Probador Certificado de ISTQB®

Programa de Estudio de Nivel Avanzado
Especialidad – Prueba de Seguridad

Versión 2016

Traducción realizada por el
Spanish Software Testing Qualifications Board
Versión ES 001.25
Basada en el Programa de Estudio
“Certified Tester Advance Level Specialist Syllabus, Security Tester, Version 2016”

International Software Testing Qualifications Board

Nota sobre Derechos de Propiedad Intelectual.

Grupo de Trabajo de Nivel Avanzado: Mike Smith (presidente)

Grupo de Trabajo de Prueba de Seguridad (“Advanced Security Tester Syllabus Working Group”):

Randall Rice (presidente)

Tarun Banga

Taz Daughtrey

Frans Dijkman

Prof. Dr. Stefan Karsch

Satoshi Masuda

Raine Moilanen

Joel Oliveira

Alain Ribault

Ian Ross

Kwangik Seo

Dave van Stein

Dr. Nor Adnan Yahaya

Wenqiang Zheng.

Historial de Revisiones

Versión	Fecha	Observaciones
0.1	24 de abril de 2015	Versión de referencia creada a partir del borrador del programa de estudio existente para probadores de seguridad, versión 3.9
0.2	15 de junio de 2015	Entrada | aportación (entrada) consolidada después de la reunión de autores de Oslo.
1.0 - Beta	20 de septiembre de 2015	Entrega beta - comentarios de la entrega alfa incorporados.
1.0 - Candidato a GA	4 de marzo de 2016	Tras la revisión del grupo de trabajo de exámenes, se ha cambiado la LO 4.1.2 de K2 y K3 y se ha redactado de nuevo de forma adecuada. El texto ya apoya adecuadamente una LO K3.
1.0 - G	18 de marzo de 2016	Publicación GA - se han incorporado los comentarios de la versión beta.

Tabla de Contenidos

Agradecimientos

Este documento ha sido elaborado por el equipo “International Software Testing Qualifications Board Advanced Level Working Group”.

El equipo principal agradece al equipo revisor y a todas los Comités Nacionales sus sugerencias y aportaciones.

En el momento en que se completó el programa de estudio de Nivel Avanzado para este módulo, el Grupo de Trabajo de Nivel Avanzado - Prueba de Seguridad tenía la siguiente composición:

Los autores del equipo principal para este programa de estudio: Randall Rice (presidente), Hugh Tazwell Daughtrey (vicepresidente), Frans Dijkman, Joel Oliveira, Alain Ribault.

Las siguientes personas participaron en la revisión, los comentarios y la votación de este programa de estudio: Tarun Banga, Clive Bates, Hugh Tazwell Daughtrey (vicepresidente), Frans Dijkman (autor), Christian Alexander Graf, Wenda Hu, Matthias Hamburg, Prof. Dr. Stefan Karsch, Sebastian Malyska, Satoshi Masuda, Gary Mogyorodi, Raine Moilanen, Joel Oliveira, Meile Posthuma, Alain Ribault, Randall Rice (Presidente), Ian Ross, Kwangik Seo, Dave van Stein, Ernst von Düring, Attila Toth, Wei Xue, Dr. Nor Adnan Yahaya, Xiaofeng Yang, Wenqiang Zheng, Ping Zuo.

Además, reconocemos y agradecemos a los responsables y miembros del Expert Level Working Party por su temprana y continua orientación: Graham Bath (Presidente del Expert Level Working Party), Judy McKay (Vicepresidenta del Expert Level Working Party).

Este documento fue publicado formalmente por la Asamblea General de ISTQB® el 18 de marzo de 2016.

Notas de la Versión en Idioma Español

El Spanish Software Testing Qualifications Board (SSTQB) ha llevado a cabo la traducción del Programa de Estudio de "ISTQB® Certified Tester, Advanced Level Syllabus, Security Tester". Este Programa de Estudio se denomina, en idioma español, "Probador Certificado de ISTQB®, Nivel Avanzado, Prueba de Seguridad, versión 2016".

El equipo de traducción y revisión para este programa de estudio es el siguiente (por orden alfabético):

Participación	Nombre y Apellidos	Organización	País
Responsable de la revisión	Luisa Morales Gómez-Tejedor	SSTQB	España
Responsable de la traducción	Gustavo Márquez Sosa	SSTQB	España

El Comité Ejecutivo del SSTQB agradece especialmente las aportaciones de los revisores.

En una siguiente versión se podrán incorporar aportaciones adicionales. El SSTQB considera conveniente mantener abierta la posibilidad de realizar cambios en el "Programa de Estudio".

Madrid, 13 de marzo de 2022

0 Introducción al Programa de Estudio

0.1 Objetivo de este Documento

Este programa de estudio constituye la base para la formación como Probador Certificado del ISTQB® de Nivel Avanzado, Prueba de Seguridad. El ISTQB proporciona este programa de estudio en los siguientes términos:

1. A los comités miembro, para traducir a su idioma local y para acreditar a los proveedores de formación. Los comités miembro pueden adaptar el programa de estudio a sus necesidades lingüísticas particulares y añadir referencias para adaptarlo a sus publicaciones locales.
2. A los organismos de certificación, para elaborar las preguntas del examen en su lengua local adaptadas a los objetivos de aprendizaje de este programa de estudio.
3. A los proveedores de formación, para desarrollar material didáctico y determinar los métodos de enseñanza adecuados.
4. A los candidatos a la certificación, para que preparen el examen de certificación (ya sea como parte de un curso de formación o de forma independiente).
5. A la comunidad internacional de ingeniería de software y sistemas, para avanzar en la profesión de prueba del software y sistemas, y como fuente de libros y artículos.

