UNIVERSIDAD TÉCNICA DE BABAHOYO

FACULTAD DE ADMINISTRACIÓN, FINANZAS E INFORMÁTICA

PROCESO DE TITULACIÓN

NOVIEMBRE 2023 – ABRIL 2024

EXAMEN COMPLEXIVO DE GRADO O DE FIN DE CARRERA

PRUEBA PRÁCTICA

PREVIO A LA OBTENCIÓN DEL TITULO DE

INGENIERO(A) EN SISTEMAS DE INFORMACIÓN

TEMA:

ANÁLISIS DE VULNERABILIDADES, PENTESTING Y ACCIONES CORRECTIVAS SOBRE EL SISTEMA ACADÉMICO INTEGRAL DE LA UNIVERSIDAD TÉCNICA DE BABAHOYO

EGRESADO:

JOSE MANUEL TOMALA VERA

TUTOR:

ING. CARLOS JULIO SOTO VALLE

AÑO

2024

Índice

RESUMEN

El sistema académico integral (SAI), al igual que otras instituciones de educación superior, se ha convertido en una aplicación esencial en la gestión y procesos de la información relacionados tanto a docentes, estudiantes, empleados, matriculación, cursos, calificaciones y demás procesos académicos y administrativos. Esta información, al ser considerada de carácter sensible, debe estar salvaguardado de algún tipo amenaza externas e internas que comprometan su seguridad y privacidad.

El análisis de vulnerabilidad, las pruebas de pentesting y las medidas preventivas son componentes esenciales de la protección de la información para el sistema académico integral de la Universidad Tecnológica de Babahoyo. La falta de seguridad en el sistema puede tener consecuencias negativas, incluidas violaciones de la privacidad y pérdida de confianza dentro de la comunidad universitaria. Al adoptar un enfoque proactivo, las universidades pueden reducir la probabilidad de violaciones de seguridad y proteger la privacidad, integridad y disponibilidad de la información confidencial de la institución.

ABSTRACT

The comprehensive academic system (SAI), like other higher education institutions, has become an essential application in the management and processes of information related to teachers, students, employees, enrollment, courses, grades and other academic processes. and administrative. This information, being considered sensitive, must be safeguarded from some type of external and internal threats that compromise its security and privacy.

The analysis of vulnerabilities, pentesting and preventive actions on the Comprehensive Academic System of the Technical University of Babahoyo is established as a fundamental part of the protection of information, the lack of security in the system could have a negative impact, including violation of privacy, loss of trust, within the university community. By taking a proactive approach, the university can reduce the likelihood of security breaches and protect the privacy, integrity, and availability of the institution's sensitive data.

Keywords: Vulnerability, Pentesting, Threats, Privacy, Security.

PLANTEAMIENTO DEL PROBLEMA

Al hablar de vulnerabilidades de sitios web lo primero que se imagina es que se refiere de un ataque el cual no puede ser detectado, es ahí cuando la palabra “hacker” tiene mayor preponderancia, en este estudio de caso analiza las vulnerabilidades, ataques más comunes y sus posibles acciones preventivas. En el panorama actual del país, en el que se realizan muchas violaciones de la información a diario, se podría decir que son pocos los sitios web están expuestos a estos peligros.

El SAI es un sistema académico integral usado en la UTB, este sistema maneja información muy sensible es por eso que es necesario un análisis exhaustivo de vulnerabilidades para poder garantizar la integridad de la información y la seguridad de la misma.

El sistema académico integral (SAI), al igual que otras instituciones de educación superior, se ha convertido en una aplicación esencial en la gestión y procesos de la información relacionados tanto a docentes, estudiantes, empleados, matriculación, cursos, calificaciones y demás procesos académicos y administrativos. Esta información, al ser considerada de carácter sensible, debe estar salvaguardado de algún tipo amenaza externas e internas que comprometan su seguridad y privacidad.

Es fundamental la auditoría o evaluación al Sistema Académico Integral para poder garantizar que los datos estén seguros y poder mitigar posibles ataques, este análisis nos permitirá tomar medidas preventivas y/o preventivas en el sistema

JUSTIFICACIÓN

En el contexto actual de la era digital, la seguridad de la información se vuelve crucial para todas las organizaciones, incluyendo las instituciones educativas de tercer nivel, la Universidad Técnica De Babahoyo se encuentra ante el desafío de asegurar su Sistema Académico Integral frente a posibles amenazas cibernéticas.

Este estudio de caso se justifica, al demostrar la importancia de la evaluación informática al Sistema Académico Integral de la UTB, ya que este análisis incluirá auditorias para verificar las vulnerabilidades presentes, así como ataques controlados para poder identificar posibles puntos débiles.

El robo o pérdida de datos e información puede afectar mucho a la comunidad universitaria de la UTB y traer consecuencias graves como pérdida de reputación hasta problemas legales. La Universidad Técnica De Babahoyo tiene la necesidad imperante de que sus datos sean seguros, debido a que la institución maneja una extensa cantidad de datos confidenciales, es posible que la falta de seguridad llegue a ocasionar pérdidas o robos de datos de estudiantes, docentes e incluso académicos.

El análisis de las vulnerabilidades, pentesting y acciones preventivas sobre el Sistema Académico Integral de la Universidad Técnica De Babahoyo se establece como una parte primordial en la protección de la información, la falta de seguridad en el sistema podría tener un impacto negativo, incluida la violación de la privacidad, pérdida de confianza, dentro de la comunidad universitaria. Al adoptar un enfoque proactivo, la universidad puede reducir la probabilidad de violaciones de seguridad y proteger la privacidad, la integridad y la disponibilidad de los datos sensibles de la institución.

OBJETIVOS

GENERAL

Evaluar la seguridad del Sistema Académico Integral (SAI) de la Universidad Técnica de Babahoyo mediante un análisis exhaustivo de vulnerabilidades, pruebas de penetración y la implementación de acciones preventivas para mitigar los riesgos identificados.

ESPECÍFICOS

Identificar y documentar las vulnerabilidades que existen en el Sistema Académico Integral de la Universidad Técnica de Babahoyo.

Determinar las herramientas de pentesting para evaluar el Sistema Académico Integral de la Universidad Técnica de Babahoyo, ante posibles ataques externos e internos a la institución.

Proponer acciones preventivas para disminuir las vulnerabilidades detectadas y mejorar la seguridad del Sistema Académico Integral de la Universidad Técnica de Babahoyo.

