar, confirmando la eficacia y seguridad del protocolo en términos de autenticación y protección contra amenazas específicas. Estos hallazgos respaldan la confiabilidad de TLS 1.3, fortaleciendo la seguridad en el intercambio de información en entornos que implementan este protocolo [18].

Este estudio tiene como objetivo analizar el rendimiento de SSL/TLS en el ámbito de Internet, la red de datos pública más extensa, abordando su función esencial en la seguridad de la comunicación de datos en aplicaciones y computadoras remotas frente a crecientes

amenazas corporativas. La metodología se enfoca en el conjunto de protocolos TCP/IP, destacando su colaboración con SSL/TLS para brindar una seguridad robusta. La evaluación del rendimiento considera diversos protocolos, centrándose especialmente en el comportamiento limitante de SSL/TLS dentro del marco de TCP/IP. Los resultados subrayan la eficacia de SSL/TLS en proporcionar seguridad sólida y revelan detalles valiosos sobre su integración con TCP/IP. En conclusión, se destaca la importancia crítica de SSL/TLS en la seguridad de la comunicación en Internet, señalando la necesidad de futuras extensiones para fortalecer la seguridad en este contexto [19].

La presente investigación se centra en abordar la pregunta fundamental de cómo los algoritmos asimétricos pueden contribuir a potenciar la seguridad de los segmentos dentro de la capa transporte. La hipótesis propuesta sostiene que la aplicación de algoritmos asimétricos efectivamente mejorará la seguridad en estos segmentos. Teniendo como objetivo general de la investigación, es evaluar los algoritmos asimétricos para la mejora de la seguridad de los segmentos en la capa de transporte. Para alcanzar el objetivo, se han delineado los objetivos específicos que abarcan en analizar la situación actual de las páginas web más visitadas, la identificación de los algoritmos operativos en la capa de transporte de cada página web, la simulación de un entorno de servidor local, la aplicación de los algoritmos identificados y, finalmente, la medición del rendimiento obtenido por dichos algoritmos. Este estudio aspira a proporcionar una comprensión profunda de cómo los algoritmos asimétricos pueden optimizar la seguridad en la capa de transporte, contribuyendo así al conocimiento y mejora de la ciberseguridad en entornos digitales.

La investigación propuesta se caracteriza por ser de naturaleza cuantitativa. Se recopilan datos numéricos para su posterior procesamiento y análisis, estableciendo conexiones con las variables independientes y dependientes. La investigación se realiza de acuerdo al método científico para probar hipótesis de modo que el resultado obtenido tenga una aplicación práctica que pueda ser utilizada para resolver problemas o sugerir mejoras,

ayudar a las organizaciones y personas a interactuar con los sistemas de servicios en línea [20]. Como diseño de investigación es Cuasi-Experimental, porque se elegirá los datos y el escenario a utilizar. En el diseño cuasiexperimental, los grupos de investigación se constituyen previamente a la realización del experimento, y se llevan a cabo comparaciones de respuestas [21].

La investigación destaca la importancia de la protección de la información frente a la evolución de riesgos cibernéticos como malware, virus y otras tácticas maliciosas [1]. Se menciona la relevancia de los algoritmos de cifrado, específicamente los asimétricos, para mejorar la seguridad de los datos en la capa de transporte. Se subraya que estos algoritmos son esenciales para proteger la integridad y confidencialidad de la información transmitida, haciendo énfasis en la necesidad de contrarrestar vulnerabilidades y asegurar una defensa sólida en la transmisión de datos

II. MATERIALES Y MÉTODO

Para llevar a cabo la evaluación de algoritmos asimétricos en la capa de transporte de las páginas web seleccionadas, este estudio se concentra en analizar los algoritmos asimétricos implementados en la capa de transporte de las 25 páginas web más influyentes a nivel mundial, identificadas mediante datos de fuentes confiables como SimilarWeb. La selección de estas páginas se basa en su destacada posición en visitas globales y promedio de duración de visita a cada página, abarcando diversos sectores, desde redes sociales hasta comercio electrónico y noticias, ofreciendo un conjunto diverso en contenido y usuarios. La muestra para el estudio de la investigación son los algoritmos asimétricos, y la población son aquellas páginas web que incorporan algoritmos de cifrado asimétricos en el marco del protocolo SSL/TLS. Esta selección posibilitará un análisis detallado de prácticas de seguridad, evaluando la efectividad y robustez de los algoritmos aplicados en la capa de transporte, garantizando coherencia y representatividad para una evaluación exhaustiva. (Anexo 01). El tipo de investigación es cuantitativa, y el diseño de la investigación es cuasi-experimental. Se utilizará la técnica de observación para recopilar los datos necesarios y comprender el funcionamiento y la efectividad de los algoritmos asimétricos. La estrategia consiste en observar y registrar sucesos específicos durante las pruebas para recopilar y analizar datos de manera sistemática. Los instrumentos empleados incluirán el análisis documental, que se utiliza para registrar los eventos que ocurren durante la implementación y prueba de los algoritmos, y un sniffer de software, que permitirá capturar y analizar los paquetes de datos transmitidos en la red. Estos métodos y herramientas asegurarán una evaluación exhaustiva y precisa de la seguridad en la capa de transporte.

Para la identificación de los algoritmos de cifrado en la capa de transporte de cada página web, se utilizó una desktop Intel(R) Core (TM) i7-2600 CPU @ 3.40GHz 16GB de RAM, y SSD de 250 GB, ejecutando Windows 10 Pro, para generar máquinas virtuales (MV). Seguidamente se habilitaron las características de Hyper-V y la plataforma de máquina virtual, terminado las configuraciones debidas para el buen funcionamiento de las MV, se continuo

con la creación de la MV con sistema operativo (SO) Ubuntu 22.04.3 LTS, terminado la instalación del SO de la MV se procedió actualizar los parches de seguridad e instalación de librerías a la última versión. Dentro de la MV se utilizó las librerías de OpenSSL a través de terminal. Para obtener la identificación de los algoritmos de cada página se ingresó la línea de comando “openssl s_client -connect www.paginaweb.com:443”, donde arroja una serie de resultados en la cual existe un apartado del cifrado del algoritmo (Cipher is) de cada página web que nos permite identificar la combinación que utiliza dicha página web, esto nos servirá para poder identificar la combinación más utilizada por las 25 páginas web analizadas, ya identificada todas las combinaciones de cada página web pudimos elegir la siguiente combinación **ECDHE-RSA-AES256-GCM-SHA384**, por lo que cuenta con algoritmos asimétricos y se encaja con la investigación que se está realizando. (Anexo 02)

Teniendo ya identificado la combinación de los algoritmos de cifrado asimétrico se continua con la simulación del entorno local, para ello se utilizó una desktop Intel® Core i7-4790 CPU – 3.60GHz, 16GB RAM, 01 SSD de 500GB, 02 HDD de 1TB, con sistema operativo Windows 10 Pro ejecutándose en el disco SSD. Seguidamente se activó la virtualización dentro de la configuración de la BIOS, para que las máquinas virtuales puedan ejecutarse sin problemas ni errores, terminada la activación previa se continuo con la instalar del software VMware Workstation Pro V17.5, que nos permitirá generar máquinas virtuales en una red local. A continuación, se levanta la primera máquina virtual (MV01) con SO. Windows 10 Pro, 2GB de RAM, 80GB HDD, 2 núcleos de Procesador con nombre **AASIMETRICOS**. Esta MV01 se ejecutará los servicios de IIS que desempeña un papel vital en la alojamiento y administración de sitios web. Su funcionalidad abarca la gestión de solicitudes HTTP, el procesamiento de páginas web dinámicas y la entrega de contenido estático. Una de sus características destacadas es su integración con la emisión y gestión de certificados de seguridad. Los certificados emitidos por una Autoridad Certificadora (CA) y utilizados en un servidor IIS son esenciales para establecer conexiones seguras mediante el protocolo HTTPS. Estos certificados aseguran la autenticidad y la integridad de la comunicación entre

cliente-servidor, garantizando que la información transmitida esté cifrada y protegida contra accesos no autorizados. El servidor de IIS no solo proporciona capacidades de alojamiento web avanzadas, sino que también colabora estrechamente con certificados de seguridad para asegurar conexiones cifradas y confiables en entornos web. Para poner en marcha el servicio de IIS, se realizó las siguientes configuraciones: (Anexo 03)

- Se habilito en la característica de Windows 10 Pro, el servicio de Internet Information Services (IIS) dentro de las cuales se tomó las opciones de herramientas de administración web y servicios de la World Wide Web (WWW), de igual modo se tiene en cuenta selecciona las características de desarrollo de aplicaciones, características de rendimiento, características HTTP comunes, estado y diagnóstico y por último seguridad, todas estas opciones nos servirían para el funcionamiento del servicio IIS.
- Se continuo con la configuración de habilitar un DNS de manera local en la siguiente ruta C:\Windows\System32\drivers\etc, donde se encuentra el archivo Hosts para poder configurar colocando el IP 192.168.174.129 del mismo equipo, junto con el nombre dns aasimetrico.com.
- A continuación, se sigue con la publicación de la página web donde se instalará el certificado con los algoritmos, para ello esta página se aloja en la siguiente ruta C:\inetpub\wwwroot, con el nombre de index.html, para hacer el llamado desde el IIS, previo a ello se eliminó todos los archivos dentro de la carpeta wwwroot.

Para la segunda máquina virtual (MV02) con SO Windows Server 2022 Standard, con 2GB de RAM, 80GB HDD, 2 núcleos de Procesador, con nombre SrvCertificador, esta MV02 tendría los servicios de Autoridad Certificadora (CA). La Autoridad Certificadora (CA) se integra estrechamente con el protocolo SSL/TLS y el estándar HTTPS para garantizar la seguridad en las comunicaciones en línea. Dentro del protocolo SSL/TLS, la CA desempeña

un papel central al emitir certificados digitales, utilizados por servidores web para autenticar su identidad ante los clientes. Estos certificados son esenciales para establecer conexiones seguras mediante el cifrado de extremo a extremo. Operando en una jerarquía, con CAs raíces y subordinadas, la CA firma digitalmente los certificados, asegurando su autenticidad e integridad. Cuando un usuario accede a un sitio web mediante HTTPS, el navegador verifica la autenticidad del certificado del servidor a través de la cadena de confianza de la CA. Este proceso permite a los usuarios confiar en la conexión segura y en la identidad del servidor. Además, la CA, al seguir estándares de seguridad y prácticas recomendadas, contribuye a la robustez del ecosistema SSL/TLS y HTTPS. La confianza en la CA, respaldada por la inclusión en listas de confianza de navegadores y sistemas operativos, es esencial para el correcto funcionamiento de la seguridad digital, proporcionando una base sólida para la autenticación y la privacidad en la comunicación web. En resumen, la CA se entrelaza con SSL/TLS y HTTPS para respaldar la autenticidad, integridad y seguridad de las transmisiones de los segmentos de datos en línea. Para llegar a generar un certificado de confianza para las páginas web se tuvo que realizar las siguientes configuraciones dentro de la MV01 - AASIMETRICOS: (Anexo 04)

- Agregar roles y características, dentro de los roles seleccionamos los servicios de certificados de active directory de igual modo los servicios de web (IIS).
- Dentro de los servicios de certificados de active directory seleccionamos los roles de entidad de certificación e inscripción web de entidad de certificación, procediendo a la instalación.
- A continuación, se realiza la configuración de servicios de certificados de active directory de los roles instalados.
- Luego se seleccionan los servicios de los roles que se configuraran que vienen hacer la entidad de certificación e inscripción web de entidad de certificación.
- Siguiendo con la configuración indicamos el tipo de instalación para la CA, en

nuestro caso hemos seleccionado CA independiente.

- Seguidamente especificamos el tipo de CA que vamos a instalar, para ello seleccionamos la CA Raíz ya que la que nos permite configurar una jerarquía de PKI (estructura de clave publica)
- Una vez selecciona el tipo de CA, creamos una clave privada nueva para el certificado.
- Llegando al punto siguiente ya podremos especificar las opciones criptográficas, seleccionando un proveedor de servicios criptográficos el cual se eligió RSA#Microsoft Software Key Storage Provider y el algoritmo Hash, SHA384 para la firma de los certificados emitidos por la CA. Se selecciono dicho proveedor por qué es lo que más se asemeja a la combinación que se ha elegido para su evaluación y análisis.

Para la aplicación de los algoritmos previamente identificados, se crea una solicitud de certificado dentro de la MV01, desde el IIS. Dentro de los servicios del IIS existe una opción de certificados de servidor, ingresando en aquella opción nos dirigimos a crear una solicitud de certificado para poder enviar al servidor CA, dentro de la solicitud se debe ingresar los datos solicitados, se debe elegir el mismo proveedor criptográficos con los mismos algoritmos y por último debemos elegir la ruta donde guardar la solitud que tendrá como extensión csr, con el nombre de aasimetricos.csr. (Anexo 05)

En la MV02 abrimos la entidad certificadora para enviar una nueva solicitud, esta solicitud se creó anteriormente en la MV01. Una vez que se crea la solicitud esta se aloja en las solicitudes pendientes, para luego poder emitirla y de esa forma ya tener el certificado listo para su instalación en las páginas web. Teniendo ya el certificado listo para su utilización, regresamos nuevamente a la MV01. (Anexo 06)

Estando en la MV01, ingresamos al servidor de IIS, para continuar y completar la solicitud, esto lo realizamos ingresando a la opción de certificados de servidor, ingresando a

las acciones de completar solitud de certificado, se debe especificar la ruta donde está la solitud de certificado (aasimetricos.csr) que se creó en la MV02, ya seleccionando la solitud se procede a crear el certificado tiene como nombre aasimetricos.cer. (Anexo 07)

Una vez ya creado el certificado, no dirigimos a nuestro sitio web que se aloja en el servidor IIS, para poder crear un nuevo enlace con el protocolo HTTPS, con puerto 443 e indicarle el certificado (aasimetricos.cer) que debe estar utilizando para asegurar la autenticidad y la integridad de la comunicación entre cliente-servidor, garantizando que la información transmitida esté cifrada y protegida contra accesos no autorizados. (Anexo 08)

Para poder medir los algoritmos que se identificaron en los 25 sitios web más visitas se tuvo que tener en cuentas las variables dependientes e independientes. Los indicadores que integran la variable independiente son cantidad de bytes entre clientes y servidor (HANDSHAKE), el Periodo de Procesamiento, la Compatibilidad de Navegadores actuales, el periodo computacional, segmentos con cifrado y segmentos sin cifrado. Para la Variable Dependiente, que es la Seguridad de los Segmentos en la capa de transporte, a través de indicadores clave como la confidencialidad, integridad y disponibilidad. Este enfoque integral permitirá una evaluación exhaustiva de cómo los algoritmos asimétricos seleccionados influyen directamente en la protección de datos durante la transmisión en la capa de transporte en entornos locales. Los resultados obtenidos ofrecerán una visión detallada de su rendimiento y su impacto en la seguridad, contribuyendo así a una comprensión más profunda y a posibles mejoras en la protección de datos en este contexto específico.

Los indicadores segmentos con cifrado y segmentos sin cifrado, se medirán a través de la página creada en el servidor IIS, a través del software Wireshark, capturando tráfico que nos permitirá obtener y observar los datos con cifrado y sin cifrado. Para poder medir todos los indicadores antes mencionados se tiene que seguir algunas técnicas e instrumentos que serían sniffer, que permite capturas datos trasmitidos, observación, esta técnica se utiliza para recopilar los datos que se obtendrán y comprender los algoritmos asimétricos. La estrategia consiste en observar sucesos para recopilar datos y analizarlos. Y por último se

tendría en cuenta el análisis documental, emplea para registrar los sucesos que tienen lugar durante la aplicación de las técnicas de prueba.

