UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

PRUEBAS DE PENETRACIÓN EN APLICACIONES WEB: CASO DE ESTUDIO PORTAL WEB DE DEPENDENCIA UNIVERSITARIA

TESIS

Que para obtener el título de

Ingeniero en Computación

P R E S E N T A

Alonso de Jesús Hernández Hernández

DIRECTORA DE TESIS

M.I.A Dulce Campos del Razo

Ciudad Universitaria, Cd. Mx., 2023

AGRADECIMIENTOS

A mi madre

Que apoyó mi sueño desde niño, apoyándome desde su corazón al darme motivación cuando ya no podía, por mi educación que me regaló para afrontarme a la vida y ser la persona que soy para poder afrontar todos los retos que vengan, te amo mamá.

A mi padre

Que me dio su apoyo en todo mi trayecto escolar y de vida, que forjó el carácter que tengo sin temor y valor a cualquier reto a enfrentar, que hizo que me definiera como un hombre de bien con concentración, compromiso y voluntad pura, te amo papá.

A mis abuelos Rafael e Isabel

Gracias por apoyarme toda mi vida, me educaron y ayudaron en proyectos que creía imposibles, les agradezco eternamente por todo su amor y cariño, los amo.

A mi tío Héctor

Gracias por enseñarme el camino de un ingeniero a mis siete años, que permitieron que definiera mi camino a esa edad y seguir tus pasos, fue un trayecto largo que con tu apoyo se hizo ligero el tiempo, gracias por enseñarme tanto.

A mis hermanos

Que me demostraron lo que es el apoyo incondicional, confiaron en mí y me apoyaron en toda adversidad para cumplir mis metas y el sueño de ser ingeniero, gracias por ser unos increíbles hermanos, los amo. Atentamente: Su hermano menor.

A Lidia

A mi futura esposa y prometida, que me apoyo en los mejores y peores momentos de mi trayectoria. gracias a su apoyo emocional y amor incondicional que alimentó mi perseverancia permitiéndome alcanzar mis sueños en la vida y profesionalmente, te amo.

A mis primos Mauricio, Fernanda y Liliana

Gracias por darme tiempo de distracción cuando más lo necesité, ideas nuevas para mis metas y sobre todo su cariño, por darme la oportunidad de ser su guía y poder aprender juntos, siempre serán mis primos favoritos, los amo a cada uno de ustedes.

A la dependencia universitaria

Que me brindó la confianza de realizar el caso práctico, creyó en mí y sobre todo abrió sus puertas hacia mis ideas para alcanzar mi meta, aprecio mucho su apoyo y siempre estaré agradecido.

A mis amigos

A mis amigos que me siguen apoyando y queriendo todos los días Fernanda, Arantza, Monserrat, Iván, Fernando e Isaí por su cariño y gran amistad constante, que llegaron en momentos oportunos de mi camino y a pesar de la adversidad se quedaron.

Mtra. Dulce Campos

Que me enseñó las bases en mi carrera profesional, mostró interés por mi enseñanza para alcanzar mis metas y logros, que tuvo la paciencia admirable de apoyarme todos los días, resolver mis dudas y gracias por dejar una gran marca en mi vida maestra Dulce.

Mtro. Rafael Sandoval

Que me apoyó con sus ideas y aclaro las mías, con su tolerancia y guía en mi trayecto profesional en todo momento, gracias, maestro Rafael.

A la Facultad de Ingeniería, a la UNAM, profesores y alumnos

Gracias a la enseñanza y educación que forjaron en mí, que me permitió conocer mi círculo de amigos, avanzar como ingeniero, profesores que apoyaron mis habilidades y compartieron sus conocimientos.

Contenido

Introducción 1
Justificación 3
Objetivo 4
Capítulo l La seguridad informática y las pruebas de penetración 4
1.1 Seguridad informática 5
1.2 Identificación de la amenaza, el riesgo y el impacto. 8
1.3 Pruebas de penetración 13
1.3.1. Metodologías de modelos de pruebas de penetración 15
1.3.2 Tipos de pruebas de penetración 18
1.3.3 Herramientas de pruebas de penetración open source 21
1.4 Reporte de resultados 25
1.5 Descubrimiento y análisis de vulnerabilidades 26
1.6 Gestión de incidentes y monitoreo 29
1.6.1. Monitoreo y gestión de vulnerabilidades encontradas 32
1.6.2. Cálculo de probabilidad y planes de acción 33
1.6.3. Gestión de incidentes 38
Capítulo ll Aplicaciones web 51
2.1 Internet e hypertext transfer protocol 52
2.2 Los protocolos de comunicación segura. 56
2.2.1 Protocolos SSL y TLS 58
2.3 Aplicaciones web 61
2.3.1 Modelos de aplicaciones web 64
2.4 Tecnologías de aplicaciones web 66
Capítulo lll Pruebas de penetración en aplicaciones web con el marco de clasificación OWASP 72
3.1 Marco de clasificación OWASP 73
3.2 Vulnerabilidades en aplicaciones web 75
3.3 Factores de clasificación de vulnerabilidades 83
3.4 Clasificación de vulnerabilidades 86
3.5 Referencias de vulnerabilidades encontradas 87
Capítulo lV Metodología de pruebas de penetración NIST SP 800 115 95

4.1 Determinación de propósito, objetivo, políticas y restricciones. 96
4.2 Técnicas de evaluación, análisis y validación de vulnerabilidades 98
4.3 Evaluación de seguridad y ejecución de prueba de seguridad 100
4.4 Actividades posteriores de la prueba de penetración 102
4.5 Fase de documentación y reporte 103
Capítulo V Caso de estudio: Prueba de Penetración a Portal Web de una dependencia universitaria con la metodología NIST SP 800-115 y marco de clasificación OWASP 108
5.1 Fase de Planificación 109
5.2 Fase de Descubrimiento 111
5.3 Fase de Ejecución 113
5.4 Fase de Documentación y Reporte 118
5.5 Resultados y evidencia del caso de estudio. 120
Conclusiones 131
Anexo A Carta de Planeación 135
Anexo B Reporte Ejecutivo 140
Anexo C Reporte Técnico 147
Glosario 164
Índice de Figuras 174
Índice de Tablas 175
Índice de Formulas 177
Referencias 178

Introducción

Este trabajo de tesis se encuentra orientado en el campo de seguridad de la información, en la cual existen diferentes actividades que se pueden implementar. Se enfoca en el ámbito de seguridad ofensiva, que aplica un servicio conocido como prueba de penetración a aplicaciones web.

El capítulo uno da la introducción sobre la seguridad de la información, sus componentes y características para identificar posibles riesgos en un medio informático. Adicionalmente se describen los tipos de prueba de penetración, metodologías que pueden emplearse, fases de cada una y sobre su gestión ante incidentes y monitoreo.

El capítulo dos, presenta los antecedentes históricos y la evolución de las páginas web, además de los inicios del internet con sus protocolos de comunicación. Explica los diferentes tipos de páginas que existen y se introduce al tema de las aplicaciones web y sus categorías. Finalmente se dan a conocer protocolos de comunicación seguros que se pueden implementar para la transferencia de información.

El capítulo tres, explica el marco de clasificación OWASP Top Ten, brinda las bases para identificar vulnerabilidades de acuerdo con su comportamiento, características y escenarios que se relacionen a ellos, adicionalmente poder asignar un probabilidad, impacto y riesgo para la mitigación de acuerdo a su severidad y finalmente conocer fuentes abiertas que proporcionan información de vulnerabilidades a tecnologías y componentes utilizadas por las aplicaciones.

El capítulo cuatro muestra como se integra cada fase de la metodología NIST SP 800 115, que se debe realizar en cada sección y como se conforma cada una para su desarrollo y obtener de una buena implementación una prueba de seguridad exitosa.

El capítulo cinco explica la aplicación de una prueba de seguridad en el caso de estudio, como se emplea cada fase de la metodología de una prueba de penetración, como se puede construir cada documento de inicio a final, el uso de herramientas de seguridad orientadas a cada fase y como clasificar una vulnerabilidad junto con una remediación genérica para esta.

Finalmente se exponen las conclusiones del trabajo de tesis asociadas al caso de estudio con la identificación de cada vulnerabilidad acorde al OWASP Top Ten y una breve explicación de que se hizo en cada fase de la metodología NIST SP 800 115.

Justificación

De acuerdo con (UAD Hispasec, 2022)[1], los ataques informáticos se han incrementado año tras año y en gran medida se deben al descubrimiento de cada vez más vulnerabilidades asociadas a nuevas tecnologías. En la actualidad los sistemas informáticos cuentan con vulnerabilidades que llegan a tener graves consecuencias. Si se identifican de una manera correcta, precisa y oportuna se puede disminuir la probabilidad de impacto al materializarse algún intento de ataque. La elaboración de esta tesis se enfoca en la realización del caso de estudio de un portal web, el cual es considerado como un sistema informático. Considerando que no se puede asegurar que un sistema esté totalmente seguro se realizará una prueba de penetración al portal web para determinar los riesgos y alcances de una intrusión o ataque informático. Con los resultados obtenidos se realizará el análisis e interpretación para generar acciones de mitigación o disminución del impacto. Esta tesis muestra la investigación sobre las metodologías que existen para realizar pruebas de penetración con ello se logrará comprender las etapas y fases para asegurar de manera correcta un sitio web.

[1] (UAD Hispasec, 2022)

Objetivo

Describir los procesos y métodos de las pruebas de penetración a sitios web, con la finalidad de analizar e interpretar los resultados que permitan conocer los riesgos y brindar recomendaciones para disminuir la probabilidad de una intrusión o ataque informático al sitio, utilizando como caso de estudio la evaluación del portal web de una dependencia universitaria.

Capítulo l
La seguridad informática y las pruebas de penetración

1.1 Seguridad informática

Según (Aguilera, 2011)[2] , "Se puede definir a la seguridad informática como la disciplina encargada de plantear y diseñar las normas, procedimientos, métodos y técnicas con el fin de lograr que un sistema de información sea seguro, confiable y, sobre todo, que tenga disponibilidad". En resumen, la seguridad de la información contiene políticas, normas, reglas y acciones preventivas implementadas por una organización de forma interna y externa, con el objetivo de mantener los datos con integridad, disponibilidad y confidencialidad donde se encuentren almacenados, protegiéndolos contra cualquier tipo de amenaza, mitigando el impacto y riesgo en medios físicos, lógicos o digitales.

El área a cargo tiene como objetivo desarrollar ciertos planes de acción en los cuales se implementan técnicas, en caso de que suceda una acción maliciosa, se tendrá que regir de acuerdo con la triada de la seguridad, siendo el enfoque principal para cualquier plan de contingencia, observando que se cumplan estos tres pilares. Si una organización logra identificar que una amenaza se materializo, interpondrá medidas cautelares de contención para disminuir algún impacto buscando que se conserven estos pilares, ya que en caso contrario puede provocar que el cliente desconfíe, perjudicar en el mercado o desconfianza a los empleados. Al sufrir un ataque, dependiendo de la magnitud y el impacto que se haya provocado, producirá incertidumbre en la trayectoria hacia el futuro para la organización si no se tiene un plan contra contingencias o incidentes de este tipo.

Según ((OWASP), 2017)[3]algunas acciones maliciosas pueden llevar a un robo de identidad por medio de técnicas de phishing hasta una inyección de código SQL que llevaría a una extracción de información sensible, provocando que cualquier información de una organización pueda ser expuesta, robada o modificada. Para conocer las debilidades de una organización y el nivel de preparación ante estas adversidades, existen las pruebas de penetración, que tienen el enfoque de la búsqueda de debilidades y vulnerabilidades a un sistema informático las cuales se dividen en dos: físicos y digitales. Un sistema físico suele ser desde la infraestructura montada para brindar el servicio como

[2] (Aguilera, 2011).
[3] ((OWASP), OWASP Top 10 - 2017 Los diez riesgos más críticos en aplicaciones web, 2017)

son el cuarto de servidores, la computadora física o hardware de un usuario. Un sistema digital es un servicio intangible que no se puede sentir pero sí ver, cómo son las aplicaciones web, aplicaciones móviles, documentos digitales, etcétera.

Para el control de dichas acciones maliciosas, se suma la protección de un sistema informático y tecnológico, por lo cual es importante entender qué mantiene a flote a una organización: a estos se les llama activos. Los activos suelen ser más propensos a ser afectados por una amenaza. El área dedicada a la seguridad en la organización tomará en consideración las acciones maliciosas que pueden ocurrir en el transcurso de cierto tiempo, desarrollando la capacidad de pensar como atacante, ya que, al tener consentimiento del valor de la organización, será más sencillo desarrollar un plan contra la probabilidad calculada para que sucedan dichas amenazas pronosticadas. El cálculo de riesgos de dichas amenazas sucede gracias a la identificación de estos. Una vez identificados, habrá que cuidarlos, activando herramientas, servicios o asignando personal al registro de cualquier acción anómala a estos activos, así se logrará tener un seguimiento de cada movimiento interno y externo para fabricar una traza ante cualquier comportamiento extraño.

Al gestionar información dentro de las organizaciones, el valor es proporcionado por los clientes y empleados que aportan datos los cuales se trabajan y almacenan. Para realizar de forma adecuada una protección de esta y buscar cuál es la mejor manera de administrarlos en el entorno, se toma en cuenta que existen un conjunto de estándares definidos como un marco que establece, implementa, gestiona y mejora la forma de administrar un sistema, siendo el método más conocido en el área de la información el SGSI (Sistema de Gestión de Seguridad de la Información). Las normas más adaptadas y seguidas en la industria de la informática son ISO/IEC 27000 (International Organization for Standardization) e IEC (International Electrotechnical Commission).

Como se ha mencionado anteriormente, la seguridad informática se emplea en un nivel práctico para reducir cualquier riesgo asociado en relación al interés de cada organización, sin desviarse del objetivo principal de proteger la información, considerando la definición de la triada (integridad, confidencialidad y disponibilidad) que se definirá a continuación:

- **Integridad**: (jurídico, DPEJ RAE (Integridad), 2019)[4] "Integridad es la propiedad o característica en la que el activo de información no ha sido alterado de manera no autorizada". Dirigiendo la definición a las TICs, es similar, ya que se debe asegurar que la información estará completa, sin alteraciones de su contenido y sin que pueda ser afectada por usuarios o procesos no autorizados dentro de una organización o un cambio realizado en un contenido de información, que podría ser tanto en su forma física como digital, así como en el ambiente en donde es utilizado o almacenado.
- **Confidencialidad**: (DPEJ RAE (Confidencialidad), 2020)[5] "Confidencialidad es el principio que impide la divulgación de cualquier dato que posibilite la identificación de información de personas u organizaciones". Aplicado a las Tecnologías de la Información y Comunicación, TIC, este principio tiene el propósito de asegurar que se prevenga el acceso o la lectura de información por parte de personas no autorizadas. Esto implica especificar qué activos serán seleccionados para que solo un individuo o un grupo controlado de personas pueda acceder a ellos y protegerlos. Además, se les asignará la responsabilidad de garantizar que los datos considerados como privados no violen este principio de confidencialidad.
- **Disponibilidad**: (Española, Diccionario de la lengua española (Disponibilidad), 2022)[6] "La disponibilidad es una cualidad o condición que se refiere a la capacidad de estar disponible o accesible." Según la RAE (adjetivo), esto significa que se puede disponer libremente de algo o que está listo para ser utilizado. Al aplicar este concepto a las TICs, se puede definir como la función que permite que la información, los servicios o los procesos de la organización sigan su curso natural o con el fin por el que fue creado, asegurando que los usuarios puedan hacer uso de ellos cuando, donde y en el momento que lo necesiten.

La seguridad informática se considera una rama de la seguridad de la información, en la cual se implementan acciones como la gestión, el almacenamiento y la transmisión de

[4] (jurídico, DPEJ RAE (Integridad), 2019)
[5] (jurídico, DPEJ RAE (Confidencialidad), 2020)
[6] (Española, Diccionario de la lengua española (Disponibilidad), 2022)

información o datos dentro de un sistema en una red. A continuación, se presentan los siguientes tipos de seguridad:

- Seguridad física: Es una forma de protección en el entorno natural que se puede observar y sentir. Se orienta a proteger los activos de la organización ante la probabilidad de que puedan existir amenazas que afecten la apariencia o integridad física de estos medios, como temblores, inundaciones, robos, incendios, etcétera.
- Seguridad lógica: Es un mecanismo implementado para proteger el flujo natural de un sistema informático. Los medios de protección para los sistemas informáticos suelen ser la criptografía, banderas lógicas, condiciones de roles, entre otros.
- Seguridad activa: Es la forma de protección encargada de prevenir, detectar y evitar que cualquier incidente en los sistemas informáticos suceda y produzca consecuencias irreparables. Para ello, se utilizan métodos de autenticación, controles de acceso, validación de datos, firewalls, entre otros.
- Seguridad pasiva: Son metodologías que aplican técnicas y procesos para disminuir los incidentes en caso de tener éxito y evitar afectaciones a la organización. Algunos ejemplos de estas medidas son las copias de seguridad, el monitoreo, las campañas de concientización, entre otros.

1.2 Identificación de la amenaza, el riesgo y el impacto.

La norma ISO 27001 menciona los criterios, indicando que una vez identificadas las causas que pueden originar un daño, se determinará si es aceptado o no por la organización. Si la causa encontrada puede afectar el proceso original de ella, se deberán tomar acciones sobre el plan de tratamiento de riesgos para poder estar preparados ante cualquier acción.

Para poder implementar medidas de protección que ayuden a identificar el valor de una organización, primero se debe analizar y comprender lo que es un activo.

Activo: Puede ser un recurso de un sistema, ya sea informático o no, el cual es imprescindible para la organización. En resumen, es todo aquello que tenga valor para el dueño y que debe ser protegido ante cualquier evento que se produzca, ya sea de un

atacante o de un usuario no malintencionado. Algunos ejemplos que se pueden considerar como activos son: hardware, empleados, software, datos, archivos, etcétera.

Para proteger los activos ante cualquier incidente, primero se deben identificar; esto permitirá establecer las medidas, mecanismos y protocolos necesarios. Estas acciones se adaptarán de acuerdo con cada activo encontrado, proporcionando una protección personalizada para cada uno de ellos.

Algunos activos ejemplificados son (Escrivá, 2013)[7]:

- Información: Es aquel recurso que puede mantener datos almacenados en un sistema informático o físico en cualquier ubicación de la organización. Puede tratarse de documentos, patentes, datos de empleados, credenciales de usuarios, entre otros.
- Software: Son programas o aplicaciones que son utilizados por la organización, ya sea para la funcionalidad completa de su servicio o para algunos procesos implementados para la organización. Algunos ejemplos son aplicaciones, sistemas operativos, programas de computadora, entre otros.
- Físicos: Es aquella infraestructura tecnológica utilizada que puede almacenar, procesar, gestionar o transmitir información importante para el buen funcionamiento de la organización. Podrían considerarse como ejemplos los archiveros, credenciales, cuarto de servidores, entre otros.
- Personal de la organización: Suelen ser los empleados que utilizan la estructura tecnológica o de comunicación para realizar el manejo y uso de los recursos disponibles de la organización.

Para cualquier activo encontrado en la organización podría existir una vulnerabilidad, la cual se puede definir como aquella debilidad que pueda provocar alguna consecuencia sobre la funcionalidad natural o flujo común para el cual fueron creados. Dichas debilidades pueden considerarse coloquialmente como "brechas de seguridad". Asimismo,

[7] (Escrivá, 2013)

se asocian con algunos fallos que pueden ser en la implementación de las medidas de seguridad o en la configuración de mecanismos, hasta la programación realizada, dependiendo del activo protegido y su función.

Para brindar protección hacia los activos, se puede establecer técnicas que logren mantener seguros los datos ante posibles atacantes. Se debe ser capaz de analizar lo que sucede en el entorno natural de la organización, cuestionar el valor actual que puede llegar a producirse en la sociedad y, a su vez, observar la probabilidad en la cual puede llegar a interesar a un atacante, dependiendo de sus motivos o razones. De esta manera, se evita que el activo pierda su valor o provoque alguna pérdida para la organización.

Para lograr esto, debe centrarse en mantener dicha protección y para conseguirlo se deben analizar tres factores que son:

- Amenaza: (Española, Diccionario de la lengua española (Amenaza), 2019)[8]: "Al venir de la conjugación, amenazar significa dar a entender con actos o palabras que se quiere hacer algún mal a alguien". Teniendo en consideración dicha definición y dando el enfoque principal en la seguridad informática, se define amenaza como aquellos eventos que ocurren con la finalidad de dañar un procedimiento, recurso o dato. En esta definición, se relacionan las debilidades, debido a que son las premisas ante cualquier acción, ya que, si un atacante conoce las vulnerabilidades de un activo, puede tener éxito ante su ataque. Asimismo, existen amenazas que son involuntarias, como desastres naturales. Dichas amenazas suelen dividirse en activas y pasivas.
    - Activas: Son aquellas amenazas que logran realizar cambios no autorizados en el sistema. Estas son consideradas las más peligrosas, ya que implican una interacción directa con el objetivo, lo que indica que se ha encontrado una debilidad y puede provocar afectaciones futuras. Algunos ejemplos de amenazas activas pueden ser el robo de credenciales, el robo de usuarios, la escalación de privilegios, etcétera.

[8] (Española, Diccionario de la lengua española (Amenaza), 2019)

- Pasivas: Son aquellas amenazas que esperan en el objetivo para obtener información relacionada con cualquier activo. Pueden ser equipos informáticos (IP, puertos, servicios, protocolos) que están en proceso de comunicación, y se utilizan programas informáticos para analizar el tráfico en la red y obtener información sin una interacción directa con el objetivo.

Según (MAGERIT, 2012)[9] presenta la siguiente clasificación ante las amenazas (véase tabla1).

Tabla 1 Ejemplos de Amenazas

Grupos de Amenazas	Ejemplos
Desastres Naturales	Incendio, daños por agua, desastres naturales.
Desastres Industriales	Incendio accidental, daños por agua, contaminación mecánica.
Errores y fallos no intencionados	Errores de usuarios, errores de configuración, etcétera.
Ataques deliberados	Manipulación de la configuración, suplantación, robo de información, entre otros.

- Riesgo: Para ello existen diversas definiciones entre las cuales se presentan las más destacadas:
  - Según (UNISDR, 2004)[10] "El riesgo es la probabilidad de que una amenaza se convierta en un desastre. Las vulnerabilidades y las amenazas por separado no representan ningún peligro."
  - Según (Organismo de Normalización en España, 2008)[11] "Un riesgo es la estimación del grado de exposición hacia una amenaza que se logre materializar ante uno o más activos provocando daños hacia la organización.
  - Según (GUÍA DE SEGURIDAD DE LAS TIC , 2010)[12] "El riesgo es la probabilidad que una amenaza se materialice logrando obtener ventaja ante

[9] (MAGERIT, 2012)
[10] (UNISDR, 2004)
[11] (Certificación, 2008)
[12] (Nacional, 2010)

una vulnerabilidad y lograr provocar un daño (impacto) de un proceso o recurso.

Al enfocarse en los sistemas de información, es importante tener en cuenta que una amenaza, al combinarse con alguna vulnerabilidad existente, puede generar un riesgo. El cual se define con la probabilidad de que una amenaza sea exitosa en su ataque, lo que podría ocasionar perjuicios para la organización, dependiendo de la magnitud de la amenaza. Un ejemplo ilustrativo sería el siguiente: suponga que se tiene un edificio antiguo con varios departamentos, cuyas estructuras datan de más de 30 años. En caso de que ocurra una amenaza natural, como un temblor, el riesgo para el edificio sería alto debido a la antigüedad de sus estructuras.

Existen diferentes niveles en los que se puede clasificar el riesgo para los activos expuestos, y estos dependen de la probabilidad de que ocurra una amenaza y del grado de impacto que podría tener. A continuación (véase tabla 2), se mencionan dichos niveles:

Tabla 2 Ejemplos de Riesgos

Nivel	Ejemplo de Riesgo
Alto	Robo de información Robo de hardware
Medio	Configuraciones incorrectas
Bajo	Fusibles de luz quemados

- Impacto: Una vez calculadas e identificadas las amenazas y el riesgo que pueden existir para los activos de una organización, según (MAGERIT, 2012), se puede clasificar el impacto en dos formas:
    - Impacto acumulado. Es el cálculo realizado sobre un activo teniendo en cuenta su valor acumulado (el propio más el acumulado de los activos que dependen de él) y las amenazas a las que está expuesto.

- Impacto repercutido. Es el cálculo realizado sobre un activo superior teniendo en cuenta su valor propio y las amenazas a las que están expuestos los activos inferiores de los que depende.

Se enfoca en el resultado acumulado, ya que la organización, al adquirir una consecuencia producida por un mal cuidado de activo, puede provocar que su funcionamiento natural se modifique y, con ello, traiga repercusiones. Para obtener la medición de la consecuencia, se implementará el cálculo del riesgo involucrando el impacto y la probabilidad con la siguiente fórmula (véase fórmula 1):

$$Impacto + Probabilidad = Riesgo$$

Fórmula 1. Nomenclatura del Riesgo

Para comprender que el impacto que se obtiene es diferente, vea cómo dos organizaciones pueden verse afectadas de manera singular dependiendo de la materialización de amenazas o de lo bien fundamentadas que estén sus políticas ante estas. Por ejemplo, si una organización programa regularmente revisiones de seguridad y utiliza métodos para respaldar su información en copias de seguridad, esto reducirá el impacto en caso de un borrado accidental de su disco duro principal. Sin embargo, si la otra organización no implementa revisiones periódicas de seguridad y experimenta un borrado de disco, su impacto será mucho mayor.

1.3 Pruebas de penetración

El uso de medios tecnológicos, internet y dispositivos inteligentes ha ido en crecimiento en los últimos años, permitiendo que cualquier individuo pueda tener acceso ilimitado a aplicaciones web o móviles, obteniendo claramente el beneficio de acceso al recurso. Las organizaciones que ofrecen estos servicios se vieron obligadas a implementar medidas de seguridad para asegurar la integridad de la información que viaja entre dispositivos hacia internet. Ahora, las organizaciones se enfocan en el desarrollo móvil o en la aplicación web para implementar un servicio y ofrecerlo. Su inversión en el desarrollo de aplicaciones web y móviles ha generado un interés que lo convierte en un servicio con alta probabilidad de ser atacado. Por ello, recurren a la contratación de servicios como las pruebas de

penetración, las cuales se efectúan en su aplicación web o móvil. Estas evaluaciones pueden realizarse en el ambiente de desarrollo o cuando la aplicación ya se encuentre en el ambiente de producción, dependiendo del enfoque que le dé la organización.

El término prueba de penetración tiene diferentes definiciones de acuerdo con la metodología que se implemente, las cuales pueden ser las siguientes:

- Según (INCIBE, Instituto Nacional de Ciberseguridad, 2019)[13] "Método para la evaluación de un sistema o red mediante la simulación de un ataque de origen hostil"
- Según (OSSTMM 3 – The Open Source Security Testing Methodology Manual, 2010)[14] "Una prueba de seguridad es la relación de una verificación de seguridad con un objetivo específico a la búsqueda de brechas seguridad y debilidades en sus mecanismos de seguridad, esta se finaliza con un tiempo establecido llamado ventana de verificación de seguridad"
- Según (Technical Guide to Information Security Testing and Assessment , 2008)[15] "Prueba de seguridad con los evaluadores que simulan ataques reales para accionar las características de seguridad de una aplicación, sistema o red."
- Según (ISO 31000:2018, 2018)[16] "Las evaluaciones de riesgo tienen el fin de detectar las debilidades de un sistema o proceso, teniendo en cuenta que nada es 100% seguro o inviolable"

Al inicio de una prueba de penetración, se recomienda contar con bases sólidas para realizarlas, las cuales tienen diferentes etapas que se definen al inicio de esta. En donde se explican las fases que tendrá cada una, consideraciones, limitaciones, restricciones y el tipo de caja de prueba, de esta manera conocerá las acciones que se podrán llevar a cabo en cada fase y obtener los resultados esperados por el cliente. Asimismo, existen

[13] (INCIBE, Instituto Nacional de Ciberseguridad, 2019)
[14] (Manual, 2010)
[15] (Technology, 2008)

[16] (Normalización, 2018)

algunas metodologías semejantes en donde se pueden definir con más detalle las fases y lo que sigue después de una u otra.

1.3.1. Metodologías de modelos de pruebas de penetración

La realización de esta verificación de seguridad debe mantener un seguimiento, estándares definidos por diferentes organizaciones para la creación de los procedimientos generales que se desarrollan dentro de ellas, hasta continuar con sus fases finales como el análisis de resultados, vulnerabilidades y riesgos, donde se puede implementar una fase de mitigación y gestión para dichos incidentes, conocida como plan de contención. Algunas de las metodologías de las pruebas de penetración son:

- OWASP (OWASP Testing Guide v4.0, 2014)[17]: Esta metodología, mejor conocida como 'OWASP Testing Guide', fue publicada en el año 2004 en su versión 1.0. Esta metodología está orientada a aplicaciones web y está dividida en varios grupos para comprobar aspectos de seguridad en aplicaciones web. Estos grupos son:
    - Recopilación de información.
    - Pruebas de seguridad en configuración y despliegue.
    - Pruebas de seguridad en la gestión de la identidad.
    - Pruebas de seguridad al proceso de autenticación.
    - Pruebas de seguridad al proceso de autorización.
    - Pruebas de seguridad al proceso de gestión de sesiones.
    - Pruebas de seguridad para la validación de entradas.
    - Pruebas de seguridad al manejo de errores.
    - Pruebas de seguridad a los mecanismos criptográficos.
    - Pruebas de seguridad a la lógica del negocio.
    - Pruebas de seguridad del lado del cliente.

[17] (OWASP Testing Guide v4.0, 2014)

- OSSTMM (OSSTMM 3 – The Open Source Security Testing Methodology Manual, 2010)[18]: La publicación de la guía fue realizada en 2010 con su versión 3 por ISECOM, en la cual sus fases son:
    - Fase de inducción: Se conoce el alcance, requerimientos de la organización y restricciones de la prueba de penetración.
    - Fase de interacción: Se encuentran las relaciones entre el alcance establecido, objetivos y los activos de la organización declarados.
    - Fase de requerimientos: Se hace la verificación de los procesos que realizan la aplicación, que configuraciones establece, capacidades y capacitación a usuarios, propiedad intelectual, información sensible y otra información a recopilar.
    - Fase de intervención: Se realiza la prueba de seguridad, el alcance de dichos hallazgos y su consecuencia o impacto ante la organización.

- ISSAF ((OISSG), 2006)[19]: Marco para la evaluación de los sistemas de tecnologías de información, lo desarrollo OISSG (Open Information Systems Security Group) y consta de las siguientes fases:
    - Planificación y Preparación: Se conjunta la información inicial del producto, se planifica y se prepara el entorno de verificación de seguridad.
    - Evaluación: Se aplican las pruebas de penetración ISSAF que son:
        - Recolección de información: Se integra toda la información posible para su explotación, para conseguir información funcional.
        - Mapeo de red de trabajo: Desde la aplicación se considera la red conectada a ella, topología de la organización y comportamiento de ésta.
        - Identificación de vulnerabilidades: Se realiza el escaneo para posibles vulnerabilidades halladas, se listan y enumeran para a continuación realizar el cálculo del posible impacto.

[18] (Manual, 2010)
[19] ((OISSG), 2006)

        - Penetración: El consultor probará las vulnerabilidades halladas para identificar qué tanto puede llegar dentro de la red y el nivel de acceso de información.
        - Obtención de acceso y escalada de privilegios: Se confirman las vulnerabilidades encontradas y se documentan.
        - Enumeración adicional: Se realiza la obtención de contraseñas que se puedan encontrar en archivos de configuración del servidor o servicio de la aplicación analizada.
        - Comprometer usuarios: Se logra el acceso a usuarios por sitios remotos y la red de la organización.
        - Mantener el acceso: Se mantiene el acceso de los usuarios ingresados a través de las vulnerabilidades halladas.
- PTES (PTES Standard, 2014)[20]: Es un estándar para la ejecución de pruebas de penetración, dicho proyecto está constituido por diversas organizaciones, las fases que lo componen son:
    - Preacuerdo: En dicha fase se plantea el alcance del proyecto y los objetivos de la prueba de penetración a alcanzar.
    - Recopilación de inteligencia: Se establecen las fuentes donde se recolectó la información para conocer a más detalle el objetivo como fuentes abiertas o de paga.
    - Modelado de amenazas: Se crean posibles estrategias con las cuales se aumentará el caso de éxito, buscando diferentes formas de explotar una vulnerabilidad y el encontrarlas.
    - Análisis de vulnerabilidades: Se enlistan las vulnerabilidades encontradas que pueden llegar a explotarse.
    - Explotación: Se realiza la verificación de vulnerabilidades encontradas.
    - Post Explotación: El consultor continúa con la explotación para encontrar el alcance de vulnerabilidades.

[20] (Standard, 2014)

- Reporte: Se recolecta la información encontrada de vulnerabilidades que permita solucionar dichos hallazgos.

- NIST SP 800-115 (Technical Guide to Information Security Testing and Assessment , 2008)[21]: Guía técnica para realizar evaluaciones y pruebas de seguridad de la información, dicha guía fue publicada en el 2008, se compone de las siguientes fases:
  - Fase de planificación: Se implementan las reglas a seguir para las pruebas de penetración junto con las condiciones y técnicas que están implementadas por la organización.
  - Fase de descubrimiento: Se comienza con el escaneo de vulnerabilidades siendo una recopilación de información a nivel infraestructura, entidad y comportamiento para el descubrimiento de vulnerabilidades.
  - Fase de Ejecución: Se realiza la comprobación de vulnerabilidades halladas en la fase anterior.
  - Fase de documentación y reporte: Se genera el reporte con los problemas de seguridad encontrados para su recomendación.

Una vez que se determina qué metodología se va a aplicar a cada proceso definido por el consultor en este caso o la empresa encargada de la verificación de seguridad, se podrán comenzar las pruebas de seguridad. Al iniciar, la organización que solicitó la evaluación deberá definir qué tipo de caja de prueba se usará, el objetivo, tiempo calculado, etcétera. El propósito de realizar una verificación es lograr entender los activos que tiene cada organización, desde un punto de vista como tester externos o internos, hasta llegar a un punto en el que se pueda demostrar que el activo y la organización podrían tener pérdidas de valor si se lograrán explotar.