El ISTQB puede permitir que otras entidades utilicen este programa de estudio para otros fines, siempre y cuando soliciten y obtengan permiso previo y por escrito por parte del ISTQB.

0.2 Visión General

La formación de probador certificado de nivel avanzado en la prueba de seguridad está dirigida a personas que ya han alcanzado un punto avanzado en su carrera de pruebas de software y desean desarrollar aún más su experiencia en la prueba de seguridad. Los módulos ofrecidos en el Nivel Avanzado cubren una amplia gama de temas referentes a la prueba.

Para recibir la certificación de Nivel Avanzado en el módulo "Prueba de Seguridad", los candidatos deben estar en posesión del certificado de Nivel Básico de Probador Certificado y satisfacer al Organismo responsable del Examen que los evalúa de que tienen suficiente experiencia práctica para ser certificados en el Nivel Avanzado, que debe ser de no menos de tres años de experiencia académica, práctica o de consultoría relevante. Consulte al Organismo responsable del Examen correspondiente para determinar sus criterios específicos relativos a la experiencia práctica.

0.3 Examen

Todos los exámenes que se realicen sobre este módulo de nivel avanzado se basarán en este programa de estudio de Prueba Seguridad de Nivel Avanzado.

El formato del examen está definido por las "Advanced Exam Guidelines of the ISTQB".

Los exámenes se pueden realizar como parte de un curso de formación acreditado o de forma independiente (por ejemplo, en un centro de exámenes). Los exámenes pueden tomarse en papel o electrónicamente, pero todos los exámenes deben ser supervisados/observados (supervisados por una persona autorizada por un Comité Nacional una Junta Nacional o el Organismo Examinador).

0.4 Cómo está Organizado este Programa de Estudio

Hay diez capítulos. A continuación del título superior se encuentra la “Duración” (en minutos) del capítulo. Por ejemplo:

1 Fundamentos de la Prueba de Seguridad

Duración: 105 minutos

Significa que está previsto que el capítulo 1 requiera 105 minutos para impartir sus contenidos. Los objetivos de aprendizaje específicos se enumeran al principio de cada capítulo.

0.5 Definiciones

Muchos términos utilizados en la literatura del software se utilizan indistintamente. Mientras que a los candidatos de los exámenes de nivel básico y avanzado se les pueden plantear preguntas basadas únicamente en el glosario de términos estándar del ISTQB, se espera, además, que los candidatos de este nivel conozcan y sean capaces de trabajar con las distintas definiciones.

NOTA: El "aseguramiento de la información" (AI) sólo se cita en la sección 2.4. En el apartado 2.4.3 se afirma que el AI debe considerarse más amplio que la "prueba de seguridad", del mismo modo que el aseguramiento de la calidad es más amplio que la prueba de software.

"Seguridad de la información" se utiliza en las secciones 2.2, 2.3.1, 2.7.2, 6 (Antecedentes), 6.1.3, y en todo el capítulo 9.

No se utiliza el término "ciberseguridad", que en algunos sectores actualmente se denomina AI.

Las palabras clave que aparecen al principio de cada capítulo de este programa de estudio de nivel avanzado se definen en el Glosario estándar de términos utilizados en las pruebas de software, publicado por el ISTQB, o se proporcionan en la bibliografía a la que se hace referencia.

0.6 Nivel de Detalle

El nivel de detalle de este programa de estudio permite mantener un alto nivel de consistencia tanto en la enseñanza como en los exámenes a nivel internacional. Para lograr este objetivo, el programa de estudio consta de:

- Objetivos generales de enseñanza que describen los objetivos del Nivel Avanzado.

- Objetivos de aprendizaje para cada área de conocimiento, que describen el resultado cognitivo del aprendizaje y la forma de pensar que se debe adoptar.
- Una lista de la información que se debe enseñar, incluyendo una descripción, y referencias a fuentes adicionales si fuera necesario.
- Una descripción de los conceptos clave que se deben enseñar, incluyendo fuentes tanto bibliográficas como estándares aceptados.
- Es posible que en este programa de estudio se mencionen determinadas herramientas, métodos y marcas comerciales. Este programa de estudio no pretende promover ni recomendar ninguna solución de seguridad en particular.

El contenido del programa de estudio no es una descripción de toda el área de conocimientos para la prueba de seguridad avanzada, sino que refleja el nivel de detalle que se debe cubrir en un curso de formación para probadores de seguridad avanzados.

0.7 Objetivos de Aprendizaje / Nivel de Conocimiento

El contenido de este programa de estudio, los términos y los elementos principales (objetivos) de todos los estándares enumerados deberán ser al menos recordados (K1) y comprendidos (K2), aunque no se mencionen explícitamente en los objetivos de aprendizaje.

Los siguientes objetivos de aprendizaje se definen como aplicables a este programa de estudio. Cada tema del programa de estudio se evaluará de acuerdo con el objetivo de aprendizaje correspondiente.

Los siguientes objetivos de aprendizaje se definen como aplicables a este programa de estudio. Cada tema del programa de estudio se evaluará de acuerdo con el objetivo de aprendizaje correspondiente.

Nivel 1: Recordar (K1)

El candidato reconocerá, recordará y evocará un término o concepto.

Palabras clave: Acordarse, recordar, reconocer, saber.