III. RESULTADOS Y DISCUSIÓN

3.1 Resultados

3.1.1. Conjunto de algoritmos criptográficos más prevalentes entre los 25 sitios web frecuentes

TABLA I
COMBINACIÓN DE ALGORITMOS DE LAS 25 PÁGINAS MÁS VISITADAS

Protocolo TCP/IP	Combinación TLS	Algoritmo Asimétrico	Página Web	Puerto
https	TLS_AES_256_GCM_SHA384	ECDSA	google.com	443
https	TLS_AES_256_GCM_SHA384	ECDSA	youtube.com	443
https	TLS_CHACHA20_POLY1305_SHA256	ECDSA	facebook.com	443
https	TLS_CHACHA20_POLY1305_SHA256	RSA-PSS	instagram.com	443
https	TLS_AES_256_GCM_SHA384	RSA-PSS	twitter.com	443
https	ECDHE-RSA-AES128-GCM-SHA256	RSA-PSS	baidu.com	443
https	TLS_AES_256_GCM_SHA384	ECDSA	wikipedia.org	443
https	TLS_AES_128_GCM_SHA256	ECDSA	Yahoo.com	443
https	TLS_AES_256_GCM_SHA384	ECDSA	yandex.ru	443
https	TLS_CHACHA20_POLY1305_SHA256	ECDSA	whatsapp.com	443
https	ECDHE-RSA-AES128-GCM-SHA256	RSA	xvideos.com	443
https	TLS_AES_128_GCM_SHA256	RSA-PSS	amazon.com	443
https	TLS_AES_256_GCM_SHA384	ECDSA	tiktok.com	443
https	TLS_AES_128_GCM_SHA256	ECDSA	pornhub.com	443
https	ECDHE-RSA-AES128-GCM-SHA256	RSA	xnxx.com	443
https	ECDHE-RSA-AES256-GCM-SHA384	RSA-PSS	live.com	443
https	TLS_AES_128_GCM_SHA256	RSA-PSS	yahoo.co.jp	443
https	TLS_AES_128_GCM_SHA256	RSA-PSS	reddit.com	443
https	TLS_AES_256_GCM_SHA384	RSA-PSS	docomo.ne.jp	443
https	ECDHE-RSA-AES256-GCM-SHA384	RSA-PSS	linkedin.com	443
https	ECDHE-RSA-AES256-GCM-SHA384	RSA-PSS	openai.com	443
https	ECDHE-RSA-AES256-GCM-SHA384	RSA-PSS	office.com	443
https	TLS_AES_256_GCM_SHA384	RSA-PSS	xhamster.com	443
https	TLS_AES_256_GCM_SHA384	ECDSA	netflix.com	443
https	ECDHE-RSA-AES128-GCM-SHA256	RSA-PSS	dzen.ru	443

Nota. La tabla detalla los algoritmos de seguridad utilizados por las 25 páginas web más visitadas en sus conexiones HTTPS. Se indican las combinaciones de cifrado TLS, través del puerto 443.

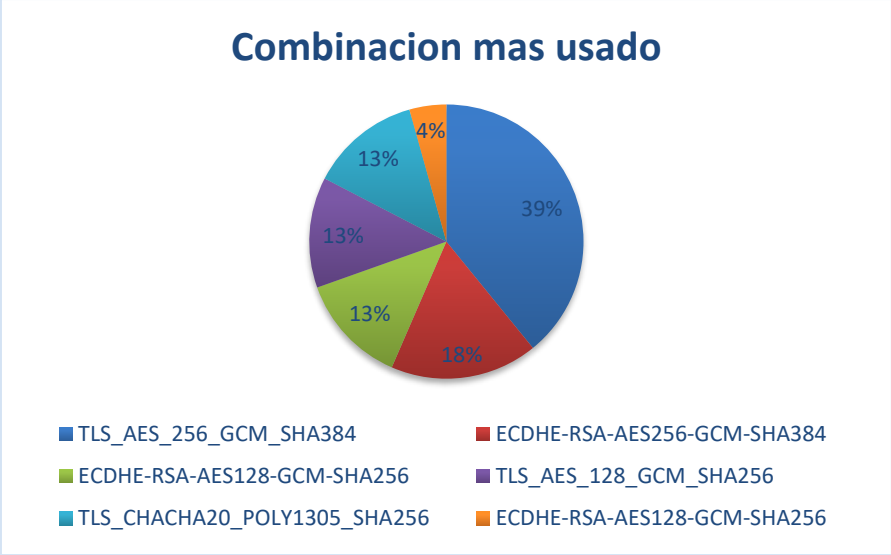


Fig. 4. Conjunto de algoritmo de cifrado más usado.

Nota. En la figura se muestra las combinaciones de cifrado TLS más utilizadas por las páginas web. La mayoría de las conexiones usan una combinación de cifrado fuerte.

De acuerdo al estudio realizado teniendo como fuente confiable la página web SIMILARWEB, se llevó a cabo un análisis de las 25 páginas web más visitadas [22], evaluando el conjunto de algoritmos de cifrado que utilizan. Según se muestra en la **figura 01**, el conjunto de algoritmos criptográficos más prevalentes entre los 25 sitios web es **TLS_AES_256_GCM_SHA384**. Esta combinación emplea AES_256_GCM para ofrecer cifrado autenticado y SHA384 como algoritmo de hash, para garantizar la integridad de los datos.

Para estudios de la evaluación de algoritmos asimétricos hemos elegido la tercera combinación más utilizada por estas 25 páginas, que sería la combinación: ECDHE-RSA-AES256-GCM-SHA384, donde emplea una versión modificada del algoritmo DiffieHellman de curvas elípticas (ECDHE) para crear una conexión confidencial a través de un medio no seguro. Además, utiliza RSA de 2048bits para verificar la identidad del servidor y SHA384, y para respaldar la integridad de los datos utiliza un algoritmo de hash (sha384). Según la página **ciphersuite.info,** cómo se puede apreciar (figura 02) la combinación de algoritmo de cifrado **ECDHE-RSA-AES256-GCM-SHA384**, es una combinación segura.

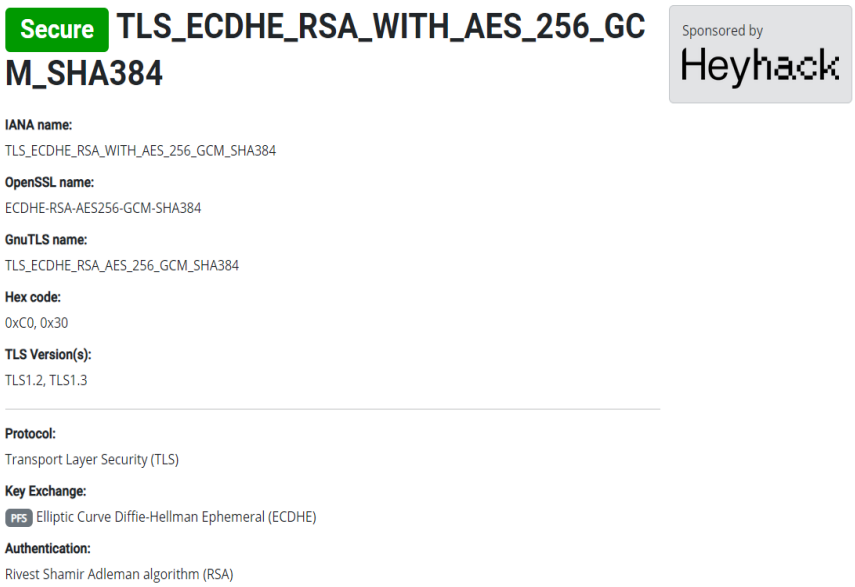


Fig. 5. Nivel de seguridad de la combinación de algoritmo de cifrado

Nota. La imagen detalla la combinación de algoritmos TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, utilizada por el protocolo TLS 1.2 y 1.3. Esta configuración emplea ECDHE para el intercambio de claves y RSA para la autenticación, ofreciendo alta seguridad en las comunicaciones en línea.

El estudio y análisis exhaustivo de estas 25 páginas web se llevó a cabo con el propósito de identificar la combinación más apropiada para la creación del certificado, permitiéndonos realizar pruebas en el ámbito de nuestra investigación. Esta evaluación minuciosa nos proporcionó información valiosa sobre las prácticas de seguridad y los algoritmos utilizados por estas páginas líderes a nivel mundial. Con estos hallazgos, estamos mejor equipados para seleccionar y configurar certificados de manera eficaz, asegurando la pertinencia y validez de nuestras pruebas de investigación en el ámbito de la seguridad web.

3.1.2. Cantidad de bytes entre clientes – servidor (HANDSHAKE)

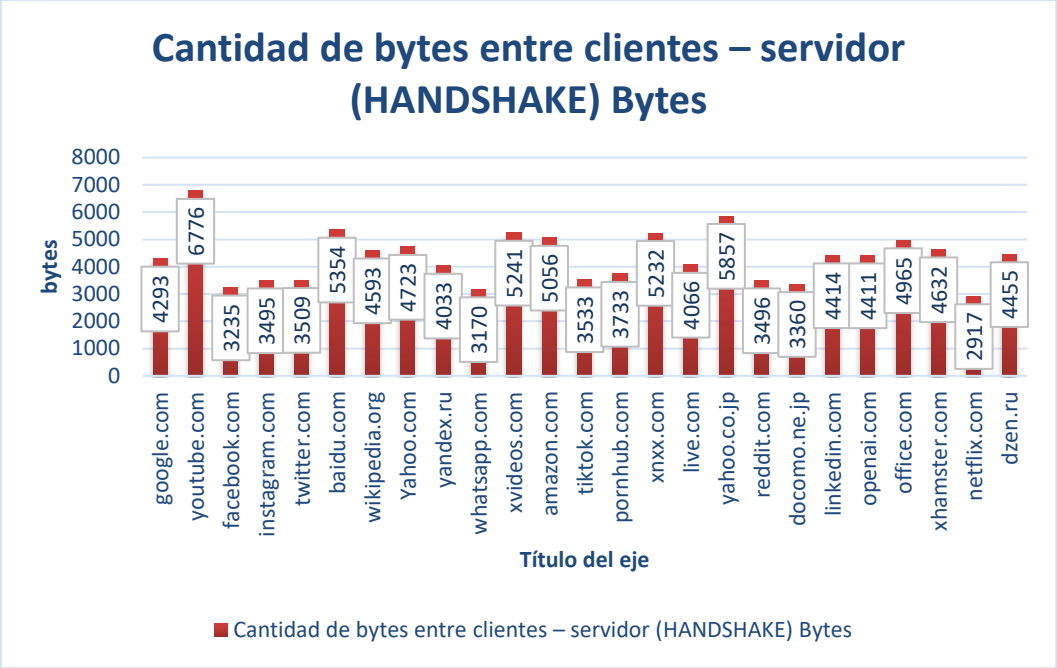


Fig. 6. Tamaño del Handshake entre cliente – servidor.

Nota. En la figura se muestra una comparación de la cantidad de bytes intercambiados durante el proceso de handshake entre clientes y servidores para varios sitios web.

El análisis reveló que el proceso de handshake en las páginas examinadas muestra un tamaño promedio, según lo ilustra la figura 03. Esto indica que, durante la fase inicial de negociación entre el usuario que visita la página web y el servidor que la hospeda, se intercambian una cantidad específica de bytes. Para obtener estos datos, se ejecutó el comando: "time openssl s_client -connect www.google.com:443", utilizando OpenSSL en un sistema operativo Ubuntu. Este procedimiento nos permitió evaluar la eficiencia de la conexión en términos del tiempo necesario para completar el handshake. Estos hallazgos son cruciales para comprender el rendimiento de seguridad de las páginas web, brindando información valiosa sobre cómo se establece la conexión inicial entre el cliente y el servidor. El handshake (apretón de manos) es un proceso fundamental en la seguridad de las comunicaciones en línea, específicamente en el contexto del protocolo SSL/TLS. Cuando un cliente intenta conectarse a un servidor seguro (por ejemplo, al acceder a una página web a

través de HTTPS), se inicia el *handshake*. Este proceso se encarga de establecer una conexión segura y autenticada entre el cliente y el servidor. Durante el *handshake*, se lleva a cabo una serie de intercambios de información, incluidos algoritmos de cifrado, claves públicas y otros parámetros cruciales para la seguridad.

3.1.3. Periodo de Procesamiento

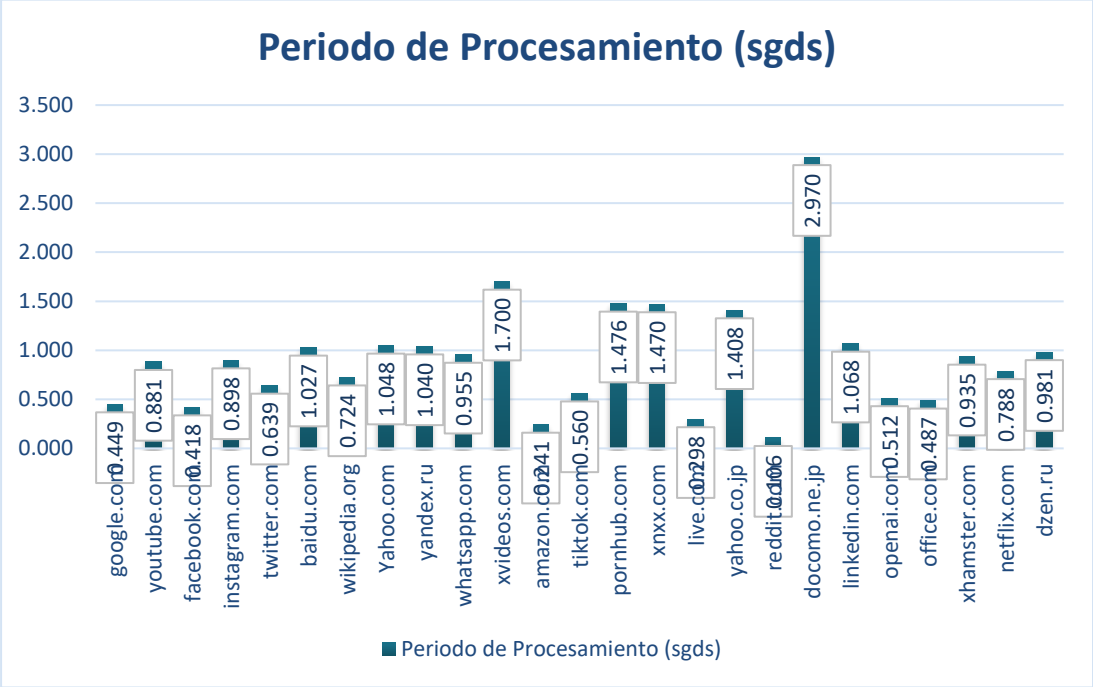


Fig. 7. Tiempo de procesamiento handshake.

Nota. El gráfico muestra el tiempo de procesamiento del handshake de TCP en diferentes sitios web. **docomo.ne.jp** tiene el mayor tiempo con **2.970 segundos**, mientras que **google.com** y **facebook.com** presentan los menores tiempos, **0.449 segundos** y **0.418 segundos** respectivamente. Estos resultados reflejan la eficiencia de la conexión segura y el impacto de los algoritmos asimétricos en el proceso de establecimiento de la conexión.

Se llevaron a cabo tres pruebas en cada página web para determinar el tiempo total del *handshake* de TCP, como se detalla en la tabla adjunta. Esta tabla refleja el tiempo requerido por cada página para establecer la negociación con el servidor a través de la capa de transporte. Cada entrada en la tabla representa el resultado de ejecutar el software Git y utilizar la línea de comando **"time curl -o /dev/null -s -w 'Tiempo total: %{time_total}\n' https://www.paginaweb.com".** Este proceso permitió medir con precisión el tiempo total del *handshake* para cada página web, proporcionando una visión detallada de la eficiencia de la

conexión en términos de negociación inicial entre el cliente y el servidor. Estos datos son esenciales para evaluar el rendimiento y la rapidez con la que se establecen conexiones seguras, contribuyendo así a comprender la experiencia del usuario al acceder a sitios web específicos. El tiempo de procesamiento de los algoritmos asimétricos se refiere al tiempo necesario para realizar las operaciones criptográficas asociadas, como la generación de claves y el cifrado. Estos algoritmos desempeñan un papel crucial en el proceso de handshake, una fase inicial en la conexión segura entre el cliente y el servidor. Durante el handshake, los algoritmos asimétricos se utilizan para la autenticación y el intercambio seguro de claves. La eficiencia en el tiempo de procesamiento de estos algoritmos directamente afecta la duración total del handshake. Cuando los algoritmos son eficientes, el tiempo de procesamiento es menor, contribuyendo a una conexión segura más rápida y optimizando la seguridad y el rendimiento de las comunicaciones seguras en línea.