1.3.2 Tipos de pruebas de penetración

Al iniciar la prueba de penetración, es fundamental tener en cuenta todas sus fases, como lo son la planeación al inicio y, al finalizar, la emisión de la carta de conformidad de

[21] (Technology, 2008)

servicios entregados. En algunas ocasiones, la fase de planificación o preacuerdo define el tipo de caja que se considerará y qué información será proporcionada. Este aspecto mencionado se suele representar con la siguiente ilustración (véase figura 1) según (OSSTMM 3 – The Open Source Security Testing Methodology Manual, 2010)[22]:

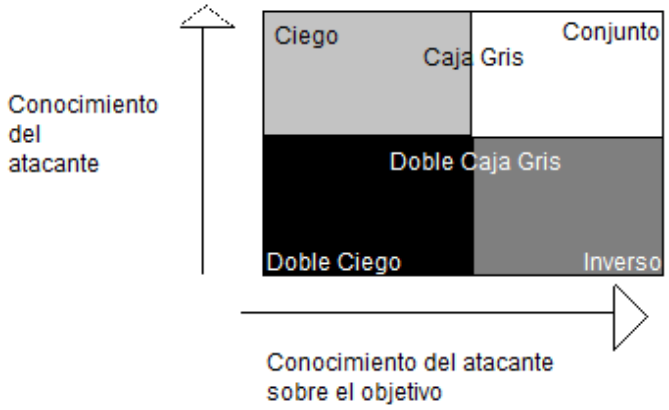


Figura 1 Tipos de pruebas de penetración

- Ciego: El consultor realizará la prueba de penetración sin tener algún conocimiento sobre el sistema, tecnología o aplicación. En esta fase, la organización se encuentra preparada para la verificación, conoce que se realizará la evaluación con anticipación. El principal objetivo es observar los conocimientos del consultor y la seguridad de la organización, ya que se intentará quebrantar la seguridad de acuerdo con el nivel de conocimiento y estrategias tanto del consultor como del personal de la organización.

- Doble Ciego: En este tipo, el consultor no tiene conocimiento alguno del sistema, tecnología o aplicación, y toda la organización no tiene notificación anticipada, a excepción de directivos o un grupo seleccionado. Se busca obtener una semejanza de un ataque externo o interno que podría suceder.

- Gray Box: En este tipo, el consultor sí tiene conocimiento limitado del sistema que se plasma en el preacuerdo, la información proporcionada dependerá de lo que

[22] (Manual, 2010)

quiera compartir la organización. El objetivo se encontrará preparado y notificado para que se realice la verificación de seguridad, teniendo conocimiento de cada etapa y proceso. Este tipo de verificaciones suelen ser solicitadas por la organización como autoevaluación de sus sistemas de seguridad.

- Doble Caja Gris: En este tipo, se tiene conocimiento limitado sobre el sistema a evaluar; así mismo, sí se notifica que se realizará la verificación de seguridad a la organización, pero no sobre qué activo o qué canales se utilizarán. La finalidad es encontrar diferentes vectores de ataque o accesos desconocidos para romper sus sistemas de seguridad.
- Conjunto: En este tipo, se suele ser minucioso, ya que tanto la organización como el consultor tienen conocimientos previos de lo que se va a analizar. La única limitante es que no se realizará la etapa 1 (preacuerdo) de las pruebas de penetración debido a que la organización está preparada. La finalidad principal de esta verificación es encontrar la protección directamente ante las amenazas o verlo desde el punto de vista de una prueba de seguridad interna.
- Inverso: En este tipo, se realiza con conocimiento previo de todos los procesos que se realizarán al objetivo a analizar, pero a su vez la organización no conoce cuándo o cómo se va a realizar la verificación. El objetivo principal para de esta es poder encontrar qué tan preparada está la organización para estos escenarios.

Para cada etapa, una vez definido el proceso, metodología y tipo de caja, y dependiendo de las técnicas del consultor, en ocasiones se tendrán que implementar el uso de herramientas mencionadas a continuación. Todo depende de qué se haya proporcionado y el tiempo que se haya asignado en el preacuerdo. Las herramientas son una ayuda importante para el consultor debido a que algunas son automatizadas y compensan el tiempo al reducirlo para encontrar brechas de seguridad. Asimismo, existen diferentes tipos de herramientas de "Software Libre", las cuales están abiertas a todo público, y las "Privadas" en las que los consultores necesitan una licencia para poder usar la herramienta de manera completa o secciones de la misma.

1.3.3 Herramientas de pruebas de penetración open source

Con la finalidad de clasificar las herramientas, se propone una metodología estándar que las relaciona con cada fase de una prueba de penetración. Así, definir las fases donde podrían ser utilizadas, permitiendo tener una visión general de los procesos dentro de esta verificación de seguridad, la cual consta de las siguientes fases:

- Preacuerdo: En esta fase se hace el acuerdo entre la organización solicitante de la prueba y el consultor que va a ejecutarla. Se define el tipo de caja, los involucrados de ambas partes y la metodología a usar. Todo esto sucede en la interacción entre la organización y el analista. Se debe tener un documento en donde se va a establecer el nombre de los involucrados junto con la aprobación de la ejecución del servicio. Dicho documento protege al consultor y a la organización en escenarios en los cuales, mientras se ejecuta la verificación y llega a suceder un ataque contra la organización, etcétera. En este mismo se plantean acuerdos de confidencialidad para el caso en que la evaluación arroje un sistema que pueda contener información sensible. Adicionalmente, limita el tipo de prueba de penetración que se va a realizar; por ejemplo, con la técnica que se podrá emplear. En esta fase no se utiliza una herramienta, debido a que todo es ejecutivo y administrativo.
- Reconocimiento: La fase en donde se da inicio a la prueba de seguridad se comienza con la recolección de información del sistema, ya sea con una interacción directa, es decir, la información es proporcionada por la organización. En caso de que no se brinde información, se comienza con la fase de recolección de información donde se usa el modelo OSINT (Open Source Intelligence) que plantea que se puede recolectar información que esté públicamente disponible de forma pasiva y pueda ser útil para esta fase y posteriores, o la fase activa en la cual se tendrá una interacción con el sistema. Para realizar esta fase, se suelen usar las siguientes herramientas:
    - Maltego: Herramienta para el mapeo de una aplicación web e identificación de DNS, servicios y subdominios.

- Shodan: Herramienta pasiva para la búsqueda de información en fuentes abiertas, desde puertos abiertos, sistemas operativos, servicios, etcétera.
- Whois: Búsqueda de información registrada al dominio que está asociado a la aplicación web, desde nombre, correo, número, por mencionar algunos.
- Exiftool: Herramienta para la búsqueda de metadatos en archivos.
- Whatweb: Herramienta para la identificación de tecnologías web, recursos y frameworks implementados en la aplicación web.
- Nmap: Herramienta de código abierto utilizada para explorar y mapear redes informáticas.
- Zenmap: Herramienta de código abierto utilizada para explorar y mapear redes informáticas.
- Google Hacking: Técnicas para utilizar motores de búsqueda como Google para encontrar información sensible o vulnerabilidades de seguridad en sitios web y sistemas.

- Modelado de Amenazas: Fase que se aplica de acuerdo con la implementación que se haya acordado en el preacuerdo, desde la definición de los activos de la organización, qué tecnologías usan, cómo funciona el sistema, operaciones, almacenamiento de información, qué tipo de información, entre otros. Todo se enlaza con la fase anterior, ya que se aplica el reconocimiento del objetivo, junto con sus activos, y se tiene que realizar el cálculo de las posibles amenazas que puedan existir hacia el activo. Para poder hacer la prueba de penetración más eficiente, las técnicas a usar son:
  - Mapeo de secciones: Técnica para realizar un mapeo del objetivo, como la identificación del número de secciones, puertos, servicios, entre otrros.
  - Tecnologías usadas: Búsqueda de servicios, frameworks y tecnologías que son utilizadas por la aplicación web o servidor.
  - Validación de versiones: Confirmación de las versiones utilizadas comparada a la información proporcionada por cliente y herramientas.
  - Consulta de vulnerabilidades en CVE: Búsqueda de vulnerabilidades conocidas asociada a las tecnologías usadas y sus versiones.

- Análisis de vulnerabilidades: En esta fase, una vez encontrados los activos, la información relacionada y cómo se manejan dichos activos e información, se realiza un análisis de vulnerabilidades con ayuda de herramientas automatizadas o pruebas manuales. Algunas de las herramientas a usar en esta fase suelen ser:
    - Nessus: Escáner de vulnerabilidades de código abierto que ayuda a identificar y evaluar posibles vulnerabilidades en sistemas informáticos y redes.
    - Nikto: Herramienta de código abierto para escanear y evaluar la seguridad de servidores web en busca de vulnerabilidades y posibles riesgos.
    - OWASP ZAP: Herramienta de código abierto para realizar escaneos automáticos y manuales, identifica riesgos de seguridad y proporciona informes detallados para mejorar la seguridad de las aplicaciones web.
    - OpenVas: Plataforma de escaneo de vulnerabilidades de código abierto que ayuda a identificar y gestionar riesgos de seguridad en redes y sistemas.
    - Burpsuite Scan: Extensión de la popular herramienta Burp Suite diseñada para realizar escaneos automáticos de seguridad en aplicaciones web.
    - Arachni: Herramienta de código abierto para el análisis automatizado en busca de vulnerabilidades y produce informes detallados sobre posibles riesgos de seguridad.

- Explotación: Una vez identificadas las vulnerabilidades, en esta fase se puede continuar con el fin de aprovechar las debilidades y lo que se identificó en el escaneo. De acuerdo con los resultados obtenidos en la fase anterior, se realizará de forma manual un seguimiento u obtención de evidencia de dichos hallazgos. Con las herramientas y nuestra prueba manual, se lograrán descartar falsos positivos o negativos. La finalidad de esta fase es poder tomar algún activo del sistema. Las herramientas a usar en esta fase son:
    - SQLmap: Herramienta de código abierto ampliamente utilizada para automatizar peticiones y evaluar parámetros vulnerables de inyección de SQL en aplicaciones web.

    - Fuzzer: Herramienta de seguridad que automatiza pruebas de penetración al enviar entradas aleatorias o específicas a una aplicación o sistema para identificar vulnerabilidades y fallas de seguridad.
    - Metasploit: Plataforma de prueba de penetración de código abierto que proporciona una amplia gama de herramientas y recursos para realizar verificaciones de seguridad en sistemas y aplicaciones.
    - Msfvenom: Herramienta incluida en el framework Metasploit que se utiliza para generar payloads (cargas útiles) personalizadas para ataques de explotación.
    - BurpSuite Proxy: Herramienta de interceptación de proxy utilizada en pruebas de seguridad de aplicaciones web.

- Post-Explotación: Una vez realizada la explotación, se debe mantener, el acceso al objetivo, realizar extracción de información en códigos fuente para crear una persistencia o ingresar una bandera en el sistema que evidencie que se logró ingresar. En este caso, la prueba suele ser manual, ya que se puede tener más control de las acciones que se toman ante el sistema relacionado a la vulnerabilidad.
- Reportaje: En esta fase se finaliza la prueba de penetración. Se ha recopilado toda la información y evidencia posible para poder realizar un reporte donde se plasman los hallazgos, vulnerabilidades, documentos, dónde se encontró y qué se realizó. Sirve esta fase como resumen breve de todos los resultados encontrados, así se puedan contabilizar y medir el riesgo.

Al finalizar cada proceso y fase, depende de la metodología utilizada y el objetivo definido por la organización cómo se realizarán los informes y reportes en los cuales se definirá qué sucede, cómo sucede y por qué sucede. Asimismo, se podrá identificar que, si fuera el caso que se lograran explotar dichas vulnerabilidades, qué sucedería a nivel interno, comercio y valor de la organización. Para ello, se realizan los reportes de entrega o informes.

1.4 Reporte de resultados

Al realizar pruebas de penetración como consultor hacia una organización, ayuda a la medición de seguridad de un sistema. Se observan de igual manera las habilidades que se necesitan ser empleadas para la obtención de resultados favorables y convincentes de la verificación de seguridad. Para poder observar de manera global y específica los resultados obtenidos, se recurre a realizar un reporte que ejemplifica las mediciones del riesgo obtenido, explicando qué sucedería si se materializa la vulnerabilidad, el proceso que se realizó para llegar a ese punto y las consecuencias. Esto se realiza en un documento escrito para poder entregar un breve resumen de lo sucedido en el tiempo acotado. Para los resultados, se implementan diferentes reportes:

- Reporte técnico: En este documento se agrega todo medio disponible y concreto que ayude a explicar de manera clara y concisa la vulnerabilidad, impacto y riesgo con ayuda de evidencia con la finalidad de recabar el material e información que demuestre cada paso realizado para las vulnerabilidades encontradas y que se haya logrado verificar. Se realiza el escrito de una manera técnica con expresiones y narraciones dirigidas hacia encargados de área de seguridad de la organización, siendo lo más técnicos posibles y que se encuentre una solución los más pronto posible. En ocasiones los reportes técnicos suelen ir con recomendaciones realizadas por los consultores para resolver dicha vulnerabilidad.

- Reporte ejecutivo: La construcción de este documento se implementa con una forma clara, dirigida al encargado de la organización, donde se toma en consideración que no tendrá los conocimientos técnicos para entender un reporte técnico. En este caso, se implementa el documento con la situación de lo que puede ocurrir y lo que puede suceder con la organización en caso de que se materialice la amenaza, cuánto tiempo puede tardar en solucionarse y a nivel monetario cuánto puede llegar a afectar. Depende del objetivo principal al que oriente la organización, se puede explicar por medio de gráficas o estimaciones en las cuales los activos de la organización sean tangibles.

Estos documentos se realizan teniendo en cuenta a todas las personas que se involucran y leerán los reportes, ya que no piensan igual y no son iguales. Al desarrollarse, debe reflejar lo que se haya realizado, la información obtenida y que cada fragmento escrito, ya sea a nivel técnico o ejecutivo, sea entendible para todo público. Al ser un reporte ejecutivo, tendrá que ser un lenguaje más formal y de alto nivel debido a que comúnmente lo que se involucra en dicho reporte es el impacto asociado a cada vulnerabilidad encontrada, en el cual deberá gestionar de manera correcta cada una, asignándolas a su equipo de trabajo. A diferencia del reporte técnico, que viene siendo una construcción de palabras técnicas, frases o incluso capturas de pantalla en las cuales se explica qué pasos se siguieron y detalladamente qué sucede en cada vulnerabilidad, este reporte se realiza de manera ligera y entendible para que así se pueda mostrar a nivel organizacional todo el impacto que se puede provocar si no se soluciona.

Cabe mencionar que, en algunas ocasiones, las herramientas que se utilicen pueden brindar la opción de crear un reporte que se relacione a la prueba de penetración y los servicios analizados asociados hacia el enfoque producido de la herramienta. Por ejemplo, si se usa una herramienta que tiene como enfoque analizar el comportamiento de los puertos utilizados para el funcionamiento, sus reportes podrán producirse con el mismo enfoque, solo al análisis de puertos. En muchas ocasiones, algunas herramientas no tienen un formato personalizado asignado para la creación de reportes, ya que solamente la herramienta realiza una recopilación de la información obtenida y la convierte a un formato de texto codificado o tipo de archivo predeterminado.

1.5 Descubrimiento y análisis de vulnerabilidades

El análisis de vulnerabilidades consiste en la investigación y detección de las debilidades, principalmente encontradas en la interacción de la aplicación de forma activa. Al presentarse una debilidad, esto orienta a que podría existir una vulnerabilidad. Por ejemplo, si al encontrar que en la sección de login de la aplicación se observa que al ingresar caracteres especiales permite el envío hasta el servidor, se podría considerar como una debilidad, ya que permite que se ingresen caracteres diferentes a los comunes, lo cual podría provocar una vulnerabilidad. Esta vulnerabilidad podría ir desde un bypass

al login hasta una inyección SQL, mediante la cual se podrían extraer los datos de la organización o de la aplicación.

De acuerdo con (INCIBE, Instituto Nacional de Ciberseguridad, 2017)[23] "una vulnerabilidad es la forma en la que se podría presentar que falle un sistema permitiendo que se pueda explotar por un atacante haciendo la relación a un riesgo que pueda afectar a la organización o a un activo de ella".

Existen diferentes tipos de vulnerabilidades, las cuales se presentan como:

- Físicas: Son aquellas que llegan a presentar fallos a nivel infraestructura. En ellas se involucran las amenazas de desastre natural y se relacionan dependiendo de cómo esté estructurada la organización o el ambiente en el que se encuentra organizada. Es decir, no es lo mismo un ambiente de un cuarto cerrado en los servidores, en los cuales puede existir una afectación a nivel de temperatura, que un cuarto con las ventilaciones asignadas y cada distribución pensada para un enfriamiento natural. Algunas vulnerabilidades relacionadas con lo físico podrían relacionarse con los controles de acceso. Si se está bien protegidos vía red externa, pero no se estableció protección a nivel físico, una persona podría realizar una intrusión en la organización si no se tiene validación para los accesos y periféricos. Por ejemplo, podrían ingresar con una USB, copiar la información e infectar la red interna.

- Lógicas: Se consideran como aquellas que pueden afectar dos campos, la infraestructura y su funcionalidad. Algunos ejemplos de vulnerabilidades lógicas son:
    - De configuración: Esto ocurre cuando el sistema operativo tiene configuraciones por defecto o el firewall tiene configuraciones incorrectas, lo que podría afectar a nivel perimetral o exponer información, accesos directos a las aplicaciones y el sistema en el servidor.

[23] (INCIBE, Instituto Nacional de Ciberseguridad, 2017)

- De actualización: En ocasiones, las organizaciones posponen las actualizaciones del sistema o a las aplicaciones debido a que podrían parar su producción o servicio. Sin embargo, es importante considerarlas, ya que ayudan a disminuir las vulnerabilidades previamente registradas. Algunas actualizaciones incluyen parches para solucionar estas vulnerabilidades.
- De desarrollo: En este caso, las vulnerabilidades se encuentran en la estructura de la aplicación a nivel de código. Pueden ser desde una inyección SQL hasta un Cross Site Scripting, dependiendo de la funcionalidad de la aplicación y las validaciones que tenga en la entrada de datos y el manejo seguro de estos en el flujo de la aplicación. En ocasiones, son relacionadas con las buenas prácticas de desarrollo que permiten evitarlas.

Lo fundamental para mitigarlas es identificarlas: conocer cuáles son y en qué parte se encuentran. Como primer paso, sería realizar un escaneo de detección de vulnerabilidades o una prueba de penetración, los cuales revisarán desde el software, sus configuraciones y cada dispositivo involucrado. Al realizar estos escaneos, se logra determinar dónde se encuentra una vulnerabilidad. Estos pasos ayudarán principalmente en la detección de cada una.

Una vez identificadas las vulnerabilidades, es necesario comenzar con un ciclo de remediación, el cual permite clasificar en un inventario cada vulnerabilidad y, a su vez, verificarlas. En ocasiones, cuando se realiza el escaneo de vulnerabilidades con una herramienta automatizada, puede mostrar las siguientes opciones: positivos, negativos, falsos positivos y falsos negativos. Con la verificación manual de las mismas, ayudará a validar los resultados mostrados por el escaneo.

Se debe tener en mente que es casi imposible mitigar cada vulnerabilidad encontrada debido a que se pueden observar diferentes parámetros como lo son vectores de entrada, secciones con diferentes acciones, servicio y cada vulnerabilidad puede tener un riesgo asociado diferente dependiendo su explotabilidad. Para ayudarnos en el proceso y hacerlo

más eficaz, es necesario clasificarlas y priorizar el riesgo de mayor a menor, de esta manera se puede realizar la mitigación de forma ordenada e importante sin perder el foco principal que es la protección de los activos de la organización. En ocasiones, se deben considerar los factores principales para un proceso, como el tiempo, el personal y el financiamiento. Estos puntos ayudarán a realizar una inversión eficiente para lograr disminuir primero las vulnerabilidades más graves en el sistema que puedan producir impactos críticos, y segundo, las menos riesgosas que puedan solucionarse con un parche de seguridad o actualización. La severidad de una vulnerabilidad se puede calcular tomando en cuenta la probabilidad de que esta ocurra y el impacto que tendría sobre los activos de una empresa, dicha relación se representa en la siguiente imagen (véase figura 2):

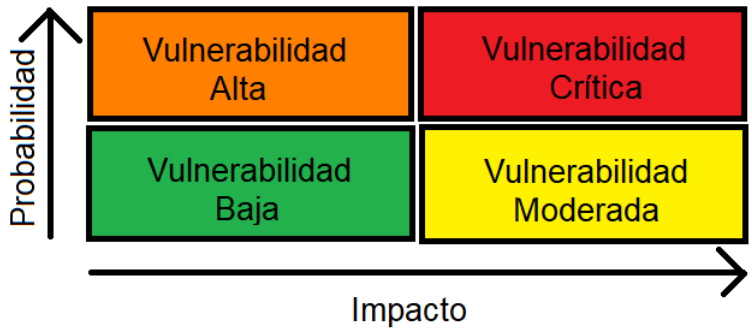


Figura 2 Relación probabilidad e impacto

1.6 Gestión de incidentes y monitoreo

La comprensión principal para un incidente de seguridad se debe entender como cualquier suceso que pueda provocar una pérdida de funcionalidad de una organización, ya sea interrumpida o que dicho hito de seguridad afecte la confidencialidad, integridad o disponibilidad.

Una buena medida de seguridad se conoce como el monitoreo de acciones a servicio o servidores. Monitorear según (Diccionario de la lengua española (Monitorear), 2022)[24] “Observar mediante aparatos especiales el curso de uno o varios parámetros fisiológicos

[24] (Española, Diccionario de la lengua española (Monitorear), 2022)

o de otra naturaleza para detectar posibles anomalías". Llevando esa definición a la seguridad informática, es la identificación de actividades y solicitudes que se puedan realizar entre un servicio y el usuario, realizando una comparación entre el comportamiento natural de las solicitudes y las que puedan arrojar una alerta ante conductas que puedan ser sospechosas por la construcción que se tiene, cabeceras o datos enviados.

En la situación empleada hacia un servicio web, las acciones maliciosas suelen ser realizadas externamente, ya que al ser un servicio web o en la nube permite que cualquier individuo con acceso a ella pueda interactuar enviando peticiones y recibiendo solicitudes del servidor como respuesta. Así que para esta situación se desarrolla el registro y almacenado de estas acciones, conocido como "Log", estableciendo la primera barrera de protección ante cualquier atacante, lo que permitirá también tener un rastreo de la anomalía en proceso o sucedida. Al registrar el movimiento de usuarios registrados y no registrados, permite realizar un seguimiento de las acciones maliciosas, teniendo un buen soporte dirigido al usuario en caso de una falla técnica, aplicando uno de los principios de la triada, mantener la disponibilidad del servicio ante cualquier situación.

La generación de guías para una gestión de incidentes como objetivo principal es plantear actividades que ayuden a cumplir con un ciclo de gestión y respuesta de incidentes, usando como guía para las actividades a definir "NIST" alineados a la norma "NTC–ISO–IEC 27035". Se recomienda a las organizaciones la creación de un equipo enfocado a definir cada proceso a realizar teniendo un enfoque al incidente, atención y gestión, administrado hacia medios internos y externos, clasificar los incidentes que pueden ocurrir y así llegar a la construcción de lo siguiente.

- Preparación
- Detección y análisis
- Contención, erradicación y recuperación
- Actividades posts-incidentes

La realización del monitoreo para encontrar hallazgos ayuda a prevenir situaciones de riesgo, su impacto y lo que puede suceder al encontrarlo en forma tardía o temprana. Todo

depende de la gestión que se pueda realizar entre hallazgos y posteriormente las vulnerabilidades encontradas para su mitigación, proceso y seguimiento de estos.

Los logs se implementan como un documento de respaldo en los que comúnmente se registran equipos de empleados, servidores, peticiones de información, etcétera. En los dispositivos conectados a la red ayudará que estos registros se realicen de forma correcta; por ejemplo, el registro puede ser por un intento malo de inicio de sesión hasta una transacción mal realizada, ya que en cualquier momento en un sistema pueden existir fallos en la aplicación y su flujo natural. El comportamiento entre aplicación y usuario se puede detectar como una anomalía, ya que no se puede asegurar que nunca pueda fallar la aplicación, la red, que el usuario inserte caracteres erróneos o maliciosos, entre otras.

Para ello, además del monitoreo y la creación de logs en los sistemas informáticos, existen a su vez los sistemas de detección de intrusos (IDS). Estos sistemas se encargarán de detectar y reaccionar de forma automática ante cualquier posible incidente o intento de intrusión. Se realiza una alerta o notificación a los administradores correspondientes del sistema informático, siendo así una forma de poder detectar cualquier actividad sospechosa mediante una serie de alarmas e informes.

Los sistemas IDS se distinguen principalmente por un diseño común y básico:

- El origen que proporciona información donde se produce algún evento en un sistema o red de informática.
- Al tener una base de datos en la cual se tendrá patrones de comportamientos relacionados al flujo normal a un perfil o conductas asociadas a ataques de acuerdo con sus solicitudes recibidas que pueden ser normales y anómalas.
- Un motor de procesamiento para el análisis obtiene la mayor parte de las evidencias y notificaciones que se produzcan en el momento, dando una ventana de tiempo y rastreabilidad.
- Un sector que sea capaz de tomar acciones determinadas a partir de las indicaciones del motor, donde se tendrá la asignación de instrucciones a los comportamientos anómalos.

A su vez existen dos diferentes tipos de IDS:

- Detección de un mal uso: Enfocado a cualquier tráfico considerado como ilegal o aquellos que son enviados con la finalidad de realizar ataques, como exploits, escanear puertos, etcétera.
- Detección de un uso anómalo: Enfocado a cualquier tráfico que tiene un análisis estadístico asignado para observar los procesos y comportamientos de usuarios, intentando detectar comportamientos diferentes de los servicios o aplicaciones para los cuales fueron desarrollados originalmente.

La funcionalidad del IDS es capaz de responder de forma automatizada a los incidentes detectados de acuerdo con sus reglas, logrando obtener dos tipos respuestas:

- Pasivas: El objetivo principal es detectar y registrar posibles intrusiones o usos anómalos identificados de acuerdo con las reglas establecidas. En este caso, se pueden generar informes y alertas dirigidas a los administradores de los servicios.
- Activas: El objetivo principal es responder ante cualquier situación o intrusión de manera inmediata, realizando acciones como anular conexiones o bloquear accesos determinados a usuarios o equipos desde la red.

1.6.1. Monitoreo y gestión de vulnerabilidades encontradas

Al realizar la creación de los logs de movimientos internos y externos, permite tomar acciones a beneficio de la organización, logrando así que esté preparada y dando seguimiento a las anomalías registradas. Adicionalmente, las herramientas que se establecen como controles dan una base para un plan contra incidentes, definiendo las acciones a realizar ante alertas. El seguimiento de los registros en distintos escenarios ayuda a anticipar lo que puede suceder; se diseñan para que la organización esté preparada para lo más probable hasta lo menos probable. Brindan las bases para tener una visualización de estos registros y realizar un seguimiento adecuado.

Una vez registrado el incidente en los logs y definidas las acciones establecidas en relación con los escenarios, se procede con un rastreo de esas posibles conductas

anómalas. Por ejemplo, si se encuentra un intento de sesión fallido, una acción a realizar podría ser bloquear al usuario durante cierto tiempo para evitar un ataque automatizado. Como segunda acción, se analiza la frecuencia de este comportamiento, y si se permitieron más de 3 intentos, esta acción se identifica como un acto anómalo en comparación con el comportamiento natural. En este caso, se realiza automáticamente el bloqueo de la cuenta hasta que se verifique que se trató de un error por parte del cliente y no un intento de ataque. Todo esto se realiza de acuerdo con el IDS, ya que es el encargado de identificar estos comportamientos y tiene establecidas reglas basadas en instrucciones previamente definidas. Además, el responsable del área también se encarga de tomar acciones, como contactar al responsable del usuario para que se comunique con el administrador del área correspondiente y realizar el proceso de desbloqueo de la cuenta en caso de ser un error humano y no un ataque.

La seguridad en un sistema y el análisis de vulnerabilidades que se realiza a la organización darán como resultado puntos débiles que deberán ser mitigados o, si no es posible, al menos controlados con un cuidado detallado. Esto significa que se deben registrar los comportamientos maliciosos o erróneos que puedan surgir ante estas vulnerabilidades no mitigadas, ya que nunca se podrá garantizar al 100% que un sistema está protegido. Sin embargo, se puede asignar un recurso para poder responder a tiempo. Un ejemplo de esto es la policía en la sociedad: no se puede tener una sociedad completamente segura siempre, lo que lleva a mantener registros de cámaras y a mantener las unidades en campo en constante movimiento, observando el comportamiento ante cualquier acto sospechoso. Una vez que se identifican actos sospechosos, se procede a realizar una detención en la cual se analiza si fue con intención de malicia o si simplemente fue un error, este comportamiento se puede identificar de acuerdo a las reglas implementadas en nuestro monitoreo y la carga útil en caso de existir o una traza realizada al comportamiento asociado a esta alerta.

1.6.2. Cálculo de probabilidad y planes de acción

Al conocer las vulnerabilidades de una organización, las probabilidades de que un atacante intente buscarlas pueden ser infinitas, aunque existan debilidades que no hayan

sido compartidas o encontradas. Tarde o temprano, un atacante intentará descubrir nuestra infraestructura y aplicaciones para posteriormente realizar una intrusión hacia ellas. Al existir una infinidad de posibles escenarios para los atacantes, debido a que existen muchos caminos por los cuales puedan analizar y explorar nuestros sistemas, la organización deberá estar preparada para un escenario global que abarque ciertas vulnerabilidades registradas e identificadas, así se podrá estimar la probabilidad de que esta suceda. La probabilidad se refiere, según (Diccionario de la lengua española (Probabilidad), 2022)[25] "Es un proceso aleatorio, la razón entre el número de casos favorables y el número de casos posibles." Entendiendo esto, al asociar un número a algo que no se sabe si pueda suceder o no, se realiza un plan de acción teniendo en cuenta el enfoque de sistemas informáticos: ((OWASP), OWASP Top 10 - 2017 Los diez riesgos más críticos en aplicaciones web, 2017)[26]

- Explotabilidad: Es la oportunidad de cuánto se conoce el sistema, tecnología o proceso. Por ejemplo, sería cuántas debilidades se conocen de un sistema que está programado para salir al público en 2025 en comparación con un sistema que salió al público en 2015. Esto se asocia en una relación, donde se puede obtener el número de personas que ya conocen y usan el producto, tomando en cuenta el tiempo que se liberó al público y si este mismo sigue obteniendo actualizaciones, ya que es un factor de interés saber si se brinda soporte por el creador.

- Prevalencia: Es la relación que se tiene con el soporte que puede dar el creador de la tecnología con un grupo determinado de reportes de vulnerabilidades registradas y cuáles fueron arregladas en sus actualizaciones. También se asocia con los diferentes grupos de testers que se encuentran al público. No sabrá con qué fines se vayan a usar estas vulnerabilidades; en ocasiones, estos grupos se dedican a realizar hallazgos o intentar vulnerar las tecnologías para posteriormente conseguir un beneficio o compartirlo en un foro de hackers. Todo esto define la capacidad de

[25] (Española, Diccionario de la lengua española (Probabilidad), 2022)
[26] ((OWASP), OWASP Top 10 - 2017 Los diez riesgos más críticos en aplicaciones web, 2017)

soporte que el creador da hacia sus usuarios, intentando sacar parches de seguridad para prevenir dichos hallazgos.

- Detectabilidad: La facilidad con la que se puede descubrir una debilidad determina la escalabilidad de tiempo requerido para su detección. Es importante considerar si se necesitan conocimientos avanzados o básicos para aprovecharla o si existe algún registro que dé información de ella y si se requiere estar registrado en la organización para acceder a dicha funcionalidad. Estos factores ayudarán a evaluar qué tan difícil es explotar estas vulnerabilidades.

Además de estos elementos para poder identificar un riesgo y probabilidad, se asocian factores que pueden indicar qué planes de acción implementar, así como calcular y observar los posibles escenarios. Para ello, el cálculo de un riesgo y la probabilidad son cruciales para armar un plan de acción en el cual se consideran los siguientes puntos (Project, OWASP Risk Rating Calculator, 2018)[27]:

- Factores de Amenaza: Los factores de amenaza son aquellos argumentos que ayudarán a identificar qué tan probable es que un ataque pueda tener éxito, lo cual mide desde el conocimiento hasta las razones de dicho ataque. De igual manera, se tiene un esquema de medición en donde se enumeran del 0 al 9 para conocer el riesgo e impacto que se produce, dicha medición ayuda a conocer cuánto se necesita para tener éxito. Los factores son los siguientes:
    - Habilidades necesarias:
        - 1. Sin habilidades técnicas.
        - 3. Algunas habilidades avanzadas de computadoras.
        - 5. Habilidades en red y programación.
        - 9. Habilidades avanzadas de prueba de seguridad.
    - Motivo:
        - 1. Recompensa baja o nula.

[27] (Project, OWASP Risk Rating Calculator, 2018)

    - 4. Posible recompensa.
    - 9. Recompensa alta.
  - Oportunidad:
    - 1. Acceso completo o recursos costosos.
    - 4. Acceso o recursos requeridos.
    - 9. Sin acceso o recursos requeridos.
  - Tamaño:
    - 1. Desarrolladores.
    - 3. Administradores.
    - 5. Usuario intranet.
    - 7. Usuario autenticado.
    - 9. Usuario anónimo.

- Factores de vulnerabilidad: Los factores de vulnerabilidad son aquellos que ayudarán a identificar desde qué tan complejo fue encontrar dicha debilidad hasta si hay rastreabilidad en nuestros movimientos. De igual manera, se tiene un esquema de medición en donde se enumeran del 0 al 9 para conocer el riesgo e impacto que se produce, dicha medición ayuda a conocer cuánto se necesita para tener éxito. Los factores son los siguientes:
  - Facilidad de descubrimiento:
    - 3. Difícil
    - 7. Fácil
    - 9. Herramienta automatizada
  - Explotación:
    - 3. Difícil.
    - 5. Fácil.
    - 9. Herramienta automatizada.
  - Conocimiento:
    - 1. Desconocido.
    - 4. Oculto.
    - 6. Obvio.

    - 9. Conocimiento público.
- Detección de Intrusión:
    - 1. Detección en aplicación.
    - 3. Registro en bitácora con revisión.
    - 7. Registro sin revisión.
    - 9. No registro en bitácora.

- Factores de Impacto Técnico: Los factores de impacto técnico son aquellos que ayudarán a identificar desde el punto de vista de la triada si viola alguno de ellos. Por ejemplo, si se logró extraer documentos sensibles o si se pueden modificar, y cuántos activos se ven afectados. De igual manera, se tiene un esquema de medición en donde se enumeran del 0 al 9 para conocer el riesgo e impacto que se produce, dicha medición ayuda a conocer cuánto se necesita para tener éxito. Los factores son los siguientes:

    - Confidencialidad:
        - 2. Datos no sensibles divulgados.
        - 7. Datos críticos divulgados.
        - 9. Todos los datos divulgados.
    - Integridad:
        - 1. Ligeramente corruptos.
        - 7. Datos corruptos.
        - 9. Todos los datos corruptos.
    - Disponibilidad:
        - 1. Servicios secundarios interrumpidos.
        - 7. Servicios primarios interrumpidos.
        - 9. Todos los servicios interrumpidos.
    - Registro:
        - 1. Totalmente rastreable.
        - 7. Posiblemente rastreable.
        - 9. Anónimo.