LÍNEAS DE INVESTIGACIÒN

Este caso de estudio se encamina a la línea de investigación "Sistemas de información y comunicación, emprendimiento e innovación", la misma que guarda una relación con el análisis de vulnerabilidades, pentesting y acciones preventivas sobre el Sistema Académico Integral de la Universidad Técnica De Babahoyo. Ambas intervienen con el objetivo de hallar soluciones tecnológicas y provocar la innovación en el ámbito de la seguridad informática.

La sublínea de investigación "Redes y tecnologías inteligentes de software y hardware" está afín al caso de estudio, ya que involucra un análisis de la interacción entre dispositivos, lo que facilita las posibles vulnerabilidades.

Estas líneas de investigación facilitan un marco teórico y conceptual apropiado para el estudio del análisis de vulnerabilidades, pentesting y acciones preventivas sobre el Sistema Académico Integral de la Universidad Técnica De Babahoyo. El enfoque tecnológico admitirá una evaluación minuciosa de las vulnerabilidades, así como la propuesta de acciones preventivas que favorezcan al fortalecimiento de la protección de los sistemas.

MARCO CONCEPTUAL

En este estudio de caso, se Analiza la información bibliográfica de varios investigadores utilizando métodos analíticos e integrales, comprender las principales características de los métodos de prueba de penetración de aplicaciones de red y obtener información relevante, analizar las herramientas de evaluación utilizadas y obtener resultados mediante análisis comparativos.

Seguridad Informática

La noción de seguridad abarca diversas aplicaciones. En líneas generales, se puede entender como la salvaguarda frente a posibles peligros, daños o riesgos. En esencia, implica fortalecer la protección de algo expuesto a amenazas. Cuando algo es seguro, se caracteriza por su solidez, seguridad e indiscutibilidad, lo que lo convierte en una certeza palpable.

En su libro, Jorge Aguirre, un experto en seguridad informática, describe la seguridad de un sistema informático como la cualidad de estar exento de peligros (Aguirre, 2018). Por lo tanto, un concepto apropiado para la seguridad informática consiste en proporcionar fiabilidad e integridad tanto al hardware como al software, mediante procesos específicos que garanticen la protección de los datos contra accesos no autorizados.

Las herramientas de análisis para equipos de red son utilizadas para determinar los servicios en ejecución en un equipo remoto (Jose, 2019). Los test de pentesting, por su parte, establecen un conjunto de técnicas y metodologías predestinadas a la evaluación de los niveles de seguridad de un sistema.

Vulnerabilidades

El hacking es una herramienta decisiva para auditar sitios web, ya que a través de ella se puede identificar las vulnerabilidades que podrían generar pérdidas significativas a nivel organizacional. Por consiguiente, la realización de estas pruebas en un sitio web antes de su

lanzamiento es necesario para una organización, ya que el seguimiento de políticas de seguridad minimiza el riesgo de ser blanco de ataques. (Hu, Beuran, & Tan, 2020)

Pentesting

El pentesting o Test de Penetración es un servicio de Ética Hacking, completamente transparente para los usuarios que admite la simulación de un ciberataque dirigido a la infraestructura tecnológica de las organizaciones, explotando las vulnerabilidades o brechas de seguridad que se descubran en la red, valorando el nivel de seguridad y respuesta ante un potencial ataque real. (Bernhard Dieber, 2020)

El objetivo del Ethical Hacking es la detección de las debilidades de seguridad existentes en los sistemas, de la organización. Esto permitirá determinar el grado de impacto en el que puede ver mermada la confidencialidad, integridad y disponibilidad de la información en los sistemas.

Fases de un Pentesting

Preparación: En esta fase se elaboran los escenarios y herramientas para la realización de las pruebas.

Reconocimiento: Esta etapa radica en concebir y estar al tanto de la real situación de la estructura tecnológica de la organización.

Escaneo y Análisis de Vulnerabilidades: Aquí se efectúa una interacción inmediata con la organización, su infraestructura tecnológica y dispositivos, escaneando, cumpliendo la búsqueda y localización de posibles vulnerabilidades.

Explotación: En esta fase se ejecuta la explotación de las vulnerabilidades descubiertas, pretendiendo acceder a los sistemas web.

Mantener acceso: Se procura estar dentro de la red de la organización sin ser detectado, con la finalidad de realizar la extracción la mayor información posible.

Limpiar Rastros: En esta fase se procede a la eliminación o encapsulamiento de cualquier tipo de información que permita revelar la existencia de personas externas tuvieron acceso a la red.

Ataques

De acuerdo a (Becerril, 2019), Los ataques que son llamados activos causan el máximo daño al sistema al obligar a los servicios a cambiar la configuración de cada servicio o detener su ejecución. Los ataques activos son visibles porque sus efectos pueden detectarse a simple vista. En esta categoría se encuentran; Denegación de servicios, Búfer Overflows, Spoofing, MITM – Man In The Middle, TCP/IP Hijacking, Ingeniería social.

Los ataques pasivos no afectan directamente a la red de la víctima; simplemente escuchan lo que se transmite a través de la red y recopilan información importante, desde conversaciones hasta claves de seguridad. (https://revistas.unesum.edu.ec/index.php/unesumciencias/article/download/316/298/, 2020)

Durante años, el software malicioso, conocido como ataques de malware, ha sido la forma más conocida de infección informática debido a su capacidad de modificarse para ser detectado y propagarse por Internet a través del correo electrónico. Electrónica, unidades USB, descargas de sitios menos conocidos. (Reid‡, 2020)

La ingeniería social es la forma más activa de propagar este tipo de infecciones, ya que, mediante enlaces o correos electrónicos con contenido atractivo, se descargan en el ordenador y permanecen allí hasta que se ejecutan y alcanzan su objetivo.

Los ataques de denegación de servicio explotan la disponibilidad de la red al sobrecargar un sistema con solicitudes simultáneas que exceden la capacidad del sitio o del sistema para responder, deteniendo su funcionamiento normal. Un desbordamiento de memoria es la forma más utilizada por atacantes, por los errores cometidos por los programadores.

El phishing suministra información inexistente sobre su identidad para conseguir acceso no autorizado a un sistema. El ejemplo más clásico es la suplantación de IP, en la que un atacante utiliza la dirección de origen de otra máquina para crear un paquete IP. (Guachamín Guevara, 2020)

Los citados Ataque Man in The Middle o también llamados ataques de intermediario capturan la red colocándose entre la puerta de enlace y el servidor o la red. Esto se logra mediante un ataque ARP (Protocolo de resolución de direcciones) que utiliza la máquina del atacante como puerta de enlace, luego cambie el Gateway Mac al Hacker Mac.