3.1.4. Costo Computacional

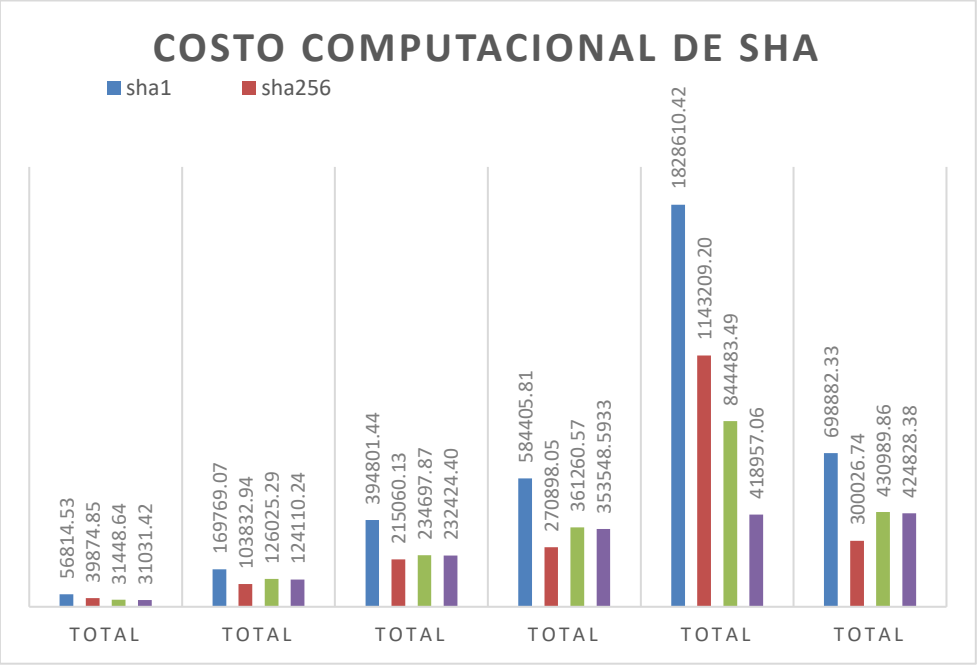


Fig. 8. Costo computacional de los algoritmos HASH.

Nota. El gráfico compara el costo computacional de los algoritmos SHA-1, SHA-256, y SHA-512, mostrando que SHA-512 tiene el mayor costo, especialmente en escenarios de alta carga, lo que lo hace más exigente en recursos.

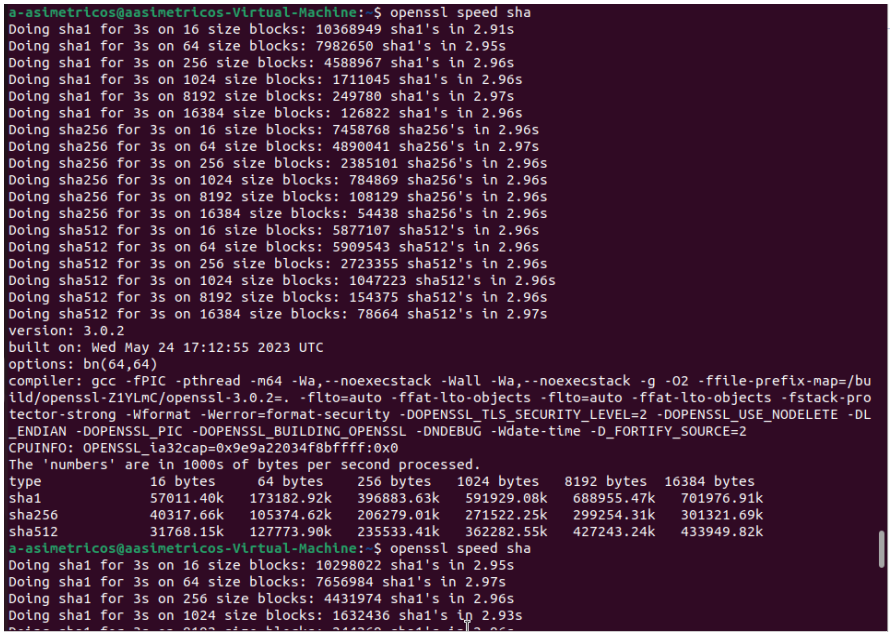


Fig. 9. Análisis de los algoritmos HASH.

Nota. En la imagen podemos apreciar los resultados del costo computacional del algoritmo hash, aplicando el comando openssl speed -evp sha384 en Openssl dentro de la plataforma de Linux.

Para llevar a cabo el costo computacional de los algoritmos Hash, se realizaron tres pruebas, con un periodo de 3 segundos en una desktop Core i7-2600 CPU a 3.40Ghz, donde se inyectaron paquetes de 16kb, 64kb, 256kb, 1024kb y 8192kb, con el fin de examinar el algoritmo de Hash (sha384) que hemos seleccionado para su análisis, es importante destacar que este algoritmo puede procesar paquetes de manera más eficiente que otros algoritmos hash, representando un costo computacional medio para un equipo de gama media/alta, con una capacidad de procesar un paquete por segundo.. Para poder obtener estos resultados se realizó pruebas con Linux (Ubuntu) mediante librería de OpenSSL con la línea de comando, para los algoritmos sha1, 256 y 512, por lo contrario, para el algoritmo sha384 se utilizó la línea de comando **openssl speed -evp sha384.**

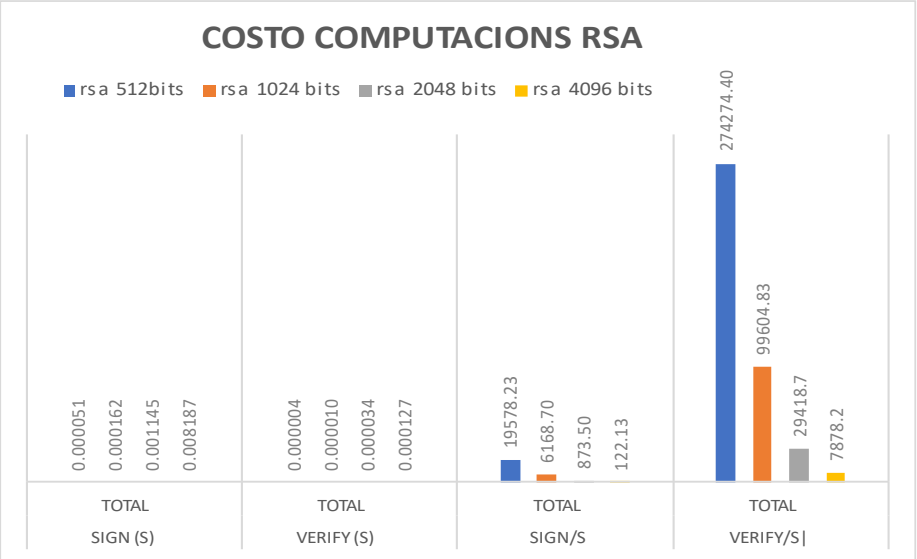


Fig. 10. Posibles vinculaciones generadas por el algoritmo RSA.

Nota. El gráfico muestra que, a medida que aumenta el tamaño de la clave RSA, el costo computacional en términos de tiempo de firma y verificación también aumenta, con una disminución significativa en la cantidad de verificaciones por segundo a medida que se incrementa el tamaño de la clave.

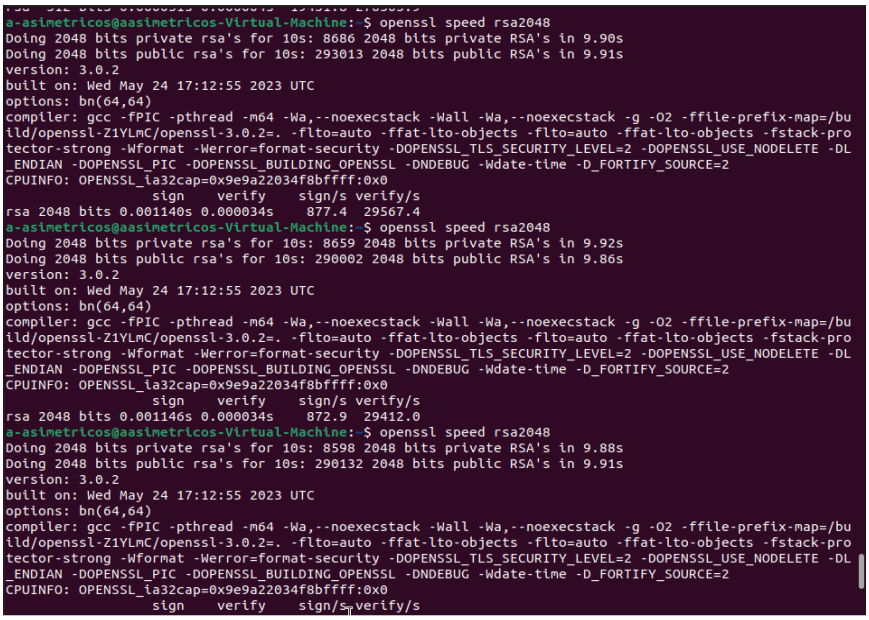


Fig. 11. Análisis del algoritmo RSA.

Nota. Las pruebas con RSA de 2048 bits en un Core i7-2600 indican que el algoritmo puede realizar una firma en 0.001145 segundos y soportar hasta 873.50 conexiones SSL por segundo, destacando su eficiencia y balance entre seguridad y rendimiento.

Se llevaron a cabo tres pruebas para cada uno de los algoritmos en un equipo con un Core i7-2600 CPU a 3.40 GHz. Se constató que el algoritmo RSA de 2048 bits, seleccionado en base a la combinación utilizada en las páginas web examinadas, permite que nuestro servidor ejecute una firma en 0.001145 s. Además, es capaz de admitir aproximadamente

873.50 conexiones SSL por segundo y una verificacion de 29418.7 s, por firmas.

El costo computacional de los algoritmos RSA y la función hash se refiere a la cantidad de recursos de cómputo necesarios para realizar sus operaciones respectivas. En el caso de RSA, la generación y manipulación de claves, así como las operaciones de cifrado y descifrado, involucran procesos matemáticos intensivos, cuyo tiempo de ejecución depende del tamaño de las claves utilizadas. Por otro lado, la función hash, como SHA-384, se encarga de generar un hash fijo de longitud a partir de datos de entrada, y su costo computacional aumenta con el tamaño de los datos. Estos algoritmos son fundamentales en seguridad informática; RSA para cifrado asimétrico y firma digital, y la función hash para verificar la integridad de datos y otras aplicaciones. La gestión eficiente de su costo computacional es esencial para equilibrar la seguridad y el rendimiento en diversos contextos de aplicación.

3.1.5. Segmentos sin cifrado

Los segmentos de datos sin cifrado se refieren a porciones de información transmitida en una comunicación, como en HTTP, que carecen de protección o cifrado de seguridad. Esto implica que la información se envía sin medidas de confidencialidad, quedando expuesta a posibles amenazas de interceptación o manipulación durante la transmisión.

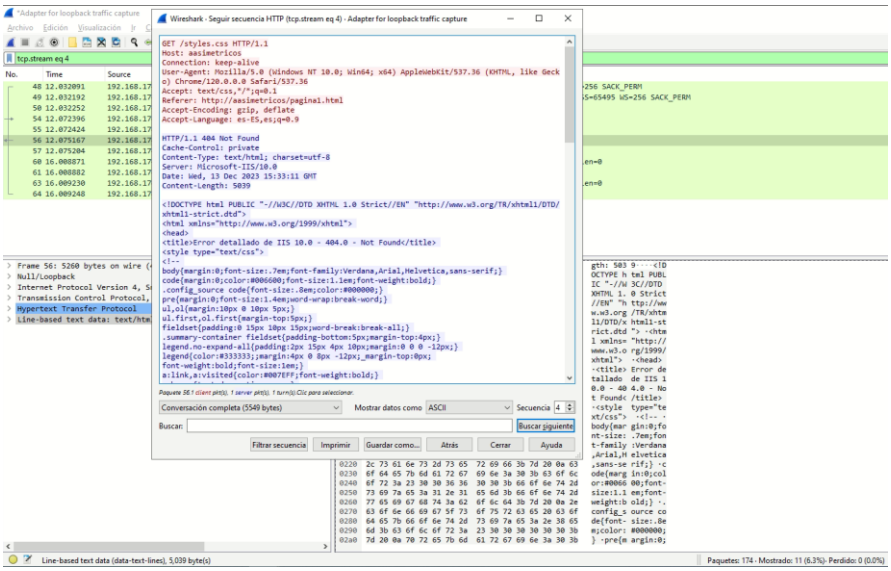


Fig. 12. Sniffeo de página web sin cifrado.

Nota. La imagen muestra una captura de tráfico HTTP en Wireshark donde se observa la transmisión de 5549 bytes de datos sin cifrar. Esto significa que la información está expuesta a posibles interceptaciones y manipulaciones, destacando la necesidad de implementar cifrado, como HTTPS, para proteger la confidencialidad de los datos.

En la **figura 12**, se aprecia que la conexión HTTP presenta 5549 bytes de información sin cifrar. Este dato visualiza la existencia de datos desprotegidos durante la comunicación, señalando la presencia de potenciales vulnerabilidades. La información no cifrada en la transmisión puede exponerse a riesgos como interceptación o manipulación, subrayando la importancia de implementar medidas de cifrado para asegurar la confidencialidad de los datos durante las comunicaciones.

3.1.6. Segmentos con cifrado

Los segmentos de datos cifrados son porciones de información transmitida en una comunicación mediante HTTPS, lo que garantiza una capa de protección y cifrado de seguridad para los datos. En contraste con la comunicación a través de HTTP, donde la información se transmite sin protección, HTTPS utiliza medidas criptográficas para salvaguardar la confidencialidad de los datos durante la transmisión, ofreciendo una capa adicional de seguridad en las comunicaciones.

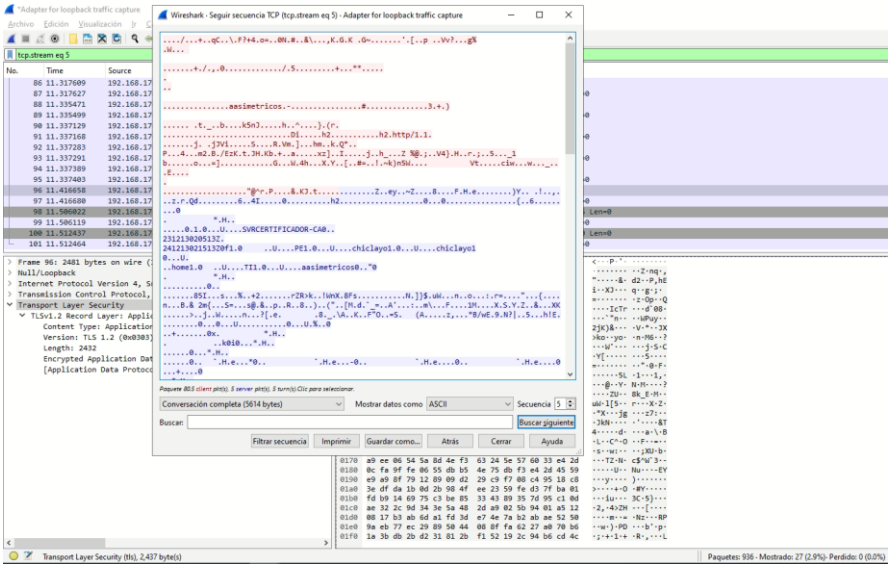


Fig. 13. Sniffeo de página web con cifrado.