- Factores de Impacto al Negocio: Los factores de impacto al negocio son aquellos que ayudarán a identificar qué tanto afectará el esquema de producción de la organización, desde un daño financiero hasta un impacto en la reputación ante clientes e inversionistas. De igual manera, se tiene un esquema de medición en donde se enumeran del 0 al 9 para conocer el riesgo e impacto que se produce, dicha medición ayuda a conocer cuánto se necesita para tener éxito. Los factores son los siguientes:

    - Daño Financiero:
        - 1. Costo menor que corregir vulnerabilidad.
        - 3. Efecto menor a ganancia anual.
        - 7. Bancarrota.
    - Daño de Reputación:
        - 1. Daño mínimo.
        - 4. Pérdida de cuentas principales.
        - 5. Pérdida de fondo de comercio.
        - 9. Daño a la marca.
    - Incumplimiento:
        - 1. Violación mínima.
        - 5. Violación clara.
        - 9. Violación de alto perfil a normas y estándares.
    - Violación de Privacidad:
        - 1. Información personal de un individuo.
        - 3. Cientos de personas.
        - 5. Miles de personas.
        - 9. Millones de personas.

1.6.3. Gestión de incidentes

Las organizaciones suelen implementar controles en los mecanismos de seguridad y acceso para poder disminuir cualquier tipo de amenaza externa que impacte gravemente. Por ello, existe la gestión de incidentes que se refiere a cuando una amenaza está en curso o está por suceder. En estos casos, se interponen planes de contención, protocolos

o procedimientos que se deben realizar una vez identificadas las amenazas y reconocidas por la organización que ha sido parte de un ataque informático o que se está en pleno ataque de intrusión, y el atacante ha logrado evadir todos los mecanismos de seguridad. Para ello, este plan de respuestas a incidentes debe contar con una serie de actividades y tareas por área, viéndose como una guía que se constituye por:

- Creación y formación de un equipo de respuesta incidentes

  Es la construcción de un conjunto de personas especializadas, el cual se llamará equipo de respuesta a incidentes. Son capaces y tienen la formación necesaria para la actuación ante amenazas y desastres que puedan afectar las bases de la información. Se tiene como objetivo que su proceso principal deberá ser ejecutado de inmediato, ya que cada segundo cuenta ante estos actos. Se definirá un directorio actualizado de teléfonos de los contactos que tienen poder de decisión, como el consejo directivo. En estas situaciones, la experiencia es una de las piezas clave para responder, ya que podría ser determinante en la forma de actuar y poder responder de forma inmediata y certera.

- Establecer y crear una guía de procedimientos.

Esta actividad la realiza el equipo y tiene como finalidad la creación de una guía que explicará las acciones, protocolos y procedimientos a seguir cuando un incidente se encuentre en curso. Esta guía ayudará a reducir el tiempo de respuesta y el impacto que pueda ocasionar. En resumen, se obtendrá una metodología donde se tendrán pasos a seguir de acuerdo con el incidente, con información detallada de lo que pueda suceder y que será analizada para futuros incidentes.

- Detección de un incidente de seguridad

La organización tendrá un enfoque en la notificación de algún comportamiento que indique que se encuentra bajo alguna amenaza, considerando globalmente que las posibilidades son infinitas, debido a que un sistema informático tiene demasiados vectores de ataque. Una intrusión al sistema mantiene una evolución de la tecnología usada, haciendo que la detección de los incidentes sea más fácil y a medida que dichas protecciones evolucionan, los ataques se vuelven más sofisticados, como herramientas open source que proporcionan la ocultación del origen del ataque. A su vez, como parte del equipo de incidentes, la información registrada en los logs ayudará a descartar falsos positivos y negativos. Contando con las herramientas necesarias y sumando las reglas y filtros, facilitarán la detección y su clasificación.

- Análisis de incidentes

Al detectar un incidente, el equipo de respuestas a incidentes procederá a analizar el registro, fecha, hora y lo que fue detectado para que se pueda clasificar de manera correcta las partes afectadas, desde la red, equipos de cómputo, aplicaciones, entre otros. Una vez identificados los dispositivos afectados, se continúa con el reconocimiento del tipo de ataque y, en caso de que un empleado haya sido detectado, proceder con el aislamiento. Adicionalmente, se clasifican las vulnerabilidades que han sido explotadas, los métodos empleados por el atacante y el escenario producido, entre otros. Emplear una matriz de diagnóstico puede ayudar

a la toma de decisiones, disminuir los errores y, a su vez, a la valoración inicial de posibles daños. El orden de prioridad ante el equipo de respuesta sería

- Proteger la vida humana y la seguridad de las mismas.
- Proteger datos e información sensible.
- Prevenir daños en los sistemas informáticos.
- Minimizar la interrupción del servicio al usuario o cliente.
- Contención, erradicación y recuperación.

Contención: Al recibir un ataque, se debe minimizar el incidente para que no tenga un mayor impacto. Se pueden tomar medidas extremas ante el incidente, como el reinicio total del servicio; ya que, si se permite más tiempo, se podrá brindar un punto más para un tercer ataque o un medio de pivote hacia otro más crítico.

Erradicación: En esta fase se seguirán al pie de la letra las actividades establecidas por la guía realizada desde un inicio por el equipo de respuesta a incidentes, donde el objetivo principal será eliminar cualquier archivo infectado y seguir el rastro que dirija hacia la causa y sus secuelas originadas. En esta fase se realiza un análisis de seguimiento de huellas, virus o accesos creados por el atacante hasta usuarios hackeados, así mismo se debe realizar una examinación exhaustiva de los diferentes sistemas de la organización.

Recuperación: Por último, se intentará la restauración de los servicios, sistemas y funciones de sistema para lograr regresar a su funcionamiento natural. Será necesario monitorear cómo inician nuevamente los servicios y, si es necesario, actualizar e instalar posibles parches de seguridad dependiendo de donde haya existido la filtración. Lo importante es volver a brindar seguridad a los clientes y a la información almacenada. Desde cambiar contraseñas hasta la reconfiguración de algunos sistemas, será un buen procedimiento para prevenir de nueva cuenta un incidente.

- Identificación del atacante y procedimiento legal.

Al recuperar el servicio, se tendrá que cambiar el enfoque a investigar a los posibles responsables del ataque, razones y consecuencias que hayan provocado el incidente y dirigirlo a sus causas. Por ejemplo, si los autores originales del suceso hicieron una

venta de la información o si hubo un delito informático. Todo eso se consigue con un análisis de los registros que se hayan hecho, ya que podrían brindar información de quiénes y cómo lograron acceder. En algunas ocasiones, entender el código da una pista de lo que realiza o las palabras asignadas por el atacante llevarán posiblemente a la nacionalidad de los autores.

- Comunicación a clientes, terceros y relaciones públicas.

Una vez minimizado el riesgo, reforzada la seguridad, identificados los posibles autores y listos los procedimientos legales, podrá comunicar a los clientes, terceros y a la sociedad que la organización ha sufrido un ataque y que es probable que los datos hayan sido afectados, o dependiendo de la afectación e impacto. El propósito del procedimiento es comunicar que el sistema fue afectado, que se identificaron los autores y la cantidad de información filtrada. Este proceso puede provocar un impacto económico y en la confianza de la organización, por lo que es importante comunicar tranquilidad y que la situación se encuentre bajo control con los parches de seguridad aplicados.

- Documentar el incidente.

Documentar lo sucedido ayudará en futuros escenarios, en caso de una represalia por parte de los atacantes o de otros motivados por el primero, ya que se dio a conocer la vulnerabilidad y podrían querer explotarla o usarla como punto de partida para un nuevo ataque. Por ello, se debe realizar una descripción detallada del incidente, registrando los hechos y eventos que se produjeron en la bitácora, todos con factores certeros y de funcionalidad. También es importante registrar los daños ocasionados en los sistemas o sectores, así como las comunicaciones realizadas ante los medios de comunicación. Es crucial listar y almacenar toda la evidencia encontrada en el análisis de la investigación. Todos estos hallazgos y documentación ayudarán en el futuro y en la preparación de una respuesta óptima y concisa.

- Análisis y revisión posterior del incidente.

Una vez recopilada la información, se recomienda crear un informe con los puntos importantes de todo el incidente, incluyendo las consecuencias asociadas. A través de este informe, podrá realizar un estudio y crear conciencia en toda la organización. Con una revisión detallada de las bitácoras, se descartan por completo los falsos positivos y se puede ver realmente el camino que tomó el atacante. Por último, el informe permitirá evaluar el costo del incidente para la organización y determinar en qué áreas invertir para mejorar la seguridad.

Capítulo ll
Aplicaciones web

2.1 Internet e hypertext transfer protocol

El internet es la red más grande de telecomunicaciones que existe, la cual permite la comunicación entre personas de diferentes países, la búsqueda de información y la facilidad en nuestras actividades diarias. El internet es un objeto no visible, y de acuerdo con su terminología, "Inter" se refiere a un enlace o conexión y "Net" proviene de la palabra "Network", que significa interconexión de redes. Asimismo, el entorno tecnológico se define como un conjunto no centrado de redes de comunicación en las cuales se interconectan utilizando protocolos de comunicación conocidos como TCP/IP.

El internet tiene su origen principal en el año 1958, cuando se funda el proyecto "Red de Agencia de Proyectos de Investigación Avanzada" (ARPA), el cual fue un proyecto realizado por el Departamento de Defensa de los Estados Unidos, en el que participaron alrededor de 200 científicos de alto nivel. Su objetivo era lograr la conexión entre computadoras de diferentes agencias localizadas en el territorio de los Estados Unidos. A lo largo de los años, ha experimentado diferentes avances tecnológicos, siendo los siguientes (Cataluña, U. P.,2008)[28]:

- 1972 ARPANET
- 1974 Telenet - Versión Comercial.
- 1979 Usenet - Sistema abierto centrado en e-mail.
- 1981 Bitnet - Unión de universidades americanas con ayuda de IBM.
- 1982 Eunet - Unía Reino Unido, Escandinavia y Holanda.

WWW (Word Wide Web)

El origen de lo que hoy es internet comienza en los inicios de 1990 con la creación de páginas web y es conocido por sus siglas WWW (World Wide Web), siendo un sistema de distribución donde se mandan documentos de hipertexto o hipermedios, permitiendo así que cualquier persona pueda acceder a ellos a través del internet. El comienzo de WWW fue por el CERN en Suiza, liderado por Tim Berners-Lee, donde se creó el primer lenguaje HTML en 1989, el cual permitió la construcción de páginas web sencillas. Un año después,

[28] (Cataluña, 2015)

el mismo equipo realizó la construcción del primer servidor web y cliente web, asignando el nombre como World Wide Web. Con los movimientos y su revolución tecnológica, las empresas comenzaron a realizar aportaciones, como el caso de Netscape en 1993 creando Mosaic y por parte de Microsoft, Internet Explorer.

Las fechas continuaron con las aportaciones, y en 1993, finalmente, lo que es el entorno web entró al dominio público, quedando disponible para cualquier persona de manera gratuita. La última generación de la revolución tecnológica conocida es la web móvil, donde los celulares implementaron por primera vez la conexión a internet, marcando el comienzo de una nueva era, ya que no se necesitaban equipos de escritorio para poder consultar sitios web. Todo podía lograrse desde un smartphone, lo que obligó a la mayoría de los creadores de páginas web a realizar una adaptación de sus aplicaciones web para que se pudiera observar de mejor manera su sitio, ajustándolo a un formato móvil debido al tamaño de pantalla que se propuso para los smartphones.

La creación de un sitio web comenzó con la integración de HTML como base principal, conocido por sus siglas "HyperText Markup Language", traducido al español como "lenguaje de marcado de hipertexto". Este lenguaje de programación fue la base para que varios sitios web pudieran integrar texto e imágenes con facilidad, estructurándolo de forma adecuada y estática. El desarrollo de un sitio web requería especificar la inserción de archivos multimedia, ya que se tenía que subir la ruta donde se consumían dichos recursos, los cuales se encontraban indexados en un listado de directorio. Para la búsqueda de recursos se implementaba bajo el mismo dominio principal, donde se creaba una ruta específica llamada "static" para colocar la multimedia. Gracias al dominio, se podía definir la URL, que se define como "una cadena de caracteres que asigna una dirección única a un recurso disponible en el espacio virtual". En términos técnicos, se conoce como "Uniform Resource Location" o "localizador uniforme de recursos" en español. Esto permitía organizar la información de la web en una dirección única que permitía acceder a un archivo, información o recurso creado en un sitio web.

La forma en que se creaban los sitios web y el conjunto de propiedades que caracterizan su construcción permitían el acceso de manera única y segura, lo que permitía procesar una transferencia de datos. Para esto, se creó HTTP, especialmente usado por la web, cuyas siglas en inglés son "HyperText Transfer Protocol", traducido al español como "protocolo de transferencia de hipertexto". Este protocolo se utilizaba para realizar transacciones entre el sitio web y el servidor, siendo la forma de comunicación en la cual se podían solicitar recursos o enviar peticiones. La solicitud y respuesta HTTP se mostraban en forma gráfica, como se puede observar en la siguiente imagen (véase figura 3).

Figura 3 Comunicación HTTP

El protocolo de transferencia de hipertexto es un conjunto de reglas que trabajan en unión para lograr la transferencia de texto, imágenes y, en general, cualquier archivo de texto o multimedia con un manejo de transferencia de archivos. Se basa en los protocolos TCP/IP, teniendo diferentes versiones de lo que se conoce como HTTP. En sus comienzos, se hizo la estructuración de un comportamiento muy simple, donde en una línea se realizaban solicitudes y solo se utilizaba un método HTTP, en este caso GET seguido de la ruta del recurso (GET /mypage.html). No se integraba toda la URL del recurso ya que en la comunicación del protocolo del servidor y el puerto no eran necesarios para conectarse al servidor. Una característica de este primer protocolo es que todo era muy sencillo y simple, en el cual solo podía transmitir archivos HTML, no existían códigos de estado o error y si se generaba un problema en el transcurso, se creaba otro archivo HTML especificando el detalle del error.

Debido a las limitaciones que generó la primera generación de HTTP, se realizaron modificaciones donde se puede observar un control de versiones, permitiendo el envío de más de una línea de código de estado al comienzo de una respuesta, así lograron que los navegadores reconocieran el éxito o el fracaso de una solicitud. Una de las características destacadas fueron las cabeceras HTTP, ya que permiten la transmisión de metadatos, haciendo flexible y abierto el envío de datos con el encabezado "content-type", permitiendo que se pueda recibir diferentes tipos de datos o archivos que no fueran HTML. Estas características se implementaron en el protocolo HTTP/1.0.

El protocolo HTTP/1.1 fue el primer protocolo que se aceptó de forma estandarizada debido a su estructura. Se integraron características que permiten conexiones diferentes y que podrían reciclarse en caso de no realizarse exitosamente, logrando que existan más de una solicitud en el mismo cliente servidor. Adicionalmente, se integraron mecanismos para realizar un control de caché y finalmente se integró la facilidad de negociar qué contenido se va a transmitir mediante las cabeceras integradas, en este caso "content-type" y la cabecera "host", lo que permitió alojar diferentes dominios en una misma dirección IP.

La última versión que se conoce es HTTP/2. Se muestra una mejora en diferentes perspectivas, ya que se representa como un protocolo binario y, a su vez, se identifica como multiplexado, lo que permite realizar solicitudes paralelas a las respuestas a través de una misma conexión. Esto es capaz de eliminar contenido similar, liberando la sobrecarga en la red, lo cual es una mejora importante ya que la mayoría de las comunicaciones se realizan de esta manera. También proporciona seguridad a las cookies con sus atributos correspondientes, como "secure" y "httponly", lo que permite trabajar con las sugerencias de una búsqueda y permitir al navegador comunicarse de manera más eficiente y segura.

2.2 Los protocolos de comunicación segura.

Toda comunicación realizada entre un cliente y un servidor se debe hacer de manera eficiente y correcta. Se deben realizar, todos los días, interconexiones que validen si existe una comunicación segura entre sistemas y redes de computadoras, ya sea al realizar una conexión vía internet mediante un navegador web o al solicitar peticiones de servicios mediante otros protocolos. Siempre se debe validar que la transmisión de datos se realice de manera segura entre el destinatario y el remitente, sin importar el origen, nación y estado. Esto se logra mediante protocolos de comunicación, los cuales se definen como un conjunto de reglas o estándares que conforman restricciones, procedimientos y formatos para el intercambio de paquetes de información y permitir la comunicación entre dos servidores o más hacia dispositivos conectados en una red. Al enviar los datos de manera correcta y exitosa, se enfatiza en que se realice de forma segura, permitiendo que los mecanismos trabajen con dispositivos que ayuden a identificarse entre ellos y a establecer una conexión segura.

Al navegar en internet, se realiza una interacción entre nuestros dispositivos y el internet, intercambiando datos en la web que se solicitan en nuestro paso. En algunas ocasiones, esto ocurre de forma natural o automática, y en otras, el usuario tiene que interactuar con ellos, lo que lleva a buscar protocolos de seguridad para garantizar la comunicación de canales seguros y la integridad de los datos en tránsito. Se hace hincapié en evitar que usuarios, aplicaciones, sitios web o servicios de internet tengan acceso no autorizado a los datos en los canales. Para ello, se han creado diferentes protocolos de internet que establecen condiciones distintas por las cuales podrían adaptarse al tipo de información a asegurar, como son:

Protocolo HTTPS

El protocolo es la versión extendida o fusionada de lo que se conoce como el Protocolo de Transferencia de Hipertexto con el protocolo SSL/TLS, por sus siglas en inglés ("Secure Sockets Layer / Transport Layer Secure"); ambos protocolos se diseñaron para que la comunicación en la red de internet sea segura, y se pueden utilizar de forma conjunta o independiente. Para representar lo que se conoce como HTTPS, en forma conjunta con

una conexión HTTP, el cliente solicita acceso al servidor, pero con la condición de tener una comunicación segura. El servidor responde que la conexión establecida entre ambos es segura. Una vez realizada esta confirmación, el cliente envía la clave pública y todos sus parámetros de seguridad. Esta fase de comunicación se establece cuando el servidor, al recibir la clave pública y sus parámetros, realiza una búsqueda completa encontrando una coincidencia con lo recibido. Así, se autentica al cliente y se valida la conexión, estableciendo una sesión SSL. Esta sesión finaliza cuando el cliente o servidor cierran la conexión. Mientras sigan conectados, su transmisión se considera segura. Algunos protocolos de comunicación son[29]:

- Protocolo SSH: Es conocido por la permisibilidad que concede para la administración remota de un dispositivo. Permite a un usuario controlar, modificar y crear servicios a través de internet o mediante un mecanismo de autenticación de forma remota. Este protocolo permite identificarse y encriptar la conexión entre dos computadoras que se conectan a internet. SSH es utilizado comúnmente para realizar administraciones de red o sistemas de forma externa, en las cuales se podrán realizar actividades como si se tuviera el dispositivo de forma física.

- Protocolo FTP: El Protocolo de Transferencia de Archivos permite, como se menciona, transferir archivos por medio de internet. Utiliza un cliente-servidor para lograr esta transferencia, siendo de forma remota la comunicación que se crea. La conexión se basa en un HTTP o HTTPS para consultar páginas web, debido a que la transferencia de archivos se realiza entre dispositivos conectados a internet.

- Protocolo DNS: El Sistema de Nombres de Dominio permite mantener un directorio de nombres que son sustituidos por una IP, un número único de identificación en la red que permite identificar y ubicar la dirección web asociada a tal dirección IP. Tanto el nombre como la dirección IP son únicos y están asociados entre sí.

Es muy importante diferenciar un protocolo de red de un protocolo de comunicación, ya que, aunque trabajen en conjunto asegurando la comunicación, no son lo mismo. Un

[29] (Doctoralia, 2018)

protocolo de red consiste en normas o estándares que fijan la forma, formato o datos que se podrán transportar, permitiendo que se pueda actuar y permitir la conexión de acuerdo a reglas establecidas y condiciones que se asignen. Por otro lado, un protocolo de comunicación puede incluir diferentes tipos de formatos, señales o datos escritos en cualquier código, con la finalidad de ser expresados con un tipo de información definido. En resumen, el protocolo de red está definido para el traspaso de información de forma segura, con la finalidad de restringir de acuerdo con reglas y condiciones para asegurar la confidencialidad e integridad de los datos en tránsito.

2.2.1 Protocolos SSL y TLS

Por sus siglas en inglés, "Secure Sockets Layer" se traduce al español como "capa de sockets seguros" y es comúnmente usado para cifrar y autenticar datos mientras son enviados entre aplicaciones desde un servidor web a un cliente. TLS, por sus siglas en inglés "Transport Layer Security", se traduce al español como "seguridad de la capa de transporte" y es el sucesor de SSL.

Protocolo SSL

Los protocolos SSL fueron desarrollados por Netscape, siendo su primera versión SSL v1.0, que en realidad no fue lanzada públicamente debido a que se consideraba uno de los protocolos más inseguros y se consideró necesario corregir las fallas encontradas. Esto dio origen al protocolo SSL v2.0, lanzado en 1995. Si bien se lograron disminuir los errores principales del protocolo anterior, con el paso del tiempo se encontraron nuevas debilidades de seguridad en este protocolo. Estas debilidades estaban relacionadas con el uso del mismo código criptográfico para la autenticación y cifrado de mensajes, así como con la construcción del código criptográfico que se consideraba débil debido al uso del hash MD5.

Un año después, en 1996, se creó SSL v3.0, que en su momento fue considerado un protocolo seguro. Sin embargo, posteriormente se convirtió en el origen para la creación de los protocolos siguientes, que son TLS.

Al construir la conexión entre un servicio y el internet, se utilizan los protocolos para asegurar la comunicación y navegación de los datos, siendo el único servicio a utilizar HTTP, con su sucesor HTTPS, que es la utilización del protocolo SSL/TLS combinado con HTTP.

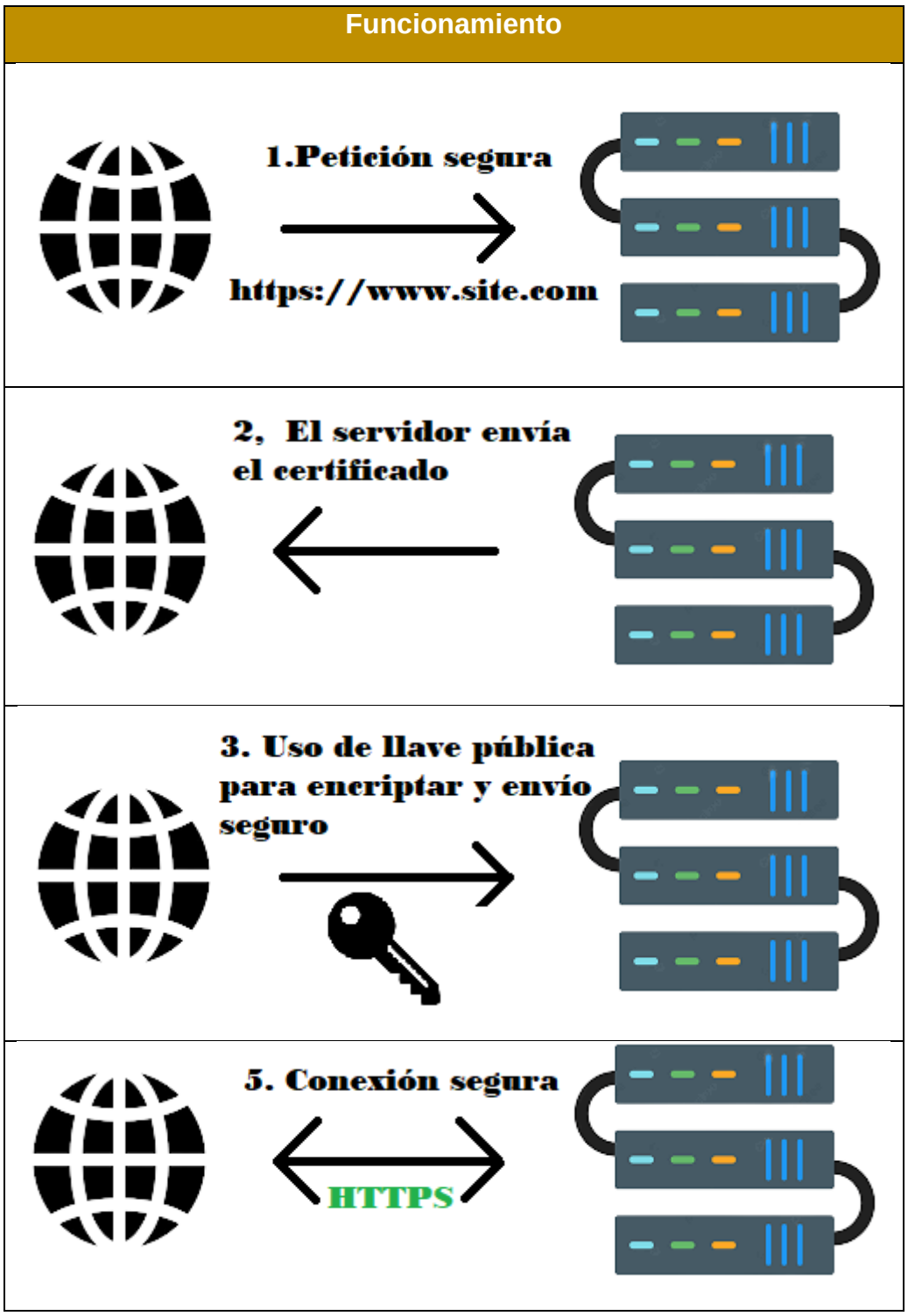
Protocolo TLS

El protocolo TLS se creó principalmente por la misma organización "Netscape", donde surge TLS v1.0 en 1999. Su aparición fue como una actualización de la versión SSL v3.0. En la actualidad, existen organizaciones que, al crear su infraestructura de red, no deshabilitan dicha conexión que, a la fecha, se considera insegura, lo que permite que se puedan explotar debilidades para materializar un ataque informático de comunicación. Este protocolo se encarga de realizar un intercambio de registros donde se envuelven datos con un formato específico. Estos registros pueden ser comprimidos, cifrados y empaquetados, y a cada uno se le asignará un código de autenticación de mensaje.

El protocolo TLS actual es TLSv1.3, que se encuentra en fase de desarrollo desde 2016. Sin embargo, el protocolo considerado como estándar es el TLSv1.2. Lo destacable de esta versión es que incluye autenticación, generación de claves, nuevos algoritmos de cifrado, entre otras características.

En resumen, HTTPS utiliza la criptografía para proteger la información transmitida entre un navegador y un servidor web, garantizando que los datos sean seguros y que la identidad del servidor sea auténtica. Esto es fundamental para proteger la privacidad y la seguridad de los usuarios en línea como se representa en la siguiente tabla (véase tabla 3).

Tabla 3 Funcionamiento HTTPS



2.3 Aplicaciones web

Las primeras generaciones que se crearon como aplicaciones web fueron la base para su desarrollo futuro. Principalmente, se conocían como sitios web en los cuales las tecnologías implementadas en esos años (1992) no estaban muy avanzadas debido a los componentes limitados de esa época, como el ancho de banda limitado, los monitores monocromos y la memoria RAM limitada en las CPUs, por mencionar algunos.

Actualmente, las aplicaciones web se derivan de esas primeras generaciones (páginas de consulta de información), ya que solo se podían realizar consultas a páginas web e interacciones mínimas con el fin de encontrar información, documentos, etcétera. Se define una aplicación web como aquella herramienta codificada que es soportada por un navegador web y capaz de ejecutarse, lo que permite una independencia en cuanto a distribución, compatibilidad y actualización, estando estrictamente restringida a que pueda ejecutarse.

El desarrollo de las aplicaciones web tuvo sus inicios en 1995, cuando Rasmus Lerdorf creó PHP, que se convirtió en la base principal. A principios de este desarrollo, algunas compañías que hoy se conocen como Google, Wikipedia y Facebook, fueron desarrolladas utilizando este lenguaje de programación.

La innovación en la creación de aplicaciones web llevó a diferentes organizaciones a pasar de páginas web de consulta a desarrollar una interacción entre la interfaz de usuario y el navegador, lo que permitió recibir y enviar información específica. En este lapso de tiempo, Netscape anunció su nueva tecnología llamada JavaScript, lo que permitió a los programadores introducir contenido más dinámico, donde se integren peticiones con más contenido o cabeceras y el envío de parámetros. Se logró que fuera diferente a solo un texto estático o con valor definido, dando la opción de que este fuera ingresado por el usuario y enviado al servidor, teniendo una interacción directa entre el usuario y el navegador web.

Debido a que una aplicación web es un conjunto de páginas web, ya sean estáticas o dinámicas, la información que es solicitada por el usuario y que no puede cambiar de ninguna manera se le conoce como una web estática. En este caso, la comunicación del servidor web con el usuario tiene una respuesta fija de lo que se solicitó y se extrae tal como se encuentra almacenada.

Por otro lado, aquello en el servidor que se modifica o se busca utilizando identificadores o valores condicionales antes de enviar una respuesta al usuario se le conoce como páginas web dinámicas. En este caso, la respuesta del servidor se genera en tiempo real y puede variar dependiendo de los datos o parámetros que se ingresen en la solicitud del usuario.

Para conocer un poco más allá lo que destaca entre las tres generaciones de sitios web se establece en el siguiente cuadro comparativo (véase tabla 4):

Tabla 4 Generaciones de sitios web

Generación de Sitios Web			
Primera Generación	**Segunda Generación**	**Tercera Generación**	**Cuarta Generación**
• Se basaban solo en texto y ningún archivo multimedia. • Se orientaban más a un contenido educativo o científico. • Páginas con contenido ampliado. • Se integraba solo texto, para dar un tiempo de carga rápida. • Se integra la tecnología CGI. 	• Se insertan iconos en lugar de títulos. • Se integra el color de fondo. • Se crean los banners. • Se integraban más imágenes o colores gráficos, haciendo el tiempo de carga lento. • Se integran más contenidos como tablas, listas, etcétera. 	• El contenido se logra modificar por CSS. • Carga de contenido rápido debido a que se logró optimizar la conexión HTML con CSS. • Se inicia la estructura de una sola pantalla, sin hacer “scroll”. • Se integran nuevas tecnologías JSP, ASP, IDC. 	• Se integran más recursos gráficos (videos, imágenes, etcétera). • Evoluciona HTML y se crea DHTML • Se emplean contenidos interactivos para el usuario 

2.3.1 Modelos de aplicaciones web

Las aplicaciones web se basan principalmente en la arquitectura cliente-servidor, mezclando una ejecución de un script del lado del servidor, creado en un lenguaje de programación conocido como "back-end", para poder procesar la petición y realizar una gestión de almacenamiento y comunicación con las funciones recibidas al servidor, para luego ser mostrado del lado cliente, escrito en un lenguaje de programación conocido como "front-end", permitiendo que el usuario pueda realizar acciones con formularios, sistemas de gestión interactiva, bases de datos, entre otros. Algunos ejemplos de aplicaciones web son:[30]:

- Aplicaciones web estáticas: Son aplicaciones que muestran poca información, orientada a mostrar contenidos o información básica. Se caracterizan por ser desarrolladas solamente en HTML y CSS, combinada con JavaScript en algunas ocasiones. Presentan contenido digital con movimiento (videos, banners, audio, GIF), pero solo muestran información concisa o permanente. La interacción suele ser nula entre el usuario y la aplicación, y la actualización suele ser manual, ya que se modifica directamente el código fuente en producción. El procesamiento de una página estática podría explicarse de la siguiente manera (véase figura 4):

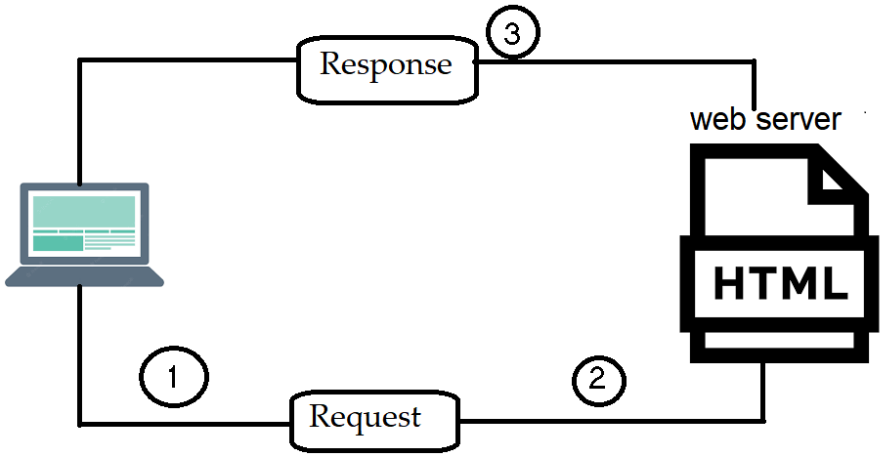


Figura 4 Esquema página estática

[30] (Maluenda, 2020)

- Aplicaciones web dinámicas: A nivel técnico, suele ser más compleja la construcción de estas, debido a que se utiliza información cargada en una base de datos y los contenidos se actualizan conforme se utiliza. Se integra un gestor de administración para que se pueda modificar el contenido y con eso se agiliza un poco la actualización de ella. En el desarrollo de estas aplicaciones, existen numerosos lenguajes de programación "back-end"; los más comunes son PHP o ASP, y sus funcionalidades son actualizables dependiendo de la interacción y del usuario destino. El procesamiento de una página dinámica podría explicarse en la siguiente imagen (véase figura 5).