Ejemplo:

Puede reconocer la definición de "riesgo" como:

- “un factor que podría dar lugar a futuras consecuencias negativas; normalmente se expresa como impacto y probabilidad”.

Nivel 2: Comprender (K2)

El candidato puede seleccionar las razones o explicaciones de los enunciados relacionados con el tema, y puede resumir, diferenciar, clasificar y dar ejemplos para los hechos (por ejemplo, comparar términos), los conceptos de prueba, los procedimientos de prueba (explicando la secuencia de tareas).

Palabras clave: Resumir, clasificar, comparar, trazar, contrastar, poner ejemplos, interpretar, traducir, representar, inferir, concluir, categorizar.

Ejemplo:

Explicar la razón por la que la prueba de seguridad debe diseñarse lo antes posible:

- Para encontrar defectos y vulnerabilidades de seguridad cuando sean más económicos de abordar
- Para evitar la construcción de un sistema o una aplicación que sea propensa a que se apliquen parches a las vulnerabilidades de seguridad de forma continua.

Nivel 3: Aplicar (K3)

El candidato puede seleccionar la aplicación correcta de un concepto o técnica y aplicarlo a un contexto determinado. El nivel K3 se aplica normalmente a los conocimientos procedimentales. No se trata de un acto creativo como la evaluación de una aplicación de software o la creación de un modelo para un programa de software determinado. Cuando se suministra un modelo, el programa de estudio explica los pasos de procedimiento necesarios para crear casos de prueba a partir de ese modelo, entonces es K3.

Palabras clave: Implementar, ejecutar, utilizar, seguir un procedimiento, aplicar un procedimiento.

Ejemplo:

- Utilizar el procedimiento genérico de creación de casos de prueba de seguridad para seleccionar los casos de prueba a partir de un diagrama de transición de estados dado con el fin de cubrir todas las transiciones.

Nivel 4: Analizar (K4)

El candidato puede descomponer información relacionada con un procedimiento o técnica en sus partes constituyentes para una mejor comprensión, y puede distinguir entre hechos e inferencias. La aplicación típica es analizar la situación de un documento, un software o un proyecto y proponer acciones adecuadas para resolver un problema o una tarea.

Palabras clave: Analizar, diferenciar, seleccionar, estructurar, concentrar, atribuir, deconstruir, evaluar, juzgar, monitorizar, coordinar, crear, sintetizar, generar, plantear una hipótesis, planificar, diseñar, construir, producir.

Ejemplo:

- Analizar los riesgos de seguridad del producto y proponer actividades de mitigación preventivas y correctivas.
- Seleccionar las herramientas de prueba de seguridad que serían más adecuadas en una situación dada con fallos de seguridad previos.

Referencia (Para los niveles cognitivos de los objetivos de aprendizaje):

Bloom, B. S. (1956). Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain, David McKay, Co. Inc.

Anderson, L. W. and Krathwohl, D. R. (eds) (2001). A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives, Allyn & Bacon.

1 Fundamentos de la Prueba de Seguridad

Duración: 105 minutos

Palabras Clave

privacidad de datos	(“data privacy”)
jáquer ético	(“ethical hacker”)
seguridad de la información	(“information security”)
prueba de penetración	(“penetration testing”)
evaluación del riesgo	(“risk assessment”)
exposición al riesgo	(“risk exposure”)
mitigación de riesgo	(“risk mitigation”)
ataque contra la seguridad	(“security attack”)
auditoría de seguridad	(“security audit”)
política de seguridad	(“security policy”)
procedimiento de seguridad	(“security procedure”)
riesgo de seguridad	(“security risk”)

Objetivos de Aprendizaje para “Fundamentos de la Prueba de Seguridad”:

1.1 Riesgos de Seguridad

AS-1.1.1	(K2)	Comprender el papel de la evaluación del riesgo en el suministro de información para la planificación y el diseño de prueba la de seguridad y la alineación de la prueba de seguridad con las necesidades de negocio.
AS-1.1.2	(K4)	Identificar los activos importantes que hay que proteger, el valor de cada activo y los datos necesarios para evaluar el nivel de seguridad que necesita cada activo.
AS-1.1.3	(K4)	Analizar el uso eficaz de las técnicas de evaluación de riesgos en una situación determinada para identificar las amenazas actuales y futuras en materia de seguridad.

1.2 Políticas y Procedimientos de Seguridad de la Información

AS-1.2.1	(K2)	Comprender el concepto de políticas y procedimientos de seguridad y cómo se aplican en los sistemas de información.
AS-1.2.2	(K4)	Analizar un conjunto concreto de políticas y procedimientos de seguridad junto con los resultados de pruebas de seguridad para determinar su efectividad.

1.3 Auditoría de Seguridad y su Papel en la Prueba de Seguridad.

AS-1.3.1 (K2) Comprender el objetivo de una auditoría de seguridad.

1.1 Riesgos de Seguridad

La prueba funcional se basa en una serie de elementos, como los riesgos, los requisitos, los casos de uso y los modelos. La prueba de seguridad se basa en los aspectos de seguridad de esas especificaciones, pero también busca verificar y validar los riesgos de seguridad, los procedimientos y políticas de seguridad, el comportamiento de los atacantes y las vulnerabilidades de seguridad conocidas.

1.1.1 El Papel de la Evaluación de Riesgos en la Prueba de Seguridad

Los objetivos de prueba de seguridad se basan en los riesgos de seguridad. Estos riesgos se identifican mediante la realización de una evaluación del riesgo de seguridad. Las técnicas generales de gestión del riesgo se describen en [ISTQB_FL_SYL] y [ISTQB_ATM_SYL].