Nota. La imagen muestra que la comunicación mediante HTTPS cifra 5614 bytes de información, indicando que los datos están protegidos por cifrado durante la transmisión. Esto asegura la confidencialidad y seguridad de la información, contrastando con HTTP, donde los datos quedan expuestos a posibles amenazas.

En la **figura 13**, se destaca que la comunicación mediante HTTPS muestra 5614 bytes de información cifrada, señalizando la existencia de datos resguardados por medidas criptográficas durante la transmisión segura. Este indicador visual subraya la implementación efectiva de un cifrado robusto, garantizando la confidencialidad de la información transmitida y ofreciendo una capa adicional de seguridad en comparación con la comunicación no cifrada a través de HTTP, donde la información queda expuesta a posibles amenazas.

3.2 Discusión

Al realizar la interpretación de los resultados, se lograron obtener información específica. Este proceso se desarrolló en el contexto de una infraestructura de clave pública (PKI) de 1 solo nivel, abarcando tres aspectos cruciales: (cliente-servidor y entidad certificadora). Esta configuración proporciona una visión clara de la implementación práctica de seguridad informática, donde se establece una relación jerárquica y roles definidos entre el cliente, el servidor y la entidad certificadora.

Según el siguiente artículo se ha desarrollado un sistema en web donde se administrar la información de la organización TELALCA, implementando medidas de seguridad por medio del protocolo SSL y HTTPS. Este proceso empleó un server (ApacheTomcat 6.0) y se generó un certificado de manera digital con una clave pública RSA 2048bits para asegurar el acceso seguro al sistema por medio del cifrado y el protocolo SSL y HTTPS [23]. En la **Figura 07** presenta el tiempo de procesamiento de esta clave, mientras que la **TABLA I** se comprueba que el algoritmo RSA de 2048bits es el que más utilizan los sitios web analizadas. La investigación destaca que los certificados digitales son generados a través de la "entidad de certificación (Active Directory Certificate Services) de Windows Server 2012". En contraste, para este estudio se empleó la herramienta Keytool, integrada en la plataforma de Java, para llevar a cabo el proceso. Este enfoque proporciona un marco sólido para la seguridad de la información en la empresa, aprovechando las capacidades de cifrado avanzadas ofrecidas por el protocolo HTTPS. La implementación de estas medidas garantiza la integridad y confidencialidad de los datos, cumpliendo con los estándares de seguridad actuales en la gestión de información empresarial [23].

La tesis examinada se centra en respaldar los registros de los servidores remotos en un servidor principal por medio de un de software de código abierto como Stunnel, rsyslog y OpenSSL. Se destaca la necesidad de cifrar los mensajes de syslog, especialmente cuando

la conexión entre los servidores se realiza mediante una red pública de datos. El objetivo principal es asegurar la integridad, la autenticidad y la confidencialidad de los datos en esta transferencia. El desarrollo se llevó a cabo en dos Desktop con procesadores Intel XeonE5320, 2 Gbytes de RAM y ejecutando Linux v2.6.27 de la partición Ubuntu v8.10 [24]. Se eligió RSA de 2048 bits para la clave pública con el fin de habilitar un cifrado seguro de los mensajes. Nuestra investigación, representada en la Figura 07, muestra el tiempo de procesamiento de esta clave. Además, en la Figura 01, se observa que el algoritmo RSA de 2048bits es el más se utiliza entre los sitios web analizadas. A pesar de identificar limitaciones en las herramientas utilizadas, como las restricciones de Stunnel en soportar solo Proxy transparente y su implementación en servicios con un único puerto, así como las complejidades en la generación de certificados con OpenSSL, la tesis concluye que la aplicación práctica se centra en el ámbito de las VPNs, demostrando la viabilidad y utilidad de la estrategia implementada [24].

En la actualidad, en el ámbito de las telecomunicaciones, es común que la transmisión de datos a través de redes, ya sea intranet o internet, se realice de manera creciente. Sin embargo, surgen problemas como la pérdida de datos durante la transmisión, ya sea debido a congestiones en la red o a la pérdida de la ruta planificada para el envío de datos. Esta situación puede ser problemática, ya que compromete la integridad y la eficiencia de la comunicación de datos en entornos conectados. Para abordar estos desafíos, es esencial implementar estrategias y tecnologías que minimicen la probabilidad de pérdida de datos y garanticen una transmisión fluida y segura. La gestión eficaz de la congestión y la planificación robusta de rutas son elementos clave para mantener la integridad y la calidad de la transmisión de datos en la red.

Este análisis nos lleva a investigar un estudio cuyo propósito es examinar, bosquejar y construir un modelo de protocolo de transporte fundamentado en la conexión

(comunicación) TCP. El objetivo central es satisfacer las demandas de transferencia de datos, asegurando la seguridad, confiabilidad y alta disponibilidad. Esta tesis se percibe como un modelo de buenas prácticas para la conexión entre el cliente, el servidor y la Autoridad de Certificación en nuestra propia investigación. La implementación del prototipo en tres máquinas virtuales proporciona una conexión segura, destacando la importancia de abordar de manera integral los aspectos de seguridad y eficiencia en la transferencia de datos en entornos de red. Este enfoque contribuye al desarrollo de soluciones robustas que garantizan la integridad y la confidencialidad de la información, crucial en contextos donde la seguridad de la comunicación es prioritaria [25].

IV. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones.

La investigación ha respondido claramente a la pregunta crítica de cómo puede contribuir el algoritmo asincrónico a mejorar la seguridad en la transferencia de segmentos en la capa de transporte. Se ha demostrado que la utilización de algoritmos asincrónicos, como la RSA con 2048 bits y ECDHE-RSA-AES256-GCM-SHA384, aumenta considerablemente la seguridad en la transmisión de datos en redes públicas. Estos algoritmos permiten un intercambio seguro de claves y una autenticación mutua, algo absolutamente necesario para establecer un enlace seguro entre cliente y servidor. Este sistema de codificación ofrece un alto grado de salvaguardia, tanto para la supervivencia de la información como para su conservación. Gracias a ellas, hackers no pueden interceptar o manipular los datos viajando por Internet ya que la aplicación de estos algoritmos lleva con ella la protección de cualquier alteración. También, a la hora de mencionar estos algoritmos en operación, es necesario referenciar que se están convirtiendo efectivamente en una garantía para la transmisión de información sin ser alterada. Este resultado subraya la necesidad de emplear técnicas de cifrado asimétrico en ambientes altamente críticos, cumpliendo al mismo tiempo los requisitos examinados anteriormente para mejorar la seguridad de los trozos de segmento en la capa de transporte.

La eficiencia de los algoritmos y los protocolos de encriptación nos ha mostrado que la adopción de combinaciones robustas de encriptación, como ECDHE-RSA-AES256-GCM-SHA384, no solo garantiza la adecuada seguridad, sino también la eficiencia operativa considerable. Esta investigación demostró que es factible mantener múltiples sesiones cifradas en paralelo sin impactar significativamente el rendimiento del servidor, lo cual es crucial para aquellos que operan en entornos de alta carga o buscan la fiabilidad superior. El empleo de dichas combinaciones garantiza que las comunicaciones sean confiables y rápidas, preservando tanto la confidencialidad de la información como su integridad. La

importancia de tales conclusiones radica en el contexto de servicios web y aplicaciones que dependen de transacciones seguras y rápidas. Además, se puede concluir que la hipótesis acerca de la capacidad de los algoritmos asimétricos de ser optimizados también fue confirmada.

El estudio ha enfatizado la relevancia del Protocolo de Seguridad de la Capa de Transporte (TLS) y la infraestructura de clave pública (PKI) en la implementación de seguridad en la capa de transporte. Al aplicar estos protocolos de seguridad, se logra no solo la autenticación de las conexiones, sino también el cifrado de los datos durante la transmisión. La infraestructura PKI, al respaldar la autenticidad de los certificados digitales, añade una capa adicional de confianza en las comunicaciones, esencial en la protección de datos sensibles. Este mecanismo es especialmente valioso para prevenir ataques de intermediarios y garantizar que la información se mantenga segura y privada. El logro de este objetivo específico de aplicar algoritmos de cifrado y medir su efectividad a través de entornos de servidor simulados refuerza la comprensión de la ciberseguridad en aplicaciones prácticas y subraya la importancia de estrategias de defensa multicapa para entornos digitales críticos.

Uno de los hallazgos críticos de la investigación es la necesidad de mantener un equilibrio óptimo entre seguridad y rendimiento en la capa de transporte. Aunque los algoritmos de cifrado más robustos, como RSA de 2048 bits y SHA-384, aumentan el tiempo de procesamiento, la investigación ha demostrado que este costo adicional puede ser justificado por el aumento en la seguridad proporcionada. El uso de estos algoritmos permite asegurar la integridad y confidencialidad de los datos sin incurrir en un tiempo de procesamiento excesivamente alto, que podría afectar la experiencia del usuario o la eficiencia del sistema. Este resultado valida el enfoque de seleccionar cuidadosamente los algoritmos de cifrado en función de las necesidades específicas de seguridad y rendimiento de cada aplicación, resaltando que, en el diseño de sistemas seguros, es fundamental encontrar un equilibrio que no comprometa la protección de los datos ni el desempeño del sistema.

4.2 Recomendaciones

Se plantean las siguientes recomendaciones a partir de los resultados y conclusiones:

1. A medida que la tecnología avanza, se vuelve fundamental explorar opciones más eficientes para proteger los datos. Los algoritmos de criptografía de curva elíptica (ECC) han ganado popularidad debido a su capacidad para ofrecer altos niveles de seguridad con menores recursos computacionales. Sugiero investigar más sobre ECC, especialmente su implementación junto con protocolos de seguridad como TLS. Esta combinación no solo puede reducir la carga en servidores y dispositivos móviles, sino que también mejora la velocidad de procesamiento sin comprometer la seguridad. Este enfoque puede ser particularmente valioso para aplicaciones que requieren una alta eficiencia, como las que funcionan en entornos con recursos limitados.

2. Utilizar protocolos de seguridad en la transferencia de datos en línea, como HTTPS, para garantizar la autenticidad, integridad, seguridad y confidencialidad de los datos. HTTPS cifra los datos que se transmiten entre los sitios web, lo que los hace más seguros. Además, verifica que el sitio web tenga un certificado SSL/TLS válido antes de compartir información confidencial. Dado que cada vez más organizaciones están migrando a infraestructuras en la nube, es crucial desarrollar métodos que aseguren una protección robusta en estos entornos. Recomiendo investigar cómo optimizar la seguridad en la capa de transporte específicamente en la nube, utilizando algoritmos asimétricos y otros mecanismos avanzados de protección. Es importante considerar cómo los diferentes algoritmos de cifrado funcionan en un ambiente de múltiples usuarios, garantizando que la privacidad y la integridad de los datos se mantengan sin importar el nivel de tráfico o uso.

3. Implementación de Algoritmos de Cifrado Híbridos para una Seguridad Mejorada, se recomienda la implementación de esquemas de cifrados híbridos. Por ejemplo, se podría utilizar RSA o ECC para el intercambio seguro de claves inicial y luego emplear algoritmos simétricos más rápidos como AES para cifrar los datos durante la sesión. Este enfoque aprovecha la alta seguridad de los algoritmos asimétricos para la autenticación y la eficiencia de los algoritmos simétricos para la transmisión de datos. Así, se garantiza tanto un alto nivel de seguridad como un rendimiento óptimo en las comunicaciones.

REFERENCIAS

[1] "Ataques contra la ciberseguridad e infracciones de la ciberseguridad." Accessed: Jun. 30, 2024. [Online]. Available: https://latam.kaspersky.com/resource-center/preemptive-safety/how-to-prevent-cyberattacks

[2] L. S. Zambrano, "Tecnologías para la Administración y Generación de Firmas Digitales Introducción a la Criptografía".

[3] "¿Qué es la criptografía?" Accessed: Jun. 30, 2024. [Online]. Available: https://latam.kaspersky.com/resource-center/definitions/what-is-cryptography

[4] M. I. Romero *et al.*, "INTRODUCCIÓN A LA SEGURIDAD INFORMÁTICA Y EL ANÁLISIS DE VULNERABILIDADES".

[5] S. Kumar, R. Mahajan, N. Kumar, and S. K. Khatri, "A study on web application security and detecting security vulnerabilities," *2017 6th International Conference on Reliability, Infocom Technologies and Optimization: Trends and Future Directions, ICRITO 2017*, vol. 2018-January, pp. 451–455, Apr. 2018, doi: 10.1109/ICRITO.2017.8342469.

[6] "A Survey on Vulnerability Assessment & Penetration Testing for Secure Communication | IEEE Conference Publication | IEEE Xplore." Accessed: Jun. 19, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/8862767

[7] David Hoffman, Daniel Rodríguez Maffioli, Andy Kotz, Sofia Bliss-Carrascosa, and Spencer Reeves, "Perspectivas de CIBERSEGURIDAD de los Líderes de la Industria," *Informe LATAM CISO*.

[8] R. Wang, Y. Zhu, J. Tan, and B. Zhou, "Detection of malicious web pages based on hybrid analysis," *Journal of Information Security and Applications*, vol. 35, pp. 68–74, Aug. 2017, doi: 10.1016/J.JISA.2017.05.008.

[9] R. Ismailova, "Web site accessibility, usability and security: a survey of government web sites in Kyrgyz Republic," *Univers Access Inf Soc*, vol. 16, no. 1, pp. 257–264, Mar. 2017, doi: 10.1007/S10209-015-0446-8/METRICS.

[10] M. Torres, D. Asesor, I. Junior, and E. C. Maco, “Algoritmos de encriptación de archivos para la transferencia en mensajería instantánea,” *Repositorio Institucional - USS*, 2020, Accessed: Dec. 08, 2023. [Online]. Available: http://repositorio.uss.edu.pe//handle/20.500.12802/6735

[11] M. AlRoubiei, T. AlYarubi, and B. Kumar, “Critical Analysis of Cryptographic Algorithms,” in *2020 8th International Symposium on Digital Forensics and Security (ISDFS)*, IEEE, Jun. 2020, pp. 1–7. doi: 10.1109/ISDFS49300.2020.9116213.

[12] L. S. Hui, A. C. Wen, O. C. Teng, Z. F. Zaaba, and A. Hussain, “Investigations and assessments on web browser security,” *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 8, pp. 392–397, Jun. 2019.

[13] K. Bhargavan, “Protecting transport layer security from legacy vulnerabilities,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9665, p. XV, Jan. 2016.

[14] M. E. C. Hurtado, D. J. A. Sarango, M. E. C. Hurtado, and D. J. A. Sarango, “Análisis de Certificados SSL/TLS gratuitos y su implementación como Mecanismo de seguridad en Servidores de Aplicación.,” *Enfoque UTE*, vol. 8, no. 1, pp. 273–286, Feb. 2017, doi: 10.29019/ENFOQUEUTE.V8N1.128.

[15] Diego Vargas, “¿Qué es TLS? Significado, uso y diferencias con SSL y HTTPS,” Hostinger tutoriales . Accessed: Oct. 13, 2023. [Online]. Available: https://www.hostinger.es/tutoriales/que-es-tls

[16] Vipul Gupta, Sheueling Chang Shantz, Stephen Fung, and Hans Eberle, “(PDF) Speeding up Secure Web Transactions Using Elliptic Curve Cryptography.” Accessed: Jun. 30, 2023. [Online]. Available: https://www.researchgate.net/publication/221655528_Speeding_up_Secure_Web_Transactions_Using_Elliptic_Curve_Cryptography

[17] Q. Luo and Y. Lin, “Analysis and comparison of several algorithms in SSL/TLS

handshake protocol," *Proceedings - 2009 International Conference on Information Technology and Computer Science, ITCS 2009*, vol. 2, pp. 613–617, Jan. 2009, doi: 10.1109/ITCS.2009.307.