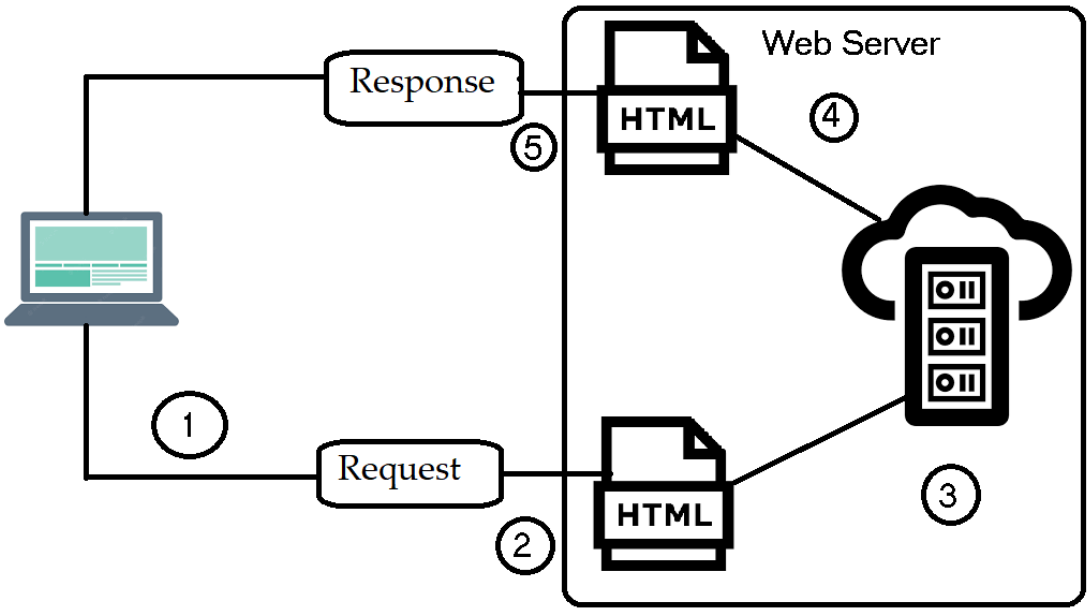


Figura 5 Esquema página dinámica

- Aplicaciones web con gestor de contenidos: Este tipo de aplicaciones suelen ser para administración, donde se busca añadir contenidos, realizar cambios y actualizaciones constantes de una forma sencilla. La construcción de estas aplicaciones es más sencilla, ya que se pueden tener bloques definidos o la opción de ingresar tu propio código HTML y CSS. Algunos ejemplos de aplicaciones web de este tipo suelen ser:
    - WordPress
    - Drupal

    - Joomla
- E-Commerce: Aplicaciones orientadas a tiendas en línea. Su desarrollo incluye complejidad, ya que se debe incluir un manejo de información sensible (direcciones, números de tarjetas, etcétera.). Se incluye un panel de gestión de administrador para poder agregar, eliminar, modificar y gestionar cualquier contenido, usuarios, productos, pagos y mantener actualizaciones de stock en todo momento.
- Portal Web App: Al usar el término "portal", se hace referencia a que la aplicación estará orientada a la interacción entre usuarios y permitirá el acceso a diferentes apartados que puedan mostrar diversas categorías, como foros, chats, correo, sin tener un contenido único establecido.

2.4 Tecnologías de aplicaciones web

Como se ha mencionado anteriormente, la comunicación se realiza entre el servidor web y la aplicación para realizar consultas e interacciones. El servicio permite que se realice la conexión desde cualquier punto como cliente. La aplicación web consta de tres componentes principales:

1. Servidor Web
2. Conexión de red
3. Clientes

Estos componentes son primordiales, ya que son la base que orientará qué tipo de tecnologías se implementarán, dependiendo del componente, la orientación de la organización o la experiencia del desarrollador. La construcción de una aplicación web se puede entender mejor a través de un ejemplo de la vida cotidiana. Por ejemplo, si se tiene una casa, la pintura y el color de las puertas y paredes serían la parte del cliente, es decir, lo que se ve externamente. La parte de la conexión de red serían el cableado eléctrico, y el servidor web sería la parte de las varillas, cimentación y estructura, ya que sería el soporte total de la construcción. En conjunto, estos componentes forman la estructura y definición de una casa, al igual que una conexión de una aplicación web.

Así funcionan las aplicaciones web y sitios web, donde se implementan bases de datos y servicios web para su gestión y almacenamiento de datos. La conexión de red se refiere a los protocolos de red, donde se definirán las reglas y algoritmos para la comunicación. Por último, el cliente sería la parte frontal de la aplicación, donde se ordenaría su apariencia, qué colores se usarían, si tendrá imágenes, contenido audiovisual o música, etcétera. Para la construcción de estos tres componentes, se pueden utilizar diferentes tecnologías, entre las que se destacan, por ejemplo:

- JavaScript: Este es considerado como un lenguaje de programación multiplataforma que permite una interacción dinámica con transiciones, movilidad automática o efectos cinemáticos entre las propiedades de los objetos, formularios, elementos, cookies, entre otros.

- HTML: En ocasiones es considerado como un lenguaje de marcado, pero es la base de la creación de sitios web, donde su principal función es la construcción de los sitios web. El proceso de evolución ha llevado al más actual, HTML5, el cual maneja la estructura mediante bloques y etiquetas.

- CSS: Es un lenguaje que tiene como objetivo darle un aspecto gráfico al entorno, ofreciendo muchas funciones gráficas para la construcción y edición de las estructuras y etiquetas HTML. Funciona para definir el aspecto visual de la aplicación web.
- PHP: Su principal función es la comunicación que realizará el sitio web con el servidor, facilitando la creación de contenido dinámico y permitiendo trabajar con bases de datos de la aplicación y su inserción de datos capturados desde la parte frontal de la aplicación.

- Python: Es un lenguaje de programación que, debido a la amplia gama de librerías gratuitas disponibles, puede ser utilizado junto con otro lenguaje de programación. Esto permite implementarlo tanto en el lado del cliente como en el lado del servidor.

- Java EE: “Java Enterprise Edition” es una tecnología para el lado del servidor que se define por la programación orientada a objetos. Se utiliza para el desarrollo de aplicaciones empresariales a gran escala, permitiendo la implementación de recursos de gran tamaño de manera eficaz y eficiente.

- MySQL: Conocido como un lenguaje de base de datos relacional, es uno de los sistemas gestores de base de datos más usados por los programadores. Es de código abierto, basado en multiplataforma y desarrollado en C. Permite el almacenamiento y organización de información en tablas múltiples, lo que facilita la extracción y recepción de datos.

- MongoDB: Una tecnología reciente que se implementa como una base de datos NoSQL, es de código abierto, lo que permite que su conexión y desarrollo sean rápidos e iterativos, facilitando su escalabilidad junto con la disponibilidad. Su orientación es más hacia documentos y fue desarrollado en C++.

Con la evolución tecnológica, se ha producido la adaptabilidad de lenguajes que anteriormente solo estaban orientados a aplicaciones nativas o ejecución en un entorno cerrado. Estos lenguajes se han unido a la creación de aplicaciones web o sitios web, permitiendo la creación de nuevos formatos de programación orientados a la parte del servidor web y su comunicación. Esto ha hecho más eficiente la comunicación. Además, existen lenguajes híbridos entre sistemas, como NodeJS, React, Flutter, etcétera.

Capítulo lll
Pruebas de penetración en aplicaciones web con el marco de clasificación OWASP

3.1 Marco de clasificación OWASP

Por sus siglas en inglés, "Open Web Application Security Project," traducido al español como "Proyecto Abierto de Seguridad de Aplicaciones Web," tiene como objetivo principal que la comunidad logre entender las causas y orígenes que pueden hacer que un sistema sea inseguro. Los integrantes del proyecto provienen de grandes empresas, organizaciones educativas y particulares de todo el mundo. Son los responsables de crear documentos, artículos, metodologías, herramientas, etcétera, con el fin de que puedan ser utilizados por cualquier persona como base o guía para fortalecer la seguridad de sus sistemas, y finalmente se aprovecha el proyecto para fomentar el desarrollo seguro. Algunos de sus proyectos más destacados son la guía OWASP de pruebas y el marco de clasificación OWASP Top 10.

El marco de clasificación es un documento de seguridad y uno de los más importantes en el ámbito tecnológico, ya que proporciona una lista pública de vulnerabilidades comunes. Se actualiza regularmente para compartir los riesgos actuales y mantenerse a la vanguardia de cada nueva vulnerabilidad o técnica de mitigación. En él se hacen referencias a escenarios de conocidos o conceptos que pueden brindar un ejemplo de los riesgos relacionados, así como clasificaciones y recomendaciones proporcionadas por diferentes organizaciones (SANS, MITRE, PCI, etcétera).

Adentrándose en el marco de autoevaluación OWASP Top 10, este es conocido por ser un documento de alto nivel que proporciona los puntos clave para construir un informe completo. Ofrece información sobre los 10 riesgos más comunes en el mundo tecnológico, que afectan a la mayoría de los sistemas en el mundo. Esto permite identificar riesgos desde los más críticos hasta los informativos, ofreciendo un amplio margen del tipo de afectaciones a un sistema. Este marco de clasificación es un documento de apoyo para buscar referencias de un hallazgo si se encuentra en el listado, debido a que engloba una categoría general en la cual se desglosan vulnerabilidades asociadas a ella. Se busca concienciar a todas las empresas sobre la importancia de realizar procesos de mitigación y lograr la minimización de los riesgos de seguridad mediante un desarrollo seguro.

OWASP Top 10 fue lanzado por primera vez en el año 2003 y ha recibido actualizaciones constantes, reflejando sus cambios en el siguiente listado (véase tabla 5):[31]:

Tabla 5 Comparativo OWASP TOP 10

Lanzamientos							
OWASP TOP TEN ENTRADAS	**2003**	**2004**	**2007**	**2010**	**2013**	**2017**	**2021**
Entrada no validada	✓	✓	✗	✗	✗	✗	✗
Desbordamiento de Buffer	✓	✓	✗	✗	✗	✗	✗
Negación de Servicio	✗	✓	✗	✗	✗	✗	✗
Inyección	✓	✓	✓	✓	✓	✓	✓
Cross Site scripting (XSS)	✓	✓	✓	✓	✓	✓	✓
Referencia insegura de objetos referenciados	✗	✓	✓	✓	✓	✓	✓
Cross Site Request Forgery (CSRF)	✗	✗	✓	✓	✓	✓	✓
Configuración de seguridad incorrecta	✓	✓	✗	✓	✓	✓	✓
Pérdida de autenticación y gestión de sesiones	✓	✓	✓	✓	✓	✓	✓
Ausencia de Control de Acceso	✓	✓	✓	✓	✓	✓	✓
Redirecciones y reenvíos no validados	✗	✗	✗	✓	✓	✓	✗
Revelación de información y gestión incorrecta de errores	✓	✓	✓	✗	✗	✗	✗
Ejecución de ficheros malintencionados	✗	✗	✓	✓	✗	✗	✗
Exposición de datos sensibles	✓	✓	✓	✓	✓	✓	✓
Comunicaciones inseguras	✗	✓	✓	✓	✓	✗	✗
Uso de componentes con vulnerabilidades conocidas	✗	✗	✗	✗	✓	✓	✓

[31] (OWASP TOP TEN, 2003)

3.2 Vulnerabilidades en aplicaciones web

Al encontrar una vulnerabilidad conocida, su impacto y riesgo suelen variar según el tipo de organización y otros factores relacionados con el entorno. Este marco de evaluación permite medir algo intangible y ajustarse de acuerdo con los factores involucrados en el escenario. Por ello, se enfocará en el OWASP Top 10 2017, ya que a pesar de que existe el OWASP Top 10 2021 (lanzado el 24 de septiembre de 2021), este último sigue siendo objeto de modificaciones constantes. A continuación, las siguientes tablas muestran una explicación de los elementos del OWASP Top 10 2017[32] que permitirán categorizar los hallazgos, de acuerdo a su descripción y características que pueden presentar las vulnerabilidades y así poderlas relacionar al marco de clasificación de OWASP.

Tabla 6 OWASP TOP 10 A1 Inyección

A1 Inyección
Las fallas de inyección suceden una vez que se envían datos y son ejecutados por un intérprete. El código inyectado suele anexarse a la información enviada en una línea ya existente por la aplicación, anexando una consulta o acción para realizarla arbitrariamente. El factor importante es que cualquier punto de entrada puede ser considerado como vector de ataque, donde se cumpla que puedan enviarse cadenas de texto del lado del cliente y sean interpretadas del lado del servidor. La información enviada suele ser dañina o tener alguna finalidad que busca engañar al intérprete, de forma que este ejecute las cadenas de caracteres añadidas involuntariamente, obteniendo así una respuesta positiva con información respecto a lo enviado.

[32] (OWASP, wiki.owasp.org, 2017)

Características • La aplicación es vulnerable si los datos de entrada no son validados, filtrados o sanitizados por la aplicación o el servidor. • Se realizan consultas dinámicas o no parametrizadas sin codificar los datos de entrada de acuerdo con el contexto. • Se realizan consultas dinámicas o no parametrizadas sin codificar los parámetros de acuerdo con el contexto.
Ejemplo En este caso es posible modificar el parámetro ID agregando una consulta lógica modificando su respuesta al cumplirse la regla de lógica. http://site.com/cuentas/cuentaedit?id=15'or '1'='1.
Remediación Genérica • Usar APIs con un intérprete con interfaces seguras. • Validar la entrada de datos en el servidor con "Listas blancas". • Realizar escape de caracteres especiales de acuerdo con el intérprete. • Utilizar controles de limitación de caracteres de acuerdo con el intérprete.

Tabla 7 OWASP TOP 10 A2 Pérdida de autenticación

A2 Pérdida de autenticación
En este caso, el objetivo es obtener acceso a usuarios registrados o acceso a la aplicación sin validación de tener un usuario. Se suele relacionar este hallazgo con la gestión de sesión, en la cual se implementa de forma insegura o no se validan elementos como los tokens de sesión o las cookies.
Características • Se pueden realizar ataques de fuerza bruta usando diccionarios expuestos por una comunidad o filtración de información. • Permite el uso de contraseñas consideradas como débiles o comunes. • No válida o expira correctamente la cookie de sesión. • Expone ID de sesión en URL.

Ejemplo http:/site.com/usuarioadmin/tokenid?=123F20RT487
Remediación Genérica • Implementar autenticación multifactor de acuerdo con las buenas prácticas de la industria. • Usar controles que no permitan la asignación de contraseñas débiles. • Usar mensajes genéricos que no permitan conocer si el usuario está registrado. • Usar un gestor de sesión del lado del servidor, seguro y que sea de forma aleatoria y tenga expiración.

Tabla 8 OWASP TOP 10 A3 Exposición de datos sensibles

A3 Exposición de datos sensibles
Se ataca a la criptografía o almacenamiento de la información, y esto puede relacionarse con el robo de claves, ataques "man-in-the-middle" o exposición de datos sensibles. Es claro que en estos casos los archivos sensibles no se protegen de forma adecuada, lo que puede afectar sectores como el financiero, de salud, información personal identificable, etcétera.
Características • Se transmiten los datos en texto claro, mediante protocolos débiles. • Se usan algoritmos criptográficos obsoletos o débiles. • Expone información del sistema o usuarios del lado del cliente.
Ejemplo Un sitio web no forza el uso de la navegación mediante HTTPS el cual permite el uso de HTTP haciendo el tráfico vulnerable a “man in the middle”.
Remediación Genérica • Usar protocolos y algoritmos seguros de comunicación de acuerdo con la industria. • No almacenar información sensible del lado del cliente. • Transmitir la información con un algoritmo de verificación de integridad.

Tabla 9 OWASP TOP 10 A4 Entidades externas XML (XXE)

A4 Entidades externas XML (XXE)
Un ataque XML es aquel en el que la aplicación es capaz de analizar y procesar las entradas de datos en este formato. Sucede cuando una entrada contiene datos referenciados hacia entidades externas (recursos fuera de dominio) y es procesado por un "parser" XML configurado de manera insegura o incorrecta.
Características • La aplicación acepta directamente XML, lo carga e inserta datos no confiables en XML. • La aplicación utiliza SOAP o Web Services en versiones inseguras. • Se logran ataques XXE podría escalar el ataque a una denegación de servicio.
Ejemplo Un atacante intenta ejecución de código remoto por XML <?xml versión=”1.0” encoding=”UTF”?> <!DOCTYPE foo[<!ELEMENT foo ANY> <!ENTITY xxe SYSTEM “file:///etc/passwd”>]> <foo>&xxe;</foo>
Remediación Genérica • Usar formatos no complejos como son JSON. • Actualizar procesadores o dependencias a su última versión. • Implementar una lista blanca, filtrado y sanitización para prevenir inserción de datos maliciosos.

Tabla 10 OWASP TOP 10 A5 Pérdida de control de acceso

A5 Pérdida de control de acceso
El control de acceso de usuarios es esencial para prevenir diferentes vulnerabilidades relacionadas con usuarios y el acceso a la aplicación. En caso de que estén presentes,

suelen provocarse por malas configuraciones, falta de validación de roles o acceso no autorizado a archivos sensibles y la escalación de privilegios.
Características • Evadir los controles de acceso por modificaciones de ID en el estado interno de la aplicación. • Manipulación de token, cookies o un campo oculto que habilite funcionalidades de un rol mayor. • Configuración incorrecta de CORS permitiendo acceso autorizado a API.
Ejemplo Las URL son similares ante un comportamiento identificado por palabras clave http://site.com/app/getinfo http://site.com/app/admin_getinfo
Remediación Genérica • Negar acceso de forma predeterminada a cualquier recurso no disponible para ese rol. • Limitar el acceso a recursos externos como APIs. • Configurar los token JWT con tiempo de expiración.

Tabla 11 OWASP TOP 10 A6 Configuración de seguridad incorrecta

A6 Configuración de seguridad incorrecta
La configuración por defecto es considerada como insegura, ya que en ocasiones es posible acceder a recursos administrativos, páginas por defecto, archivos o directorios no protegidos, lo que puede brindar información sobre la organización o el comportamiento de la aplicación.
Características • Se encuentran características habilitadas que no son necesarias para la aplicación. • El manejo inseguro de error por defecto o trazas de la aplicación revela a los usuarios demasiada información.

• Configuración del servidor web se encuentran configurados como defecto.
Ejemplo El servidor web tiene habilitado la página de administración http://102.142.22.94/admin.php
Remediación Genérica • Deshabilitar las páginas por defecto que no sean necesarias. • Restringir el acceso a páginas de administración o importantes para su uso. • Cambiar las credenciales de las cuentas predeterminadas.

Tabla 12 OWASP TOP 10 A7 Secuencias de comandos entre sitios (XSS)

A7 Secuencias de comandos entre sitios (XSS)
La aplicación se desarrolla en tecnologías web, en las cuales su principal intérprete suele ser HTML y JavaScript. Mediante la inserción de código en una entrada no sanitizada, podría permitir la ejecución de acciones maliciosas.
Características • Reflejado: La aplicación interpreta los datos sin validar de manera correcta haciendo inserción de código HTML o JavaScript en el código original. • Almacenado: La aplicación almacena los datos de entrada sin ser validados el cual permite que sean visualizados por otro usuario. • DOM: La aplicación incluye datos de forma dinámica en las cuales se manejan por medio de APIs para el proceso de datos y control, permitiendo agregar código malicioso.
Ejemplo La aplicación utiliza los datos no confiables para la construcción del HTML <input name='name' type='TEXT' value="user1">" value="><script>alert("Inserción")</script>" el cual permitirá la inserción de datos al código HTML
Remediación Genérica • Utilizar "frameworks" seguros por diseño y reglas de codificación de contenido.

• Codificar los datos insertables en HTML o JavaScript. • Habilitar políticas de seguridad de contenido.

Tabla 13 OWASP TOP 10 A8 Deserialización Insegura

A8 Deserialización Insegura
La aplicación permite que los ataques logren deserializar una estructura de datos y objetos, lo que permite al atacante modificar su lógica, datos o roles escritos en dicha estructura, permitiendo así acciones maliciosas ante ella.
Características • Se utiliza la serialización de forma sencilla permitiendo que cualquier herramienta la pueda convertir en texto plano. • Manipulación de datos para el control de acceso mediante una estructura de datos. • El envío de la estructura de datos se realiza con un cifrado básico o lógica predecible.
Ejemplo Una aplicación web usa objetos para almacenar token de sesión y roles. a:00(i:2;l:212;l:1;s:8:”Admin”;i:2;l:212;l:1;s:8”;)
Remediación Genérica • Registrar los fallos y excepciones en la deserialización. • Aislar el código de modo que se ejecute en un ambiente con pocos privilegios. • Verificar la integridad del objeto recibido con firmas digitales.

Tabla 14 OWASP TOP 10 A9 Uso de componentes con vulnerabilidades conocidas

A9 Uso de componentes con vulnerabilidades conocidas
Las aplicaciones suelen ser programadas de modo que, al ser desarrolladas, pueden no implementar las tecnologías actualizadas o que son consideradas obsoletas, lo que lleva a que ciertos componentes puedan tener vulnerabilidades registradas como críticas y cuenten con un exploit público.

Características • El software es vulnerable, no posee soporte o se encuentra desactualizado. • No se han aplicado los parches de seguridad implementados por el proveedor. • No se analizan los componentes periódicamente.
Ejemplo El servidor web utilizado en la aplicación web es un Apache con vulnerabilidades conocidas y adicionalmente no tiene soporte por parte del proveedor.
Remediación Genérica • Monitorear componentes continuamente con bases de datos de vulnerabilidades. • Obtener componentes de fuentes oficiales para instalar la última versión. • Remover dependencias, componentes desactualizados no necesarios.

Tabla 15 OWASP TOP 10 A10 Registro y Monitoreo Insuficientes

A10 Registro y Monitoreo Insuficientes
Los controles de respuesta a incidentes y una configuración oportuna no son suficientes mecanismos de seguridad, ya que puede existir que estén de forma incorrecta o mal configurados, lo que permite al atacante realizar acciones maliciosas sin ser detectado a tiempo.
Características • Registros de aplicación o APIs no son monitoreados para detección de actividades sospechosas. • La aplicación no logra detectar o alertar ataques en tiempo real. • Advertencias y errores en registros sin trazas o con información nula.
Ejemplo El registro de la aplicación tiene logs con información casi nula en la cual no es posible identificar si es un comportamiento natural o malicioso
Remediación Genérica

- Asegurar que los inicios de sesiones, controles de acceso y validación de entrada en el servidor, se logren registrar para identificar acciones sospechosas.
- Asignar un ID a los registros que permitan trazar la actividad sospechosa a un usuario.
- Establecer monitorización y alertas efectivas que permitan la respuesta oportuna.

En resumen, las tablas previamente presentadas proporcionan una visión resumida de los factores a tener en cuenta, ejemplos y posibles recomendaciones relacionadas con vulnerabilidades en los controles o mecanismos de seguridad que están configurados de forma insegura o por defecto. Estas vulnerabilidades pueden dar lugar a debilidades y representan un riesgo importante para la seguridad.

3.3 Factores de clasificación de vulnerabilidades

Las vulnerabilidades que se encuentran se pueden clasificar de diferentes maneras u obtener un riesgo similar dependiendo del comportamiento. Algunos factores a considerar para determinar el impacto son los ámbitos de la organización, ya que en ciertos casos pueden ser considerados altos y en otros bajos para el mismo hallazgo en otra aplicación, dependiendo del tipo de sitio web y su alcance. Además, al encontrar una vulnerabilidad, se buscará si en otras secciones se puede obtener un comportamiento similar para tener en cuenta la cantidad de veces que fue encontrada. Se debe considerar el flujo natural y compararlo con ejemplos globales. Existen diversos factores para su clasificación, por ejemplo, la probabilidad que exista la vulnerabilidad y la profundidad del impacto que se obtendría. Tomando en cuenta estos factores, se crea su clasificación, la cual se compone de los siguientes puntos:

1. Agentes de amenaza

   Son individuos o conjuntos de individuos que tienen la capacidad de afectar la seguridad a nivel externo con fines comunes, políticos, personales, etcétera. Este

factor suele ser variado debido a que existen muchas personas con la intención de querer acceder a los datos o servicios.

2. Específico de la organización y aplicación
   El factor principal que ayudará a definir la clasificación será especificar el tipo de dependencia o aplicación a la que está orientado principalmente, ya que dependiendo del sector en el que esté enfocado, será el riesgo. La información almacenada no se trata de la misma forma, por ejemplo, entre un banco y un supermercado.
3. Vectores de ataque
   Son los puntos de entrada que tienen las organizaciones ante posibles atacantes, en los cuales se tendrá que pensar como uno de ellos, buscando dónde se encuentran las formas de extraer información o que permitan al menos realizar algún envío de peticiones e interacción con el objetivo. Por ejemplo, los puntos de inserción de datos enviados por el usuario, la modificación de parámetros, la lógica de negocio y muchos factores asociados a los mecanismos de seguridad son los que pueden resultar en una vulnerabilidad.
4. Explotabilidad
   Al encontrar los puntos de entrada en los cuales se pueden realizar los envíos de carga útil con fines maliciosos, no quiere decir que cierta inserción de carga útil pueda funcionar, ya que depende de la aplicación y su nivel de seguridad implementada ante estos intentos.
5. Debilidades de seguridad

   El desarrollo de una aplicación puede hacer que se limiten a las mejores prácticas de la industria para lograr un desarrollo seguro. Sin embargo, en ocasiones los desarrolladores hacen uso de tecnologías y servidores web que contienen vulnerabilidades conocidas con explotación de estas, lo cual hace completamente inseguro el entorno. Por ello, es importante tener la última versión de cualquier componente tecnológico y cambiar las configuraciones por defecto por una personalizada.
6. Detección de vulnerabilidad

Todo depende de la información que se puede observar a simple vista con una interacción pasiva o activa con la aplicación. En ocasiones, algunas cabeceras de respuesta muestran información sobre las tecnologías y versiones que se utilizan en ella, lo que permitirá conocer o crear una carga útil que se pueda usar para aprovechar dicha vulnerabilidad.

7. Prevalencia de vulnerabilidad

   Una vez que se ha logrado explotar la vulnerabilidad, se busca mantener disponible dicha vulnerabilidad para usarla como pivote. En ocasiones, se guían por las vulnerabilidades públicas que se pueden aprovechar en el objetivo, así obteniendo una persistencia en caso de lograr que se interprete nuestra carga útil.

8. Controles de seguridad

   Son los mecanismos de seguridad que tiene la aplicación ante un comportamiento anormal. Si se logró detectar a tiempo la recepción de una carga útil del lado del servidor, dependerá de qué tan protegidos estén y qué reglas existan. Sin embargo, si sus controles son muy débiles, permitirá conocer qué controles de seguridad se tienen establecidos. Finalmente, al identificarse, el atacante intentará evadir dichos controles y el peor escenario sería que el impacto aumente debido a que, a pesar de los controles establecidos, se permita la ejecución de la carga útil.

9. Impacto técnico

   Es la forma de visualizar el ataque en un punto donde se tomará en cuenta qué podrá afectar, ya que depende de la explotabilidad, el nivel de conocimiento técnico y los controles de seguridad que existan. Básicamente, es el factor que medirá el impacto generado de acuerdo con la CIA (Confidencialidad, Integridad y Disponibilidad).

10. Impacto al negocio

    Después del impacto técnico, se continúa clasificando el impacto en la producción. Se define la afectación frente a la vulnerabilidad con relación a la orientación de la organización y si se afectó la estructura, los datos, la disponibilidad, entre otros. En esta fase, se piensa en el futuro, considerando que, en caso de llevar el peor escenario a sus extremos, se analizará el resultado que se obtiene y si se puee

disponer de la información obtenida. Cada fase ayudará a clasificar el hallazgo encontrado, lo que finalmente permitirá obtener un resultado asociado al escenario de la vulnerabilidad y así poder asignar un riesgo final.

3.4 Clasificación de vulnerabilidades

Una vez que se haya identificado qué vulnerabilidad se ha encontrado en la aplicación y conociendo los factores que se involucran, ayudará a poder identificar y diferenciar cada vulnerabilidad, lo que permitirá tener una medición dentro de ella. La asignación de cada fase ayudará a obtener el impacto que puede provocarse; en caso de explotar con éxito, se asignará una severidad de acuerdo con un número y rango que ayudará a ordenar de la más crítica a la más baja, logrando así tener un proceso de mitigación ordenado. Para ello, diferentes organizaciones proporcionan calculadoras de riesgo; se enfocan en el "OWASP Risk Rating Calculator", el cual ayudará a tener una clasificación de las vulnerabilidades y contiene los siguientes elementos[33]:

- Factores de agentes de amenaza
- Factores de vulnerabilidad
- Factores de Impacto Técnico
- Factores de Impacto al Negocio

Adicionalmente, una vez identificada la vulnerabilidad con sus factores asociados, se obtendrá un resultado que servirá como base de clasificación, el cual se podrá usar para determinar la categoría de acuerdo con el escenario y mediante los factores obtenidos de acuerdo a los escenarios asociados al objetivo se podrán obtener resultados que ayuden a asignar un riesgo a nuestras vulnerabilidades. Un ejemplo del uso se muestra en la imagen (véase figura 6).[34]

[33] (Project, OWASP Risk Rating Calculator, 2018)
[34] (Project, OWASP Risk Rating Calculator, 2018)

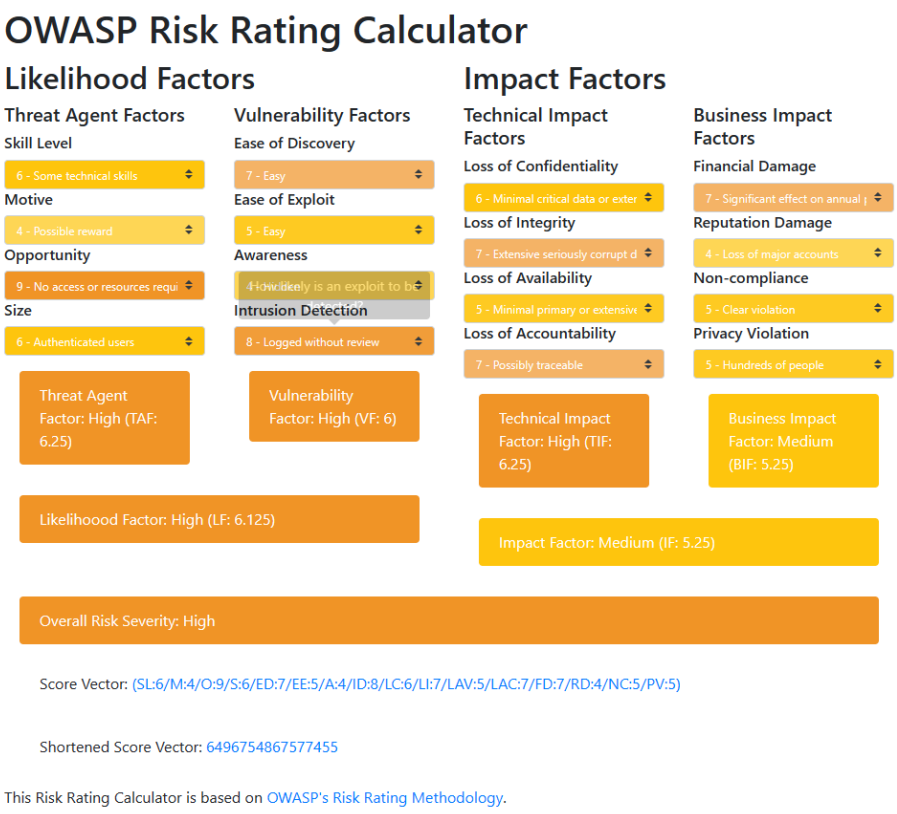


Figura 6 OWASP Calculator

3.5 Referencias de vulnerabilidades encontradas

CVE, por sus siglas en inglés (Common Vulnerabilities and Exposures), es un proyecto definido y mantenido por MITRE. En realidad, el nombre originario es "MITRE CVE List", el cual se encarga de mantener actualizada dicha documentación. Es producto de la unión con el gobierno de Estados Unidos y presenta una lista de vulnerabilidades comunes y su exposición en las versiones de software afectadas. Se pretende brindar información sobre estas, en las que se pueden identificar tecnologías, componentes o dependencias propensas a ataques con exploits conocidos. A su vez, se anexa una posible recomendación o mitigación que suele relacionarse con la actualización de la versión del componente afectado.

Las vulnerabilidades publicadas suelen tener referencias asociadas a publicaciones en foros o blogs, donde se demuestra la explotación o pruebas de concepto. Dicha lista contiene un identificador que se organiza de acuerdo al año. Se presenta de la siguiente

forma, iniciando por las siglas CVE, seguido según el año en el que fue descubierto y seguido por un identificador numérico, siguiendo la siguiente nomenclatura (véase formula 2).

$$CVE - YYYY - NNNN \rightarrow CVE - 2022 - 2576$$

Formula 2 Nomenclatura CVE

Por otro lado, se encuentra el CWE, por sus siglas en inglés (Common Weakness Enumeration), el cual se conoce como una base de datos con categorías de debilidades y vulnerabilidades asociadas a ellas. Este proyecto es comunitario, donde se comparte con la comunidad y tiene como objetivo principal comprender las fallas en el software, recomendar herramientas que ayuden a la comunidad a identificar, corregir y prevenir dichas fallas. Este programa se clasifica en más de 600 categorías para la corrección y desarrollo seguro del software, y está asociado a la CVE debido a que es de la misma organización con la cual se definió, patrocinó y mantuvo en marcha el proyecto MITRE. Fue una forma preliminar de entender qué puede provocar un error y entenderlo profundamente a un nivel genérico, sin importar en qué lenguaje de programación se haya basado. De igual forma, su representación para identificar la debilidad es sus siglas CWE seguido por un identificador único asignado, el cual ayudará a obtener la información correcta en su base de datos. La nomenclatura es la siguiente (véase formula 3):

$$CWE - NNNN \rightarrow CWE - 7870$$

Formula 3 Nomenclatura CWE

NIST NVD, por sus siglas en inglés "NIST National Vulnerability Database", tiene el mismo objetivo que CVE: ayudar a identificar las vulnerabilidades, pero se basa más en la clasificación y asignación de riesgo, lo que permite clasificar qué riesgo puede ocurrir relacionado con la calculadora de riesgo CVSS, la cual ayuda a conocer, a nivel técnico, qué habilidad se ocuparía para explotar dicha vulnerabilidad. Igualmente, viene referenciado en el listado CVE y CWE para un trabajo en conjunto, en el cual se anuncian soluciones y herramientas para esta vulnerabilidad publicadas en distintos foros. Para su identificación, tiene el mismo formato que CVE, iniciando por las siglas CVE, seguido por

el año en que se encontró la vulnerabilidad y, por último, un número asignado para la identificación. Los identificadores son los mismos en la base de datos CVE y NIST NVD, teniendo el siguiente formato (véase formula 4):

$$CVE - YYYY - NNNN \rightarrow CVE - 2022 - 2562$$

Formula 4 Nomenclatura NIST

Exploit DB, por sus siglas en inglés "Exploit Database", es una base de datos de exploits donde se recopilan los posibles programas utilizados por los hackers con el objetivo de poder aprovechar las vulnerabilidades y lograr identificar o servir como base para la construcción de un ataque. Gracias a los listados anteriores, se puede conocer qué software se ve afectado, si es posible explotarlo y qué puede llegar a afectar, además de proporcionar información más detallada a nivel técnico para su explotación. En ocasiones, este listado contiene bloques de código para la ejecución y explotación. Esta base de datos es más amplia, ya que permite la clasificación por plataforma, software, etcétera. Adicionalmente, permite realizar consultas por medio del listado CVE.

Capítulo lV
Metodología de pruebas de penetración NIST SP 800 115

4.1 Determinación de propósito, objetivo, políticas y restricciones.

En lo que corresponde una evaluación de seguridad se brinda el o los objetivos a analizar (host, sistema, red, procedimiento, entre otros.) de los cuales algunos aspectos a revisar son que se cumpla los estándares, regulación de un manejo de procesos seguro y protección de datos, a su vez tener la determinación y conocimiento de una evaluación de riesgos brindando resultados que proporcionan un contexto global y entendible con relación a los activos de la organización.