Manual de Metodología OWASP

Es uno de los estándares profesionales más completos que brinda orientación sobre métodos abiertos de prueba de penetración de seguridad

Este método se divide en varias partes:

Seguridad de la información.

Seguridad de los procesos.

Seguridad en las tecnologías de internet.

Seguridad en las comunicaciones.

Seguridad inalámbrica.

Seguridad Física.

Las clasificaciones de ataques basadas en la vulnerabilidad incluyen:

Inyección: los errores de inyección ocurren cuando se envían datos no confidenciales al intérprete como parte de un comando o consulta, como SQL, OS, LDAP, en un intento de engañar al intérprete para que realice comandos no deseados o acceda a datos no autorizados.

Secuencias de comandos entre sitios: se produce un error XSS cuando una aplicación obtiene datos no confidenciales y los envía a un navegador web sin la validación y el cifrado adecuados. XSS permite a un atacante ejecutar scripts en el navegador de la víctima, secuestrando así sesiones de usuario, desfigurando sitios web o redirigiendo a los usuarios a sitios web maliciosos. (Catuto Pilay, 2020)

Configuración de seguridad incorrecta. Una buena seguridad requiere definir e implementar configuraciones de seguridad para aplicaciones, marcos, servidores de aplicaciones, servidores web, bases de datos y plataformas. Todas estas configuraciones deben definirse, implementarse y mantenerse porque a menudo son inseguras de forma predeterminada. (Miranda Jiménez, 2021)

Exposición de datos confidenciales: varias aplicaciones web no resguardan convenientemente los datos confidenciales, como números de tarjetas de crédito o información de autenticación. Los datos confidenciales demandan métodos de protección especiales, como el cifrado de información y previsiones en lo referente al intercambio datos con los navegadores.

Falsificación de solicitudes entre sitios (CSRF): un ataque CSRF hace que el navegador de una víctima autenticada envíe una solicitud HTTP falsificada a una aplicación web vulnerable, incluso la sesión del usuario y alguna otra información de autenticación.

Se han definido un conjunto de métodos para detectar vulnerabilidades, tales como:

Black Box: Este método se basa en descubrir vulnerabilidades en aplicaciones web y probar la aplicación desde la perspectiva de un atacante.

Caja Blanca: Está en el lado del servidor. Con este tipo de enfoque se puede acceder a información esencial sobre la organización.

Análisis de código estático (auditoría de código fuente): Este es un método que no requiere la ejecución del programa y realiza un análisis directo del código fuente para identificar vulnerabilidades de seguridad

Análisis de código dinámico: interactúa con una aplicación web a través del front-end de la aplicación para identificar posibles agujeros y debilidades de seguridad en la arquitectura de la aplicación web

Pruebas de penetración: Este proceso implica un escaneo proactivo del sistema en busca de posibles vulnerabilidades que pueden deberse a una configuración deficiente o inapropiada del sistema, errores de hardware o software conocidos y desconocidos, errores de rendimiento del proceso o contramedidas técnicas.

Pruebas pasivas: Están delineadas para el análisis del tráfico de información. Permite detectar errores y vulnerabilidades de seguridad examinando los paquetes capturados. (https://revistas.unesum.edu.ec/index.php/unesumciencias/article/download/316/298/, 2020)

Verificación aciva: utilice un programador de subprocesos asignado aleatoriamente para verificar que las advertencias informadas por el análisis predictivo de aplicaciones sean errores verdaderos (JA Ovallos-Ovallos, 2020).

Pruebas de fugas (pruebas de caja negra): Implica el uso de datos aleatorios o mutados para estimular el sistema bajo prueba para detectar comportamientos no deseados, como la violación de la privacidad (Chowdhary, y otros, 2020).

Las principales herramientas de detección de vulnerabilidades son:

Auditoría de seguridad del sitio web - WSSA: Le permite verificar páginas web, aplicaciones y servidores web en busca de vulnerabilidades de seguridad.

Nessus Vulnerability Scanner: además de las vulnerabilidades de OWASP, también le permite escanear servidores y servicios web en busca de vulnerabilidades. Además de

comprobar errores de configuración del sistema y parches faltantes. Muestre informes personalizados en formatos XML, CVS, PDF nativo y HTML (Colque, 2020).

Whatweb: identifica tecnologías de redes y sitios web, incluidos sistemas de gestión de contenidos (CMS), bibliotecas de JavaScript y servidores web. (Robayo García, 2021)

Contar aplicaciones en un servidor web. Esta prueba está diseñada para conocer la cantidad de aplicaciones que se ejecutan en un servidor. Mediante esto podemos determinar que estamos ejecutando la aplicación solamente en el servidor Apache, mediante la herramienta Nmap.

Kali Linux

Kali Linux, es un sistema operativo de código open source diseñado para llevar a cabo auditorías de seguridad informática. Junta más de 300 herramientas manejadas en pruebas de pentesting, lo que ayuda a los administradores valorar la efectividad de sus medidas de seguridad y mitigación de riesgos (Caballero, 2022). Dentro de las muchas herramientas que posee este sistema operativo, se encuentran Metasploit para pruebas de pentesting, Nmap para escaneo de puertos, Wireshark para monitorear el tráfico de (Caballero, 2022).

The Harvester

La herramienta The Harvester, que se encuentra en la distro Kali Linux, es utilizada en la recolección de direcciones de correo electrónico asociadas a un dominio en particular, lo que puede ser de ayuda en la realización de ataques dirigidos (Stratton-Porter, 2019)

MARCO METODOLÓGICO

Métodos De Investigación

Método Documental

La metodología documental nos permitió utilizar un estudio de caso, obtener recursos de información a través de citas bibliográficas, artículos académicos y revistas sobre el tema "Análisis de vulnerabilidades, pentesting y acciones preventivas sobre el Sistema Académico Integral de la Universidad Técnica De Babahoyo.", la metodología de este estudio proporcionará la base técnica para la definición de las herramientas a utilizar, comenzando por el análisis y presentación de los objetivos planteados..

Método deductivo

Mediante el uso de un método de investigación deductivo se identificó un estudio de caso basado en los elementos temáticos "Análisis de vulnerabilidades, pentesting y acciones preventivas sobre el Sistema Académico Integral de la Universidad Técnica De Babahoyo.", conclusiones y recomendaciones basadas en resultados de entrevistas, concisas y útiles para el desarrollo del proyecto.