[18] B. ; Dowling *et al.*, "A Cryptographic Analysis of the TLS 1.3 Handshake Protocol," *Journal of Cryptology*, vol. 34, no. 4, doi: 10.3929/ethz-b-000438744.

[19] S. Riaz, A. Sajid, and M. Kanwal, "Performance Analysis of SSL/TLS," *IJISET-International Journal of Innovative Science, Engineering & Technology*, vol. 1, no. 6, 2014, Accessed: Dec. 15, 2023. [Online]. Available: www.ijiset.com

[20] R. Hernández Sampieri, C. Fernández Collado, D. María del Pilar Baptista Lucio, and S. Méndez Valencia Christian Paulina Mendoza Torres, "Metodología de la Investigación," 2014.

[21] B. N. Moreno, N. Yamilett, B. Rimarachin, E. N. Rut, R. Moscol, and M. Fernando, "Evaluación de técnicas de cifrado para el intercambio de datos en internet de las cosas en el ámbito de la salud," *Repositorio Institucional - USS*, 2023, Accessed: Dec. 08, 2023. [Online]. Available: http://repositorio.uss.edu.pe//handle/20.500.12802/10593

[22] "Top Websites Ranking - Most Visited Websites in November 2023 | Similarweb." Accessed: Dec. 16, 2023. [Online]. Available: https://www.similarweb.com/top-websites/

[23] Ordoñez Calero and Hernán David, "Desarrollo del módulo de gestión de información técnica para TELALCA S.A. e implementación de seguridad mediante cifrado SSL del protocolo HTTPS." Accessed: Dec. 17, 2023. [Online]. Available: https://biblioteca.epn.edu.ec/cgi-bin/koha/opac-detail.pl?biblionumber=12496&shelfbrowse_itemnumber=12890

[24] R. Hernández Ortiz, N. Peña Blanco, C. M. Ramírez Amaya, and V. F. Rodríguez Baños, "Implementación de un túnel con cifrado para transporte de datos," Oct. 2015, Accessed: Dec. 17, 2023. [Online]. Available: http://tesis.ipn.mx/xmlui/handle/123456789/7520

[25] I. De Sistemas, J. E. Chapaca, G. Jairo, and D. R. Bustamante, “Análisis, diseño y desarrollo de un prototipo de protocolo de transporte basado en comunicación TCP con capacidad de cubrir las necesidades de transferencia de datos seguros, confiables y de alta disponibilidad.,” 2013, Accessed: Dec. 17, 2023. [Online]. Available: http://dspace.ups.edu.ec/handle/123456789/6354

ANEXOS

1. Esquema de la Capa física

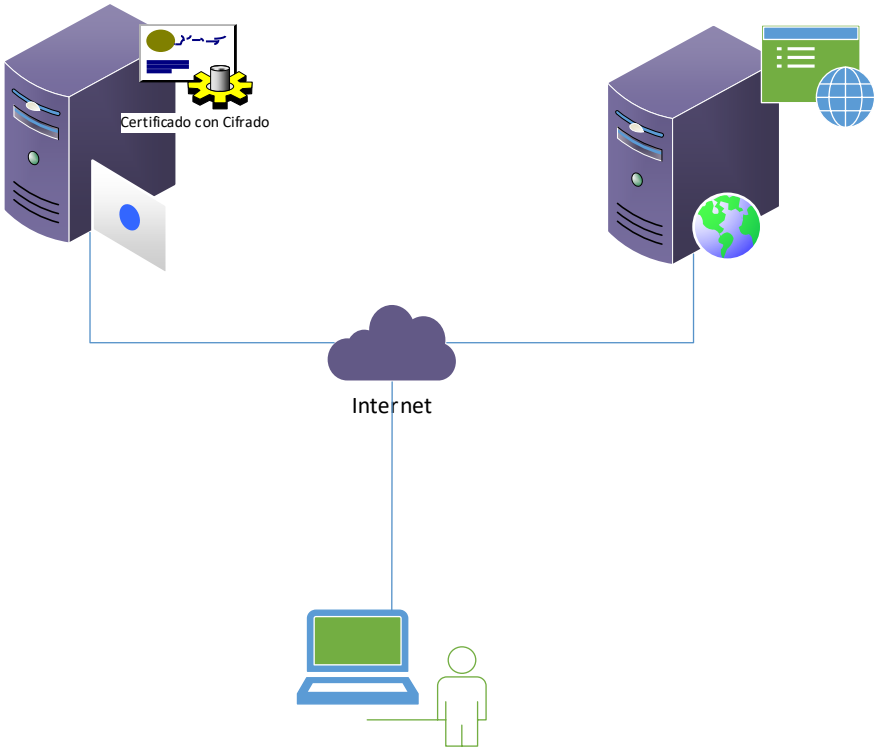


Fig. 14. Esquema de la capa física propuesto

Nota. El gráfico muestra el esquema de capa física propuesto para la implementación de servidores. Este esquema representa la topología física que se ha diseñado para la instalación, detallando cómo se conectarán los servidores a la red y cómo se dispondrán los componentes de hardware en el entorno.

2. Diseño de la estructura física y lógica de la red

a. Diseño de la estructura física.

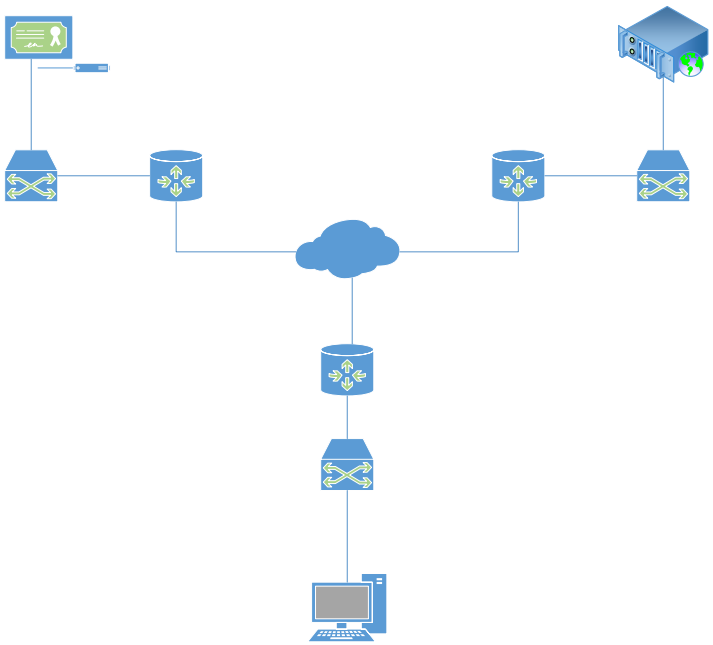


Fig. 15. Topología Física

Nota. El gráfico ilustra la topología física implementada para el desarrollo de la tesis, mostrando cómo se han conectado los servidores y dispositivos de red.

b. Diseño de la estructura Lógica.

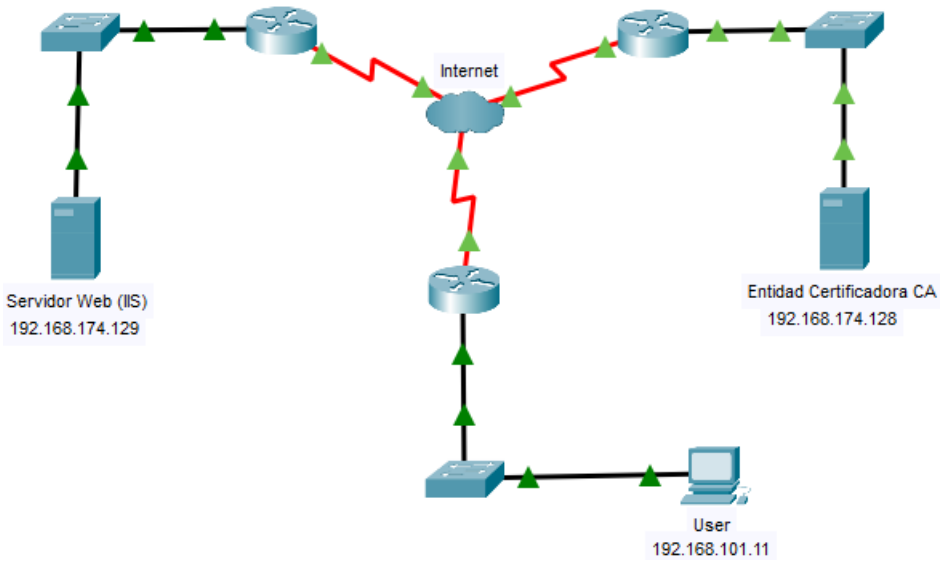


Fig. 16. Topología Lógica

Nota. El gráfico muestra la topología lógica que se ha implementado. En esta representación, se detallan las conexiones entre los distintos dispositivos de red, como servidores, routers, y el usuario final, indicando cómo se comunican entre sí a través de Internet.

c. Desarrollo y aplicación del procedimiento cifrado:

Se estableció una estructura de Infraestructura de Clave Pública (PKI) para llevar a cabo la aplicación del protocolo TLS de cifrado en un sitio web. La infraestructura comprendió un servidor de Autoridad de Certificación (CA) con Windows Server 2022 Standard y un servidor web IIS con Windows 10 Pro. La elección de este sistema operativo se fundamentó en la disponibilidad de la combinación particular de algoritmos de cifrados asociados a TLS 1.3.

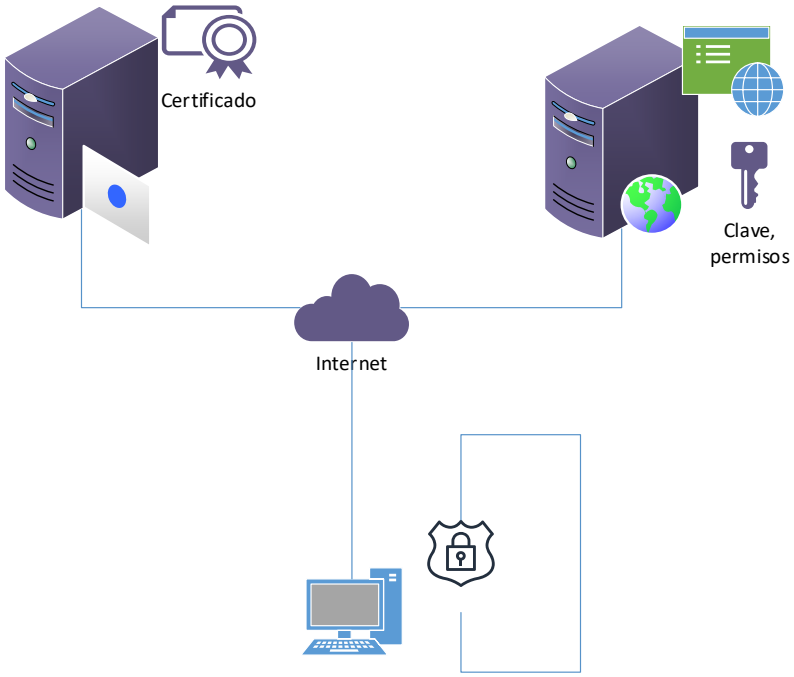


Fig. 17. Infraestructura de PKI

Nota. La imagen ilustra una infraestructura de Public Key Infrastructure (PKI) donde se emplea un certificado digital emitido por una Autoridad Certificadora (CA) raíz para asegurar la comunicación entre un servidor web y los usuarios finales a través de Internet.

La aplicación de esta infraestructura de PKI implica la utilización de una Root CA, mediante la cual solicitaremos directamente a la CA la firma digital de nuestro servidor web. Se realizará la configuración de un servidor web que utilizará el protocolo HTTPS para la transferencia de hipertexto. Para lograrlo, emplearemos una entidad certificadora de tipo Root para generar el certificado correspondiente. Posteriormente, este certificado será instalado en nuestro servidor web, asegurando así una comunicación segura y autenticada entre los usuarios y el servidor. Este enfoque fortalece la integridad y seguridad de la información transmitida a través del servidor web, garantizando la confianza de los usuarios.

Los elementos que se usaran para la implementación son los siguientes: Entidad Certificadora (Windows Server 2022 Estándar), Servidor web IIS e Usuario (Windows 10 Pro) y por último como usuario se usara (Ubuntu 22.04.3 LTS).

Fig. 18. Información básica de los servidores CA y IIS.

Nota. En la imagen se muestra las características de dispositivos que se han utilizado para el desarrollo de los servidores.

Anexo 01:

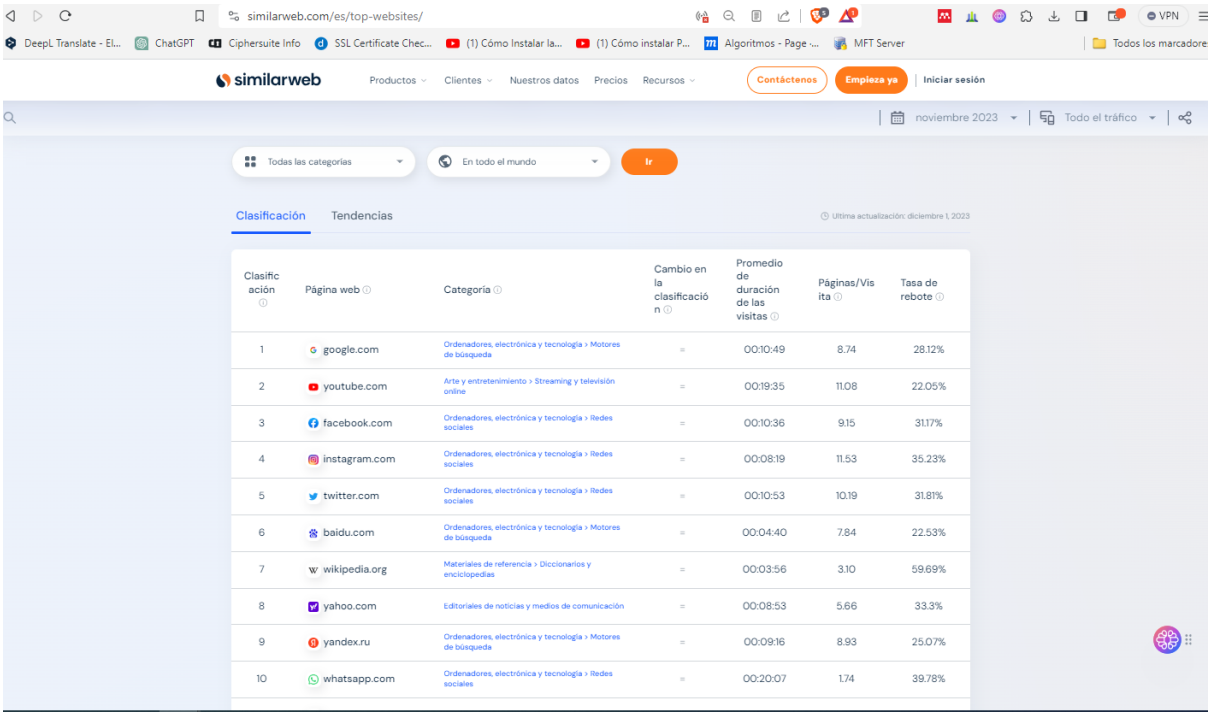


Fig. 19. páginas web más influyentes a nivel mundial

Nota. La imagen muestra un ranking de las páginas web más visitadas a nivel mundial, según Similarweb.com, una fuente confiable de análisis de tráfico web.

Similarweb.com es una página confiable que nos brinda un top global de las 50 páginas más visitadas hasta noviembre del 2023, para nuestro estudio se analizaron las top 25 páginas web.