NIST SP 800-115 es una guía en donde se puede revisar los aspectos técnicos de un sistema orientado a la seguridad de la información, donde se observarán las fases a realizar, clasificaciones y evaluaciones para brindar un resultado evaluando la correcta configuración de mecanismos de seguridad, midiendo el impacto que se puede provocar en los sistemas de la organización, la primera fase que se indica en la guía es:

Propósito y Objetivo

En esta parte la organización orienta y da contexto al consultor sobre el análisis de seguridad, la estancia en la que se trabajará y cuáles son sus objetivos. Es muy importante revisar con el equipo de trabajo de la organización qué técnicas son permitidas, los propósitos del análisis y el número de sistemas o enlace de referencia al que se realizará la prueba de penetración, ya que habrá políticas y restricciones que se expresarán de acuerdo al tipo de caja y funcionalidades como son: el no realizar denegaciones de servicios, sobrecarga de proceso, ingeniería social, etcétera. Debe de existir una cláusula de confidencialidad en caso de extraer información sensible, con ella se refuerza y protege a la no divulgación de información. Al igual en esta sección se establece que tipo de caja de penetración se realizará y si se brindará información adicional (credenciales de acceso, IP, enlace, archivos, entre otros.), cabe mencionar que la guía es una forma de orientación de cómo obtener una verificación de seguridad efectiva, existen factores clave que brindan un análisis exitoso y las técnicas implementadas para un resultado satisfactorio.

Políticas de evaluación

Fase importante en donde se asignan los roles y matrices de los responsables en caso de surgir un inconveniente, por ejemplo en caso de tener un equipo de respuesta a incidentes y se alerte de un posible ataque o la aplicación deje de responder acercarnos a realizar el aviso correspondiente lo antes posible, esto se realiza con la finalidad de tener un responsable al que se le hará la entrega de la documentación creada al final del análisis para así cumplir con el objetivo de entrega de reportes de resultados, de igual manera si se comparte de manera interna los reportes ya es responsabilidad del responsable a firmar de recibido que sucede con ellos dentro de la organización. Se asignan los nombres con roles de la organización, responsabilidades de cada uno y escalamiento de responsables.

Restricciones de Ejecución.

Es una forma de establecer que el análisis se realizará de manera controlada, segura y efectiva, donde el consultor asume la responsabilidad de las acciones que llevará a cabo en la ventana de ejecución, así como de las cargas útiles que se envíen a la red o aplicaciones, ya que, en ocasiones, si no se tiene cuidado en este paso, se podría afectar a la organización. En algunos casos, se define la información técnica del dispositivo a evaluar, como pueden ser tecnologías implementadas, servidores, componentes, URL, entre otros. Se mencionan los métodos de conexión a utilizar, si es necesario el uso de una conexión personalizada para alcanzar el sistema de forma local, como lo es una VPN, o si se requiere que el consultor asista de manera presencial para su evaluación.

La primera fase es importante para lograr una ejecución exitosa, ya que es necesario revisar todo lo que se va a involucrar, quiénes serán los responsables, qué estará prohibido y qué insumos se proporcionarán para su ejecución. Al definir un inicio estable sin dudas, dejará al consultor libre para llevar a cabo su ejecución y aplicar técnicas asociadas conforme a las restricciones establecidas al inicio.

4.2 Técnicas de evaluación, análisis y validación de vulnerabilidades

Una vez realizado el documento inicial, se procede con la ejecución de la prueba sobre los mecanismos de seguridad en su sistema. En este caso, depende de la definición de las técnicas que se aplicarán. Conforme se avance en la ejecución, se conocerán las acciones a realizar después de un análisis de respuesta obtenida y finalmente se observará si es explotable. La fase que se conoce como la planificación es donde se recolecta la mayor cantidad de información que pueda extraerse del objetivo. De acuerdo con esa información, se valida que realmente el sistema esté basado en los datos proporcionados y así enfocarnos en posibles técnicas que pueden ser de gran ayuda al realizar esta fase.

Descubrimiento de red

Se utilizan métodos de reconocimiento para encontrar los hosts activos en una red o, en su caso, de una aplicación web, el host activo y su IP. En el caso de las aplicaciones web, la IP puede compartirse, ya que puede albergar más de 2 aplicaciones web en una sola; el dominio es único, el cual no puede compartirse, por lo tanto, se tiene un dominio diferente por aplicación. El descubrimiento de red o host es lograr identificar cómo responde ante las solicitudes, identificar debilidades e interactuar con el sistema para lograr reconocer el comportamiento natural, cuáles son sus recursos usados y cómo procesa la información ingresada.

Existen técnicas de reconocimiento, como las pasivas y las activas, que sirven para identificar una red. La técnica pasiva consiste en la recolección de información de manera indirecta de la red o aplicación, utilizando técnicas como sniffers de red, monitoreo, registro de direcciones IP de los hosts activos, búsquedas en internet sobre información de la aplicación web, etcétera. Por otro lado, están las técnicas activas, en las cuales se realiza una interacción directa con el sistema y se envían constantemente solicitudes para conocer los hosts activos. Esta técnica es más agresiva, ya que en ocasiones se envían cargas útiles para identificar el sistema operativo y los recursos tecnológicos que se implementan.

Identificación de red y servicio

Al identificar los hosts activos que se encuentran en el sistema, se puede realizar una enumeración de la red. Esta fase se implementa para conocer cuántos puertos están en funcionamiento y los tipos de servicios que se están ejecutando en los puertos abiertos. Si es posible, se busca identificarlos con su versión y nombre. Si se utiliza un servidor web, en este caso es importante considerar que para el funcionamiento de una aplicación web se utilizan comúnmente los puertos HTTP (80) o HTTPS (443). Esta información ayuda a conocer qué tipo de servidor se encuentra implementado. En caso de que haya un servicio web en otro puerto no común y si se utiliza tecnología que complementa a la aplicación, también se puede identificar posibles vulnerabilidades o validar si tiene configuraciones por defecto.

Para llevar a cabo esta tarea, se utilizan herramientas para identificar el tipo de servicios implementados. En caso de que el sistema tenga puertos abiertos para su uso, se tomará en cuenta un análisis posterior sobre la seguridad de estos puertos. Una vez identificados los servicios en ejecución, se procede a conocer qué se está ejecutando y se envían cargas útiles para explotar posibles malas configuraciones o vulnerabilidades asociadas.

Escaneo de vulnerabilidades

Al realizar el escaneo de servicios y puertos, se implementa una técnica similar; en esta fase, se tomarán en cuenta los resultados obtenidos anteriormente para lograr identificar vulnerabilidades. Puede ser mediante la identificación de la versión del servicio o usando un escáner de vulnerabilidades. En ocasiones, solo con tener el host activo relacionado a la aplicación, permite que se pueda llevar a cabo la fase de reconocimiento con la herramienta usada. Además, en algunos casos, se podrá realizar un escaneo de vulnerabilidades identificadas o registradas en una base de datos propia de la misma. A veces, proporcionará cierta información en un reporte producido internamente, como son:

1. El host de la aplicación y las políticas de red que están implementadas.
2. Información sobre los objetivos identificados (Sistema Operativo, Puertos Abiertos).
3. Vulnerabilidades encontradas con posible exploit.

4. Información sobre la mitigación de vulnerabilidades descubiertas.

Validación de vulnerabilidades

Las vulnerabilidades obtenidas por los escáneres de vulnerabilidades no son suficientes para realizar un reporte, ya que en ocasiones la carga útil enviada es solamente para la identificación de una posible vulnerabilidad y para identificar el comportamiento de la aplicación ante esta carga útil. En posibles escenarios, se pueden confundir los hallazgos en la aplicación y marcarlos como positivos, otras veces como falsos positivos. Esto lleva a emplear técnicas adicionales para su explotación hasta obtener una respuesta positiva ante estas vulnerabilidades.

En resumen, la validación de vulnerabilidades es importante, ya que su objetivo implica un riesgo. Se requieren técnicas para ampliar su potencial y la implementación de otras técnicas para su identificación completa.

4.3 Evaluación de seguridad y ejecución de prueba de seguridad

En esta fase, se utiliza la información que se obtuvo en la anterior para poder aumentar la explotación y la probabilidad de encontrar una vulnerabilidad. En este caso, se busca demostrar que existe alguna vulnerabilidad y poder exponer su impacto, así como considerar futuros escenarios en caso de que sea explotada. Se mencionan algunos métodos a considerar:

Pruebas de penetración

Es una fase en la que se involucran las anteriores de la guía, en la cual, a partir de la recolección de Hosts, IP, interacción del sistema y análisis de vulnerabilidades, se lleva a cabo la validación de vulnerabilidades de forma manual. En esta etapa, se identifica la existencia de vulnerabilidades y, al mismo tiempo, se realiza el reconocimiento de la explotabilidad que se tiene. Según el orden a seguir, la primera fase sería la recolección de información, la segunda el descubrimiento y la tercera el ataque o prueba de penetración, en la cual ya se realiza un análisis de la seguridad del sistema y se evalúan las reglas implementadas para detectar un posible ataque. En esta fase, se incluye lo que

se puede realizar en el sistema, desde la recolección de archivos hasta el ingreso a la base de datos e identificación de contraseñas.

Descifrado de contraseñas

Es una técnica que involucra las contraseñas almacenadas que fueron encontradas en la prueba de penetración y, en caso de que se hayan encontrado en formato hash, se procedería a su descifrado. Un ejemplo podría ser, haber interceptado el tráfico de un inicio de sesión de una aplicación en la cual se observa el envío de credenciales. La contraseña va hasheada, pero si se logra descifrarla, se podría obtener acceso a un usuario e ingresar al sistema. Por lo general, se utiliza para asegurar la integridad del dato, pero dependiendo del tipo de hash utilizado y también podría asegurar la contraseña.

Fase de ataque

La aplicación de un ataque es un descubrimiento adicional de acuerdo a la fase de penetración. Se podrá realizar un reconocimiento extra si es posible acceder al sistema, buscando tener una post-explotación o un acceso persistente. En resumen, se podría decir que las fases de esta etapa son las siguientes:

- Ganar acceso: Se realizan acciones en la fase de penetración para obtener acceso al sistema.
- Escalación de privilegios: Dependiendo de las acciones que se hayan realizado y el acceso obtenido, es cómo se lleva a cabo esta fase, ya que se busca obtener control total del sistema.
- Exploración del sistema: A nivel aplicativo, se logró acceder y obtener un perfil administrativo. El siguiente paso sería realizar una exploración de la infraestructura y de los recursos implementados para su funcionamiento.
- Instalación: Se busca la persistencia del acceso o alguna bandera que permita demostrar al consultor hasta qué punto logró llegar. Esto se reportará para que el administrador sea capaz de eliminarla, en caso de ser necesario.

4.4 Actividades posteriores de la prueba de penetración

Siendo una de las fases más importantes, debido a que es la penúltima, en ella se encuentran las actividades que ayudarán a mostrarle a la organización el impacto y la información obtenida, de acuerdo a los rubros y lo que se logrará obtener. A su vez, son las bases para la construcción de los reportes de entrega. Algunos puntos por considerar son:

Carga de archivos bandera.

La carga de archivos bandera es importante si se quiere obtener un acceso persistente. En ocasiones, el sistema se encuentra mal configurado, lo que permite la carga de archivos modificados que se encuentran codificados en el mismo lenguaje base para que puedan ser interpretados y así lograr la persistencia. Esto ayuda a elevar el impacto de la vulnerabilidad que a simple vista podría verse sencilla, pero al permitir la persistencia y la ejecución de comandos remotos, permite tener un impacto alto ante la evaluación.

Descarga de archivos.

Al obtener acceso completo al sistema y realizar la escalación de privilegios, es posible que se pueda acceder a recursos restringidos para un usuario con privilegios bajos. La descarga y apertura de estos recursos podrían brindar información tecnológica, credenciales válidas, códigos fuente y archivos sensibles. Esto puede aumentar el impacto de la prueba de penetración y también permitir el acceso persistente.

Persistencia directa.

La obtención de códigos fuente del sistema, permite que el consultor los pueda interpretar y llevar a cabo una persistencia directa. Esto puede ser mediante la instalación directa de un "backdoor" a través de una conexión SSH que se encontró en el código fuente. También es posible encontrar las credenciales de la base de datos del sistema, lo que permitiría registrar un usuario con perfil de administrador y obtener acceso directo sin necesidad de subir un archivo bandera o realizar un pivote para tal técnica.

Vulnerabilidades públicas.

Una fase que permite al consultor realizar consultas en las bases de datos de vulnerabilidades públicas y encontrar un exploit público que facilite el acceso o descarga de información. Al realizar esta búsqueda, abre diferentes opciones además de nuestro conocimiento personal, ya que existen muchos foros de la comunidad de hacking ético en los que se comparten pruebas de concepto o listas de palabras que ayudan en la identificación de ciertas brechas de seguridad y adjuntan posibles cargas útiles.

Recolección de evidencia

La recolección de evidencia es la fase final el que permitirá demostrar a la organización qué tan vulnerable se encuentra su red, sistema y activos. La evidencia es un factor crucial, que puede consistir en archivos descargados, capturas de pantalla tomadas durante la interacción con una vulnerabilidad específica e incluso el acceso de un usuario administrador demostrado con un nombre específico. Esto aumentará la severidad de la vulnerabilidad y, a su vez, la credibilidad de la existencia del hallazgo. Es una parte fundamental para la fase posterior.

4.5 Fase de documentación y reporte

La fase final de la guía es aquella en la que se realiza la documentación del análisis de seguridad. En este documento se proporciona un resumen de lo que se encontró en los dispositivos analizados, aplicaciones vulnerables o servicios expuestos. Todo esto se describe en un documento que explica la información obtenida, así como información sobre el objetivo. Además, se incluye la descripción de las vulnerabilidades encontradas, su clasificación, impacto y posible remediación.

La documentación de las pruebas de seguridad ayuda a las organizaciones a mantener un registro de los hallazgos, permitiendo, en caso de un análisis posterior, encontrar el estado del hallazgo, si está mitigado o no. En cuanto al contenido del documento, se incluye un resumen sobre los dispositivos vulnerables y sus ocurrencias encontradas en la prueba de penetración. Se brinda una explicación concisa y coherente de una sola

categoría de vulnerabilidades, en este caso, de acuerdo al OWASP Top Ten. Además, se utiliza la métrica basada en el "OWASP Risk Calculator" para asignar el impacto a la vulnerabilidad, lo que guiará la gestión para su mitigación. Por lo general, se realizará la mitigación de acuerdo con un impacto crítico, alto, medio o bajo, según las necesidades de la organización.

El reporte estará constituido por las fases mencionadas anteriormente, de forma resumida o detallada en ciertos fragmentos, integrando evidencia seleccionada para apoyar al responsable del área a identificar los hallazgos y los pasos a seguir para replicar dicha vulnerabilidad. El orden de la evidencia dependerá del impacto o relación entre ellas y las fases de las pruebas de penetración.

La evidencia permitirá al consultor expresar de forma técnica los procedimientos realizados para llegar a la identificación y explotación de la vulnerabilidad. Se utilizará lenguaje técnico de acuerdo con la descripción y construcción del primer reporte, conocido como reporte técnico. Posteriormente, de forma estadística, se realizará una explicación clara y formal detallando su impacto, lo cual funcionará como guía para orientar la gravedad que se involucra, conocido como un reporte ejecutivo.

Capítulo V
Caso de estudio: Prueba de Penetración a Portal Web de una dependencia universitaria con la metodología NIST SP 800-115 y marco de clasificación OWASP

5.1 Fase de Planificación

Como caso de estudio, se realizará una prueba de penetración a una institución educativa, la cual tiene como alcance realizarse a una aplicación web solamente en la búsqueda de afectaciones frontales o cuyas excepciones realicen un consumo de servicio, protocolos de comunicación, procesamiento de datos y representación, etcétera, entre el “Front-End” y el “Back-End” debido a que el objetivo principal y único será la el aplicativo móvil con la finalidad de buscar debilidades en sus mecanismos de seguridad, configuraciones tecnológicas, controles de accesos, por mencionar algunos y definido a no realizar verificación de seguridad a la infraestructura o servidores de su segmentación de red donde se encuentra montada la aplicación. Dicha aplicación es responsable de dar servicio a los alumnos del sistema a distancia, donde se estableció que se evaluaran las tecnologías con las cuales trabaja y estas se encuentren actualizadas, sin debilidades o exploits públicos al igual que toda información proporcionada se consideró como un activo a analizar, se estableció como caja gris, dicha información proporcionada se representa en la siguiente tabla (véase tabla 16).

Tabla 16 Tabla de información proporcionada

Título	Dato
Nombre de la aplicación	Sistema de Registros
URL / Enlace descarga	https://sistemaregis.edu.com.mx
Lenguaje y/o frameworks utilizados y versión	1. PHP 7.4 2. MariaDB 10.4 3. Apache 2.4

En la fase de planificación es importante tener una base sólida sobre la administración de proyectos y el manejo con cliente, debido a una buena comunicación con la otra parte se puede dar una verificación de seguridad exitosa, el conocimiento en redacción permite la realización de un documento donde se establece en formalidad las partes involucradas con la información proporcionada por ambos, todo esto basado a los acuerdos de confidencialidad y ética. El conocer a grandes rasgos las tecnologías implementadas permitirá saber un contexto superficial sobre el objetivo y a que se enfrentan.

Como todo servicio, se debe incluir información de fácil alcance por ejemplo datos de contacto del consultor y de parte de la institución para una comunicación rápida ante cualquier percance, la metodología implementada se establece al inicio de la prueba junto el rango de clasificación dando un porcentaje numérico que permita medir la severidad y explicar el marco de clasificación a usar en este caso el OWASP Top Ten 2017 el cual se deberá de tener conocimientos en aplicación de pruebas de seguridad éticas para su uso y finalmente explicar la metodología a usar en este caso la NIST-SP 800-115 la cual permite organizar la verificación de seguridad mediante fases que se representan en la siguiente imagen (véase figura 7).

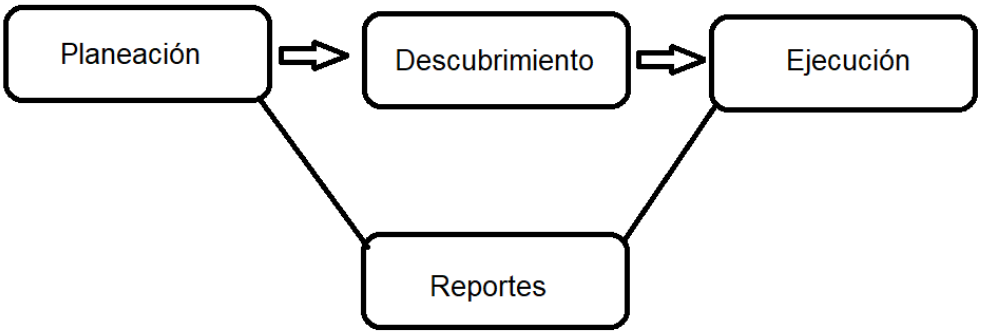


Figura 7 Esquema de metodología NIST

El documento de planeación se encuentra en el “Anexo A”, así mismo se puede realizar un documento de planeación tomando las siguientes bases por las que está compuesto, explicado por el siguiente esquema:

1. Objetivos y Alcance: Información plasmada por la institución definida al objetivo de búsqueda de vulnerabilidades en sus activos de mayor valor.
2. Información de contacto: Información del consultor y los responsables del área de la institución en la cual se integrarán el nombre, teléfono, correo y puesto para una comunicación de alto nivel.
3. Información del aplicativo: Información de la aplicación a evaluar proporcionada por la institución, la cual se integra por URL, información tecnológica, tipo de conexión, usuarios y privilegios.

4. Matriz de Escalación: Tabla organizada para los responsables del área y el analista, para una comunicación directa ante las adversidades integradas, solicitud de información ante cualquier alerta o reuniones imprevistas.
5. Firmas: Firmas de aceptación de ambas partes, las cuales aceptan un acuerdo establecido entre el consultor y la institución para la ejecución del servicio, fechas, alcances, entre otros. Con la finalidad de tratar cualquier información obtenida de forma confidencial.

5.2 Fase de Descubrimiento

Para iniciar esta fase, de acuerdo con la metodología establecida se tienen opciones que considerar, de inicio en primera estancia se tiene información directa sobre la aplicación, por la cual se tendrán dos fases a implementar:

Descubrimiento Pasivo

Se usa esta técnica para un reconocimiento en fuentes abiertas con los navegadores web asociando sus palabras reservadas para una búsqueda a profundidad, implementando conocimiento sobre la técnica “OSINT” y las herramientas “Open Source” que permitieron una búsqueda pasiva con alcance de las herramientas a utilizar, finalmente la búsqueda de información pública dio visibilidad de posibles ataques con relación a brechas de seguridad.

Descubrimiento Activo

En esta técnica se realiza la identificación y validación de las tecnologías que consume la aplicación web, haciendo una comparación entre lo que proporcionó la institución en el documento de planeación y lo encontrado en la interacción activa. El tipo de las tecnologías se pudo identificar con herramientas de líneas de comandos, respuesta en peticiones web y a su vez con extensiones de navegadores de consulta, con ella se pueden utilizar los recursos obtenidos como pivote para ataques posteriores o usar exploits públicos. En esta fase se deberá de encontrar información disponible que permita tener un

mapeo del sitio u organigrama de cómo son sus comunicaciones, secciones dentro de ellas, etcétera.

De igual manera se recomienda iniciar por la fase pasiva para observar cuanta información es posible obtener en recursos públicos y sin interacción directa con el objetivo, ya que eso permite conocer si fue vulnerable anteriormente y se expuso información, en adición la fase activa permite realizar una búsqueda directa con el objetivo donde se buscará información interactuando directamente.

La información obtenida en esta fase se observa en la siguiente tabla donde hay una relación sobre el conocimiento base para la interpretación correcta sobre los resultados (véase tabla 17).

Tabla 17 Conocimiento base asociada al reconocimiento obtenido

Información recolectada	Herramientas Usada	Tecnologías identificadas	Conocimiento base
Protocolos de comunicación identificados.	TestSSL	• SSLv3. • TLSv1.0. • TLSv1.1. • TLSv1.2.	• Protocolos de comunicación. • Análisis de protocolos TLS/SSL. • Seguridad en canales de comunicación. • Configuración de infraestructura.
Sistema Operativo.	Censys	CentOS.	• Comunicación TTL. • Puertos por defecto. • Sistemas Operativos.
Servidor Web.	Whatweb	Apache 2.4.	• Configuración en servidores. • Servidores web.

Ambiente de desarrollo	Proxy	PHP 7.4.	• Entornos de desarrollo. • Extensiones de archivo. • Lenguajes de programación.
Frameworks	Whatweb	Javascript 1.0.2k. OpenSSL. jQuery 3.6.1. Bootstrap 4.6.0.	• Sintaxis de programación. • Desarrollo web y móvil. • Lectura de directorios. • Sistemas web.
Puertos	NMAP	TCP/80/HTTP. TCP/443/HTTPS.	• Configuración IP. • Servicios y servidores.

5.3 Fase de Ejecución

El análisis del siguiente movimiento es importante en esta fase ya que ayuda a hallar puntos de entrada para inserción de carga útil que da indicios para encontrar malas configuraciones o pistas que orienten a una vulnerabilidad asociada, algunas de las técnicas que se pueden implementar son las siguientes:

Escaneo automatizado

Las herramientas o script programados por uno mismo o públicos que permitieron automatizar las peticiones realizadas para la búsqueda de documentos, directorios, archivos indexados o comportamientos que permitieron identificar configuraciones por defecto y a su vez el servidor web asociada y conocer si el sistema es vulnerable de acuerdo a la versión, adicionalmente permiten la búsqueda a profundidad y ahorrar tiempo para encontrar archivos sensibles que puedan traer credenciales en texto claro, este método dependiendo las técnicas asociadas se consideran como intrusivas o pasivas,

adicionalmente pueden hacer que se provoque una denegación de servicio, por lo cual se usan con precaución.

Escaneo manual

El uso de un proxy permitió la visualización de la comunicación entre el navegador y la aplicación web, ya que al usar una herramienta de captura de peticiones HTTP da oportunidad a la modificación de cualquier solicitud enviada a los dispositivos, esta técnica permite la inserción de carga útil de forma manual y controlada, en la cual se pueden provocar errores que muestre información útil, inyección de código, encontrar métodos HTTP no comunes, entre otros.

Análisis de vulnerabilidades

El escaneo de vulnerabilidades es una técnica que permitió encontrar configuraciones por defecto o inseguras. El encontrar una vulnerabilidad se debe validar ya que puede mostrar falsos positivos, falsos negativos, positivos, negativos, dependiendo de la prueba de concepto enviada por la herramienta se van a validar si el sitio web es vulnerable o no.

La siguiente tabla (véase tabla 18) da la relación de las vulnerabilidades encontradas en la prueba de penetración en conjunto con todas las fases de la metodología, adicionalmente se muestra junto con su severidad y los conocimientos para su descubrimiento y explotación:

Tabla 18 Conocimiento base asociada a las vulnerabilidades encontradas

OWASP Top 10 2017	Nombre de vulnerabilidad	Herramienta Usada	Severidad	Conocimiento base
A3 Sensitive Data Exposure	Exposición de información sensible	Dirbruster	Crítica	Códigos de respuesta HTTP
A7 Cross-Site Scripting	Secuencias de comando entre sitios (XSS-Almacenado)	Burpsuite Proxy	Alta	Programación web
A5 Broken Access Control	Falsificación de solicitud entre sitios (CSRF)	Burpsuite Proxy	Alta	Cabeceras HTTP y programación web
A6 Security Misconfiguration	Carga de Archivos Insegura (Tipo de Archivo)	Burpsuite Proxy	Alta	Extensión de archivos y

A2 Broken Authentication	Identificador en URL	Burpsuite Proxy	Alta	Buenas configuraciones y comunicación HTTP
A6 Security Misconfiguration	Protocolo Obsoleto Habilitado (SSLV3)	TestSSL	Alta	Protocolos de comunicación
A9 Using Components with Known Vulnerabilities	Servidor web con vulnerabilidades públicas	Whatweb	Moderada	Buenas prácticas de la industria
A9 Using Components with Known Vulnerabilities	Componente con vulnerabilidades públicas	Whatweb	Moderada	Buenas prácticas de la industria
A6 Security Misconfiguration	Paginas por defecto habilitadas	Burpsuite Proxy	Moderada	Configuración web y buenas practicas
A6 Security Misconfiguration	Métodos HTTP no necesarios habilitados	Burpsuite Proxy	Moderada	Configuración web y comunicación HTTP
A6 Security Misconfiguration	Algoritmos Inseguros Habilitados	TestSSL	Moderada	Protocolos de comunicación
A6 Security Misconfiguration	Protocolo Obsoleto Habilitado (TLSV1.1)	TestSSL	Moderada	Protocolos de comunicación
A6 Security Misconfiguration	Protocolo Inseguro Habilitado (TLSV1.0)	TestSSL	Moderada	Protocolos de comunicación
A2 Broken Authentication	Política de contraseña débil no habilitada	Burpsuite Proxy	Moderada	Buenas prácticas y programación web
A6 Security Misconfiguration	Autocompletado Habilitado	Burpsuite Proxy	Moderada	Buenas prácticas y programación web
A2 Broken Authentication	Atributo HTTPOnly en Cookie no habilitada	Burpsuite Proxy	Baja	Buenas prácticas y configuración web
A6 Security Misconfiguration	Redirección de Sitios Insegura	Burpsuite Proxy	Baja	Buenas prácticas y configuración web
A6 Security Misconfiguration	Metadata en Archivos	Exiftool	Informativa	Buenas prácticas y programación web

A continuación, se realiza una relación de las vulnerabilidades halladas con el marco de clasificación OWASP Top Ten 2017 y su calculadora "OWASP Risk Rating", el cual permitió clasificar los resultados del caso práctico de la siguiente manera, representada en la siguiente imagen (véase figura 8):

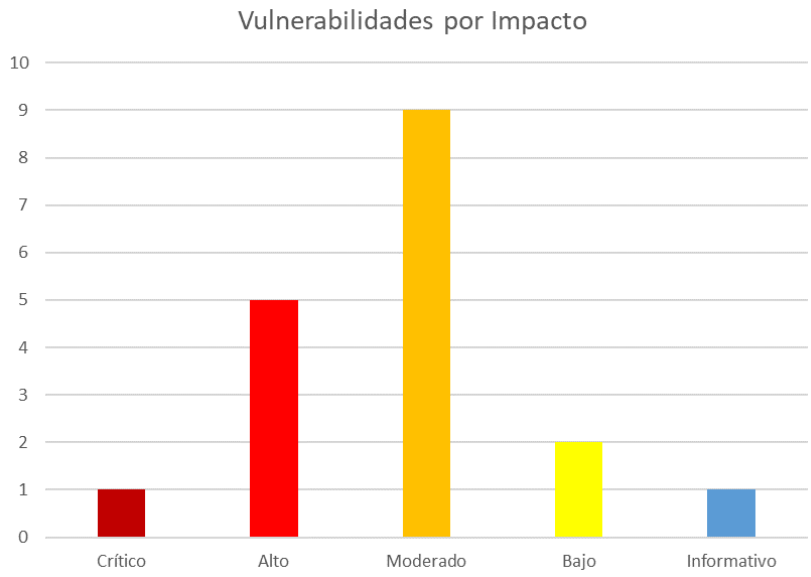


Figura 8 Gráfica de vulnerabilidades

TTPs (Tácticas, Técnicas y Procedimientos)

Para realizar el descubrimiento de vulnerabilidades y explotación relacionada a la brecha de seguridad, se realizan actividades y métodos empleados como actores maliciosos para intentar comprometer la CIA del sistema informático, para ello se realiza la tabla siguiente (véase tabla 19) con relación a la CIA, TTPs y la vulnerabilidad encontrada.

Tabla 19 Tabla de relación TTPs y vulnerabilidades

Tácticas	Técnica	Procedimiento	Vulnerabilidad	CIA Afectada
Reconocimiento	Escaneo activo	Escaneo de lista de palabras	Exposición de información sensible	Confidencialidad
Ejecución	Intérprete de comandos y scripting	Javascript	Secuencias de comando entre sitios (XSS-Almacenado)	Integridad

Ejecución	Explotación para la ejecución del cliente	Explotación basada en navegador	Falsificación de solicitud entre sitios (CSRF)	Integridad
Desarrollo de recursos	Capacidades del escenario	Carga de archivos maliciosos	Carga de Archivos Insegura (Tipo de Archivo)	Integridad
Exfiltración	Exfiltración sobre canal C2	Exposición de token en URL	Identificador en URL	Confidencialidad
Comando y Control	Protocolo de capa de aplicación: Protocolos Web	Uso de protocolos obsoletos o sin soporte	Protocolo Obsoleto Habilitado (SSLV3)	Confidencialidad
Obtener capacidad	Vulnerabilidades	Vulnerabilidades conocidas	Servidor web con vulnerabilidades públicas	Integridad
Obtener capacidad	Vulnerabilidades	Vulnerabilidades conocidas	Componente con vulnerabilidades públicas	Integridad
Descubrimiento	Páginas por defecto	Exposición de páginas por defecto	Paginas por defecto habilitadas	Confidencialidad
Inyección de proceso	Proceso de inyección	Llamadas al sistema trace	Métodos HTTP no necesarios habilitados	Integridad
Evasión de defensa	Ofuscar/decodificar archivos o información	Uso de algoritmos obsoletos o inseguros	Algoritmos Inseguros Habilitados	Confidencialidad
Comando y Control	Protocolo de capa de aplicación: Protocolos Web	Uso de protocolos obsoletos o sin soporte	Protocolo Obsoleto Habilitado (TLSV1.1)	Confidencialidad
Comando y Control	Protocolo de capa de aplicación: Protocolos Web	Uso de protocolos obsoletos o sin soporte	Protocolo Inseguro Habilitado (TLSV1.0)	Confidencialidad
Acceso a credenciales	Fuerza bruta: Adivinación de contraseñas	Política de contraseña débil	Política de contraseña débil no habilitada	Integridad
Acceso inicial	Cuentas validas	Obtención de cuentas locales	Autocompletado Habilitado	Confidencialidad
Acceso a credenciales	Robar cookie de sesión web	Recolección de cookies	Atributo HTTPOnly en Cookie no habilitada	Confidencialidad

Flujo de ejecución	Ejecución de secuestro	Redireccionamiento a sitios maliciosos	Redirección de Sitios Insegura	Integridad
Reconocimiento	Recopilar información del anfitrión de la víctima	Exposición de software o metadata	Metadata en Archivos	Confidencialidad

5.4 Fase de Documentación y Reporte

La documentación y el reporte de la prueba de penetración son importantes para todo el caso práctico, ya que es la base para fundamentar todos los hallazgos encontrados. En él se implementan de forma técnica y ejecutiva, con un lenguaje de alto nivel que permite describir con detalle las vulnerabilidades.

- Reporte Ejecutivo: El documento de reporte ejecutivo se encuentra en el "Anexo B" para su consulta y complementación. Este informe está dirigido a niveles altos de administración, directivos o gerencia pueden no estar familiarizados con los detalles técnicos, pero necesitan comprender las implicaciones y riesgos generales, este puede estar conformado por:
    - Resumen Ejecutivo: Una breve descripción de los objetivos del pentest, el alcance, la duración y un resumen de los hallazgos más críticos.
    - Metodología: Una descripción general de la metodología utilizada para realizar el pentest.
    - Hallazgos Clave: Un resumen de los principales problemas de seguridad descubiertos durante el pentest, junto con su impacto potencial en la organización. En él se muestra numéricamente el impacto y el nombre genérico de las vulnerabilidades para su consulta posterior asociada a una representación gráficamente.
    - Nivel de Riesgo: Clasificación de los hallazgos según su gravedad y el riesgo que representan para la organización.
    - Conclusiones: Un resumen de las conclusiones generales y un llamado a la acción para abordar los problemas de seguridad.

- Reporte Técnico: El documento de reporte ejecutivo se encuentra en el “Anexo C” para su consulta y complementación, por cuestiones de confidencialidad no se adjuntó evidencia y procedimientos que se realizaron en dichos escenarios. Este informe está dirigido a desarrolladores o CISO que sean los administradores del equipo de desarrolladores de las aplicaciones web, en el cual se realiza una explicación detallada de cada vulnerabilidad que se conforma por:
    - Introducción: Descripción de los objetivos del pentest, el alcance, las restricciones y la metodología.
    - Hallazgos encontrados: Resumen de las vulnerabilidades encontradas en el pentest y su ejecución, la cual puede verse representada en el siguiente párrafo, dicha relación se conforma por las siguientes partes.
        - Descripción de Hallazgos: Detalles técnicos de cada hallazgo, incluyendo la descripción del problema, la ruta de ataque seguida y los sistemas afectados.
        - Evidencia de Explotación: Si es posible, proporciona evidencia detallada de cómo se pudo explotar cada hallazgo.
        - Impacto y Riesgo: Explica el impacto potencial de cada hallazgo y el riesgo que representa para la organización. Adicionalmente, se puede integrar una tabla que ilustre las vulnerabilidades encontradas de acuerda a su color y la cantidad de ocurrencias encontradas.
        - Recomendaciones de Mitigación: Proporciona instrucciones detalladas sobre cómo remediar cada hallazgo, incluidos los pasos técnicos a seguir.
        - Apoyo Técnico: Proporciona enlaces a recursos, herramientas o guías adicionales que puedan ayudar en la mitigación.
    - Conclusiones Técnicas: Un resumen de las conclusiones técnicas y la eficacia general de las defensas de la organización.