El riesgo es una medida del grado en que una entidad se ve amenazada por una circunstancia o evento potencial, y en general es una función de:

- Los impactos adversos que surgirían si la circunstancia o evento ocurre, y
- La probabilidad de que ocurra.

Los riesgos de seguridad de la información son aquellos que surgen de la pérdida de confidencialidad, integridad o disponibilidad de la información o de los sistemas de información y reflejan los posibles impactos adversos para las operaciones de la organización (es decir, la misión, las funciones, la imagen o la reputación), los activos de la organización, los individuos, otras organizaciones y un país. [NIST 800-30]

El papel de una evaluación del riesgo de seguridad es permitir que una organización comprenda qué áreas y activos pueden estar en peligro, y determinar la magnitud de cada riesgo. Para los probadores de seguridad, una evaluación del riesgo de seguridad puede ser una importante fuente de información a partir de la cual se pueden planificar y diseñar las pruebas de seguridad. Además, una evaluación del riesgo de seguridad puede utilizarse para priorizar las pruebas de seguridad de modo que el mayor nivel de rigor y cobertura de las pruebas pueda concentrarse en las áreas con mayor exposición al riesgo.

Al establecer prioridades en las pruebas de seguridad basadas en una evaluación del riesgo de seguridad, las pruebas se alinean con los objetivos de seguridad del negocio. Sin embargo, para que esta alineación se produzca, la evaluación del riesgo de seguridad debe reflejar con exactitud las amenazas a la seguridad de la organización, los implicados y los activos a proteger.

Es importante comprender que cualquier evaluación del riesgo (de seguridad o de otro tipo) es sólo una instantánea en un momento dado y se basa en información limitada que puede llevar a suposiciones y conclusiones no válidas. Los riesgos de seguridad cambian continuamente en una organización y en los proyectos, ya que cada día surgen nuevas amenazas. Por lo tanto, las evaluaciones de los riesgos de seguridad deben realizarse a intervalos regulares. El intervalo de tiempo exacto para realizar las evaluaciones de riesgos de seguridad varía en función de la organización y del grado de cambio que experimente. Algunas organizaciones realizan evaluaciones de riesgos de seguridad cada tres o seis meses, mientras que otras las realizan anualmente.

Otro problema de las evaluaciones del riesgo es el nivel de conocimientos de los participantes. Algunos riesgos pueden pasar desapercibidos debido a la falta de información detallada. Además, los riesgos pueden pasar desapercibidos si la gente no entiende las amenazas y los riesgos de seguridad. Por esta razón, es bueno solicitar la aportación de diversas personas y prestar mucha atención al nivel de detalle de la información que proporcionan.

Es previsible que se hagan suposiciones erróneas que pueden hacer que algunos riesgos de seguridad importantes pasen desapercibidos en la evaluación. Entre las formas de hacer frente a la posibilidad de que falte información sobre los riesgos o de que ésta sea incompleta se incluye el uso de una metodología establecida de evaluación de riesgos de seguridad como lista de comprobación y la obtención de aportaciones de diversas personas. Una de estas metodologías puede encontrarse en [NIST 800-30].

1.1.2 Identificación de Activos

No toda la información que hay que asegurar está en formato digital, como los documentos copiados (contratos, planos, notas escritas, registros y contraseñas en forma escrita). Aunque no esté en formato digital, esta información puede tener un gran valor. Por lo tanto, es necesario preguntarse qué información es digital y cuál no lo es. Tal vez el activo a proteger exista tanto en formato digital como físico. A la hora de identificar los activos que hay que proteger, hay que plantearse las siguientes preguntas:

¿Qué activos son valiosos para la organización?

Entre los ejemplos de información sensible de alto valor se encuentran:

- Datos de clientes.
- Planes de negocio.
- Software propietario desarrollado por la empresa.
- Documentación del sistema.
- Imágenes y diagramas que son propiedad de la empresa.
- Propiedad intelectual (por ejemplo, procesos, secretos comerciales).
- Hojas de cálculo financieras.
- Presentaciones y cursos de formación.
- Documentos.
- Correos electrónicos.
- Registros de los empleados.
- Declaraciones fiscales.

Aunque muchos activos están basados en información, es posible que algunos activos de una organización sean de naturaleza tanto física como intangible. Entre los ejemplos de estos activos se encuentran:

- Prototipos físicos de nuevos dispositivos en desarrollo.
- La capacidad de prestar servicios.
- La reputación y credibilidad de la organización.

¿Qué valor tiene el activo?

Muchos activos sensibles tienen un valor material. Otros se miden más por los costes y las consecuencias de su pérdida. Por ejemplo, ¿qué haría un competidor con el plan de negocio de un rival?

El valor puede ser difícil de evaluar con certeza; sin embargo, entre los métodos para determinar el valor de los activos digitales se encuentran:

- Los ingresos futuros que generará el activo.
- El valor para un competidor que pueda obtener la información.
- El tiempo y el esfuerzo necesarios para recrear el activo.
- Las multas y sanciones por no poder presentar la información cuando se necesite, por ejemplo, para una auditoría o un juicio.
- Multas y sanciones por la pérdida de datos de clientes.

¿Dónde se encuentran ubicados los activos digitales?