Anexo 02:

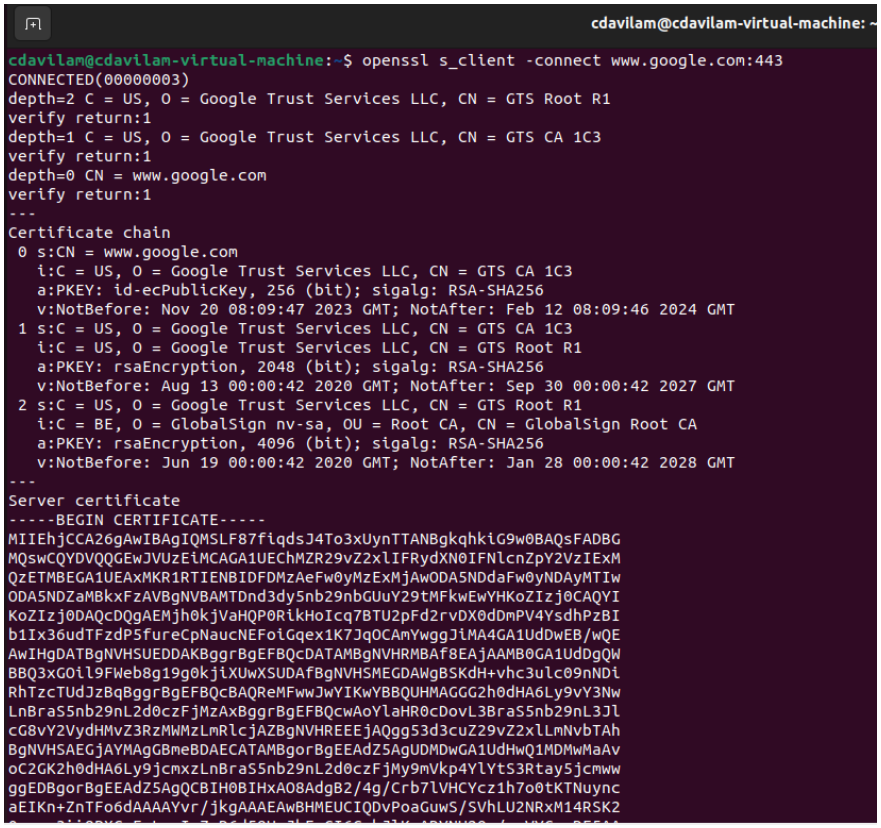


Fig. 20. Evaluación de los algoritmos de las páginas web.

Nota. La imagen muestra el resultado de ejecutar un comando de OpenSSL en una terminal de Linux, utilizado para evaluar la seguridad de las conexiones HTTPS y los algoritmos criptográficos de una página web.

Para obtener la identificación de los algoritmos de cada página se ingresó la línea de comando “openssl s_client -connect www.paginaweb.com:443”.

Anexo 03:

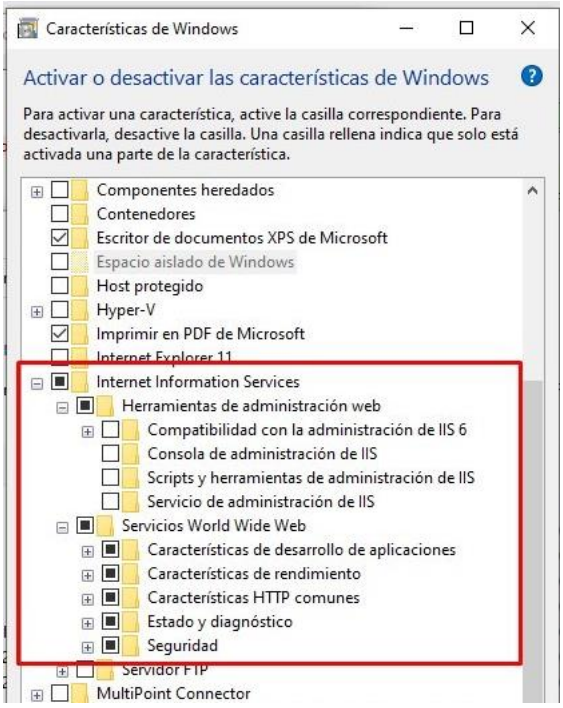


Fig. 21. Activación de los servicios de IIS

Nota. En la imagen se muestra la ventana de "Características de Windows" donde se han activado los roles y características de Internet Information Services (IIS). *Esta activación es esencial para tener acceso y poder levantar nuestras páginas web.*

Activamos los roles y características de IIS para tener acceso y poder levantar nuestras páginas web, el IIS desempeña un papel vital en la alojamiento y administración de sitios web. Su funcionalidad abarca la gestión de solicitudes HTTP, el procesamiento de páginas web dinámicas y la entrega de contenido estático.

Anexo 04:

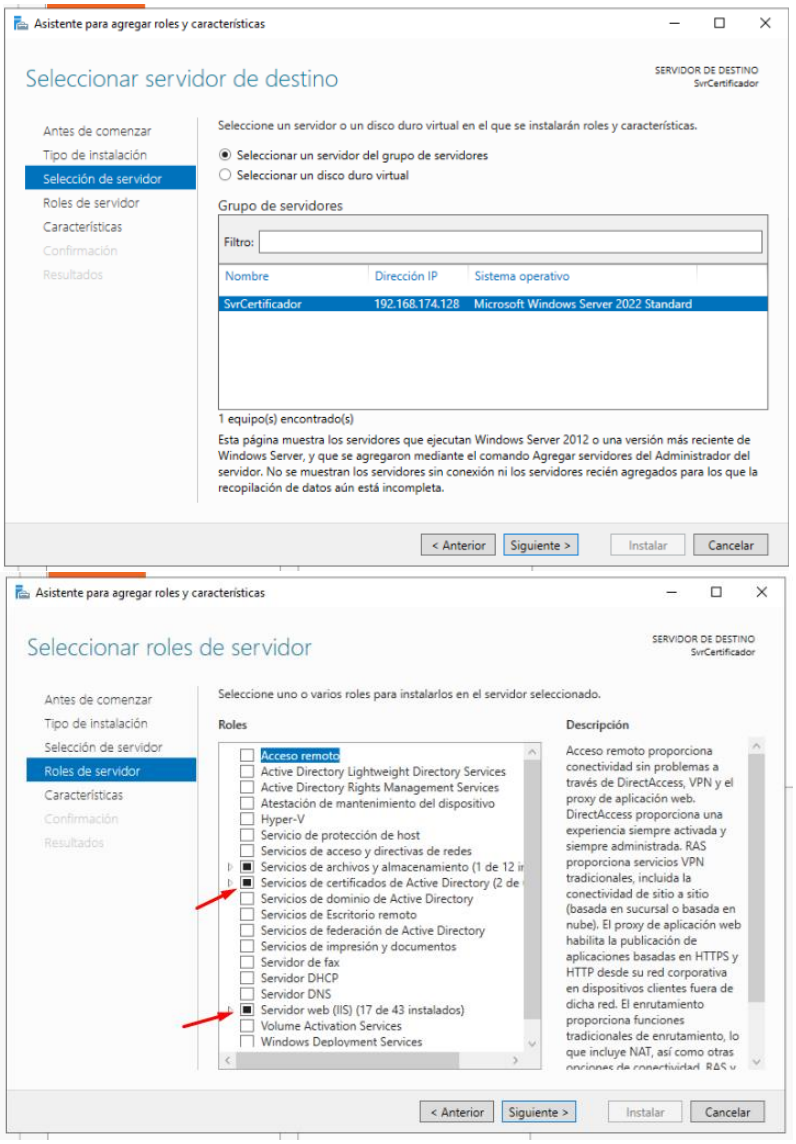


Fig. 22. Agregamos roles y características de la CA y IIS

Nota. En la imagen se observa la ventana de “Agregar roles y características” con los roles de Servicios de Certificados de Active Directory (CA) y Servicios web (IIS) seleccionados, esenciales para configurar y elaborar certificados en los servidores.

Se selecciona los roles y característica de la CA y IIS para poder tener acceso a los servidores para la configuración y elaboración del certificado.

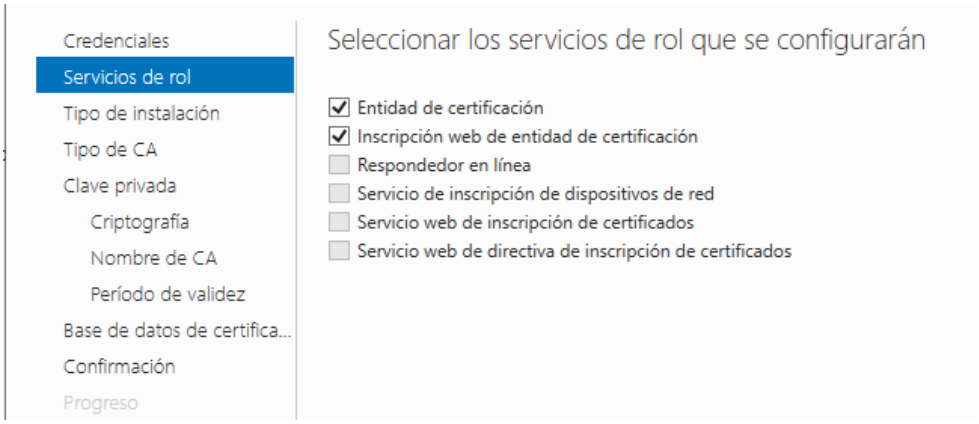


Fig. 23. Servicios de rol a configurar.

Nota. En la imagen se muestra la interfaz de configuración de servicios de rol, donde seleccionaremos los servicios necesarios para la entidad de certificación. En esta configuración, la entidad de certificación será responsable de emitir y revocar certificados desde el servidor.

Seleccionaremos los servicios del rol que estamos configurando, donde la entidad de certificación será responsable de emitir y revocar certificados desde el servidor. Asimismo, haremos uso de la inscripción web de la entidad de certificación para establecer conexión con la entidad certificadora mediante cualquier navegador web. Esto nos facilitará la descarga de certificados de la CA y la solicitud de certificados, siendo especialmente útil para solicitar un certificado destinado al servidor web.

Fig. 24. Tipo de instalación de CA

Nota. En la imagen se muestra la configuración de ADCS, donde se seleccionan servicios de rol y se elige entre una CA empresarial integrada con Active Directory o una CA independiente sin necesidad de AD DS.

Fig. 25. Seleccionamos el tipo de CA

Nota. La imagen muestra la configuración de ADCS, donde se elige entre una CA empresarial integrada con Active Directory o una CA independiente sin AD DS. También se detalla la jerarquía de la PKI con una CA raíz y una CA subordinada.

Después de instalar los servicios de la entidad certificadora de Active Directory (ADCS), es posible modificar la configuración del tipo de entidad certificadora que estamos utilizando. En nuestro escenario, con un dominio existente, emitiremos los certificados directamente. Optaremos por emplear una entidad certificadora empresarial de tipo Root.

Fig. 26. Tipo de Clave privada

Nota. La imagen muestra la configuración de ADCS para elegir entre crear una nueva clave privada o usar una existente.

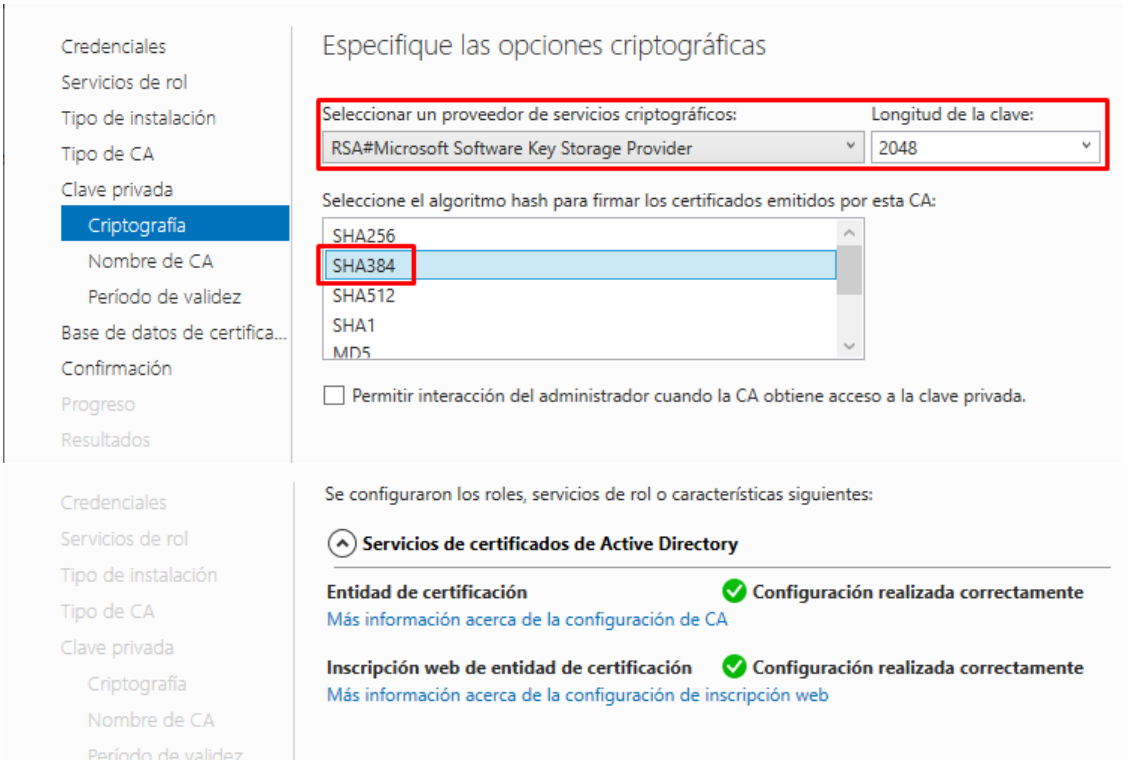


Fig. 27. Selección de algoritmos de cifrado

Nota. La imagen muestra la configuración criptográfica en ADCS, donde se selecciona el proveedor de servicios criptográficos, la longitud de la clave y el algoritmo hash para firmar certificados.

Procederemos a ajustar la clave privada que emplearemos, creando una nueva. Luego procedemos seleccionaremos el proveedor de servicios criptográficos, el tamaño de la clave pública y el algoritmo de hash.