5.5 Resultados y evidencia del caso de estudio.

En esta sección se representa las vulnerabilidades en las tablas siguientes con una breve descripción a un escenario asociado a su posible riesgo y severidad, explicando la probabilidad y el impacto relacionado a cada una, la cual permitió clasificarlas y representárselas de mayor severidad a menor, cabe mencionar que por cuestiones de confidencialidad no se adjuntó evidencia y procedimientos que se realizaron en dichos escenarios.

La importancia y objetivo de estas pruebas de seguridad permite reforzar los mecanismos de seguridad implementados e identificar las configuraciones inseguras permitiendo reforzar buenas prácticas en el aspecto de desarrollo, actualización y administración de una aplicación web.

Exposición de información sensible -> Severidad Alta

Tabla 20 Vulnerabilidad - Exposición de información sensible

Descripción:	Se expone información confidencial del usuario, como contraseñas, nombre de usuarios, nombre de bases de datos y certificados SSL, por mencionar algunos.
Riesgo:	Probabilidad: Moderada: La vulnerabilidad es sencilla de encontrar, al hallarse con una herramienta automatizada y una técnica conocida como “Fuzzing”. Impacto: Alto: La exposición de posibles nombres de usuarios y contraseñas en texto claro, son criticas para escalar un ataque, adicionalmente no se debe de almacenar información sensible del lado del cliente.
Recomendación	• Audite cualquier código para detectar una posible divulgación de información como parte de su control de calidad o procesos de compilación. • Se recomienda ampliamente que no se exponga la información confidencial de usuarios, servicios o diferentes contextos, en los cuales se puedan observar en la respuesta HTTP. • Asegúrese de que todos los involucrados en la producción del sitio web sean plenamente conscientes de qué información se considera confidencial.

Secuencias de comando entre sitios (XSS-Almacenado) -> Severidad Alta

Tabla 21 Vulnerabilidad - Secuencias de comando entre sitios (XSS-Almacenado)

Descripción:	Es una vulnerabilidad donde un atacante logra insertar código malicioso, generalmente en forma de JavaScript o HTML. Se inyecta y es ejecutado por el servidor web, se muestra a los usuarios después de ser recuperado del almacenamiento volviéndolo persistente. El navegador ejecuta este código dado que proviene de un servidor considerado como confiable.
Riesgo:	Probabilidad: Alta - La aplicación no es sencilla de encontrar, se hallaron de forma manual por medio de una herramienta conocida como “Proxy”. Impacto: Moderada – Afecta a usuarios registrados dentro de la aplicación, ya que la vulnerabilidad permitió que se puede insertar código HTML con redireccionamiento a un sitio externo.
Recomendación	• La codificación de datos de entrada y salida debe aplicarse directamente antes de que los datos controlables por el usuario se escriban en una página, ya que el contexto en el que escribe determina qué tipo de codificación necesita usar • Valide los datos de entrada de acuerdo con la industria y su estándar • Validación del lado del servidor • Validar la entrada lo más estrictamente posible en el momento en que se recibe por primera vez de un usuario, usando diccionarios de listas blancas.

Falsificación de solicitud entre sitios (CSRF) -> Severidad Alta

Tabla 22 Vulnerabilidad - Falsificación de solicitud entre sitios (CSRF) -> Alta

Descripción:	Ataque que obliga a un usuario especifico a realizar acciones no deseables en lo que está autenticado. Utilizando técnicas de ingeniería social, un atacante podría generar peticiones maliciosas para que se realicen por él en un perfil valido y sin su consentimiento.
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que se tienen que autenticar para acceder a funcionalidades, realiza una traza al ataque, se hallaron de forma manual por medio de una herramienta conocida como “Proxy”. Impacto: Alto – Afecta a usuarios registrados dentro de la aplicación, ya que la vulnerabilidad permite la modificación de información de perfil sin que el dueño de la cuenta tenga consentimiento.

Recomendación	• Emplee un token dentro de las solicitudes relevantes. El token debe cumplir con los siguientes criterios: • Impredecible con alta entropía, como para tokens de sesión en general. • Vinculado a la sesión del usuario.

Carga de Archivos Insegura (Tipo de Archivo) -> Severidad Alta

Tabla 23 Vulnerabilidad - Carga de Archivos Insegura (Tipo de Archivo)

Descripción:	La carga de archivos dentro de la aplicación web permite cargar archivos al servidor de cualquier tipo (extensión).
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que se tienen que autenticar para acceder a funcionalidades, realiza una traza al ataque, se hallaron de forma manual por medio de una herramienta conocida como “Proxy”. Impacto: Alto – Afecta a usuarios registrados, por lo cual podría realizar la carga de archivos maliciosos, la aplicación permite la carga de archivos, el segmento que se marco fue dentro de la carga una imagen de perfil permitiendo subir un archivo ejecutable, posteriormente descargable.
Recomendación	• Verifique la extensión del archivo con una lista blanca de extensiones permitidas. • Asegúrese de que el nombre del archivo no contenga subcadenas que puedan interpretarse como un directorio o una secuencia transversal (../). • No cargue archivos en el sistema de archivos permanente del servidor hasta que hayan sido completamente validados.

Identificador en URL -> Severidad Alta

Tabla 24 Vulnerabilidad - Identificador en URL

Descripción:	Se envía token de identificación considerados como sensibles por medió de la URL, utilizando el método HTTP GET.
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que las funcionalidades que exponen el token se encuentran al autenticarse, se hallaron de forma manual por medio de una herramienta conocida como “Proxy”. Impacto: Alto – Afecta a usuarios registrados dentro de la aplicación, ya que la vulnerabilidad transmite un token y estos quedan expuestos los cuales podrían ser usados por un atacante.
Recomendación	• Use un mecanismo para transmitir tokens de sesión, como cookies HTTP o campos ocultos en formularios que se envían mediante el método POST.

Protocolo Obsoleto Habilitado (SSLV3) -> Severidad Alta

Tabla 25 Vulnerabilidad - Protocolo Obsoleto Habilitado (SSLV3)

Descripción:	El protocolo HTTPS hace uso de un protocolo obsoleto conocido como SSLV3, el cual ya no recibe soporte del proveedor.
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que se necesitan técnicas de análisis de protocolos e identificación de estas, se hallaron de forma técnica por medio de una herramienta para el análisis de protocolos en canales de comunicación. Impacto: Alto – Afecta a toda la información enviada y recibida en el canal de comunicación, dicha vulnerabilidad puede usarse para lograr descifrar la información que viaja dentro del canal de comunicación
Recomendación	• Deshabilite los protocolos SSLv3 e inferiores. • Habilite protocolos TLSv1.2 o superiores con algoritmos seguros. • Use algoritmos con longitud mayor a 128 bits • No habilite los algoritmos con vulnerabilidades asociadas como son: RC4, SHA-1, MD5.

Servidor web con vulnerabilidades públicas -> Severidad Media

Tabla 26 Vulnerabilidades - Servidor web con vulnerabilidades públicas

Descripción:	El servidor actual que se encuentra operando sobre una versión con vulnerabilidades conocidas que debilitan la seguridad implementada. Para más información consulte el siguiente enlace: https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-66/version_id-514578/opbyp-1/Apache-Http-Server-2.4.6.html
Riesgo:	Probabilidad: Moderada - La vulnerabilidad es sencilla de encontrar, pero no explotable sencillamente debido a que son vulnerabilidades conocidas, pero con técnicas avanzadas para una explotación, se hallaron de forma técnica por medio de una herramienta de identificación de los recursos tecnológicos implementados en la aplicación web. Impacto: Moderado – Afecta a toda la aplicación operando en el servidor, se encuentran vulnerabilidades públicas, se necesitan técnicas avanzadas y se tienen fortalecidas las medidas de seguridad, dichas vulnerabilidades conocidas pueden ser explotadas por un usuario malicioso para ataques posteriores.
Recomendación	• Aplique un inventario de los componentes tecnológicos en el despliegue de la aplicación. • Aplique un plan de mantenimiento. • Aplique un plan de actualización y monitoreo, si se requiere aplique parches de seguridad.

Componente con vulnerabilidades públicas -> Severidad Media

Tabla 27 Vulnerabilidad - Componente con vulnerabilidades públicas

Descripción:	El componente actual que se encuentra operando en la aplicación web tiene vulnerabilidades conocidas que debilitan la seguridad implementada. Para más información consulte el siguiente enlace: https://www.cvedetails.com/vulnerability-list/vendor_id-217/product_id-383/version_id-577438/Openssl-Openssl-1.0.2k.html
Riesgo:	Probabilidad: Moderada - La vulnerabilidad es sencilla de encontrar, pero no explotable fácilmente debido a que son vulnerabilidades conocidas, pero con técnicas avanzadas para una explotación, se hallaron de forma técnica por medio de una herramienta de identificación de los recursos tecnológicos implementados en la aplicación web. Impacto: Moderado – Afecta a toda la aplicación operando en el servidor, se encuentran vulnerabilidades públicas se necesitan técnicas avanzadas y se tienen fortalecidas las medidas de seguridad, dichas vulnerabilidades conocidas pueden ser explotadas por un usuario malicioso para ataques posteriores.
Recomendación	• Aplique un inventario de los componentes tecnológicos en el despliegue de la aplicación. • Aplique un plan de mantenimiento. • Aplique un plan de actualización y monitoreo, si se requiere aplique parches de seguridad.

Páginas por defecto habilitadas -> Severidad Media

Tabla 28 Vulnerabilidad - Páginas por defecto habilitadas

Descripción:	Se tienen paginas por defecto habilitadas por el servidor web o páginas que muestran información de tecnologías usadas o configuraciones.
Riesgo:	Probabilidad: Moderada - La vulnerabilidad es sencilla de encontrar, debido a que se encuentran expuestas, se realizan ataques automatizados para identificarlas, por lo que podría generar una alerta en el monitoreo, se hallaron de forma técnica por medio de una herramienta para el análisis de recursos o “fuzzing”. Impacto: Moderado – Afecta de forma general la aplicación web debido a que pueden ser páginas que muestren información actual o pasada que ya no se encuentre en uso en la aplicación, dicha vulnerabilidad puede usarse para un reconocimiento profundo a la aplicación en caso de encontrar información actual.
Recomendación	• Deshabilite o elimine páginas que se instalen por defecto o no sea requeridas por el servidor, para que no sean accedidas por un usuario no autorizado.

Métodos HTTP no necesarios habilitados -> Severidad Media

Tabla 29 Vulnerabilidad - Métodos HTTP no necesarios habilitados

Descripción:	Se encuentran habilitados métodos HTTP, que no son comúnmente usados para la finalidad que tienen las aplicaciones web, métodos habilitados “TRACE, HEAD”.
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que se necesita conocimiento técnico en los métodos HTTP y sus peticiones disponibles, se hallaron de forma manual por medio de una herramienta conocida como “Proxy”. Impacto: Moderada – Afecta de forma general a la aplicación debido al tipo de métodos HTTP encontrados, ya que se necesitan técnicas avanzadas y podría funcionar como pivote de vulnerabilidades expuestas para una explotación compleja.
Recomendación	• Deshabilite los métodos que no sean necesarios en el servidor web, por el estándar de la industria los métodos más comunes para una aplicación web son GET y POST.

Algoritmos Inseguros Habilitados -> Media

Tabla 30 Vulnerabilidad - Algoritmos Inseguros Habilitados

Descripción:	El canal de comunicación en HTTPS usa algoritmos en sus protocolos TLS que son considerados como inseguros,
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que se necesitan técnicas de análisis de protocolos e identificación de estas, se hallaron de forma técnica por medio de una herramienta para el análisis de protocolos en canales de comunicación. Impacto: Moderado– Afecta a toda la información enviada y recibida en el canal de comunicación debido a que los métodos de cifrados son diferente y un poco sofisticados disminuye el impacto, dicha vulnerabilidad puede usarse para explotar un ataque conocido como “Man in the middle”.
Recomendación	• Use algoritmos con longitud mayor a 128 bits • No habilite los algoritmos con vulnerabilidades asociadas como son: RC4, SHA-1, MD5.

Protocolo Obsoleto Habilitado (TLSV1.1) -> Severidad Media

Tabla 31Vulnerabilidad - Protocolo Obsoleto Habilitado (TLSV1.1)

Descripción:	El protocolo HTTPS hace uso de un protocolo obsoleto conocido como TLSV1.1 por las técnicas de cifrado implementadas.
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que se necesitan técnicas de análisis de protocolos e identificación de estas, se hallaron de forma técnica por medio de una herramienta para el análisis de protocolos en canales de comunicación. Impacto: Moderado– Afecta a toda la información enviada y recibida en el canal de comunicación debido a que los métodos de cifrados son diferente y un poco sofisticados disminuye el impacto, dicha vulnerabilidad puede usarse para explotar un ataque conocido como “Man in the middle”.
Recomendación	• Deshabilite los protocolos TLSv1.1 e inferiores. • Habilite protocolos TLSv1.2 o superiores con algoritmos seguros. • Use algoritmos con longitud mayor a 128 bits • No habilite los algoritmos con vulnerabilidades asociadas como son: RC4, SHA-1, MD5.

Protocolo Inseguro Habilitado (TLSV1.0) -> Severidad Media

Tabla 32 Vulnerabilidad - Protocolo Inseguro Habilitado (TLSV1.0)

Descripción:	El protocolo HTTPS hace uso de un protocolo inseguro conocido como TLSV1.0 por las técnicas de cifrado implementadas y los algoritmos permitidos.
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que se necesitan técnicas de análisis de protocolos e identificación de estas, se hallaron de forma técnica por medio de una herramienta para el análisis de protocolos en canales de comunicación. Impacto: Moderado– Afecta a toda la información enviada y recibida en el canal de comunicación debido a que los métodos de cifrados son diferentes y los algoritmos aceptados son más sofisticados disminuye el impacto, dicha vulnerabilidad puede usarse para explotar un ataque conocido como “Man in the middle” y adicionalmente podría descifrar conexiones por medio del uso de un servidor SSLV2.
Recomendación	• Deshabilite los protocolos TLSv1.0 e inferiores. • Habilite protocolos TLSv1.2 o superiores con algoritmos seguros. • Use algoritmos con longitud mayor a 128 bits • No habilite los algoritmos con vulnerabilidades asociadas como son: RC4, SHA-1, MD5.

Política de contraseña debilidades no habilitada -> Severidad Media

Tabla 33 Vulnerabilidad - Política de contraseña debilidades no habilitada

Descripción:	La sección de cambio de contraseña no forza el uso de una política segura debido a que la generación de contraseñas no cuenta con el estándar mínimo de la industria.
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que se necesita una cuenta registrada, se hallaron de forma manual por medio de la navegación de la aplicación web. Impacto: Moderado – Afecta el mecanismo de autenticación debido a que la aplicación no forza el uso de una política segura de acuerdo con el estándar, dicha vulnerabilidad puede permitir a un atacante encontrar cuentas vulnerables y realizar un ataque de fuerza bruta o cracking de contraseñas.
Recomendación	• Establezca una política de contraseñas segura con las siguientes características: • Mínimo de 12 caracteres • Contenga como mínimo una letra mayúscula • Contenga como mínimo una letra minúscula • Contenga como mínimo un carácter especial • Contenga como mínimo un numero

Autocompletado Habilitado -> Severidad Media

Tabla 34 Vulnerabilidad - Autocompletado Habilitado

Descripción:	Se encuentra habilitada la funcionalidad de autocompletado lo que da la opción de que el navegador web guarde la información que se ha escrito en un formulario.
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que se necesita conocimientos en código HTML, se hallaron de forma manual por medio de la navegación de la aplicación web. Impacto: Moderado – Afecta la seguridad de los usuarios, debido a que permite observar la información escrita anteriormente, lo cual un atacante podría identificar datos validos como usuarios en la sección de login.
Recomendación	• Para evitar que los navegadores almacenen las credenciales ingresadas en formularios HTML, incluya el atributo autocomplete="off" dentro de la etiqueta FORM (para proteger todos los campos del formulario) o dentro de las etiquetas INPUT relevantes (para proteger campos individuales específicos).

Atributo HTTPOnly en Cookie no habilitada -> Severidad Baja

Tabla 35 Vulnerabilidad - Atributo HTTPOnly en Cookie no habilitada

Descripción:	Atributo HTTPOnly no se encuentra habilitado en la cookie de sesión.
Riesgo:	Probabilidad: Baja- La vulnerabilidad es sencilla de encontrar, debido a que al ingresar configura una cookie de sesión, pero se necesita conocimiento sobre las configuraciones y manejo de sesiones para su identificación. Impacto: Moderado – Afecta la seguridad de identificadores de sesión de los usuarios, debido a que permite a la cookie de sesión se envié por medio de un Cross Site Scripting en el cual se podría extraer el valor de los identificadores de sesión.

Recomendación	• Configure las cookies de sesión con los atributos de configuración segura por la industria como lo son
	  ○ Secure
	  ○ HTTPOnly
	Con la finalidad de realizar la comunicación más segura de las cookies.

Redirección de Sitios Insegura -> Severidad Baja

Tabla 36 Vulnerabilidad - Redirección de Sitios Insegura

Descripción:	Se puede redireccionar a los usuarios hacia sitios internos, es modificable el valor del parámetro el cual podría redireccionarse a archivos públicos locales que podrían afectar las configuraciones de los usuarios o equipos personales.
Riesgo:	Probabilidad: Moderada - La vulnerabilidad no es sencilla de encontrar, debido a que se necesita una cuenta registrada, se hallaron de forma manual por medio de la navegación de la aplicación web. Impacto: Bajo – Afecta la seguridad de los usuarios, debido a que permite la redirección a cualquier URL local, adicionalmente el valor es modificable, a realizar la redirección localmente no permite una explotación clara, pero que podría funcionar como pivote.
Recomendación	• Evite incorporar datos controlables por el usuario en los objetivos de redirección. • Elimine la función de redirección de la aplicación y reemplace los enlaces a ella con enlaces directos a las URL de destino relevantes. • Mantenga una lista del lado del servidor de todas las URL que están permitidas para la redirección. • En lugar de pasar la URL de destino como parámetro al redirector, pase un índice a esta lista. • La aplicación debe utilizar URL relativas en todos sus redireccionamientos y la función de redireccionamiento debe validar estrictamente que la URL recibida es una URL relativa.

Metadata en Archivos -> Informativa

Tabla 37 Vulnerabilidad - Metadata en Archivos

Descripción:	Se encuentran archivos alojados o producidos por las aplicaciones web, los cuales contiene información que ayudan a identificar que software los produjo.
Riesgo:	Probabilidad: Baja - La vulnerabilidad no es sencilla de encontrar, debido a que se necesita una cuenta registrada, se hallaron de forma manual por medio de la navegación de la aplicación web. Impacto: Bajo – Afecta la información de software usados, debido a que muestra información que le puede servir a un atacante a identificar versiones o software usado por la aplicación web.
Recomendación	• Para archivos de Windows, utilice las opciones correspondientes de propiedades para eliminar los metadatos integrados. • Para software utilice sus opciones y propiedades correspondientes para eliminar metadatos. • Para archivo cargados y almacenados para su descarga utiliza una herramienta para eliminar los metadatos integrados.

Con el caso práctico a la institución educativa, se observó que los mecanismos de seguridad se encuentran con debilidades, en este caso se relacionan a configuraciones por defecto lo cual afectó a la confidencialidad de la aplicación mostrando credenciales en texto plano, la falta de validación en los datos de entrada hallada permitió que el flujo natural se modificara afectando a la integridad y finalmente la disponibilidad no se afectó debido a que no se encontraron vectores de ataque para un escenario relacionado, adicionalmente se estableció en el documento de planeación que no se afectaría la disponibilidad de la aplicación por medio de un ataque de denegación de servicio o cualquier escenario de riesgo relacionado que afecte este factor de la triada, un factor a considerar es el tiempo de ejecución que se asocia al tamaño de la aplicación haciendo que la prueba dure desde 1 semana hasta 6 meses.

Finalmente se encontraron fortalezas (véase tabla 38) que no permitieron que se extrajera información por su configuración personalizada y su firewall perimetral que no permitía cualquier tráfico provocado dentro del flujo normal, ante cualquier anomalía reducía el tráfico aceptado o bloqueaba el origen de este.

Tabla 38 Tabla de Fortalezas

Nombre de Fortaleza	Descripción
Firewall perimetral	Se identificó en la fase de reconocimiento activo debido a que no se permitió la conexión al encontrar un segmento de red, debido al firewall perimetral establecido entre el servidor web y la infraestructura.
Balanceador de carga	Se identificó debido en la fase de descubrimiento con la técnica “Fuzzing” debido a que al realizar peticiones constantes esta detenía el tráfico repetido por host origen por cierto tiempo.
Puertos exclusivos a servicio web	Se identificó en el reconocimiento activo al analizar la IP de la aplicación web, la cual solo tenía habilitados los puertos destinados a HTTP/HTTPS.

Conclusiones

La realización de esta tesis permitió ejecutar una prueba de seguridad al portal web de una institución educativa, en esta se plasmó a nivel práctico como se puede usar una metodología para una prueba de penetración y se expresó cómo hacer uso de las herramientas de acuerdo a cada fase, permitiendo encontrar vulnerabilidades y emplear un marco de clasificación en este caso el OWASP Top Ten, finalmente con los riesgos asociados a cada vulnerabilidad, se brindan las bases a la alta gerencia para generar estrategias de mitigación de acuerdo a las remediaciones asociadas a cada una.

La relación entre la seguridad informática y el aumento del uso de la tecnología siempre va de la mano, ya que, con la innovación de estas, las debilidades y vulnerabilidades se han vuelto más complicadas con el tiempo. Sin embargo, a su vez, el software ha evolucionado permitiendo disminuir esta complejidad. Tanto la relación con las herramientas de protección como las de ataque se han visto envueltas en un crecimiento tecnológico.

Por lo tanto, se empleó un caso práctico para relacionar el uso de una metodología, herramientas de software libre y un marco de clasificación para la realización de una prueba de seguridad en una institución educativa. Esto permitió incluir las fases pertinentes, las herramientas relacionadas en cada una, las clasificaciones que se asocian con la probabilidad e impacto que ayudan a establecer la severidad de cada vulnerabilidad y, finalmente, la documentación que, con la ayuda de todo lo recopilado, permitió construir un informe que relaciona estadísticamente los hallazgos técnicos con la demostración de su explotación y sus posibles recomendaciones genéricas para su mitigación.

La siguiente tabla (véase tabla 39) representa la relación y descripción realizada entre la metodología NIST SP 800-115 y el caso de estudio:

Tabla 39 Relación NIST con caso de estudio

Fases NIST 800-115	Relación Caso de estudio
Determinación de objetivo y propósito	Se realizó la creación de un documento donde se establecían, objetivos, restricciones, involucrados, llamada carta de planeación.
Técnicas de evaluación, análisis y validación de vulnerabilidades	Se involucraron las técnicas de evaluación y ejecución de pruebas de seguridad para implementar la fase de descubrimiento, reconocimiento para la identificación de tecnologías involucradas.
Evaluación de seguridad y ejecución de prueba de seguridad	Con las posibles brechas de seguridad, y su evaluación se realizó la ejecución de carga útil en búsqueda de validar las vulnerabilidades encontradas, buscando asociar un control total del sistema en el cual fuera por una de estas.
Actividades posteriores de la prueba de penetración	Se debe de considerar que, si existe la posibilidad de carga de archivos, sean con un nombre común o archivos bandera, con la explotación de vulnerabilidades se validaron con relación a las tecnologías y búsqueda de vulnerabilidades conocidas.
Fase de documentación y reporte	En la documentación y reporte al finalizar el caso práctico permitió la realización de un reporte ejecutivo y técnico el cual relaciona todos los hallazgos encontrados en la exploración de estos.

Por último, se logró asociar el marco de clasificación OWASP con las vulnerabilidades halladas en el caso de estudio las cuales se representan en la siguiente tabla (véase tabla 40):

Tabla 40 Tabla de relación OWASP y hallazgos encontrados

OWASP Top 10 2017	Vulnerabilidades asociadas.
A1- Injection	No aplicó, debido a que limpiaban cualquier inyección de código relacionado al intérprete.
A2 – Broken Authentication	• Falsificación de solicitud entre sitios (CSRF) • Identificador en URL • Política de contraseña débil no habilitada
A3 - Sensitive Data Exposure	• Exposición de información sensible
A4 - XML External Entities (XXE)	No aplicó, debido a que su contenido no se enviaba por medio de XML.
A5 - Broken Access Control	No aplicó, debido a que sus mecanismos de control de acceso se encontraban bien configurados.
A6 - Security Misconfiguration	• Carga de Archivos Insegura (Tipo de Archivo) • Protocolo Obsoleto Habilitado (SSLV3) • Paginas por defecto habilitadas • Métodos HTTP no necesarios habilitados • Algoritmos Inseguros Habilitados • Protocolo Obsoleto Habilitado (TLSV1.1)

	• Protocolo Inseguro Habilitado (TLSV1.0) • Autocompletado Habilitado • Atributo HTTPOnly en Cookie no habilitada • Redirección de Sitios Insegura • Metadata en Archivos
A7 - Cross-Site Scripting (XSS)	• Secuencias de comando entre sitios (XSS-Almacenado)
A8 - Insecure Deserialization	No aplicó, debido a que no usaban objetos serializados.
A9 - Using Components with Known Vulnerabilities	• Servidor web con vulnerabilidades públicas • Uso de Componente con vulnerabilidades públicas
A10 - Insufficient Logging & Monitoring	No aplicó, debido a que se tienen establecidos logs, herramientas de traza y sesiones rastreadas.

Finalmente, mantener un medio seguro durante su uso es posible siguiendo las recomendaciones de la industria, estandarizando procesos y cumpliendo con las normas que se aplican en nuestro país a los medios tecnológicos. Asimismo, siempre se debe buscar la protección de los usuarios como prioridad ante cualquier desarrollo de un servicio, ya que toda la información es vital para la organización y existirán muchas amenazas relacionadas con ella, que buscarán atacar los activos de la organización con la finalidad de extraerlos y, posteriormente, venderlos o, en su caso, un consultor con ética definida puede reportar dicha debilidad a la organización.

Anexo A Carta de Planeación

Requerimientos de Prueba de Seguridad

Análisis de Vulnerabilidades en Aplicaciones Web

Autor

Hernández Hernández Alonso de Jesús

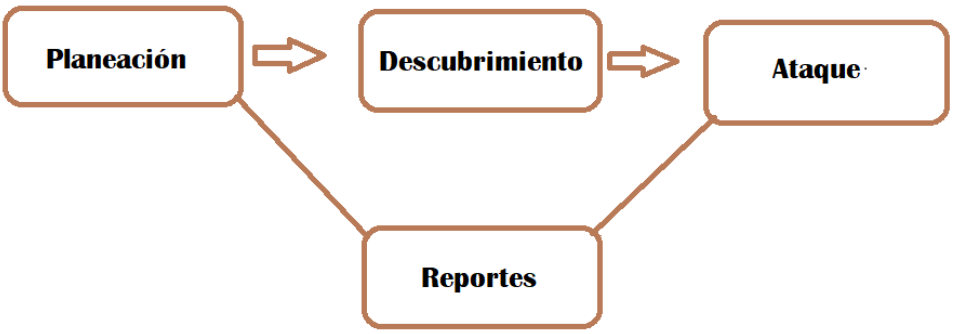
Declaración de Confidencialidad

Este documento es propiedad exclusiva de la institución. Este documento contiene información patentada, registrada y confidencial. La duplicación, redistribución o uso, total o parcial, en cualquier forma, requiere el consentimiento del responsable del documento. La institución puede compartir este documento con auditores bajo acuerdo o sociedad interna perteneciente a la organización con la finalidad de no divulgación para demostrar el cumplimiento de confidencialidad.

Descripción General

En julio de 2023, La institución aceptó la realización de la auditoria establecida en las fechas anteriores, para evaluar la postura de seguridad de su infraestructura y aplicación web de "Sistema de Registro" en comparación con las mejores prácticas actuales de la industria que incluyeron una prueba de penetración de forma externa y el tipo de prueba como caja gris. Todas las pruebas realizadas se basan en la guía técnica NIST SP 800-115 para pruebas y evaluación de seguridad de la información y el marco de clasificación OWASP TOP 10 2017.
Las fases de las actividades de la prueba de penetración incluyen lo siguiente:

- Planificación: Se recopilan los objetivos del cliente, se obtienen la carta de aceptación, se establecen restricciones, credenciales de prueba y las reglas de participación.
- Descubrimiento: Se realiza el análisis de reconocimiento y enumeración para identificar tecnologías, posibles vulnerabilidades, áreas débiles y explotaciones.
- Ataque: Se confirman las vulnerabilidades potenciales a través de la explotación o pruebas de concepto.
- Reportes: Se documentan todas las vulnerabilidades y explotaciones encontradas, los intentos fallidos y las fortalezas y debilidades de la empresa.



Clasificación de Vulnerabilidades

La siguiente table define niveles de severidad y el rango de puntuación de acuerdo con el OWASP Risk Rating, adicional se utiliza el OWASP Top Ten 2017 para evaluar la vulnerabilidad y el impacto del riesgo.

Severidad	OWASP Risk Rating	Definición
Critica	9.0-10.0	Explotación sencilla o extracción de información sensible y genera como resultado un compromiso al sistema, Se aconseja formar un plan de acción y parche inmediatamente
Alta	7.0-8.9	Explotación más difícil, pero podría causar privilegios elevados o una potencial perdida de datos, por mencionar algunos. Se recomienda formar un plan de acción y parche tan pronto como sea posible.
Moderada	4.0-6.9	Las vulnerabilidades existen, pero se requieren pasos adicionales, como herramientas o conocimientos de auditoría. Se recomienda formar un plan de acción y aplicar un parche después de resolver los de alta prioridad.
Baja	0.1-3.9	Las vulnerabilidades no escalan a una explotación alta, pero muestra información que pueda servir para un ataque posterior. Se recomienda formar un plan de acción y aplicar un parche en la próxima ventana de mantenimiento.
Informativa	N/A	No existe una vulnerabilidad, pero muestra información que pueda ayudar a identificar posibles brechas de seguridad. Se recomienda formar un plan de acción y realizarse en la próxima ventana de mantenimiento.

El propósito de este documento describe los requerimientos para el inicio de prueba de seguridad a la aplicación web, de acuerdo con las ventanas de prueba de julio del 2023. Para su ejecución del servicio se debe establecer línea directa entre los involucrados y el consultor, proporcionando información sobre los responsables y la aplicación a revisar.

Matriz de escalamiento, datos de contacto de responsables de la institución.

Requerimiento		Dato
Responsable del Área	Nombre completo	Luisa Jiménez Guzmán
	Puesto	CISO
	Correo electrónico	ljimenez.CISO@institucion.com.mx
Contacto Adicional	Nombre completo	Miguel Ortiz de Domínguez
	Puesto	Gerente de Seguridad
	Correo electrónico	m.ortiz@institucion.com.mx

Consultor, datos de contacto de responsables consultoría.

Requerimiento		Dato
Responsable de Consultoría	Nombre completo	Isabel Fernández Cantú
	Puesto	Coordinadora de Seguridad
	Correo electrónico	i.fernandez@auditoria.com.mx
Responsable de Auditoría	Nombre completo	Mario Fernán González
	Puesto	Consultor
	Correo electrónico	mario.fernan@consultor.com.mx

Información para la prueba de seguridad

Requerimiento		Dato
Aplicación	Nombre de la aplicación	Sistema de Registros
	URL / Enlace descarga	https://sistemaregis.edu.com.mx
	Lenguaje y/o frameworks utilizados y versión	4. PHP 7.4 5. MariaDB 10.4 6. Apache 2.4
	Restricciones de la prueba	No realizar DDoS
	Credenciales de prueba	Admin:4dm1n% Administrador User1:Us3rs= Solo Lectura

Nota: Toda información proporcionada, se registrará en el documento redactado y laprotección de dicha información se regirá mediante un contrato de confidencialidad proporcionado por la institución.

_____

Firma del responsable del Área

_____

Firma de responsable de Consultoría

_____

Firma de responsable de Auditoría

Anexo B Reporte Ejecutivo

REPORTE EJECUTIVO

Prueba de Penetración en Aplicaciones Web

Universidad Nacional Autónoma de México

Autor: Alonso de Jesús Hernández Hernández

Declaración de Confidencialidad

Este documento es propiedad exclusiva de la institución. Este documento contiene información patentada, registrada y confidencial. La duplicación, redistribución o uso, total o parcial, en cualquier forma, requiere el consentimiento del responsable del documento. La institución puede compartir este documento con auditores bajo acuerdo o sociedad interna perteneciente a la organización con la finalidad de no divulgación para demostrar el cumplimiento de confidencialidad.

Responsabilidad

Los hallazgos y recomendaciones se reflejan en la información recopilada durante la evaluación y no los cambios o modificaciones aplicados fuera del periodo de prueba de **Julio de 2023**. El compromiso por la búsqueda completa de todos los controles de seguridad, que la institución autorizo se vio reflejada en este documento. Se priorizó la evaluación para identificar controles de seguridad débiles que podrían ser explotados por un atacante. Por lo que se recomienda realizar evaluaciones similares anualmente o después de aplicar el plan de remediación por evaluadores interno o externos para garantizar el éxito continuo de los controles.