En el pasado, los activos digitales residían en servidores, ordenadores de sobremesa o periféricos como discos o CD's. Aunque se trata de un enfoque anticuado y desorganizado, todavía puede haber datos sensibles en viejos CD, DVD y unidades USB. Un medio más seguro de almacenar los activos digitales es el uso de servidores empresariales seguros, que utilizan un cifrado fuerte para todos los datos sensibles. Para acceder a los datos sensibles almacenados en los servidores seguros, debe exigirse autenticación y autorización. Además, puede ser necesaria otra protección de seguridad, como los certificados digitales para acceder a la información sensible a través de Internet.

El almacenamiento está cambiando. Ahora pueden existir grandes cantidades de datos empresariales en dispositivos móviles como teléfonos inteligentes, tabletas y unidades de memoria USB. Cuando la información digital se ha trasladado al almacenamiento en la nube, existe un nuevo conjunto de preocupaciones de seguridad basadas en el acceso a los datos.

La importancia de la cuestión del almacenamiento de datos surge de casos ocurridos en el pasado en los que personas a las que se les habían confiado datos sensibles simplemente salieron del edificio de una empresa con un disco duro lleno de datos privados de clientes y de la empresa. Uno de estos casos en Estados Unidos fue el de un disco duro robado de una zona segura en una agencia de seguridad gubernamental, que incluía información bancaria y de nóminas de más de 100.000 trabajadores actuales y antiguos. [Washington Post, 2007].

¿Cómo se accede a los activos digitales?

Entre los métodos habituales para acceder a los activos digitales se encuentran:

- Acceso por ordenador a través de una red de área local o redes Wi-Fi.
- Acceso remoto a través de una Red Privada Virtual (VPN) o una unidad en la Nube.
- La transmisión de almacenes de datos físicos (CD, DVD, unidades USB) de persona a persona, que es una práctica de baja tecnología pero muy común.
- Envío de archivos por correo electrónico

¿Cómo se aseguran los activos digitales?

Hay varias formas de asegurar los activos digitales, entre las que se incluyen:

- Cifrado (¿Qué tipo y fuerza, quién tiene las claves?)
- Autenticación y tokens (¿Se requieren certificados digitales? ¿Son adecuadas y se siguen las políticas de contraseñas?)
- Autorización (¿Qué niveles de privilegio se han concedido a los usuarios que manejan activos digitales?)

1.1.3 Análisis de las Técnicas de Evaluación del Riesgo

El proceso de evaluación del riesgo de seguridad es muy similar a una evaluación del riesgo estándar, con la principal diferencia de que se concentra en las áreas relacionadas con la seguridad.

Una evaluación del riesgo de seguridad debe incluir las perspectivas de los implicados externos en la prueba de seguridad (es decir, personas o partes involucradas en el proyecto o con el producto que están fuera de la empresa y tienen un claro interés en la seguridad del proyecto/producto). Entre estos implicados se encuentran:

- Clientes y usuarios - útiles para comprender la perspectiva, obtener aportaciones para la prueba de seguridad y establecer una buena comunicación.
- El público y la sociedad - importante para transmitir que la seguridad de la información es un esfuerzo y una responsabilidad de la comunidad.
- Los organismos reguladores - necesarios para asegurar el cumplimiento de las leyes aplicables en materia de seguridad de la información.

La preparación de una evaluación del riesgo incluye las siguientes tareas [NIST 800-30]:

- Identificar el propósito de la evaluación.
- Identificar el alcance de la evaluación.
- Identificar las suposiciones y restricciones asociadas a la evaluación.

- Identificar las fuentes de información que se utilizarán como entradas (entradas) en la evaluación.
- Identificar el modelo de riesgo y los enfoques analíticos (es decir, los enfoques de evaluación y análisis) que se emplearán durante la evaluación.

La realización de evaluaciones del riesgo incluye las siguientes tareas específicas [NIST 800-30]:

- Identificar las fuentes de amenaza que son relevantes para la organización.
- Identificar los eventos que representan una amenaza que podrían ser provocados por esas fuentes.
- Identificar las vulnerabilidades dentro de la organización que podrían ser explotadas por fuentes de amenaza a través de eventos que representan amenazas específicas y las condiciones predisponentes (i.e. factores) que podrían afectar al éxito de la explotación.
- Determinar la probabilidad de que las fuentes de amenaza identificadas inicien eventos que representan amenazas específicas y la probabilidad de que los eventos que representan amenaza tengan éxito.
- Determinar los impactos adversos para las operaciones y activos de la organización, los individuos, otras organizaciones y el país, resultantes de la explotación de las vulnerabilidades por parte de la amenaza.

Comunicar y compartir la información consiste en las siguientes tareas específicas [NIST 800-30]:

- Comunicar los resultados de la evaluación del riesgo.
- Compartir la información desarrollada durante la ejecución de la evaluación del riesgo para apoyar otras actividades de gestión del riesgo.

1.2 Políticas y Procedimientos de Seguridad de la Información

1.2.1 Comprender Políticas y Procedimientos de Seguridad

Es habitual que las políticas de seguridad de la información varíen entre las organizaciones en función del modelo de negocio, la industria específica y los riesgos de seguridad únicos a los que se enfrenta la organización. Incluso con una amplia gama de variaciones, los objetivos de las políticas de seguridad son similares. La base de todas las políticas de seguridad debe ser una evaluación del riesgo de seguridad que examine las amenazas específicas a la seguridad y cómo afectan a la organización. [Jackson, 2010]

Entre los ejemplos de políticas de seguridad se encuentran, entre otros, los siguientes [Jackson, 2010]:

Uso Aceptable - Esta política define las prácticas que un usuario de un sistema informático debe seguir para cumplir con las políticas y procedimientos de seguridad de la organización. Esta política abarca tanto el comportamiento aceptable como el no aceptable en el uso de los recursos digitales, como las redes, los sitios web y los datos. Además, la política puede aplicarse tanto a los usuarios internos como externos de los sistemas de una organización. Es importante que los usuarios del sistema entiendan y sigan la política en todo momento. Para evitar la confusión y las violaciones accidentales de la política, ésta debe definir reglas específicas relativas al comportamiento aceptable, al comportamiento inaceptable y al comportamiento requerido.