Tipos de Investigación

Investigación Bibliográfica

El uso de la investigación bibliográfica es de notable importancia, ya que permitirá tener de manera clara las bases teóricas por el autor del estudio de caso en referencia a la propuesta planteada, lo que admite la obtención de los elementos sobre el tema "Análisis de vulnerabilidades, pentesting y acciones preventivas sobre el Sistema Académico Integral de la Universidad Técnica De Babahoyo.", este estudio de caso está basado en las citas bibliográficas de revistas, artículos académicos y sitios web, lo que ayudara en la fundamentación teórica del estudio de caso a desarrollar.

Investigación Aplicada

El desarrollo de estudio de caso sobre el tema “Análisis de vulnerabilidades, pentesting y acciones preventivas sobre el Sistema Académico Integral de la Universidad Técnica de Babahoyo.”, mediante La metodología Abierta de Testeo de Seguridad (OWAP) y especificaciones técnicas de herramientas open source (Kali Linux, nmap), como resultado conseguiremos un test de pentesting correcto y documentar posibles vulnerabilidades dentro del Sistema Académico Integral (SAI).

Población

Para el desarrollo de este trabajo de investigación se realizaron entrevistas a 2 personas que trabajan en la Dirección de Tecnologías y Sistemas Informáticos de la UTB, al Ing. Alexander Izquierdo Lara, y a el analista Ing. Dennis Álava quienes forman parte de la Unidad de Proyectos y Soluciones Tecnológicas, como encargados de desarrollo y soporte del Sistema Académico Integral de la Universidad Técnica De Babahoyo.

Técnica de Investigación

Entrevista

A través de esta técnica se junta la información más relevante en el desarrollo de un cuestionario creado por el investigador para la conversación a los entrevistados y conocer sus apreciaciones en base a la aplicación y evaluación del sistema SAI, mediante la aplicación de pentesting en la Universidad Técnica de Babahoyo.

Metodología OWASP

Esta metodología OWASP recopila y destaca las brechas de seguridad en un sistema web y cómo los atacantes manipulan estos puntos de entrada. En este caso de estudio, se apeló a el uso de una metodología llamada OWASP ZAP, para efectuar el análisis de vulnerabilidades del Sistema Académico Integral, los cuales amenazan la información de la comunidad universitaria.

RESULTADOS

Escaneo de Vulnerabilidades del Sistema Académico Integrado (SAI)

Metodología OWASP

Fases de un pentesting

Reconocimiento

Fase principal a través de la cual hemos procedido a la recolección de la información de sitio web, usando diferentes técnicas como:

Recopilación de IPs

Recopilación de servicios

Recopilación de puertos

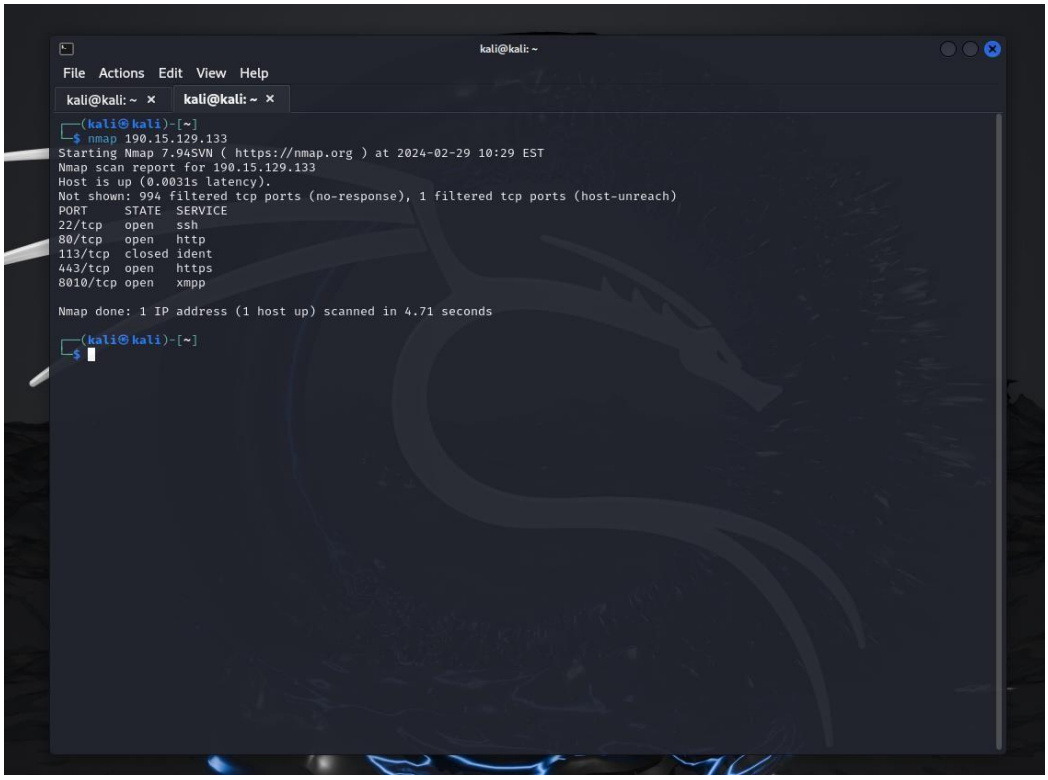

Ilustración 1Recopilación de información

Análisis De Vulnerabilidades

Aquí realizamos y analizamos la información adquirida en la etapa anterior y procedemos a visualizar de las vulnerabilidades y poder intentar la búsqueda de CVEs (Common Vulnerabilities and Exposures) conocidos y/o fáciles de explotar.

Explotación

Tuvo como propósito realizar todas las acciones que intenten comprometer al sistema web auditado, a los usuarios o la información que manipula (Laprovittera, 2023). Especialmente se demuestra que no se consigan realizar ataques tipo:

Inyección de código

Inclusión de ficheros

Evasión de autenticación

Falta de controles de autorización

Realización de comandos en el lado del server

Ataques tipo Cross Site Reques Forey

Control de errores

Gestión de sesiones

Fugas de información

Secuestros de sesión

Comprobación de las condiciones para realizar una denegación de servicio

Carga de ficheros maliciosos

Explotación Con La Herramienta ZAP

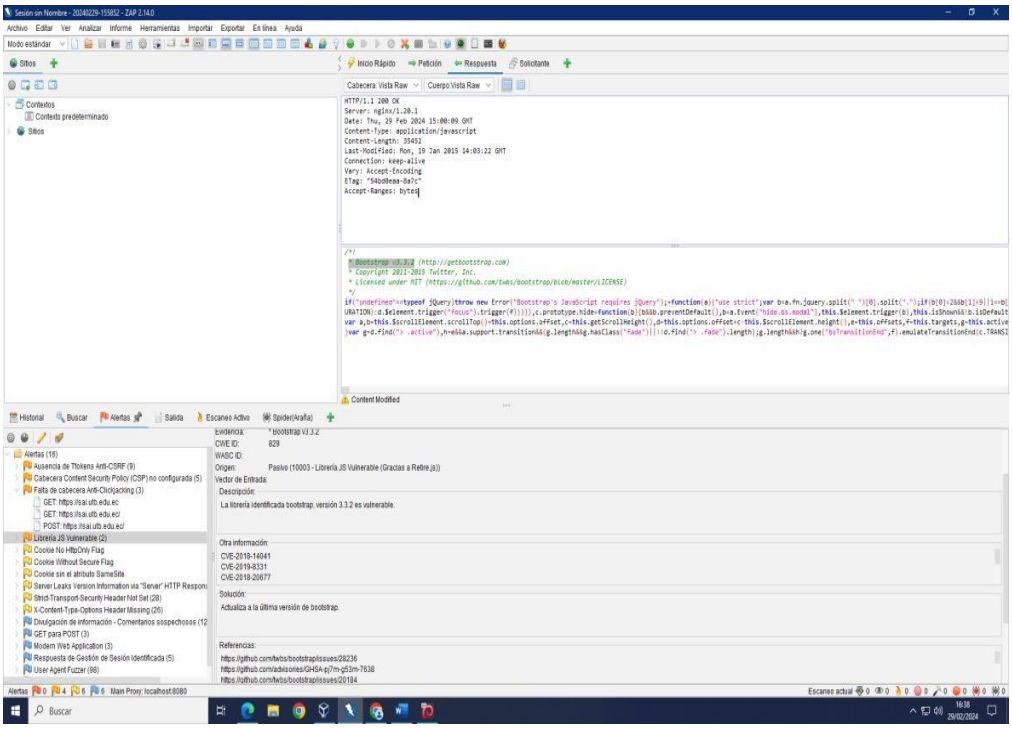

Ilustración 2 Explotación con La Herramienta ZAP

Post Explotación

Esta fase se caracterizó por la búsqueda de vulnerabilidades que permita realizara controles adicionales con el objeto de comprobar la criticidad de esta.

A continuación, se listan las acciones de post-explotación:

Obtención de información confidencial

Evasión de mecanismos de autenticación

Realizar acciones del lado de los usuarios

Realizar ejecuciones de los comandos en el servidor que aloja la aplicación

Privilegios disponibles en el servidor, si se consigue acceso al mismo

Servicios accesibles desde la aplicación comprometida

Realizar acciones sin el consentimiento o conocimiento de los usuarios

Reporte De Riesgos

Parámetros del informe

Contextos

No se seleccionó ningún contexto, por lo que todos los contextos se incluyeron de forma predeterminada.

Sitios

Se incluyeron los siguientes sitios:

- https://sai.utb.edu.ec

(Si no se seleccionaba ningún sitio, todos los sitios se incluían de forma predeterminada).

Un sitio incluido también debe estar dentro de uno de los contextos incluidos para que sus datos se incluyan en el informe.

Niveles de riesgo

Included: Alto, Medio, Bajo, Informativo

Excluido: Ninguno

Niveles de confianza

Included: Confirmado por Usuario, Alta, Media, Baja

Excluded: Confirmado por Usuario, Alta, Media, Baja, Falso positivo

Ilustración 3Reporte de Riesgos

Resúmenes

Recuentos de alertas por riesgo y confianza

En esta tabla se muestra el número de alertas para cada nivel de riesgo y confianza incluido en el informe.

(Los porcentajes entre paréntesis representan el recuento como un porcentaje del número total de alertas incluidas en el informe, redondeado a un decimal).

		Confianza				
		Confirmado por Usuario	Alta	Medio	Baja	Total
Riesgo	Contralto	0 (0,0 %)	0 (0,0 %)	0 (0,0 %)	0 (0,0 %)	0 (0,0 %)
	Medio	0 (0,0 %)	1 (6,2 %)	2 (12,5 %)	1 (6,2 %)	4 (25,0 %)
	Bajo	0 (0,0 %)	2 (12,5 %)	4 (25,0 %)	0 (0,0 %)	6 (37,5 %)
	Informativo	0 (0,0 %)	2 (12,5 %)	3 (18,8 %)	1 (6,2 %)	6 (37,5 %)
	Total	0 (0,0 %)	5 (31,2 %)	9 (56,2 %)	2 (12,5 %)	16 (100%)

Ilustración 4 Reporte de alertas por riesgo y confianza

Alertas De Riesgos

Recuentos de alertas por sitio y riesgo

Esta tabla muestra, para cada sitio para el que se generaron una o más alertas, el número de alertas generadas en cada nivel de riesgo.

Las alertas con un nivel de confianza de "falso positivo" se han excluido de estos recuentos.

(Los números entre paréntesis son el número de alertas generadas para el sitio en o por encima de ese nivel de riesgo).

		Riesgo			
		Alto (= Alto)	Medio (>= Medio)	Bajo (>= Bajo)	Informativo (>= Informativo)
Sitio	https://sai.utb.edu.ec	0 (0)	4 (4)	6 (10)	6 (16)