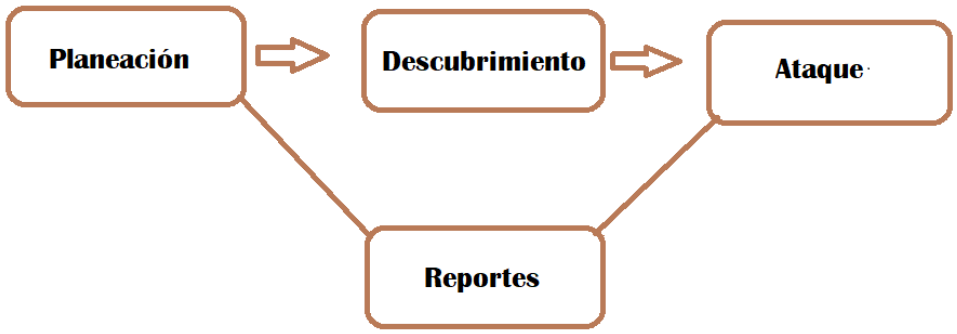
Información de Contacto

Nombre	Puesto	Información de Contacto
Institución		
Luisa Jiménez Guzmán	CISO	Email: ljimenez.CISO@institucion.com.mx
Miguel Ortiz de Domínguez	Gerente de Seguridad	Email: m.ortiz@institucion.com.mx
Consultoría		
Isabel Fernández Cantú	Coordinadora de Seguridad	Email: i.fernandez@auditoria.com.mx
Mario Fernán González	Consultor	Email: mario.fernan@consultor.com.mx

Descripción General

En julio de 2023, La institución acepto la realización de la auditoria establecida en las fechas anteriores, para evaluar la postura de seguridad de su infraestructura y aplicación web de "Sistema de Registro" en comparación con las mejores prácticas actuales de la industria que incluyeron una prueba de penetración de forma externa y el tipo de prueba como caja gris. Todas las pruebas realizadas se basan en la guía técnica NIST SP 800-115 para pruebas y evaluación de seguridad de la información y el marco de clasificación OWASP TOP 10 2017.
Las fases de las actividades de la prueba de penetración incluyen lo siguiente:

- Planificación: Se recopilan los objetivos del cliente, se obtienen la carta de aceptación, se establecen restricciones, credenciales de prueba y las reglas de participación.
- Descubrimiento: Se realiza el análisis de reconocimiento y enumeración para identificar tecnologías, posibles vulnerabilidades, áreas débiles y explotaciones.
- Ataque: Se confirman las vulnerabilidades potenciales a través de la explotación o pruebas de concepto.
- Reportes: Se documentan todas las vulnerabilidades y explotaciones encontradas, los intentos fallidos y las fortalezas y debilidades de la empresa.



Componentes de la evaluación
Prueba de Penetración Externa

Una prueba de penetración externa emula el papel de un atacante que intenta obtener acceso a una red interna, al ser de **caja gris** se proporcionaron recursos y conocimiento internos. El auditor intentará recopilar información a través de la consultoría para asegurar realmente están en uso las tecnologías compartidas anteriormente, las actividades se realizarán a través de inteligencia de código abierto (OSINT), se realizarán escaneos y enumeración para identificar posibles vulnerabilidades para su probable explotación.

Clasificación de Vulnerabilidades

La siguiente tabla define niveles de severidad y el rango de puntuación de acuerdo con el OWASP Risk Rating que se utiliza en todo el documento para evaluar la vulnerabilidad y el impacto del riesgo.

Severidad	OWASP Risk Rating	Definición
Critica	9.0-10.0	Explotación sencilla o extracción de información sensible y genera como resultado un compromiso al sistema, Se aconseja formar un plan de acción y parche inmediatamente
Alta	7.0-8.9	Explotación más difícil, pero podría causar privilegios elevados o una potencial perdida de datos, por mencionar algunos. Se recomienda formar un plan de acción y parche tan pronto como sea posible.
Moderada	4.0-6.9	Las vulnerabilidades existen, pero se requieren pasos adicionales, como herramientas o conocimientos de auditoría. Se recomienda formar un plan de acción y aplicar un parche después de resolver los de alta prioridad.
Baja	0.1-3.9	Las vulnerabilidades no escalan a una explotación alta, pero muestra información que pueda servir para un ataque posterior. Se recomienda formar un plan de acción y aplicar un parche en la próxima ventana de mantenimiento.
Informativa	N/A	No existe una vulnerabilidad, pero muestra información que pueda ayudar a identificar posibles brechas de seguridad. Se recomienda formar un plan de acción y realizarse en la próxima ventana de mantenimiento.

Alcance

Evaluación	Objetivo
Prueba de Penetración Externa	URL: https://sistemaregis.edu.com.mx IP: 192.168.80.216 Tecnologías: Apache 2.4.6, jQuery 3.6.1, OpenSSL 1.0.2k, PHP 7.4.33

Restricción

Por parte de la institución, no autoriza cualquier prueba relacionada a una denegación de servicio durante la ventana de prueba

Resumen Ejecutivo

En la consultoría se evaluó la postura de seguridad externa del portal “Sistema de Registro” a través de una prueba de penetración de red externa del **julio de 2023.** Al aprovechar una serie de ataques, en la consultoría se encontraron **1** vulnerabilidad de severidad **crítica, 5** vulnerabilidades de severidad **alta**, **9** vulnerabilidades de severidad **moderada**, **2 vulnerabilidades** de severidad **baja** y por último **1** vulnerabilidad de severidad **informativa**

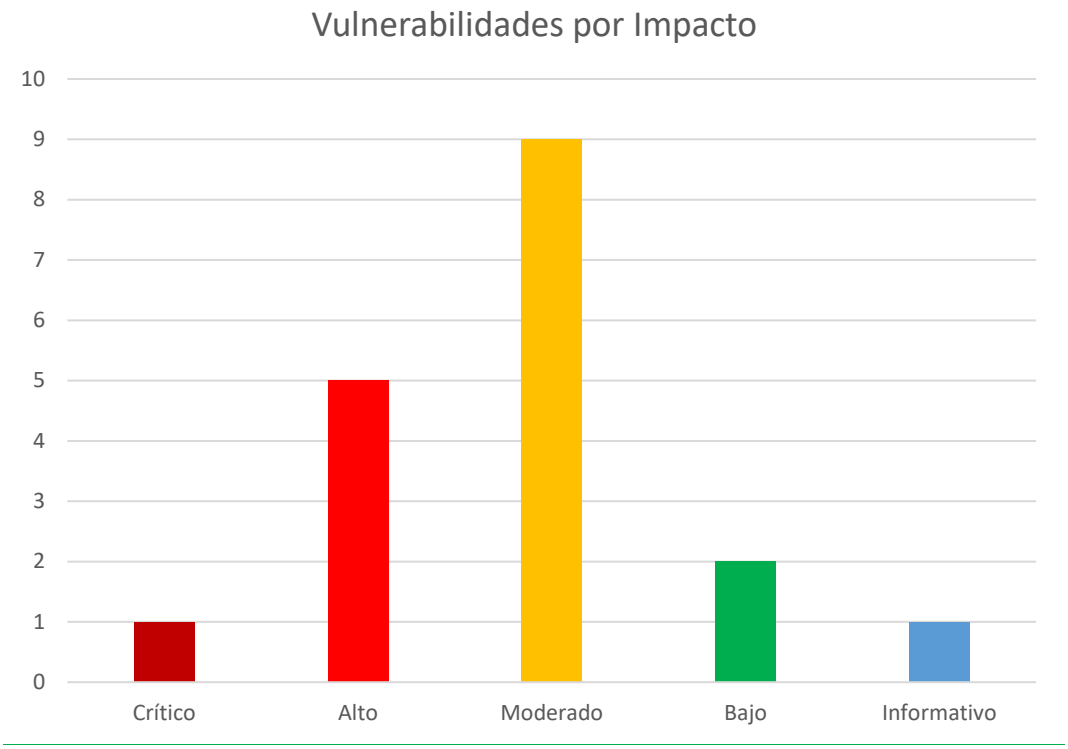
Resumen de Vulnerabilidades

La siguiente tabla da un resumen de vulnerabilidades encontradas:

#	Nombre	Severidad
1	Exposición de información sensible	Crítica
2	Secuencias de comando entre sitios (XSS-Almacenado)	Alta
3	Falsificación de solicitud entre sitios (CSRF)	Alta
4	Carga de Archivos Insegura (Tipo de Archivo)	Alta
5	Identificador en URL	Alta
6	Protocolo Obsoleto Habilitado (SSLV3)	Alta
7	Servidor web con vulnerabilidades públicas	Moderada
8	Uso de Componente con vulnerabilidades públicas	Moderada
9	Paginas por defecto habilitadas	Moderada
10	Métodos HTTP no necesarios habilitados	Moderada
11	Algoritmos Inseguros Habilitados	Moderada
12	Protocolo Obsoleto Habilitado (TLSV1.1)	Moderada
13	Protocolo Inseguro Habilitado (TLSV1.0)	Moderada
14	Política de contraseña débil no habilitada	Moderada
15	Autocompletado Habilitado	Moderada
16	Atributo HTTPOnly en Cookie no habilitada	Baja
17	Redirección de Sitios Insegura	Baja
18	Metadata en Archivos	Informativa

Vulnerabilidades por su impacto

La siguiente grafica ilustra las vulnerabilidades encontradas por impacto:



Anexo C Reporte Técnico

REPORTE TÉCNICO

Prueba de Penetración en Aplicaciones Web

Universidad Nacional Autónoma de México

Autor: Alonso de Jesús Hernández Hernández

Declaración de Confidencialidad

Este documento es propiedad exclusiva de la institución. Este documento contiene información patentada, registrada y confidencial. La duplicación, redistribución o uso, total o parcial, en cualquier forma, requiere el consentimiento tanto de la institución. Esta puede compartir este documento con auditores bajo acuerdo o sociedad interna perteneciente a la organización con la finalidad de no divulgación para demostrar el cumplimiento de confidencialidad.

Responsabilidad

Los hallazgos y recomendaciones se reflejan en la información recopilada durante la evaluación y no los cambios o modificaciones aplicados fuera del periodo de prueba del **Julio de 2023**. El compromiso por la búsqueda completa de todos los controles de seguridad, que SEA Acatlán autorizo se vio reflejada en este documento. Se priorizó la evaluación para identificar controles de seguridad débiles que podrían ser explotados por un atacante. Por lo que se recomienda realizar evaluaciones similares anualmente o después de aplicar el plan de remediación por evaluadores interno o externos para garantizar el éxito continuo de los controles.

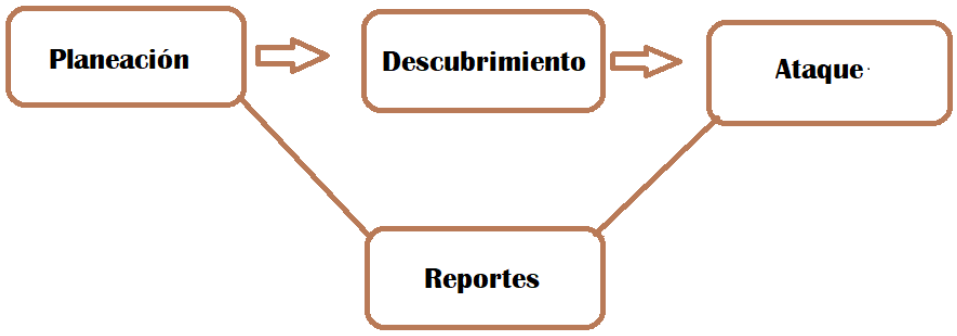
Información de Contacto

Nombre	Puesto	Información de Contacto
Institución		
Luis Jiménez Guzmán	CISO	Email: ljimenez.CISO@institucion.com.mx
Miguel Ortiz de Domínguez	Gerente de Seguridad	Email: m.ortiz@institucion.com.mx
Consultoría		
Isabel Fernández Cantú	Coordinadora de Seguridad	Email: i.fernandez@auditoria.com.mx
Mario Fernán González	Consultor	Email: mario.fernan@consultor.com.mx

Descripción General

Desde el Julio de 2023, la institución acepto la realización de la auditoria establecida en las fechas anteriores, para evaluar la postura de seguridad de su infraestructura y aplicación web de "Sistema de Registros" en comparación con las mejores prácticas actuales de la industria que incluyeron una a la prueba de penetración de forma externa y el tipo de prueba como caja gris. Todas las pruebas realizadas se basan en la guía técnica NIST SP 800-115 para pruebas y evaluación de seguridad de la información y el marco de clasificación OWASP TOP 10 2017. Las fases de las actividades de la prueba de penetración incluyen lo siguiente:

- Planificación: Se recopilan los objetivos del cliente, se obtienen la carta de aceptación, se establecen restricciones, credenciales de prueba y se obtienen las reglas de participación.
- Descubrimiento: Se realiza el análisis de reconocimiento y enumeración para identificar tecnologías, posibles vulnerabilidades, áreas débiles y explotaciones.
- Ataque: Se confirman las vulnerabilidades potenciales a través de la explotación o pruebas de concepto.
- Reportes: Se documentan todas las vulnerabilidades y explotaciones encontradas, los intentos fallidos y las fortalezas y debilidades de la empresa.



Componentes de la evaluación
Prueba de Penetración Externa

Una prueba de penetración externa emula el papel de un atacante que intenta obtener acceso a una red interna, al ser de caja gris se proporcionaron recursos y conocimiento internos. El auditor intentará recopilar información a través de la consultoría para asegurar el uso correcto de las tecnologías compartida, las actividades se realizarán a través de inteligencia de código abierto (OSINT), se realizarán escaneos y enumeración para identificar posibles vulnerabilidades para su probable explotación.

Clasificación de Vulnerabilidades

La siguiente table define niveles de severidad y el rango de puntuación de acuerdo con el OWASP Risk Rating que se utiliza en todo el documento para evaluar la vulnerabilidad y el impacto del riesgo.

Severidad	OWASP Risk Rating	Definición
Critica	9.0-10.0	Explotación sencilla y genera como resultado un compromiso al sistema, Se aconseja formar un plan de acción y parche inmediatamente
Alta	7.0-8.9	Explotación más difícil, pero podría causar privilegios elevados o una potencial perdida de datos, por mencionar algunos. Se recomienda formar un plan de acción y parche tan pronto como sea posible.
Moderada	4.0-6.9	Las vulnerabilidades existen, pero se requieren pasos adicionales, como herramientas o conocimientos de auditoría, por mencionar algunos. Se recomienda formar un plan de acción y aplicar un parche después de resolver los de alta prioridad.
Baja	0.1-3.9	Las vulnerabilidades no escalan a una explotación alta, pero muestra información que pueda servir para un ataque posterior. Se recomienda formar un plan de acción y aplicar un parche en la próxima ventana de mantenimiento.
Informativa	N/A	No existe una vulnerabilidad, pero muestra información que pueda ayudar a identificar posibles brechas de seguridad. Se recomienda formar un plan de acción y realizarse en la próxima ventana de mantenimiento.

Factores de Riesgo

El riesgo se mide por dos factores: probabilidad e impacto:

Probabilidad

La probabilidad mide el potencial de una vulnerabilidad que se pueda explotar. Las calificaciones se otorgan según la dificultad del ataque, las herramientas disponibles, el nivel de habilidad del atacante y el entorno del cliente.

Impacto

El impacto mide el efecto potencial de la vulnerabilidad en el negocio, incluye la confidencialidad, la integridad y la disponibilidad de los sistemas y/o datos del cliente, el daño a la reputación y la pérdida financiera.

Alcance

Evaluación	Objetivo
Prueba de Penetración Externa	URL: https://sistemaregis.edu.com.mx IP: 192.168.80.216 Tecnologías: Apache 2.4.6, jQuery 3.6.1, OpenSSL 1.0.2k, PHP 7.4.33

Restricción

Por parte de SEA Acatlán, no autoriza cualquier prueba relacionada a una denegación de servicio durante la ventana de prueba

NOTA: Por cuestiones de confidencialidad no se adjuntó evidencia y procedimientos que se realizaron en dichos escenarios

Resumen Ejecutivo

En la consultoría se evaluó la postura de seguridad externa del portal “Sistema de Registros” a través de una prueba de penetración de red externa del **Julio de 2023**. Al aprovechar una serie de ataques, en la consultoría se encontraron **1** vulnerabilidad de severidad **crítica, 5** vulnerabilidades de severidad **alta**, **9** vulnerabilidades de severidad **moderada**, **2 vulnerabilidades** de severidad **baja** y por último **1** vulnerabilidad de severidad **informativa**

Resumen de vulnerabilidad e informe.

Las siguientes tablas ilustran las vulnerabilidades encontradas por impacto y las soluciones recomendadas:

Resultados de la prueba de penetración

1	5	9	2	1
Crítica	Alta	Moderada	Baja	Informativa

La siguiente tabla da un resumen de vulnerabilidades encontradas:

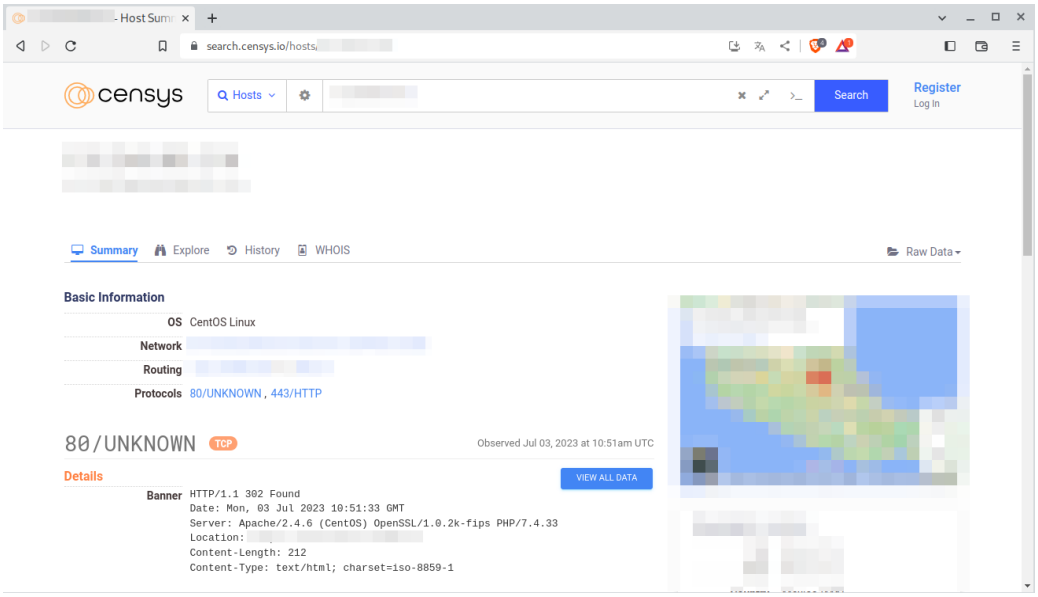
Nombre	Severidad	Ocurrencias
Exposición de información sensible	Crítica	1
Secuencias de comando entre sitios (XSS-Almacenado)	Alta	2
Falsificación de solicitud entre sitios (CSRF)	Alta	1
Carga de Archivos Insegura (Tipo de Archivo)	Alta	1
Identificador en URL	Alta	5
Protocolo Obsoleto Habilitado (SSLV3)	Alta	1
Servidor web con vulnerabilidades públicas	Moderada	1
Uso de Componente con vulnerabilidades públicas	Moderada	1
Paginas por defecto habilitadas	Moderada	20
Métodos HTTP no necesarios habilitados	Moderada	1
Algoritmos Inseguros Habilitados	Moderada	1
Protocolo Obsoleto Habilitado (TLSV1.1)	Moderada	1
Protocolo Inseguro Habilitado (TLSV1.0)	Moderada	1
Política de contraseña débil no habilitada	Moderada	1
Autocompletado Habilitado	Moderada	1
Atributo HTTPOnly en Cookie no habilitada	Baja	1
Redirección de Sitios Insegura	Baja	1
Metadata en Archivos	Informativa	1

Etapas de la prueba

Fase de Descubrimiento

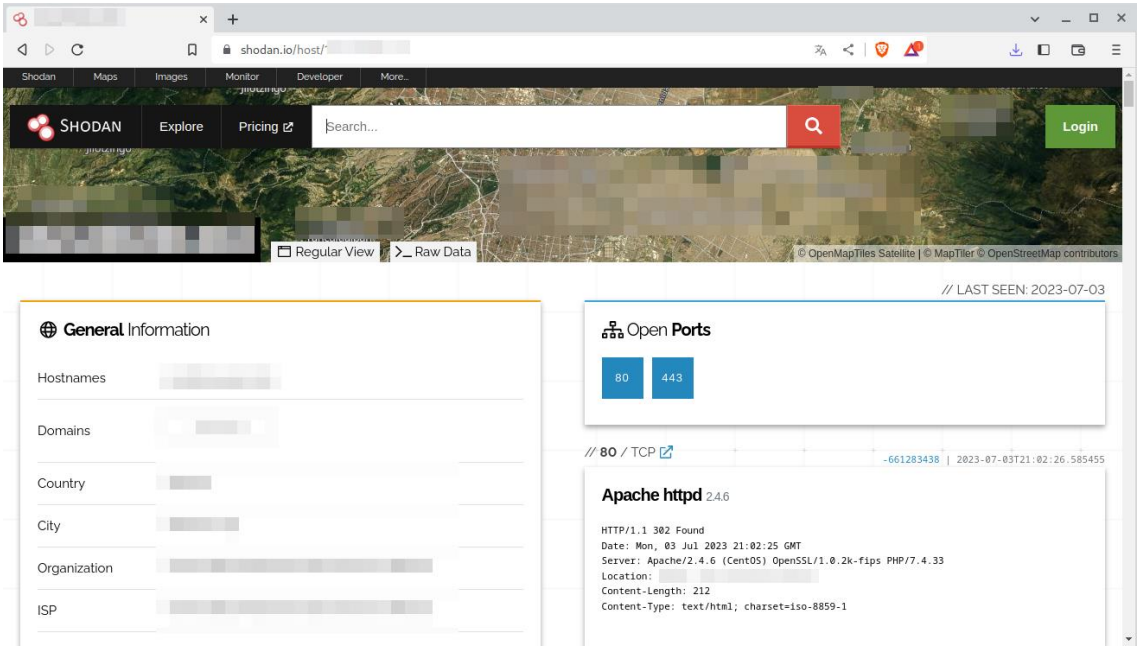
CENSYS

Se utilizó la herramienta para encontrar información de servicios o dispositivos relacionados al host, donde nuestro objetivo fue buscar las tecnologías, puertos abiertos, sistema operativo o cualquier tipo de información sin una interacción directa. En este caso, nos ayudó a identificar puertos, localización del host, país y el posible sistema operativo como se muestra a continuación:



SHODAN

Herramienta de reconocimiento pasivo de fuente abierta que nos proporciona información directa del host y sus configuraciones, así como tecnologías asociadas. En la siguiente imagen podemos observar que nos muestra información sobre posibles puertos abiertos, las cabeceras implementadas en la respuesta de una petición HTTP e información relacionada al dueño del dominio como lo es ciudad, país, organización al que está relacionado el host, por mencionar algunos.



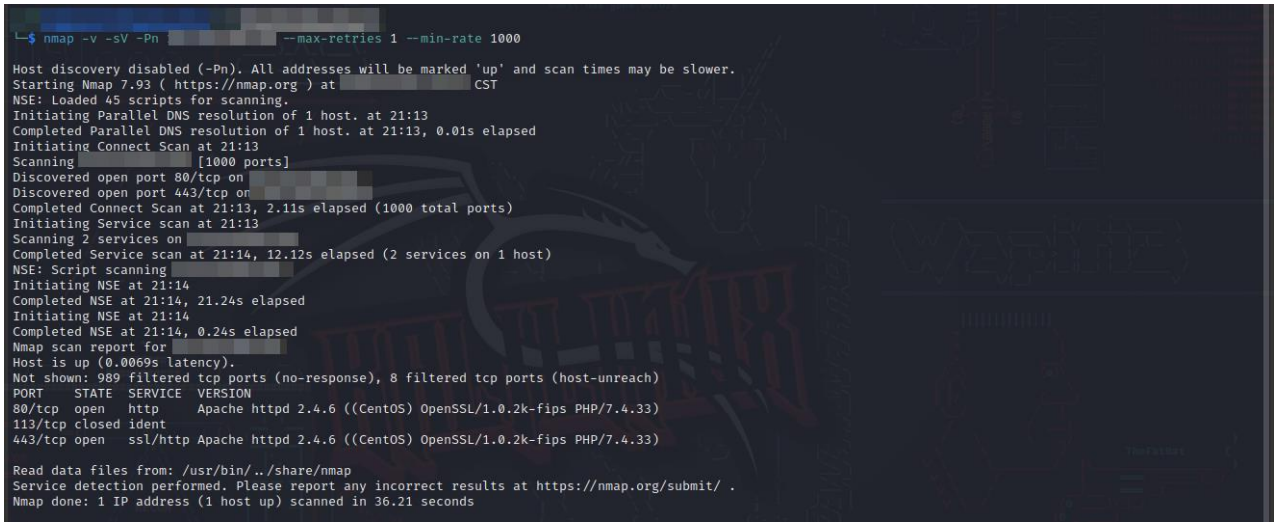
WHOIS

Esta herramienta de código abierto fue empleada durante la fase de reconocimiento y facilitó la recopilación de información vital sobre el propietario del dominio. Entre los datos obtenidos se encuentran el nombre completo del dueño, su correo electrónico oficial y su número telefónico, por mencionar algunos. La precisión y el detalle de los datos proporcionados por esta herramienta abrieron nuevas oportunidades para la identificación en futuras interacciones de manera estratégica.

Fase de Ejecución

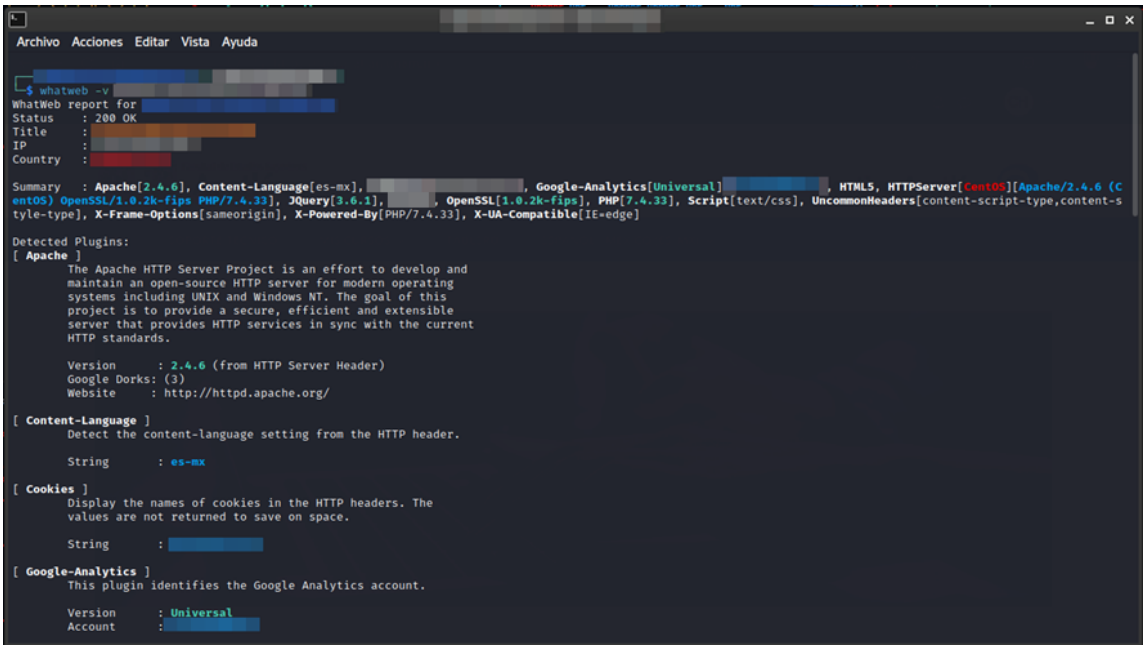
NMAP

Network Mapper, es una herramienta que se basa en comandos de Linux de código abierto, al encontrar posible información de puertos y el sistema operativo del host, con las herramientas anteriormente usadas, se empleó el uso de NMAP para realizar un escaneo activo a la IP de la aplicación web, así extraer los servicios y puertos de red utilizados, para validar que en realidad se encuentran abiertos dichos puertos que nos arrojaron las fuentes abiertas, su uso es como se muestra en la siguiente imagen.



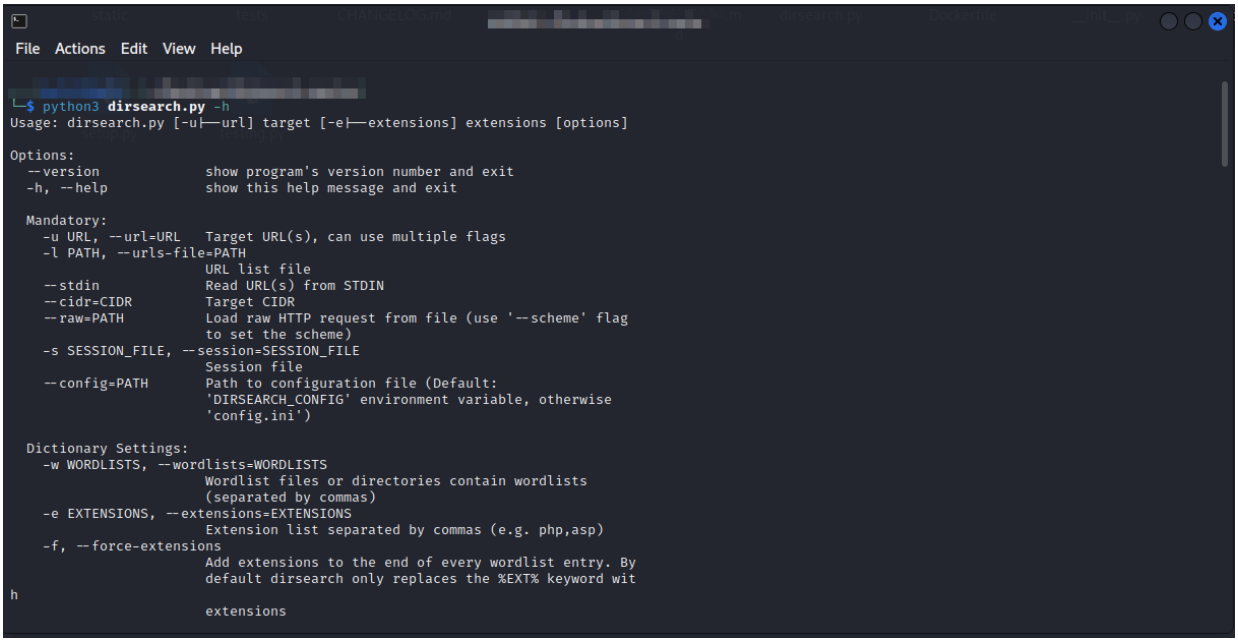
WHATWEB

Herramienta de código abierto que ayudó a la validación de la tecnología proporcionada por la institución, ya que nos permite recopilar todo sobre los recursos tecnológicos implementados y sus versiones. Escanea toda la información sobre complementos y muestra gran parte de los detalles de la aplicación, dependiendo de la configuración del servidor. En este caso al validar el tipo de versiones y observar que es la misma del documento de planeación, podemos buscar debilidades asociadas a las tecnologías ya que logramos obtener el nombre y su versión, con ayuda de las referencias de vulnerabilidades.



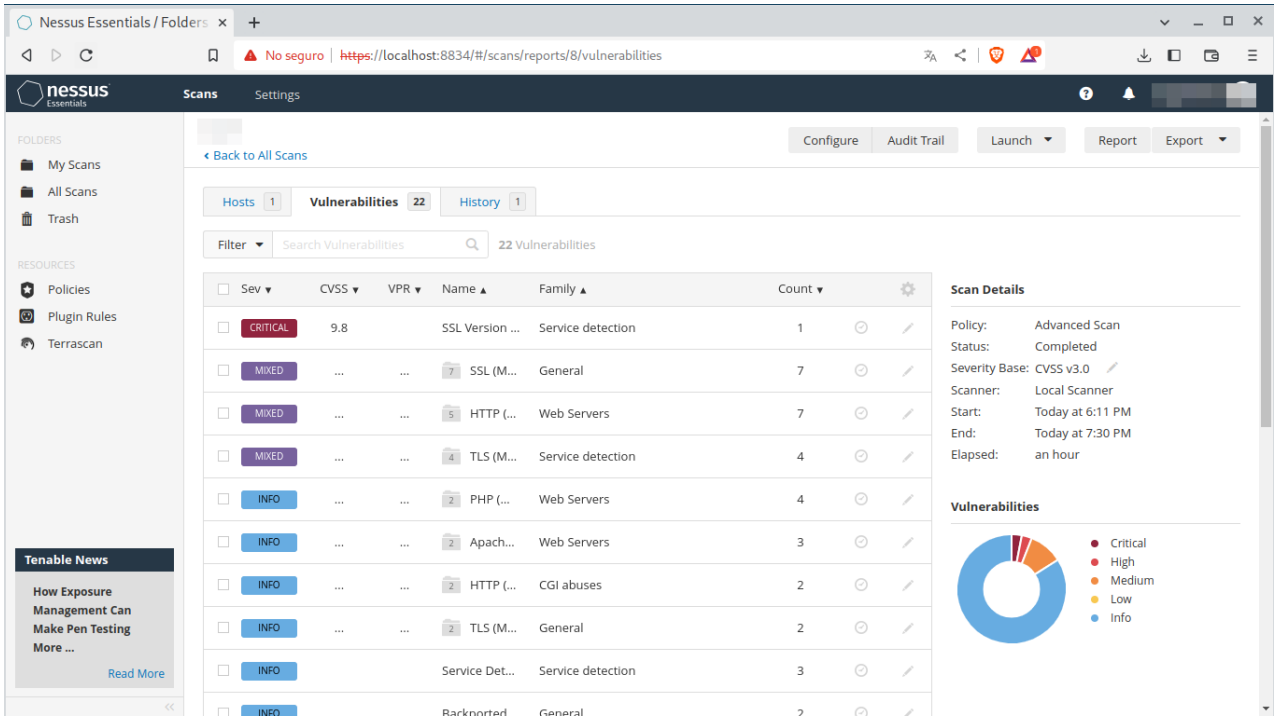
Dirsearch

Herramienta escrita en Python, que se utilizó para realizar un escaneo automatizado de directorios en la aplicación web, el método usado fue mediante un diccionario de posibles palabras para realizar la búsqueda de recursos. Esta herramienta nos permite buscar información o páginas por defecto habilitadas, en este caso logramos encontrar páginas de configuración XML, información sensible almacenada del lado del cliente, entre otras.