Acceso Mínimo - Esta política define los niveles mínimos de acceso que se necesitan para realizar determinadas tareas. El objetivo de esta política es evitar que se concedan derechos de acceso superiores a los necesarios para realizar sus tareas. Tener derechos de acceso superiores a los necesarios puede dar lugar a un abuso inadvertido o intencionado de los privilegios de los usuarios.

Acceso a la Red - Esta política define los criterios para acceder a varios tipos de redes, como las redes de área local (LAN) y las redes inalámbricas. Además, esta política puede definir lo que está permitido y lo que no está permitido mientras se está en la red. Esta política suele prohibir a los usuarios que añadan a la red dispositivos no autorizados, como enrutadores[1] y puntos calientes[2].

Acceso Remoto - Esta política requiere lo necesario para que se pueda conceder el acceso remoto a la red tanto a los empleados internos como a los usuarios externos (no empleados). El uso de la RPV[3] suele estar contemplado en esta política.

Acceso a Internet - Esta política define el uso permitido de Internet por parte de los empleados e invitados de una organización. El alcance de esta política incluye los tipos de sitios web a los que se puede y no se puede acceder, como los sitios de juegos de azar o de pornografía, y también aborda si se permite el uso de Internet para fines no comerciales. Aunque algunos de los elementos contemplados en esta política pueden tratarse también en la política de uso aceptable, algunas organizaciones deciden definir esta política por separado debido al número de personas que desarrollan actividades de negocio en Internet.

Gestión de cuentas de usuario - Esta política define la creación, el mantenimiento y la supresión de cuentas de usuario. La auditoría periódica de las cuentas de usuario también está contemplada en esta política para garantizar su cumplimiento.

Clasificación de Datos - Hay muchas formas de clasificar datos desde el punto de vista de la seguridad. En este programa de estudio, el término "datos sensibles" se utiliza como término general para cualquier dato que deba ser protegido para evitar su pérdida. Una política de clasificación de datos define los diferentes tipos de datos que se consideran sensibles y que deben ser protegidos. Al tener una política de clasificación de datos, una organización puede crear controles para proteger los datos en función de su valor para la organización y sus clientes. Normalmente, el área de negocio que crea los datos es responsable de su clasificación basándose en una estructura de clasificación estándar.

A continuación, se presenta un ejemplo de estructura de clasificación de datos (desde un contexto de negocio):

- Público: Cualquier persona, ya sea perteneciente o ajena a la organización, puede ver estos datos (por ejemplo, los documentos y las páginas web de cara al exterior).
- Confidencial: Esta es normalmente la clasificación por defecto para cualquier documento creado internamente. Estos documentos pueden incluir correos electrónicos, informes y presentaciones que se utilizan internamente en la organización. Un ejemplo de esto sería un informe de ventas. Sólo los usuarios autorizados deberían poder trabajar con este nivel de información. A menudo se

[1] enrutador es la traducción del término en inglés “router”.

[2] punto caliente es la traducción del término “hot spot” o “hotspot” o “hot-spot”.

[3] RPV es el acrónimo del término Red Privada Virtual. Red Privada Virtual es la traducción del término en inglés “Virtual Private Network - VPN”.

requieren acuerdos de no divulgación (o acuerdos de confidencialidad)[4] antes de compartir este tipo de información con terceras partes, como los consultores.

- Altamente confidencial: Este es un nivel de confidencialidad más alto para la información sensible que debería estar disponible sólo para ciertas personas de la organización. Esto incluiría información como secretos comerciales, planes estratégicos, diseños de productos y datos financieros no públicos. No se permite compartir este tipo de datos salvo con el permiso explícito del propietario de los datos.

- Privado: Se trata de información que suele estar restringida a los directivos de la organización que deben estar específicamente autorizados a tener acceso a ella. Si se divulga, esta información podría tener importantes repercusiones negativas para la organización, como por ejemplo un perjuicio financiero. Debido al alto riesgo asociado a la pérdida, la información privada debe protegerse con extremo cuidado. Estos datos pueden incluir información de investigación y desarrollo, planes de fusiones y adquisiciones, así como información de los clientes, como datos de tarjetas de crédito y cuentas.

- Secreto: En el contexto corporativo, se trata de información que una organización obtiene de una parte externa para realizar cambios, pero que no se permite que se conozca dentro o fuera de la organización. Un ejemplo en el contexto corporativo sería un documento de diseño creado por un consultor que trabaja en un nuevo tipo de tecnología que implica la colaboración con otras empresas, cada una de las cuales debe mantener la información en un nivel de secreto hasta que la tecnología esté lista para ser revelada. Es comparable a la alta confidencialidad con la diferencia de que puede no tener un valor tangible para la propia organización. En ese sentido, es diferente de un secreto comercial. Sin embargo, la revelación de la información secreta podría causar daños a la organización, a otras organizaciones o al país. En el contexto militar y gubernamental, se trata de información que puede ser desarrollada u obtenida, pero que debe ser conocida sólo por personas con ciertos niveles de autorización de seguridad. En el contexto militar, esto incluiría detalles de proyectos científicos o de investigación que incorporan nuevos desarrollos tecnológicos o técnicas que tienen aplicaciones militares directas de vital importancia para la defensa de un país.