Ilustración 5Alerta de riesgos del SAI

Vulnerabilidades Que Se Deben Corregir

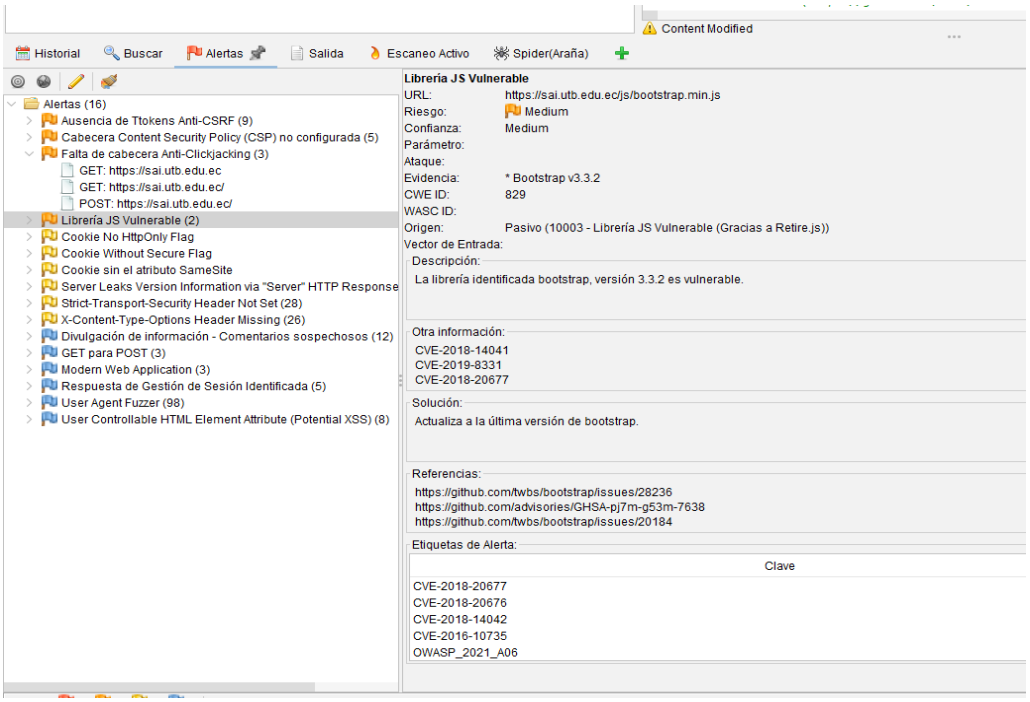

Ilustración 6 Vulnerabilidades

Divulgación De Información Análisis Sospechosos

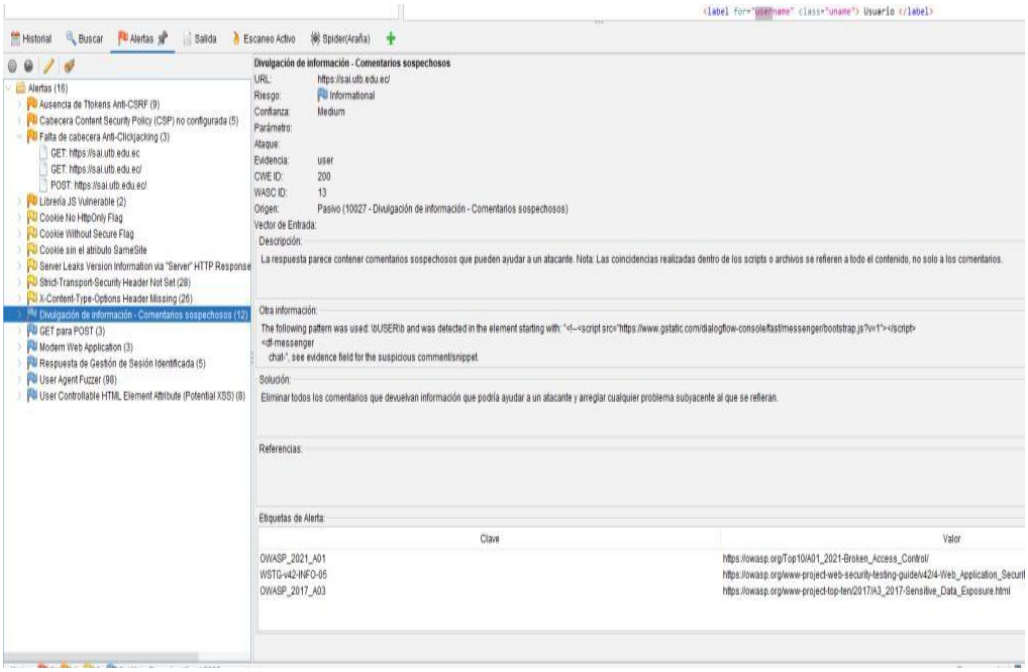

Ilustración 7Divulgación De Información Análisis Sospechosos

Alertas De Riesgos Por Tipos

Recuentos de alertas por tipo de alerta

En esta tabla se muestra el número de alertas de cada tipo de alerta, junto con el nivel de riesgo del tipo de alerta.

(Los porcentajes entre paréntesis representan cada recuento como un porcentaje, redondeado a un decimal, del número total de alertas incluidas en este informe).

Tipo de alerta	Riesgo	Contar
Ausencia de Ttokens Anti-CSRF	Medio	9 (56,2 %)
Política de Seguridad de Contenidos (CSP) de Cabecera no configurada	Medio	5 (31,2 %)
Falta de cabecera Anti-Clickjacking	Medio	3 (18,8 %)
Librería JS Vulnerable	Medio	2 (12,5 %)
Cookie No HttpOnly Flag	Bajo	1 (6,2 %)
Cookie sin bandera de seguridad	Bajo	1 (6,2 %)
Cookie sin el atributo SameSite	Bajo	

Ilustración 8Alertas De Riesgos Por Tipos

Tipos De Riesgos Y Niveles De Confianza

Ilustración 9Tipos De Riesgos y Niveles de Confianza

DISCUSIÓN DE RESULTADOS

En la fase de reconocimiento se hizo uso de la página Virus Total para escanear la dirección del sitio web y de esta forma reconocer los dominios y subdominios que tiene este sitio, así mismo ver la dirección ir en la que se encuentra, entre otros datos. Con la herramienta nmap se revisaron los puertos abiertos del sitio web en la cual se puede observar que tiene 4 puertos abiertos.

Para realizar el análisis de vulnerabilidades, se utilizó la metodología OWASP en el Sistema académico integral (SAI), encontrándonos con ausencia de Tokens y además no se encontraron fichas Anti-CSRF en ningún formulario HTML.

Cabecera CSP (Content Security Policy) No configurada

La Cabecera CSP es una capa de seguridad que permite detectar y atenuar ciertos tipos de ataques, incluidos los de secuencias de comandos entre sitios (XSS) y de inyección de datos.

Estos ataques se pueden utilizar para cualquier cosa, desde robar datos hasta destruir sitios web o difundir malware. CSP facilita un conjunto de encabezados HTTP estándar que admiten a los propietarios de sitios web exponer fuentes autorizadas de contenido que los navegadores deberían soportar al momento de cargar en una página; Los tipos cubiertos contienen elementos como JavaScript, CSS, marcos HTML, fuentes y objetos incrustados. Por ejemplo, subprogramas de Java, ActiveX, archivos de audio y vídeo.

Librería JS vulnerables

La librería identificada bootstrap, versión 3.3.2 es vulnerable.

CVE

CVE-2023-45802 Cuando un cliente restableció una secuencia HTTP/2 (trama RST), hubo una ventana de tiempo en la que los recursos de memoria de la solicitud no se recuperaron inmediatamente. En su lugar, la designación se aplazó hasta el cierre de la conexión.