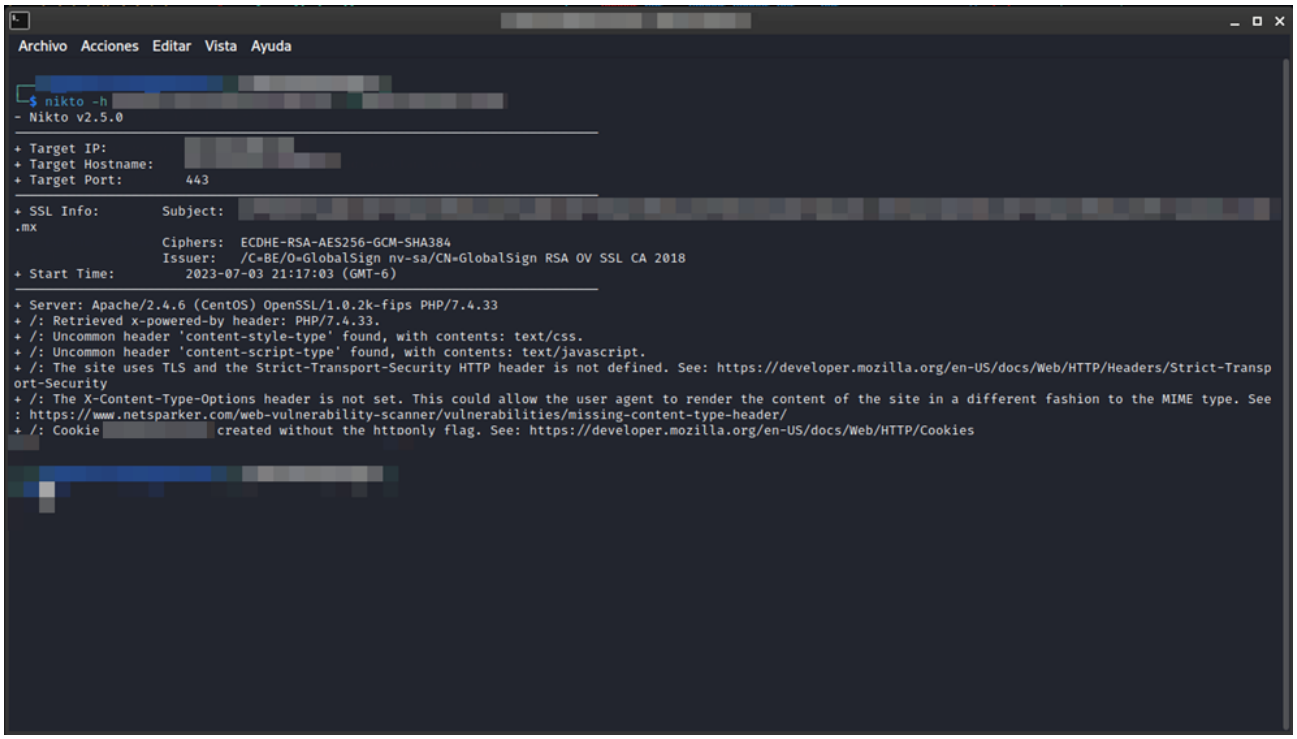
NESSUS

Esta herramienta especializada se seleccionó para llevar a cabo el escaneo de vulnerabilidades en el host. Su principal objetivo es detectar y analizar exhaustivamente todos los servicios y puertos activos en los sistemas, así como evaluar las configuraciones de seguridad en busca de debilidades que puedan representar riesgos potenciales. La identificación de estos radica en su extensa base de datos de firmas, que contiene una amplia gama de plugins y módulos diseñados para detectar vulnerabilidades conocidas y emergentes.



NIKTO

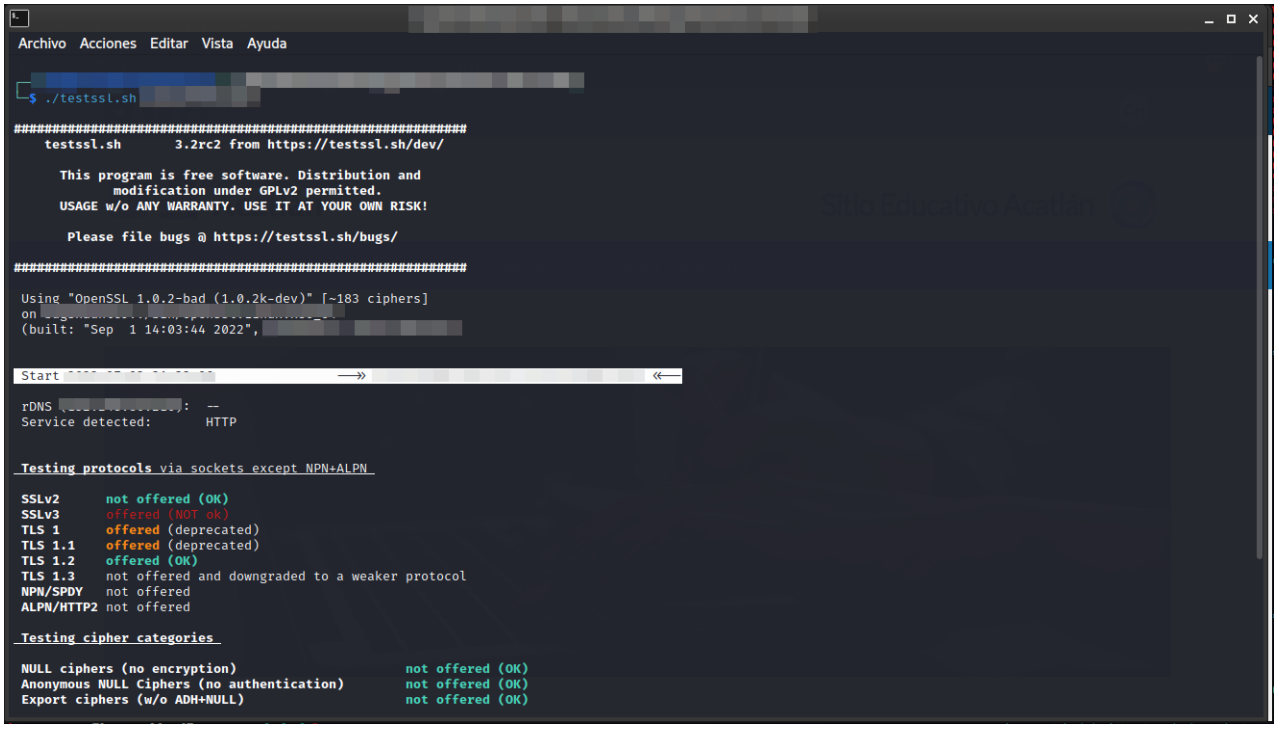
Herramienta de software libre con la finalidad de escanear vulnerabilidades, configuraciones inseguras o exposición de información crítica, así como reconocer firmas CVE conocidas o almacenadas en la herramienta con base en un comportamiento específico. Adicionalmente, funciona como un buscador de directorios para "Fuzzing", la cual nos permitió identificar páginas con configuración incorrecta, vulnerabilidades asociadas a sus tecnologías y por último exposición de información mediante sus cabeceras HTTP.



Fase de Explotación

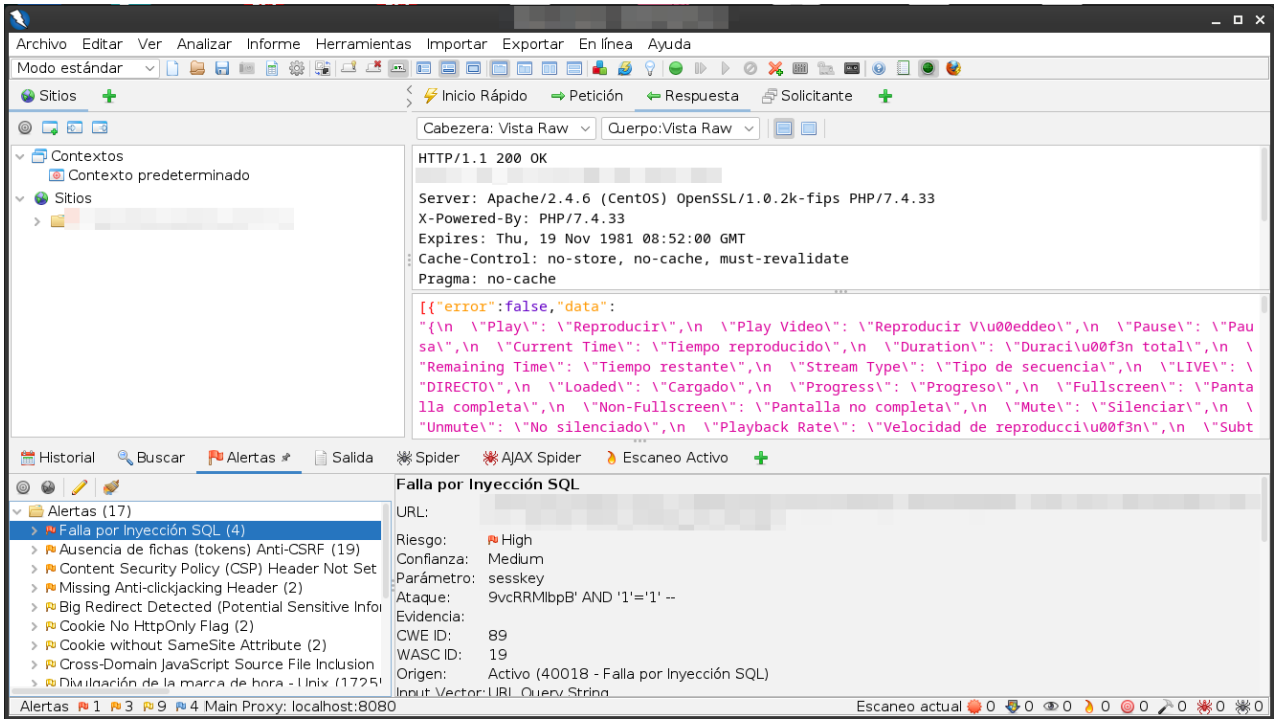
TESTSSL

Herramienta basada en bash de línea de comando que verifica los certificados de comunicaciones en el servicio de cualquier puerto, para el soporte de cifrados TLS/SSL y fallos criptográficos. Realiza la identificación y comprueba si no es vulnerable ante las firmas comunes de CVE. En este caso, se realizó el escaneo de protocolos en la comunicación del puerto 443, el cual permitió la identificación de protocolos inseguros, obsoletos y algoritmos, asociándola a una firma CVE respectivamente.



OWASP ZAP

Herramienta de escaneo de vulnerabilidades de forma automatizada o manual, que puede realizarse de forma activa, en la cual se lleva a cabo una exploración de directorios de acuerdo con la opción seleccionada. También puede realizarse de forma manual, utilizando esta herramienta como un proxy para la edición de peticiones HTTP. Nos permitió la búsqueda de directorios, exploración de la aplicación web, recursos críticos, encontrar debilidades, configuraciones incorrectas, cabeceras que exponen información, etcétera. Se consideró como una herramienta de complemento para la validación de vulnerabilidades asociadas a la aplicación web, facilidad de descubrimiento y explotación.



Glosario

A

Acceso: Control y gestión de permisos y derechos que tienen los usuarios y sistemas para acceder a recursos, datos o servicios dentro de un entorno informático.

Acciones de mitigación: Estrategias y medidas que se implementan para reducir o limitar los riesgos y el impacto de posibles amenazas o incidentes de seguridad.

Acciones preventivas: Medidas proactivas que se implementan para evitar o reducir la probabilidad de que ocurran amenazas o incidentes de seguridad antes de que puedan materializarse.

Acción maliciosa: Cualquier acción intencional llevada a cabo por individuos malintencionados o "actores maliciosos" con el objetivo de causar daño, comprometer la seguridad de sistemas o redes, robar información confidencial o realizar actividades ilegales.

Activos: Son a los recursos valiosos y críticos que posee una organización o entidad, y que son objeto de protección frente a posibles amenazas y riesgos cibernéticos.

Amenaza: Es cualquier evento, acción o entidad que tiene el potencial de causar daño, explotar vulnerabilidades o comprometer la seguridad de los activos, sistemas o datos de una organización o individuo.

Ambiente: Conjunto de elementos y condiciones que componen el entorno tecnológico en el cual se desarrollan, operan y protegen los sistemas, redes, datos y activos de una organización.

Análisis: Proceso de examinar, evaluar y estudiar detalladamente diferentes aspectos relacionados con la seguridad de sistemas, redes o datos con el objetivo de comprender, identificar y resolver problemas o posibles vulnerabilidades.

Anomalías: Se refieren a desviaciones o desajustes inusuales que se detectan dentro de un sistema, red o conjunto de datos en comparación con un comportamiento considerado normal o esperado.

Aplicaciones móviles: Programas de software diseñados específicamente para ser utilizados en dispositivos móviles, como teléfonos inteligentes y tabletas.

Aplicaciones web: Programas de software que se ejecutan en un servidor web y se acceden a través de un navegador web en dispositivos como computadoras, teléfonos inteligentes o tableta.

Ataque informático: Es una acción maliciosa o intento deliberado realizado por personas o sistemas con el objetivo de comprometer, dañar, explotar o interferir con la seguridad de sistemas, redes, datos o activos de una organización o individuo.

B

Banderas lógicas: Variables o marcadores utilizados en el código de un programa o sistema para indicar o activar ciertas condiciones o comportamientos específicos durante su ejecución.

Brecha de seguridad: Situación en la cual se produce una vulneración o violación en las medidas de seguridad de una organización, lo que permite el acceso no autorizado a datos, sistemas, redes o activos valiosos.

Bypass: Término que se refiere a la acción de evitar o eludir medidas de seguridad o restricciones para acceder a recursos, sistemas o datos sin cumplir con las condiciones establecidas.

C

Clasificación: Proceso de categorizar o etiquetar información y recursos en función de su nivel de confidencialidad, sensibilidad o importancia para una organización.

Clave: Secuencia de caracteres que se utiliza para cifrar o descifrar datos en un proceso de criptografía.

Código: Conjunto de instrucciones y comandos escritos en lenguaje de programación que forman parte de un programa o software.

Configuración por defecto: Es el ajuste que se establece automáticamente si el usuario no realiza ningún cambio o personalización durante el proceso de instalación o puesta en marcha.

Confidencialidad: Propiedad que garantiza que la información y los datos se mantengan privados y sean accesibles únicamente por personas o entidades autorizadas.

Copias de seguridad: Práctica de realizar duplicados de los datos, archivos y sistemas críticos almacenados en un dispositivo o servidor con el propósito de protegerlos contra pérdida de información, daños, fallas técnicas o ataques cibernéticos.

Consultor: Profesional o entidad independiente que realiza evaluaciones y revisiones sistemáticas de los sistemas, políticas, controles y prácticas de seguridad de una organización para verificar su cumplimiento con estándares, regulaciones, mejores prácticas y políticas internas.

Credenciales: Información de identificación que se utiliza para autenticar y verificar la identidad de un usuario o entidad que intenta acceder a un sistema, red, aplicación o servicio.

Criptografía: Disciplina de seguridad informática que se ocupa del estudio y desarrollo de técnicas y algoritmos para proteger la información y los datos mediante el uso de transformaciones matemáticas.

D

Datos: La información digital o electrónica que se almacena, procesa, transmite o utiliza en sistemas informáticos y aplicaciones. Los datos pueden ser de diferentes tipos, incluyendo texto, imágenes, videos, audio, bases de datos, documentos, correos electrónicos y cualquier otra forma de información digital.

Debilidad: Cualquier fallo o punto vulnerable en un sistema, aplicación, red o infraestructura que podría ser explotado por un atacante para comprometer la seguridad y permitir un acceso no autorizado o causar daños.

Desarrollo: Creación, diseño y construcción de sistemas, aplicaciones y software que cumplen con los requisitos funcionales y de seguridad establecidos.

Disponibilidad: Asegurar que los sistemas, aplicaciones, datos y recursos estén accesibles y funcionando correctamente cuando sean necesarios por los usuarios autorizados.

Dispositivos: Cualquier equipo o aparato electrónico utilizado para procesar, almacenar, transmitir o acceder a datos e información.

E

Entorno de prueba: Ambiente de pruebas o sandbox se refiere a un entorno controlado y aislado utilizado para probar aplicaciones, software o sistemas sin poner en riesgo el ambiente de producción o el entorno real donde se encuentran los datos y sistemas en funcionamiento.

Escaneo: Técnica utilizada para analizar y detectar posibles vulnerabilidades y debilidades en sistemas, redes o aplicaciones

Entrada de datos: Proceso mediante el cual se ingresan datos o información en un sistema, aplicación o dispositivo electrónico.

Error: Refiere a un defecto o problema en un sistema, aplicación, código o configuración que puede provocar un comportamiento no deseado o inesperado.

Evaluación: Proceso de examinar, analizar y valorar la seguridad de sistemas, redes, aplicaciones o infraestructuras con el objetivo de identificar posibles debilidades, vulnerabilidades o áreas de mejora-

Evidencia: Es cualquier tipo de información, datos o registros que se utilizan para demostrar o respaldar la existencia de un evento, acción o incidente relacionado con la seguridad de la información y sistemas.

Excepción: Una situación especial o un caso atípico en el cual se permite una desviación o una acción diferente de lo que se considera normal o permitido dentro de una política, norma o regla de seguridad establecida.

Estándar: Conjunto de normas, directrices, protocolos o reglas que se establecen para garantizar la seguridad, interoperabilidad y buenas prácticas en los sistemas, redes, aplicaciones o dispositivos electrónicos.

Escenario: Conjunto de situaciones hipotéticas o casos prácticos que se utilizan para analizar, planificar y probar la respuesta a diferentes eventos de seguridad.

Explotación: Acción de aprovechar una vulnerabilidad o debilidad en un sistema, red o aplicación con el fin de realizar acciones maliciosas o no autorizadas.

F

Fallo: Defecto o error en un sistema, aplicación, software o hardware que puede dar lugar a una vulnerabilidad o debilidad en la seguridad.

Firewall: Barrera de seguridad que se utiliza para proteger una red o sistema informático al controlar y filtrar el tráfico de datos que entra y sale de la red.

Funcionalidad: Capacidad y el conjunto de características que posee un sistema, aplicación, software o dispositivo para realizar tareas específicas o proporcionar servicios.

G

Gestión: Planificación, organización, implementación y supervisión de medidas y estrategias destinadas a proteger los activos, datos e información de una organización contra posibles amenazas y riesgos cibernéticos.

H

Herramienta: Software, aplicación o utilidad que se utiliza para ayudar en la identificación, análisis, protección, detección o respuesta a amenazas y vulnerabilidades de seguridad en sistemas y redes informáticas.

I

IEC (International Electrotechnical Commission): Organización internacional de normalización que se encarga de establecer estándares en el campo de la electrónica, la electricidad y las tecnologías relacionadas.

Impacto: Consecuencias y repercusiones que pueden surgir como resultado de un evento de seguridad, un ataque cibernético o una brecha de seguridad en un sistema, red, aplicación o infraestructura tecnológica.

Incidente: Cualquier evento o suceso que representa una amenaza, violación o intento no autorizado de acceder, modificar, robar, destruir o interferir con los sistemas, datos o recursos de una organización o individuo.

Información: Datos, conocimientos o detalles que se generan, almacenan, transmiten o procesan en sistemas y redes informáticas.

Infraestructura: Conjunto de recursos físicos, lógicos y tecnológicos que conforman una red o sistema informático.

Inyección de código: Técnica de ataque en seguridad informática que consiste en introducir código malicioso o no autorizado en una aplicación, sistema o sitio web con el objetivo de alterar su comportamiento, obtener información confidencial o comprometer su integridad.

Integridad: Calidad o estado de los datos, sistemas o recursos informáticos que asegura que estos no han sido alterados, manipulados o corrompidos de manera no autorizada o accidental

Intrusión: Cualquier intento no autorizado y malicioso de acceder, interferir o comprometer un sistema, red o recurso informático.

ISO (International Organization for Standardization): La ISO es una organización internacional independiente que se encarga de establecer normas y estándares en diversas áreas, incluida la seguridad de la información.

L

Licencia: Contrato legal que establece los términos y condiciones bajo los cuales se permite el uso de un software, aplicación o recurso informático específico.

M

Mapeo de red: Proceso de descubrir y visualizar la topología y estructura de una red informática.

Materializar: Acto relacionado a tiempo que indica que va a suceder o se efectuó una consecuencia provocada por acciones.

Medidas de seguridad: Son las acciones, procedimientos y controles que una organización o individuo implementa para proteger sus sistemas, redes, datos y activos de posibles riesgos y amenazas cibernéticas.

Medios tecnológicos: Los recursos y herramientas tecnológicas utilizados para proteger y fortalecer la seguridad de la información, sistemas, redes y activos digitales.

Metodología: Enfoque sistemático y estructurado para llevar a cabo actividades relacionadas con la seguridad de la información.

Métodos de autenticación: Técnicas y procedimientos utilizados para verificar y confirmar la identidad de un usuario o entidad que intenta acceder a un sistema, red o recurso informático.

Matriz: Es una tabla o estructura que representa los permisos y derechos que tienen los usuarios o grupos de usuarios a diferentes acciones en un sistema o red.

Monitoreo: Proceso de supervisar y observar de manera continua y sistemática los sistemas, redes, aplicaciones y recursos informáticos para identificar posibles amenazas, anomalías o incidentes de seguridad.

N

NIST (National Institute of Standards and Technology): Instituto Nacional de Estándares y Tecnología de los Estados Unidos. Es una agencia federal del Departamento de Comercio de los Estados Unidos que se dedica a desarrollar y promover estándares, guías y buenas prácticas en diversos campos, incluyendo la seguridad informática y la ciberseguridad.

O

Organización: Una entidad o empresa que opera sistemas informáticos, redes y recursos tecnológicos para llevar a cabo sus actividades comerciales u operativas.

OSINT (Open Source Intelligence): La recopilación, análisis y uso de información proveniente de fuentes abiertas y públicas disponibles en línea para obtener conocimientos relevantes y valiosos.

P

Parche de seguridad: Una actualización o corrección de software que se implementa para resolver vulnerabilidades o problemas de seguridad en un sistema, aplicación o dispositivo.

Plan de contención: Conjunto de estrategias, políticas y procedimientos diseñados para contener y mitigar los efectos de un incidente de seguridad una vez que ha sido detectado.

Phishing: Ataque cibernético donde se busca engañar a las personas para que revelen información personal confidencial, como contraseñas, números de tarjetas de crédito, números de seguridad social u otra información sensible.

Políticas: Conjunto de reglas, directrices, normas y principios que establecen las pautas y comportamientos que deben seguirse en una organización para proteger la seguridad de la información y los sistemas.

Prueba de seguridad: Proceso de evaluación y análisis que se lleva a cabo para identificar y evaluar las vulnerabilidades y debilidades de seguridad presentes en sistemas, aplicaciones, redes o infraestructuras tecnológicas de una organización.

Probabilidad: Posibilidad de que ocurra un evento o incidente de seguridad específico.

Producción: Entorno o fase operativa donde los sistemas, aplicaciones y servicios informáticos están en pleno funcionamiento y son utilizados por los usuarios finales o clientes.

Proceso: Serie de acciones, pasos o actividades ordenadas y coordinadas que se llevan a cabo para lograr un objetivo específico relacionado con la seguridad de la información y los sistemas.

Procedimientos: Instrucciones detalladas y secuenciales que describen cómo llevar a cabo tareas específicas relacionadas con la seguridad de la información y los sistemas.

Puertos: Canales de comunicación específicos que se utilizan para transmitir datos entre dispositivos y aplicaciones en una red de computadoras.

R

Recomendaciones: Sugerencias y directrices proporcionadas para mejorar la seguridad de la información y los sistemas en una organización.

Remediación: Proceso de identificar, abordar y corregir las vulnerabilidades, debilidades o problemas de seguridad que se han detectado en los sistemas, aplicaciones o infraestructuras de una organización.

Reporte: Documento o informe detallado que contiene información relevante sobre eventos, incidentes, actividades, hallazgos o resultados relacionados con la seguridad de la información y los sistemas.

Respaldar: Proceso de crear una réplica de los datos, archivos y configuraciones importantes almacenados en un sistema o dispositivo.

Riesgo: Probabilidad de que ocurra un incidente o evento no deseado que pueda tener un impacto negativo en la confidencialidad, integridad o disponibilidad de los activos de información y sistemas de una organización.

Red: Conjunto de dispositivos electrónicos (como computadoras, servidores, enrutadores, conmutadores, impresoras, teléfonos, etcétera.) que están interconectados para compartir datos, recursos y servicios.

S

Servicios: Funcionalidades, aplicaciones o procesos que se ejecutan en una red, sistema o dispositivo para proporcionar determinadas funciones o recursos a los usuarios o a otros sistemas.

Servidores: Tipo de computadora o sistema de computación diseñado para proporcionar servicios, recursos y funcionalidades a otras computadoras o dispositivos conectados a una red.

Severidad: Nivel de gravedad o impacto que tiene un incidente de seguridad o una vulnerabilidad en un sistema o red.

SGSI (Sistema de Gestión de Seguridad de la Información): Es un enfoque integral para administrar y proteger la seguridad de la información en una organización.

Simulación: Representación de situaciones, eventos o escenarios de seguridad de la información mediante la imitación de condiciones reales, pero en un entorno controlado.

Sistema informático: Conjunto de componentes físicos y lógicos que trabajan de manera conjunta para procesar, almacenar y transmitir información en formato digital.

Sistemas operativos: Tipo de software fundamental que actúa como intermediario entre el hardware de una computadora y las aplicaciones y programas que se ejecutan en ella.

Software: Los programas y aplicaciones informáticas que se ejecutan en una computadora o dispositivo electrónico

T

Tecnología: Conjunto de conocimientos, técnicas, herramientas, equipos y procesos utilizados para diseñar, crear, operar y mejorar productos y servicios, así como para resolver problemas y satisfacer necesidades humanas.

Texto codificado: Cadena de caracteres que ha sido transformado mediante algún algoritmo o método de codificación para hacerlo ilegible o ininteligible para personas que no posean la clave o el conocimiento necesario para descifrarlo.

TICs (Tecnologías de la Información y Comunicación): Conjunto de tecnologías, herramientas, dispositivos y recursos que se utilizan para procesar, almacenar, transmitir y compartir información de manera digital.

Topología: Disposición o estructura física y lógica de una red de computadoras. Describe cómo están conectados los dispositivos y cómo se comunican entre sí dentro de la red.

Transmisión de información: Proceso de enviar datos y comunicaciones electrónicas entre diferentes dispositivos o sistemas a través de una red, ya sea local o global.

Triada de la seguridad: Tres principios clave que deben ser considerados y equilibrados para proteger adecuadamente los recursos de información en cualquier sistema o red.

V

Vectores de ataque: Diferentes formas o vías a través de las cuales un atacante puede comprometer un sistema, red o dispositivo para obtener acceso no autorizado, robar información sensible, dañar o interferir con el funcionamiento normal de los recursos informáticos.

Ventana de pruebas: Período de tiempo específico durante el cual se realizan pruebas de seguridad en un sistema, aplicación o red antes de su implementación en producción o puesta en marcha.

Verificación de seguridad: Proceso sistemático y exhaustivo de evaluación y análisis de los sistemas, procedimientos y controles de seguridad de una organización con el objetivo de verificar su efectividad, identificar posibles vulnerabilidades y asegurarse de que se cumplan los requisitos de seguridad y las mejores prácticas.

Vulnerabilidad: Debilidad o fallo en un sistema, aplicación, red o dispositivo que puede ser explotada por un atacante para comprometer la seguridad y obtener acceso no autorizado, dañar o interferir con el funcionamiento normal de los recursos informáticos.

Índice de Figuras

Figura 1 Tipos de pruebas de penetración 19
Figura 2 Relación probabilidad e impacto 29
Figura 3 Comunicación HTTP 54
Figura 4 Esquema página estática 64
Figura 5 Esquema página dinámica 65
Figura 6 OWASP Calculator 87
Figura 7 Esquema de metodología NIST 110
Figura 8 Gráfica de vulnerabilidades 116

Índice de Tablas

Tabla 1 Ejemplos de Amenazas 11
Tabla 2 Ejemplos de Riesgos 12
Tabla 3 Funcionamiento HTTPS 60
Tabla 4 Generaciones de sitios web 63
Tabla 5 Comparativo OWASP TOP 10 74
Tabla 6 OWASP TOP 10 A1 Inyección 75
Tabla 7 OWASP TOP 10 A2 Pérdida de autenticación 76
Tabla 8 OWASP TOP 10 A3 Exposición de datos sensibles 77
Tabla 9 OWASP TOP 10 A4 Entidades externas XML (XXE) 78
Tabla 10 OWASP TOP 10 A5 Pérdida de control de acceso 78
Tabla 11 OWASP TOP 10 A6 Configuración de seguridad incorrecta 79
Tabla 12 OWASP TOP 10 A7 Secuencias de comandos entre sitios (XSS) 80
Tabla 13 OWASP TOP 10 A8 Deserialización Insegura 81
Tabla 14 OWASP TOP 10 A9 Uso de componentes con vulnerabilidades conocidas 81
Tabla 15 OWASP TOP 10 A10 Registro y Monitoreo Insuficientes 82
Tabla 16 Tabla de información proporcionada 109
Tabla 17 Conocimiento base asociada al reconocimiento obtenido 112
Tabla 18 Conocimiento base asociada a las vulnerabilidades encontradas 114
Tabla 19 Tabla de relación TTPs y vulnerabilidades 116
Tabla 20 Vulnerabilidad - Exposición de información sensible 120
Tabla 21 Vulnerabilidad - Secuencias de comando entre sitios (XSS-Almacenado) 121
Tabla 22 Vulnerabilidad - Falsificación de solicitud entre sitios (CSRF) -> Alta 121
Tabla 23 Vulnerabilidad - Carga de Archivos Insegura (Tipo de Archivo) 122
Tabla 24 Vulnerabilidad - Identificador en URL 122
Tabla 25 Vulnerabilidad - Protocolo Obsoleto Habilitado (SSLV3) 123
Tabla 26 Vulnerabilidades - Servidor web con vulnerabilidades públicas 123
Tabla 27 Vulnerabilidad - Componente con vulnerabilidades públicas 124
Tabla 28 Vulnerabilidad - Páginas por defecto habilitadas 124
Tabla 29 Vulnerabilidad - Métodos HTTP no necesarios habilitados 125
Tabla 30 Vulnerabilidad - Algoritmos Inseguros Habilitados 125
Tabla 31Vulnerabilidad - Protocolo Obsoleto Habilitado (TLSV1.1) 126
Tabla 32 Vulnerabilidad - Protocolo Inseguro Habilitado (TLSV1.0) 126
Tabla 33 Vulnerabilidad - Política de contraseña debilidades no habilitada 127
Tabla 34 Vulnerabilidad - Autocompletado Habilitado 127
Tabla 35 Vulnerabilidad - Atributo HTTPOnly en Cookie no habilitada 128
Tabla 36 Vulnerabilidad - Redirección de Sitios Insegura 128
Tabla 37 Vulnerabilidad - Metadata en Archivos 129
Tabla 38 Tabla de Fortalezas 130
Tabla 39 Relación NIST con caso de estudio 132

Tabla 40 Tabla de relación OWASP y hallazgos encontrados 133

Índice de Formulas

Fórmula 1. Nomenclatura del Riesgo 13
Formula 2 Nomenclatura CVE 88
Formula 3 Nomenclatura CWE 88
Formula 4 Nomenclatura NIST 89

Referencias

Numero de pie de pagina	Referencia
2	Aguilera, P. (2011). *Redes Seguras (Seguridad informática).* Madrid, España: Editex.
7	Escrivá, R. (2013). *Seguridad informática.* Madrid,España: MacMillan.
12	Nacional, C. C. (2010). GUÍA DE SEGURIDAD DE LAS TIC . Editor y Centro Criptológico Nacional.
14,18,22	Manual, O. S. (2010). *OSSTMM 3 – The Open Source Security Testing Methodology Manual.* ISECOM.
15,21	Technology, N. I. (2008). *Technical Guide to Information Security Testing and Assessment .* Maryland: NIST Special Publication 800-115
19	(OISSG), O. I. (2006). *Information System Security Assesment Framework (ISSAF).* Open Information Systems Security Group.
3	(OWASP), O. W. (2017). *OWASP Top 10 - 2017 Los diez riesgos más críticos en aplicaciones web.* https://wiki.owasp.org/images/5/5e/OWASP-Top-10-2017-es.pdf
1	*UAD Hispasec*. (Marzo de 2022). Tabla tomada de: Jespases, (24 de Marzo,2022), Vulnerabilidades reportadas ascienden a más de 172.000, https://unaaldia.hispasec.com/2022/03/vulnerabilidades-reportadas-ascienden-a-mas-de-172-000.html
4	jurídico, D. P. (Marzo de 2019). *DPEJ RAE (Integridad)*. https://dpej.rae.es/lema/integridad
5	jurídico, D. P. (Julio de 2020). *DPEJ RAE (Confidencialidad)*. https://dpej.rae.es/lema/confidencialidad
6	Española, R. A. (Octubre de 2022). *Diccionario de la lengua española (Disponibilidad)*. https://dle.rae.es/disponibilidad
8	Española, R. A. (Marzo de 2019). *Diccionario de la lengua española (Amenaza)*. https://dle.rae.es/amenaza
9	MAGERIT, C.-C. (2012). *Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información.* Madrid: Ministerio de Hacienda y Administraciones Públicas. https://www.ccn-cert.cni.es/documentos-publicos/1789-magerit-libro-i-metodo
10	UNISDR. (2004). *Oficina de las Naciones Unidas para la Reducción del Riesgo de Desastres.* https://www.unisdr.org/2004/campaign/booklet-spa/page9-spa.pdf
11	Certificación, A. E. (2008). *Organismo de Normalización en España.* https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0041430
13	INCIBE. (Julio de 2019). *Instituto Nacional de Ciberseguridad*. https://www.incibe.es/empresas/blog/el-pentesting-auditando-seguridad-tus-sistemas

16 Normalización, O. I. (2018). *ISO 31000:2018*. https://www.iso.org/obp/ui#iso:std:iso:31000:ed-2:v1:es

17 (OWASP), O. W. (2014). *OWASP Testing Guide v4.0.* https://owasp.org/www-project-web-security-testing-guide/assets/archive/OWASP_Testing_Guide_v4.pdf

20 Standard, P. T. (Agosto de 2014). *PTES Standard*. http://www.pentest-standard.org/index.php/Main_Page

23 INCIBE. (Marzo de 2017). *Instituto Nacional de Ciberseguridad*. https://www.incibe.es/empresas/blog/amenaza-vs-vulnerabilidad-sabes-se-diferencian

24 Española, R. A. (2022). *Diccionario de la lengua española (Monitorear)*. https://dle.rae.es/monitorear#PecQiEQ

25 Española, R. A. (2022). *Diccionario de la lengua española (Probabilidad)*. https://dle.rae.es/probabilidad

26 (OWASP), O. W. (2017). *OWASP Top 10 - 2017 Los diez riesgos más críticos en aplicaciones web.* https://wiki.owasp.org/images/5/5e/OWASP-Top-10-2017-es.pdf

27,33,34 Project, O. W. (Agosto de 2018). *OWASP Risk Rating Calculator*. https://owasp-risk-rating.com/

28 Cataluña, U. P. (Septiembre de 2015). *Retro Informática el pasado del futuro*. https://www.fib.upc.edu/retro-informatica/historia/internet.html

29 Doctoralia. (Agosto de 2018). *Clinic Cloud*. https://clinic-cloud.com/blog/protocolos-de-seguridad-de-la-informacion/

30 Maluenda, R. (Agosto de 2020). *Profile, blog de Raquel Maluenda de Vega*. https://profile.es/blog/desarrollo-aplicaciones-web/

31 Project, O. W. (Diciembre de 2003). *OWASP TOP TEN*. https://owasp.org/www-project-top-ten/

32 OWASP. (Diciembre de 2017). *wiki.owasp.org.* https://wiki.owasp.org/images/5/5e/OWASP-Top-10-2017-es.pdf