Anexo 05:

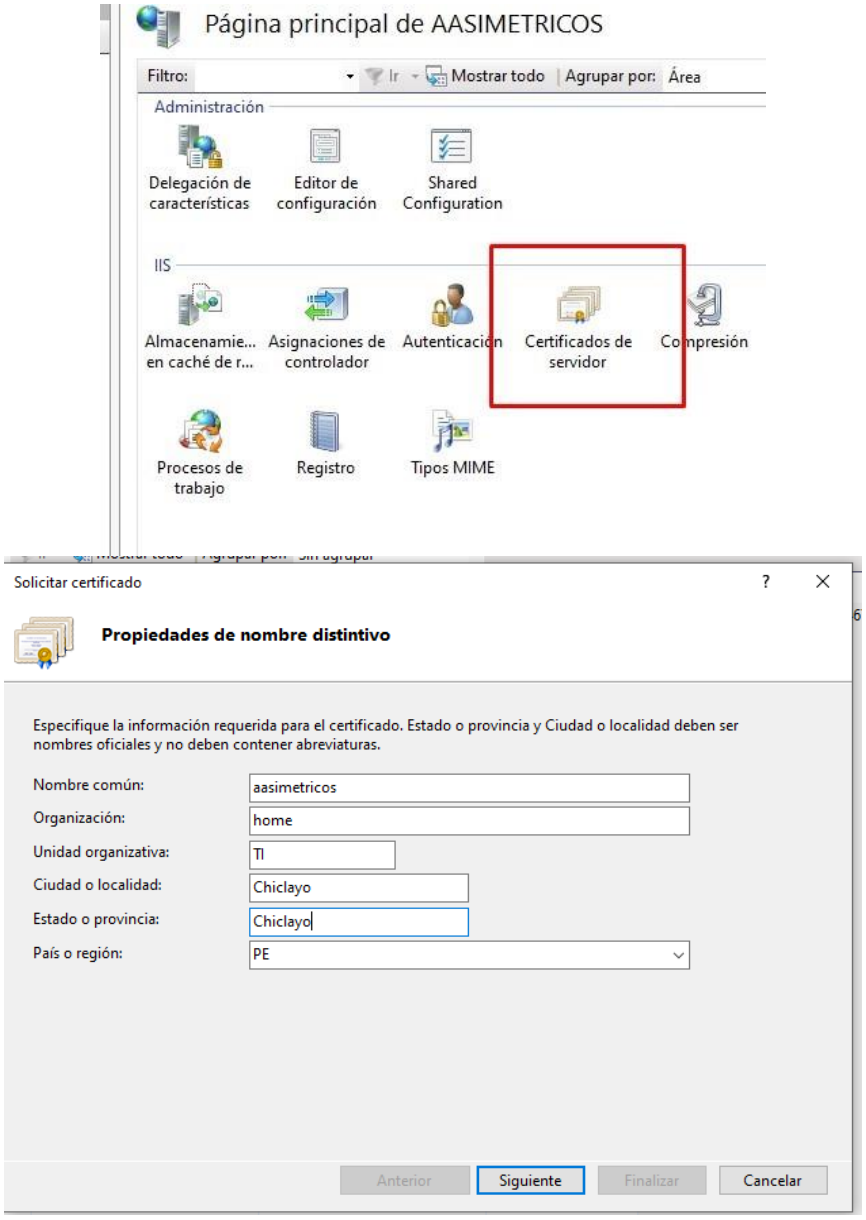


Fig. 28. Formulario de solicitud de certificado

Nota. En la imagen se logra visualizar el certificado de servidor, y un formulario con el nombre que llevara el certificado y sus diferentes datos que son requisitos llenar.

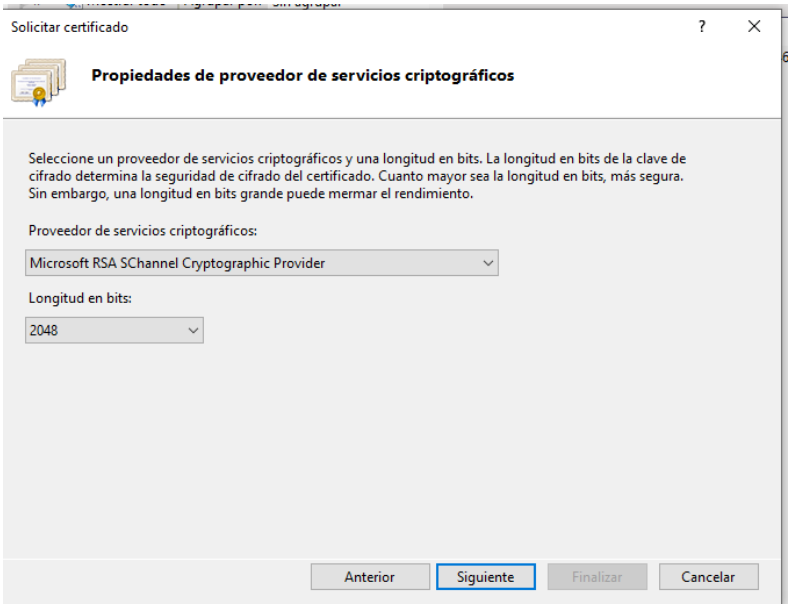


Fig. 29. Selección de los algoritmos de cifrado

Nota. La imagen muestra la interfaz para solicitar un certificado en ADCS, donde se elige el proveedor de servicios criptográficos y la longitud de la clave.

Seleccionados dentro del servidor IIS la opción de certificados de servidor, para poder crear la solicitud, dentro de ello se elige los algoritmos que se implementaran, para ello se elige los algoritmos asimétricos con sha384, y nos quedaría un archivo de la solicitud con extensión .csr.

Anexo 06:

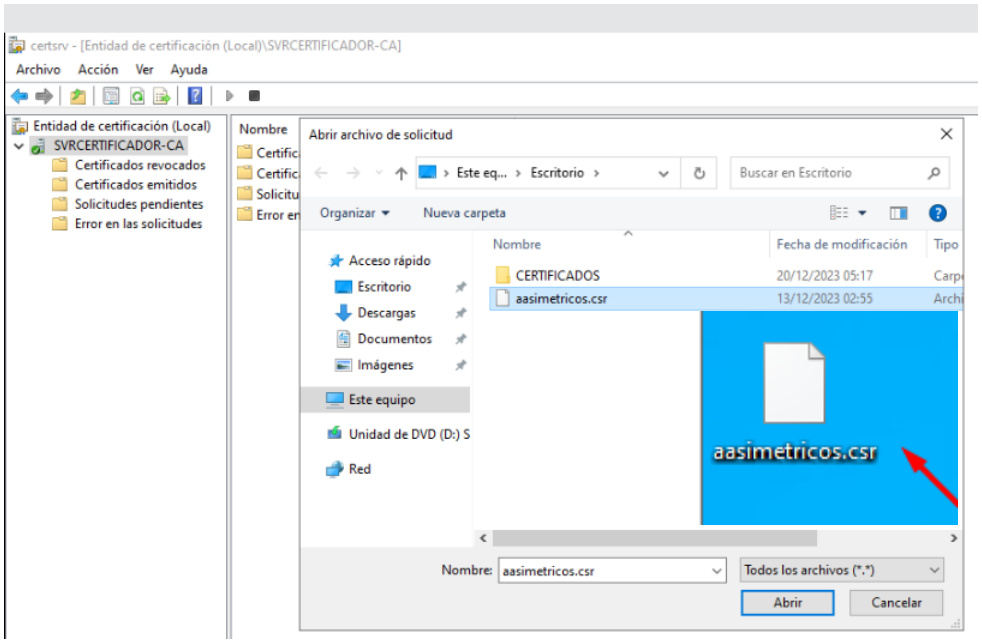


Fig. 30. Emisión de solicitud de certificado.

Nota. En la imagen se muestra la emisión del certificado con los algoritmos seleccionado anteriormente, listo para ser emitido.

Anexo 07:

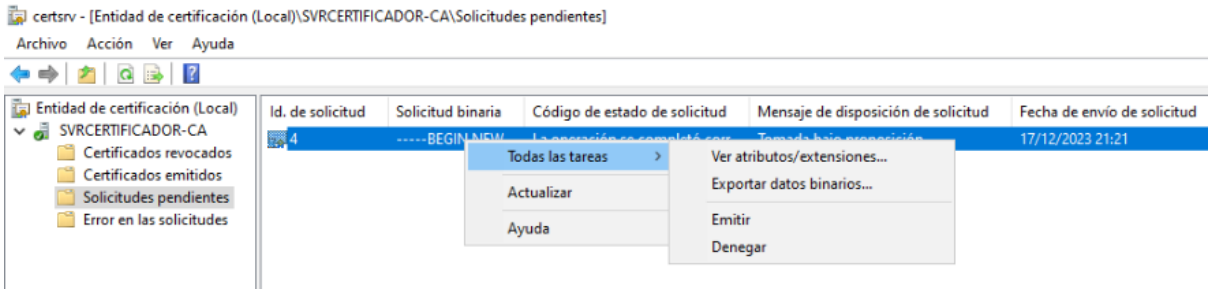


Fig. 31. Emisión de certificado

Nota. La imagen muestra el certificado en el certsrv, en el apartado de solicitudes pendientes, para poder ser emitido.

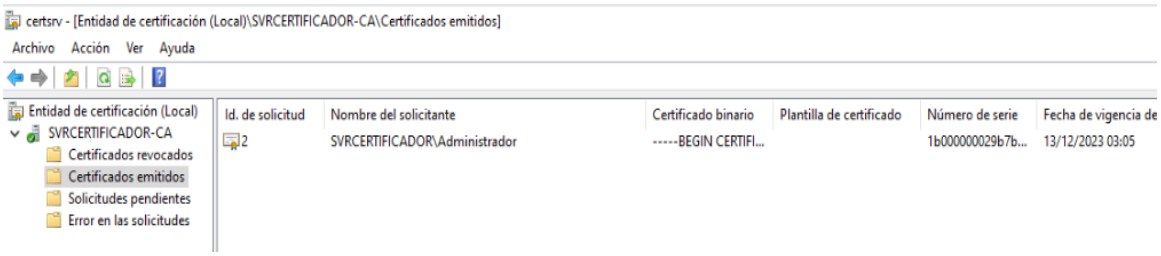


Fig. 32. Certificado emitido

Nota. En la imagen mostramos el certificado en el apartado de certificado emitidos, listo para su uso.

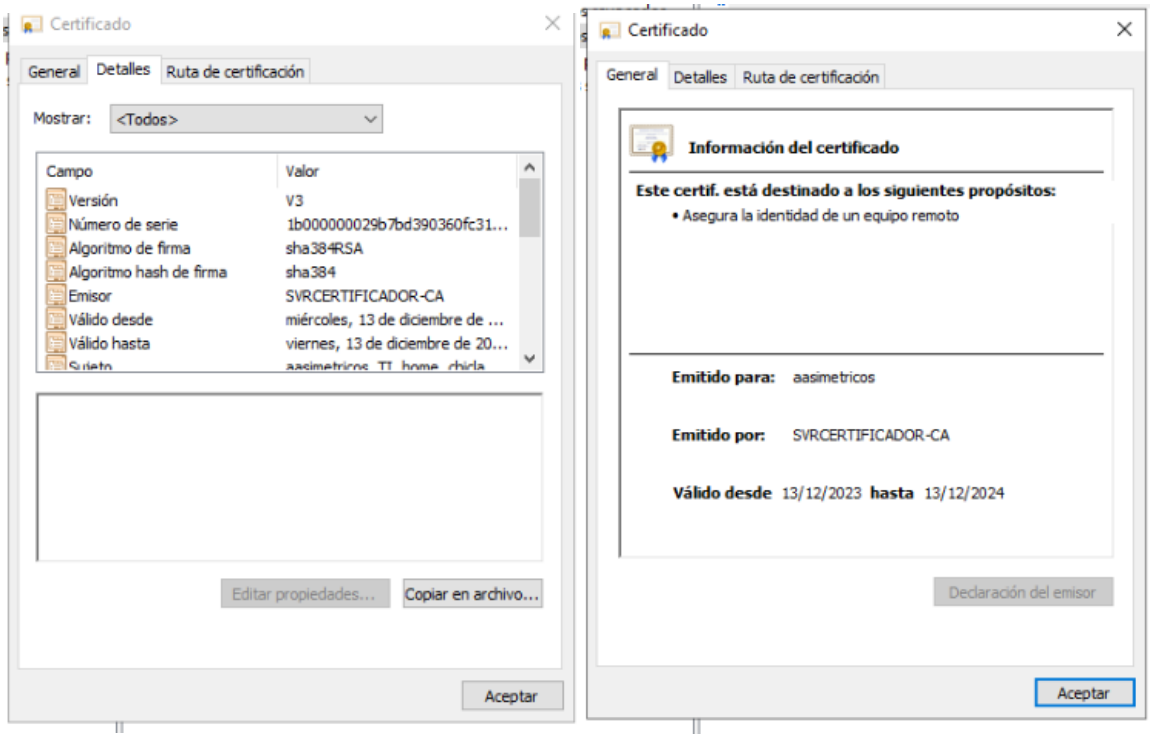


Fig. 33. Detalle de certificado

Nota. En la imagen podemos apreciar los detalles del certificado ya emitido, donde se visualiza la combinación de algoritmos que tiene dicho certificado.

Anexo 08:

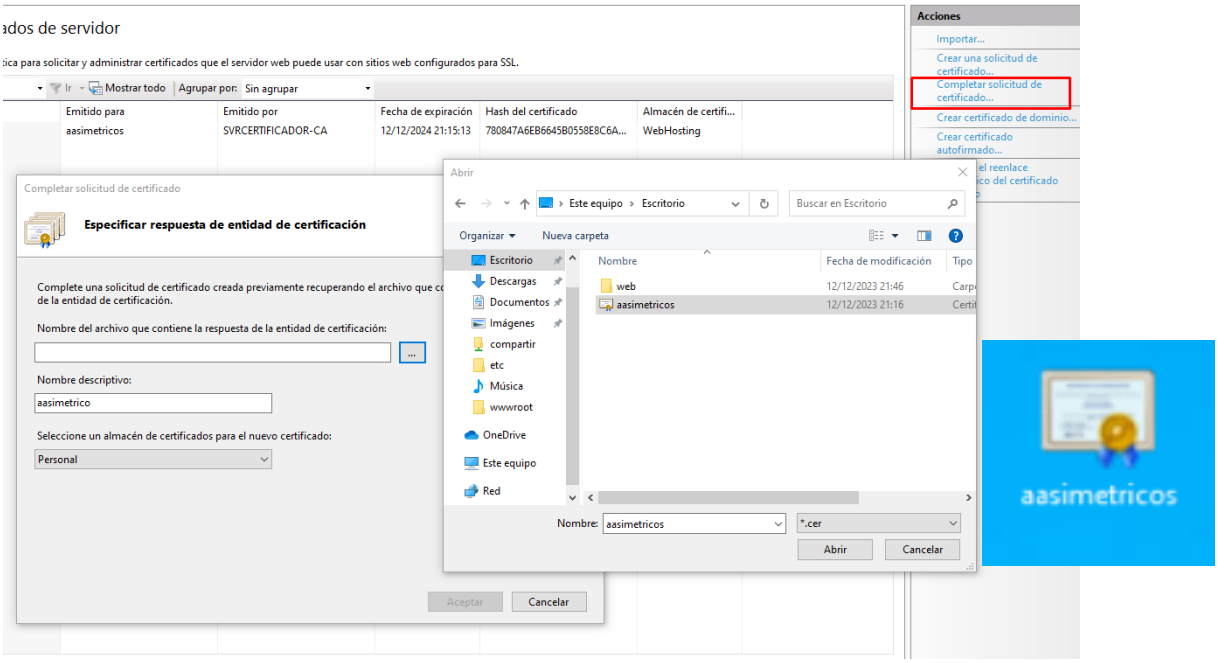
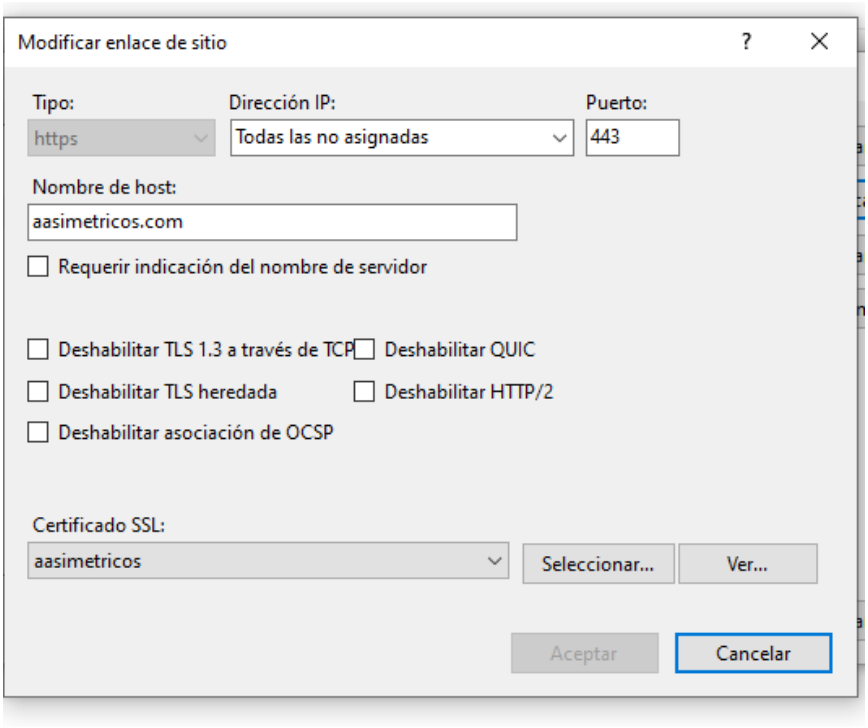


Fig. 34. Complementación de la solicitud de certificado.