Gestión de la Configuración y del Cambio - Esta política puede tener un contexto operativo normal, como la descripción de cómo se gestionan y configuran los cambios en los sistemas, con el fin de evitar las interrupciones debidas a un impacto inesperado. Desde el punto de vista de la seguridad, la gestión de la configuración controla cómo se aplican los ajustes de seguridad a los dispositivos y aplicaciones seguros. El riesgo es que un cambio no autorizado en un dispositivo seguro podría causar una vulnerabilidad de seguridad que podría pasar desapercibida.

Otro riesgo es que un cambio no autorizado en la configuración del código o de la aplicación podría crear una vulnerabilidad de seguridad. Esta política incluye las configuraciones estándar que deben utilizarse, un proceso de aprobación de todos los cambios y un proceso de reversión si se producen problemas. Esta política puede aplicarse a todos los servicios, aplicaciones y dispositivos de TI de una organización.

Seguridad de Servidores - Esta política transmite la responsabilidad al propietario o propietarios del servidor de seguir las prácticas de seguridad corporativas, así como las mejores prácticas de la industria

[4] acuerdo de no divulgación o acuerdo de confidencialidad es la traducción del término “non-disclosure agreement”.

para la instalación, configuración y funcionamiento de los servidores y sistemas. Además, es obligatorio definir y mantener configuraciones de referencia. Algunos ejemplos de prácticas descritas en esta política son los requisitos de seguridad, las copias de seguridad y la recuperación, y la limitación de los servicios activos a los necesarios para la ejecución de las aplicaciones. También pueden incluirse en esta política los requisitos de monitorización y auditoría para garantizar que el servidor está configurado y actualizado correctamente.

Dispositivos Móviles - Los dispositivos móviles tienen un conjunto único de asuntos de interés relativos a la seguridad, por lo que puede ser necesaria una política separada sólo para los dispositivos móviles. Por ejemplo, los ordenadores portátiles y los teléfonos inteligentes pueden perderse o ser robados con facilidad, lo que podría provocar la pérdida de datos privados y de la empresa. Estos dispositivos también tienen un alto riesgo de contacto con el software malicioso. Estos riesgos requieren normas y precauciones específicas que deben seguirse para mitigar los riesgos y limitar la exposición de la organización a las amenazas de seguridad. Esta política puede incluir requisitos sobre los datos que deben encriptarse, la instalación y el mantenimiento de versiones actuales de software antimalware y cuándo se necesitan contraseñas para acceder al dispositivo. Además, en esta política se definen los tipos de información de la organización que pueden residir en los dispositivos móviles. También se puede abordar la seguridad física, como disponer de candados para los ordenadores portátiles y contar con procedimientos para informar de la pérdida o el robo de dispositivos.

Acceso de Invitados - Esta política define las prácticas que deben aplicarse para proteger a la organización, al tiempo que permite a la empresa acoger a invitados y a otras personas en las redes de la organización. Un aspecto de esta política es exigir a los invitados que lean y acepten las políticas de uso aceptable antes de concederles acceso a la red. Esta política puede aplicarse de varias maneras, como haciendo que los invitados firmen una política de uso aceptable y proporcionando después un código para el acceso temporal. La intención principal de esta política es hacer cumplir las normas de seguridad de la organización y seguir proporcionando procedimientos para permitir que los invitados accedan a la red o a Internet.

Seguridad Física - Esta política define los controles necesarios para las instalaciones físicas, ya que estar en la proximidad física de dispositivos seguros puede aumentar el riesgo de una brecha de seguridad. Esta política también puede cubrir otros riesgos, como la pérdida de energía, el robo, el incendio y los desastres naturales. También se aborda aquí qué dispositivos pueden sacarse o introducirse en la empresa, especialmente para las áreas que albergan información sensible.

Política de Contraseñas - Esta política define los requisitos mínimos para las contraseñas fuertes y otras prácticas de seguridad con respecto a las contraseñas, como el tiempo permitido entre los cambios obligatorios de contraseñas, la forma en que las personas protegen la privacidad de sus contraseñas (como no usar la función "recordar contraseña" en los navegadores, prohibir que se compartan las contraseñas y prohibir la transmisión de contraseñas por correo electrónico). Esta política puede aplicarse a las aplicaciones, a las cuentas de usuario y a cualquier otro lugar en el que se necesiten contraseñas.

Protección Contra Software Malicioso - Esta política define un marco de defensas y comportamientos para prevenir, detectar y eliminar el software malicioso. Dado que el software malicioso y el software espía pueden provenir de diversas fuentes, esta es una política importante que todos los miembros de la organización deben entender y seguir. Por ejemplo, esta política podría restringir el uso de unidades USB.

Respuesta a Incidencias - Esta política describe cómo responder a una incidencia relacionada con la seguridad. Estas incidencias pueden ir desde el descubrimiento de software malicioso y violaciones de la política de uso aceptable hasta el acceso no autorizado a datos sensibles. Es importante contar con esta política antes de que se produzca una incidencia para evitar tener que determinar las respuestas

adecuadas caso por caso. Esta política también aborda la comunicación, incluyendo las respuestas a los medios de comunicación y la notificación a las fuerzas del orden.