Un cliente podría enviar nuevas solicitudes y reinicios, manteniendo la conexión ocupada y abierta y haciendo que la huella de memoria siga creciendo. Al cerrar la conexión, se recuperaron todos los recursos, pero es posible que el proceso se quede sin memoria antes de eso.

Esto fue descubierto por el reporte durante las pruebas de CVE-2023-44487 (HTTP/2 Rapid Reset Exploit) con su propio cliente de prueba. Durante el uso "normal" de HTTP/2, la probabilidad de encontrar este error es muy baja. La memoria guardada no se notaría antes de que se cierre la conexión o se agote el tiempo de espera. Se recomienda a los usuarios que actualicen a la versión 2.4.58, que soluciona el problema.

CVE-2023-44487 El protocolo HTTP/2 permite una denegación de servicio (consumo de recursos del servidor) porque la cancelación de solicitudes puede restablecer muchos flujos rápidamente, como se explotó en la naturaleza entre agosto y octubre de 2023.

CVE-2019-9514 Algunas implementaciones de HTTP/2 son vulnerables a una inundación de reinicio, lo que puede provocar una denegación de servicio. El atacante abre una serie de secuencias y envía una solicitud no válida a través de cada secuencia que debería solicitar una secuencia de RST_STREAM tramas del par. Dependiendo de cómo el par ponga en cola las tramas RST_STREAM, esto puede consumir un exceso de memoria, CPU o ambos.

CONCLUSIONES

La fundamentación teórica y metodológica sobre las vulnerabilidades que existen en el Sistema Académico Integral de la Universidad Técnica de Babahoyo, lograron establecer que existen diversas vulnerabilidades que logran a ser catalogadas como una amenaza constante para el sitio web, no posee un nivel de seguridad aceptable.

A través del análisis de vulnerabilidad, se seleccionó las herramientas de seguridad de la información ideales para este caso de estudio. Para el análisis de vulnerabilidades se utilizó la herramienta FOCA, para el análisis de dependencias se utilizó OWASP y Network Mapper para el análisis de la aplicación.

La utilización de la metodología OWASP, se efectuó mediante la obtención de información decisiva del del Sistema Académico Integral de la Universidad Técnica de Babahoyo, alcanzando la focalización del sitio web.

RECOMENDACIONES

Se recomienda a la Universidad Técnica de Babahoyo efectuar frecuentemente análisis de seguridad informática para impedir vulnerabilidades, garantizando el funcionamiento correcto del sitio web a la comunidad universitaria.

Es recomendado la exploración de las herramientas a ser utilizados, para el análisis del sitio web, por cuanto algunas de ellas son muy intrusivas, ocasionando el congestionamiento del ancho de banda de la red y un funcionamiento extraño en la misma como por ejemplo alguna consulta extensa con el comando de Linux NMAP.

Para un posterior análisis del sitio web, es ideal la utilización de la metodología OWASP porque efectúa un análisis exhaustivo de los archivos y procesos, incluyendo los servidores existentes en la DTSI de la Universidad Técnica de Babahoyo.

REFERENCIAS

Becerril, A. (2019). La ciberseguridad en los Tratados de Libre Comercio. *chilena de derecho y tecnología*, 8.

Bernhard Dieber, R. W. (2020). *Penetration Testing ROS.*

Caballero, A. E. (2022). *Hacking con Kali Linux.* Lima.

Catuto Pilay, R. M. (2020). *Análisis de amenazas y vulnerabilidades informáticas basado en la Norma ISO 27002, en el proceso de citas del servidor web de una Institución.* Universidad Estatal Península de Santa Elena: Universidad Estatal Península de Santa Elena.

Chowdhary, A., Huang, D., Mahendran, J. S., Romo, D., Deng, Y., & Sabur, A. (2020). Autonomous Security Analysis and Penetration Testing. Tokyo, Japan: Conference on Mobility, Sensing and Networking (MSN).

Colque, S. I. (2020). Escáner de vulnerabilidades aplicando nessus. . *Ciencia Y Tecnologia Informatica* .

Guachamín Guevara, S. D. (2020). *Análisis de herramientas para el control de vulnerabilidades informáticas e implementación de un sistema de monitoreo y alerta temprana para una empresa de construcción.* Quito: Universidad de las Américas.

https://revistas.unesum.edu.ec/index.php/unesumciencias/article/download/316/298/. (2020). Análisis de las herramientas y técnicas utilizadas en prueba de penetracion en aplicaciones web. *Revista Científica Multidisciplinaria*, 42.

Hu, Z., Beuran, R., & Tan, Y. (2020). *Automated Penetration Testing Using Deep Reinforcement Learning.* Genoa, Italy.

JA Ovallos-Ovallos, D. R.-B. (2020). Guía práctica para el análisis de vulnerabilidades de un entorno cliente-servidor GNU/Linux mediante una metodología de pentesting. *Revista Ibérica de Sistemas e Tecnologias de Informação*, 17.

Jose, M. (2019).

Laprovittera, C. (29 de Diciembre de 2023). *Guía de Hacking y Pentesting – Capítulo 9: Reporte del Pentest.* Obtenido de Guía de Hacking y Pentesting – Capítulo 9: Reporte del Pentest: https://achirou.com/guia-de-hacking-y-pentesting-capitulo-9-reporte-de-resultados-del-pentest/

Long, J. (2022). *Google Hacking for Penetration Testers, Third Edition.* Reino Unido: 9780128029640.

Miranda Jiménez, J. N. (2021). *Mapeo sistemático de metodologías de Seguridad de la Información para el control de la gestión de riesgos informáticos.* Guayaquil: Repositorio Institucional de la Universidad Politécnica Salesiana.

Reid‡, J. A. (2020). *Cone penetration testing on silty tailings using a new small calibration chamberç.*

Robayo García, L. A. (2021). *Vulnerabilidades informáticas en implementaciones con el cms wordpress.* Bogota.

Stratton-Porter, G. (2019). *The Harvester.* Estados Unidos: 9781942885214.

ANEXOS

Entrevista

Estas preguntas proporcionaran una base sólida para evaluar el Análisis de vulnerabilidades, pentesting y acciones correctivas sobre el Sistema Académico Integral de la Universidad Técnica De Babahoyo.