Nota. La imagen muestra la interfaz para especificar la respuesta de la entidad de certificación en ADCS. Aquí se completan detalles como el nombre de la entidad de certificación y se confirma la solicitud de certificado. Esta configuración es crucial para gestionar y validar certificados digitales en un entorno de red.



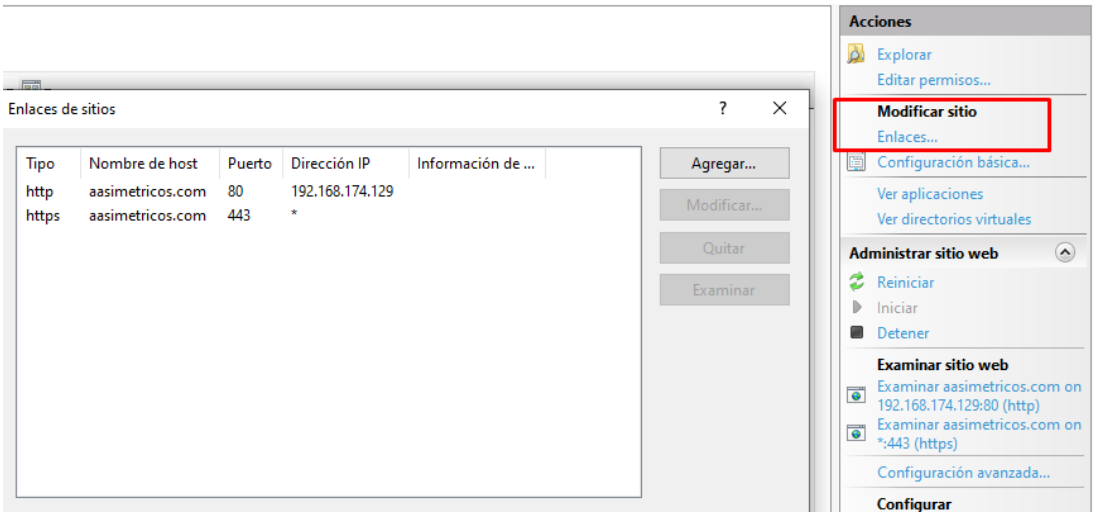


Fig. 35. Agregar enlace HTTPS.

Nota. La imagen muestra la configuración de un enlace de sitio en un servidor, donde se especifica la dirección IP, el puerto (443), y el nombre del host (aasimetricos.com). También se pueden habilitar o deshabilitar opciones avanzadas como TLS 1.3, HTTP/2, y la asociación de OCSP. Además, se selecciona y visualiza el certificado SSL correspondiente.

En el servidor IIS, incorporamos el certificado proporcionado por la entidad certificadora (CA), lo que nos permite utilizar la página web de forma segura mediante HTTPS.

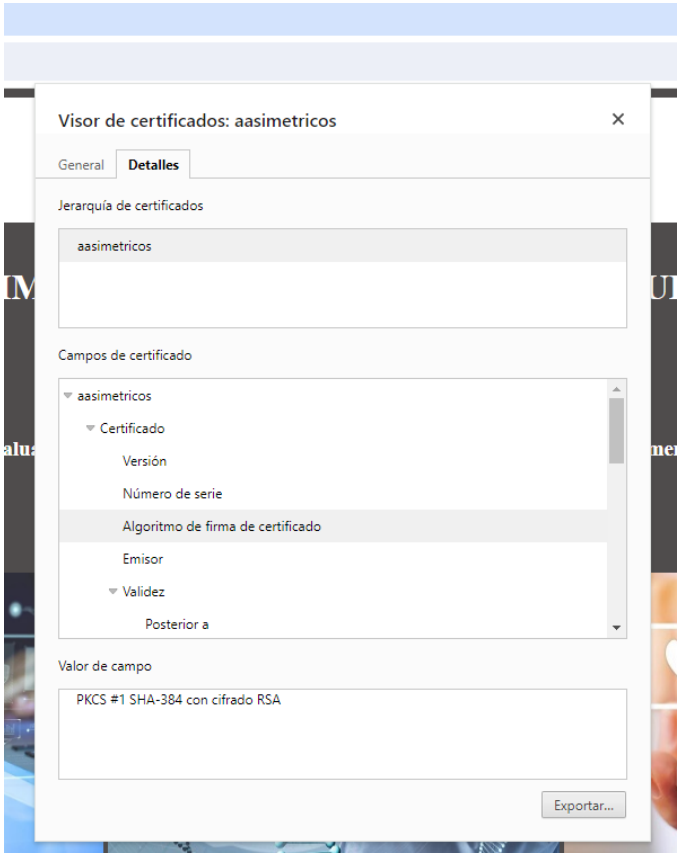


Fig. 36. Página web con HTTPS

Nota. La imagen muestra el visor de certificados para “aasimetricos”, donde se pueden ver detalles importantes del certificado como la versión, el número de serie, el algoritmo de firma, el emisor y la validez. También se ofrece la opción de exportar el certificado.

Ingresamos desde otro explorador a nuestro servidor IIS y observamos que la entidad certificadora reconoce el sitio, asegurando que la conexión con el servidor está encriptada.

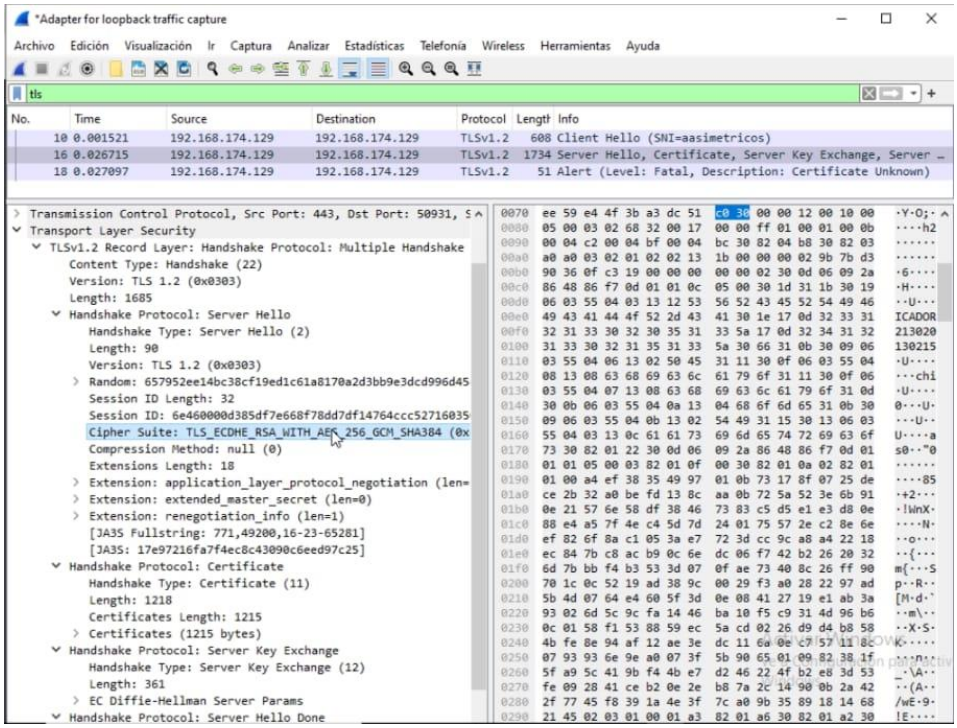


Fig. 37. tráfico SSL/TLS.

Nota. La imagen muestra una captura de tráfico de red que ilustra el handshake inicial del protocolo SSL, esencial para establecer una comunicación segura.

La captura de tráfico muestra que la figura 38 representa la negociación inicial del protocolo SSL, conocido como handshake, que se establece en la primera instancia.

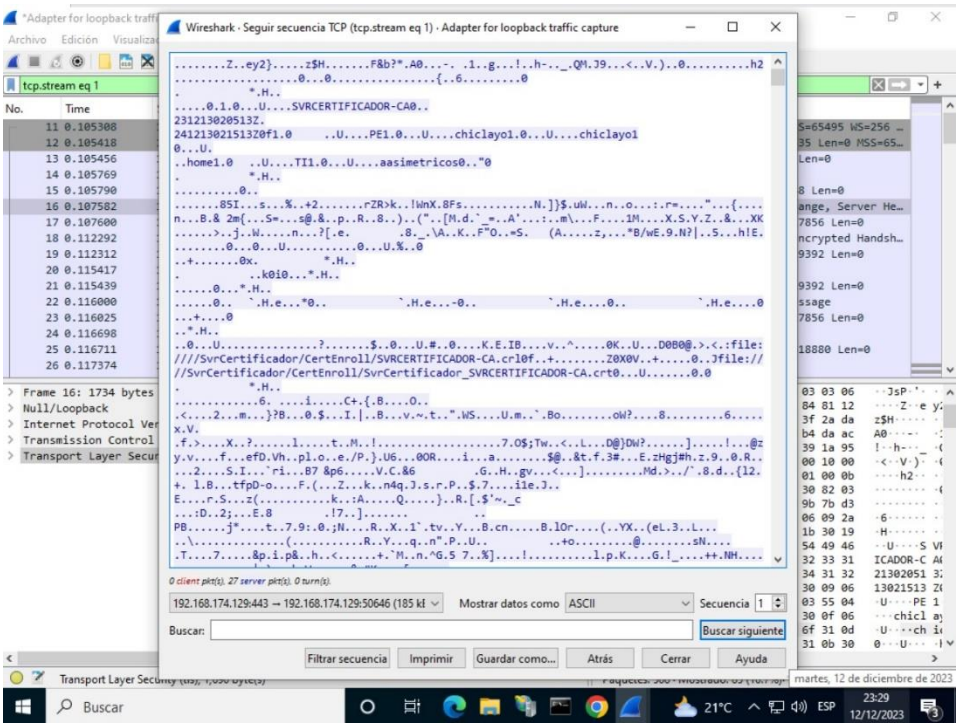


Fig. 38. Datos transmitidos con cifrado.

Nota. La imagen muestra una captura de tráfico de red en Wireshark, indicando que los datos transmitidos están cifrados. Esto es crucial para asegurar la protección de la información contra accesos no autorizados

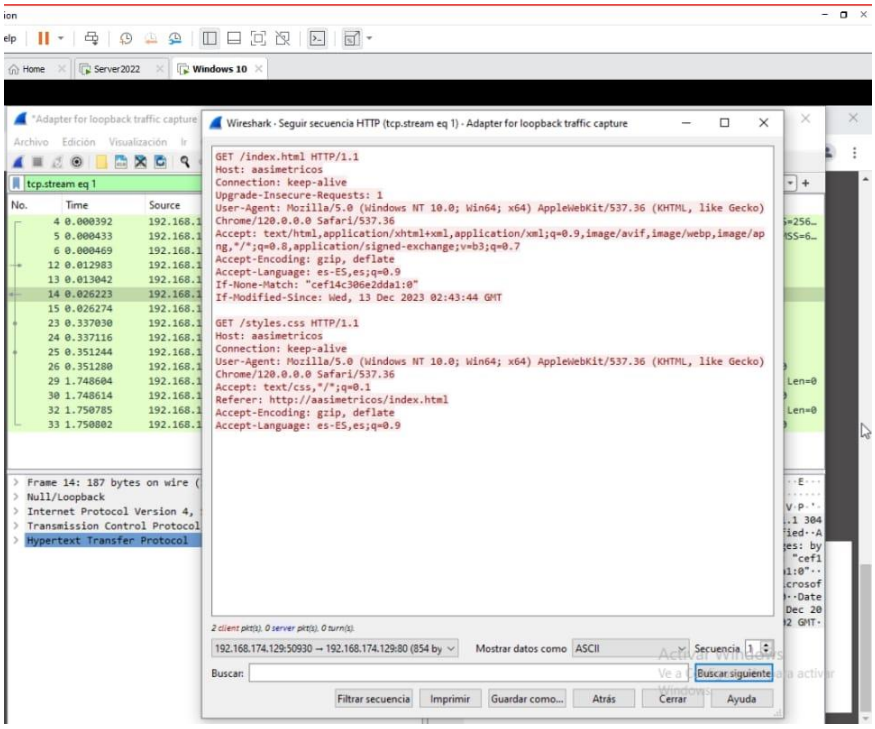


Fig. 39. Datos transmitidos sin cifrado.

Nota. La imagen muestra una captura de tráfico de red en Wireshark, indicando que los datos transmitidos no están cifrados. Esto es crucial para entender que la información puede ser interceptada y leída fácilmente, lo que representa un riesgo de seguridad.

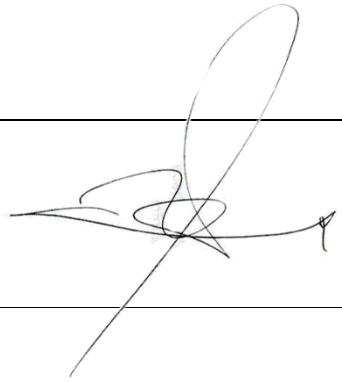
ANEXO 02: ACTA DE REVISIÓN DE SIMILITUD DE LA INVESTIGACIÓN

Yo **Samillan Ayala, Alberto Enrique** docente del curso de **Investigación II** del Programa de Estudios de **Ingeniería de Sistemas** y revisor de la investigación del (los) estudiante(s), **Campos Davila Marcos Eduardo, Castro Quesquen Jaime Elton**, titulada:

EVALUACIÓN DE ALGORITMOS ASIMÉTRICOS PARA MEJORAR LA SEGURIDAD DE LOS SEGMENTOS EN LA CAPA DE TRANSPORTE

Se deja constancia que la investigación antes indicada tiene un índice de similitud del **10%**, verificable en el reporte final del análisis de originalidad mediante el software de similitud TURNITIN. Por lo que se concluye que cada una de las coincidencias detectadas no constituyen plagio y cumple con lo establecido en la Directiva sobre índice de similitud de los productos académicos y de investigación en la Universidad Señor de Sipán S.A.C., aprobada mediante Resolución de Directorio N° 145-2022/PD-USS.

En virtud de lo antes mencionado, firma:

Samillan Ayala, Alberto Enrique	DNI: 18134651	

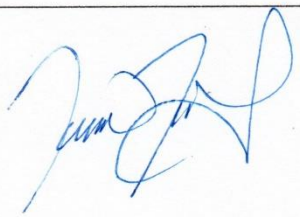
Pimentel, 19 de diciembre del 2023.

Universidad
Señor de Sipán

ANEXO 03: ACTA DE APROBACIÓN DEL ASESOR

Yo **Bravo Ruiz Jaime Arturo**. quien suscribe como asesor designado mediante Resolución de Facultad N° **0771-2023/FIAU-USS**, del proyecto de investigación titulado **Evaluación de Algoritmos Asimétricos para mejorar la Seguridad de los Segmentos en la Capa de Transporte.**, desarrollado por los estudiantes: **Campos Dávila Marcos Eduardo, Castro Quesquén Jaime Elton**., del programa de estudios de **Ingeniería de Sistemas**, acredito haber revisado, realizado observaciones y recomendaciones pertinentes, encontrándose expedito para su revisión por parte del docente del curso.

En virtud de lo antes mencionado, firman:

Bravo Ruiz Jaime Arturo (**Asesor**)	DNI: 17610253	
Campos Dávila Marcos Eduardo (**Autor 1**)	DNI: 47418197	
Castro Quesquén Jaime Elton (**Autor 2**)	DNI: 45602729	

Pimentel, 19 de diciembre del 2023