1. ¿Cuál es tu experiencia en el análisis de vulnerabilidades y pentesting en entornos académicos como el sistema integral de una universidad?
2. ¿Cuáles son los pasos clave que seguirías para realizar una evaluación exhaustiva de la seguridad en el sistema académico integral de la Universidad Técnica de Babahoyo?
3. ¿Qué herramientas y metodologías utilizarías para identificar y analizar posibles vulnerabilidades en el sistema académico?
4. ¿Cómo evaluarías la efectividad de las medidas de seguridad actuales implementadas en el sistema académico? ¿Qué métricas utilizarías?
5. ¿Podrías proporcionar ejemplos de vulnerabilidades comunes que podrían afectar a un sistema académico como el de la Universidad Técnica de Babahoyo?
6. ¿Qué acciones tomarías para remediar las vulnerabilidades identificadas durante el proceso de pentesting?
7. ¿Cómo te asegurarías de que las correcciones implementadas no introduzcan nuevas vulnerabilidades o problemas en el sistema?
8. ¿Qué medidas proactivas sugerirías para fortalecer la seguridad del sistema académico a largo plazo?
9. ¿Cómo abordarías las preocupaciones de privacidad y cumplimiento normativo al realizar análisis de vulnerabilidades y pentesting en un entorno académico?

10. ¿Qué estrategias utilizarías para concienciar a los usuarios y administradores del sistema académico sobre las buenas prácticas de seguridad cibernética?

Babahoyo, 20 de febrero de 2024
D-FAFI-UTB-0191-2024

Ingeniero.
Marcos Oviedo Rodríguez, Ph.D.
RECTOR DE LA UNIVERSIDAD TÉCNICA DE BABAHOYO
En su despacho. -

De mis consideraciones:

Reciba un cordial saludo por parte de la Facultad de Administración, Finanzas e Informática de la Universidad Técnica de Babahoyo, donde formamos profesionales altamente capacitados en los campos de Tecnologías de la Información y de Administración, competentes, con principios y valores cuya practica contribuye al desarrollo integral de la sociedad, es por ello que buscamos prestigiosas Empresas e Instituciones Públicas y Privadas en las cuales nuestros futuros profesionales tengan la oportunidad de afianzar sus conocimientos.

El señor **JOSÉ MANUEL TOMALA VERA,** con cédula de identidad No. **120817312-8** estudiante de la Carrera de Ingeniería en Sistemas de Información, matriculado en el proceso de titulación en el periodo NOVIEMBRE 2023 – ABRIL 2024, trabajo de titulación modalidad examen de carácter complexivo, previo a la obtención del grado académico profesional universitario de tercer nivel como INGENIERO EN SISTEMAS DE INFORMACIÓN, solicita por intermedio del Decanato de esta Facultad el debido permiso para realizar su Estudio de Caso, en el Departamento de Dirección de Tecnológicas y Sistemas de Información de la Universidad Técnica de Babahoyo, en el cual su tema es: **"ANÁLISIS DE VULNERABILIDADES, PENTESTING Y ACCIONES CORRECTIVAS SOBRE EL SISTEMA ACADÉMICO INTEGRAL DE LA UNIVERSIDAD TÉCNICA DE BABAHOYO".**

Atentamente,

Lcdo. Eduardo Galeas Guijarro MAE.
DECANO
cc: Archivo

UNIVERSIDAD TÉCNICA DE BABAHOYO FACULTAD ADMINISTRACION FINANZAS E INFORMATICA DECANATO

Recibido
27-02-2024
10:38
Rectorado.

CERTIFICADO DE ANÁLISIS
magister

JOSE TOMALA VERA

10%
Textos sospechosos

8% **Similitudes**
< 1% similitudes entre comillas
< 1% entre las fuentes mencionadas
3% **Idiomas no reconocidos**
4% **Textos potencialmente generados por la IA (ignorado)**

Nombre del documento: JOSE TOMALA VERA.docx
ID del documento: db3300a36fdfab84fe95ab5223ec08a4196f4246
Tamaño del documento original: 1,04 MB

Depositante: SOTO VALLE CARLOS JULIO
Fecha de depósito: 3/3/2024
Tipo de carga: interface
fecha de fin de análisis: 3/3/2024

Número de palabras: 4744
Número de caracteres: 32.757

Ubicación de las similitudes en el documento:

Fuentes principales detectadas

Nº	Descripciones	Similitudes	Ubicaciones	Datos adicionales
1	dspace.utb.edu.ec http://dspace.utb.edu.ec/bitstream/handle/49000/12692/E-UTB-FAFI-SIST.INF-000084.pdf?sequence... 6 fuentes similares	3%		Palabras idénticas: 3% (163 palabras)
2	www.hiberus.com | Pentesting con OWASP: fases y metodología - Blog de hiberus https://www.hiberus.com/crecemos-contigo/pentesting-owasp-fases-metodologia/#:~:text=Fases de... 3 fuentes similares	2%		Palabras idénticas: 2% (76 palabras)
3	dspace.utb.edu.ec http://dspace.utb.edu.ec/bitstream/49000/11697/3/E-UTB-FAFI-SIST.INF-000003.pdf.txt 6 fuentes similares	< 1%		Palabras idénticas: < 1% (43 palabras)
4	revistas.unesum.edu.ec https://revistas.unesum.edu.ec/index.php/unesumciencias/article/download/316/298/	< 1%		Palabras idénticas: < 1% (43 palabras)
5	dspace.udla.edu.ec | Repositorio Digital Universidad De Las Américas: Análisis de ... https://dspace.udla.edu.ec/handle/33000/12116	< 1%		Palabras idénticas: < 1% (28 palabras)

Fuentes con similitudes fortuitas

Nº	Descripciones	Similitudes	Ubicaciones	Datos adicionales
1	repository.usta.edu.co https://repository.usta.edu.co/bitstream/handle/11634/52592/2023jairrodriguez.pdf?sequence=1	< 1%		Palabras idénticas: < 1% (19 palabras)
2	TITULACION-CHIGUANO 3.docx | TITULACION-CHIGUANO 3 #60b5ce El documento proviene de mi grupo	< 1%		Palabras idénticas: < 1% (16 palabras)
3	dspace.utb.edu.ec http://dspace.utb.edu.ec/bitstream/49000/12658/1/E-UTB-FAFI-SIST.INF-000059.pdf	< 1%		Palabras idénticas: < 1% (10 palabras)
4	www.doi.org https://www.doi.org/10.1680/jGELE.20.00037	< 1%		Palabras idénticas: < 1% (11 palabras)
5	dspace.utb.edu.ec http://dspace.utb.edu.ec/bitstream/49000/2406/1/-E-UTB-FAFI-SIST-000004.pdf	< 1%		Palabras idénticas: < 1% (10 palabras)