Política de Auditoría - Esta política autoriza a los auditores a solicitar acceso a los sistemas con el fin de realizar una auditoría. El equipo de auditoría puede necesitar acceso a datos de registro, registros de tráfico de red y otros datos forenses.

Suministro y Uso de Licencias de Software - Esta política aborda cómo la organización obtiene y licencia el software que utiliza. Si se violan las licencias de software comercial, la organización corre el riesgo de ser sancionada con multas y acciones legales. Por ello, es importante que se identifiquen y supervisen las licencias. La descarga e instalación de software no aprobado es una prohibición clave que se encuentra a menudo en esta política.

Monitorización electrónica y privacidad - Las organizaciones tienen el derecho y la responsabilidad de monitorizar las comunicaciones electrónicas a través del hardware y los recursos de la empresa. Esto incluye la correspondencia por correo electrónico y los medios sociales. Esta política describe qué monitorización realiza la organización y qué datos son objeto de recopilación. Las leyes varían según los países, por lo que es necesario contar con asesoramiento legal antes de redactar esta política. [Jackson, 2010]

Procedimientos de Seguridad

Los procedimientos de seguridad especifican los pasos que se deben dar para aplicar una política o un control específico, y los pasos que se deben dar en respuesta a una incidencia de seguridad concreta. Los procedimientos formales y documentados facilitan la aplicación de las políticas de seguridad y los controles obligatorios.

Las políticas, las normas y las directrices describen los controles de seguridad que deben aplicarse, mientras que un procedimiento describe los aspectos específicos, explicando cómo aplicar los controles de seguridad paso a paso. Por ejemplo, se podría redactar un procedimiento para explicar cómo conceder niveles de acceso a los usuarios, detallando cada paso que hay que dar para garantizar que se concede el nivel correcto de acceso, de modo que los derechos de usuario satisfagan la política, las normas y las directrices aplicables.

1.2.2 Análisis de las Políticas y Procedimientos de Seguridad

Antes de evaluar un conjunto de políticas y procedimientos de seguridad es importante determinar el objetivo o los objetivos de la evaluación y definir un conjunto de criterios con los que juzgar la idoneidad de las políticas y los procedimientos. En algunos casos, los criterios pueden estar definidos por estándares como COBIT [COBIT], ISO27001 [ISO27001] o PCI [PCI].

Además, es necesario definir:

Qué recursos se necesitan en términos de competencias y conocimientos en las áreas particulares que se evalúan.

- Cómo medir la adecuación de las políticas y los procedimientos.
- Qué hay que medir y evaluar (por ejemplo, la efectividad, la eficiencia, la usabilidad, la adopción).
- Dónde se encuentran las políticas y los procedimientos en la organización.

- Una lista de comprobación para guiar la evaluación y aportar consistencia.

La lista de comprobación actúa como una guía que orienta al auditor sobre dónde buscar y qué esperar. Las herramientas, como las de auditoría de contraseñas, pueden ser útiles para probar ciertos controles y determinar si están cumpliendo sus objetivos y para generar datos que puedan utilizarse posteriormente en la evaluación del riesgo. El auditor busca encontrar "pruebas" del cumplimiento de las políticas, los controles y los estándares. Algunas de las tareas de la siguiente lista son de naturaleza estática, mientras que otras, como la observación de los procesos en acción, son dinámicas. El auditor hace lo siguiente:

- Revisa la documentación del sistema.
- Encuesta a las personas sobre su percepción de la efectividad de las políticas y los procedimientos.
- Entrevista al personal clave que participa en los procesos que se controlan.
- Presencia los sistemas y procesos que se están llevando a cabo.
- Analiza los resultados de auditorías anteriores para descubrir tendencias.
- Analiza los registros e informes.
- Revisa la configuración del control técnico, como la configuración del cortafuegos y la del sistema de detección de intrusos.
- Toma muestras de las transacciones de datos para detectar cualquier anomalía o transacción sospechosa [Jackson, 2010].

Controles

Los controles de seguridad son salvaguardas o contramedidas técnicas o administrativas para evitar, contrarrestar o minimizar las pérdidas o la indisponibilidad debidas a las amenazas que actúan sobre su vulnerabilidad de coincidencia, es decir, el riesgo de seguridad. [Por ejemplo, un control de seguridad en un sistema de nóminas puede ser que dos personas deban aprobar por separado un cambio en la información de la retribución de un empleado. Los probadores de seguridad deben conocer los controles específicos de su organización e incluir pruebas para ellos en las pruebas de seguridad.

Los principales tipos de control de seguridad son administrativos, técnicos y físicos. Dentro de cada categoría, los controles específicos que se pueden implementar son preventivos, de detección, correctivos o de recuperación. Estos tipos de control funcionan conjuntamente y, en general, deben proporcionarse controles de cada categoría para proteger eficazmente un activo. [Jackson, 2010]

Puede encontrar una lista de los 20 controles de seguridad críticos ("Top 20 Critical Security Controls") en www.sans.org. [Web-1]

Pruebas de Seguridad

La principal diferencia de las pruebas de seguridad con respecto a un análisis estático de las políticas y procedimientos de seguridad es el uso de los resultados de las pruebas diseñadas específicamente para verificar o validar la efectividad de las políticas y procedimientos de seguridad. Estas pruebas se concentran en el riesgo de que una política de seguridad pueda estar en vigor, pueda cumplirse, pero no sea eficaz para proteger los activos.

También es posible que al realizar las evaluaciones de la política de seguridad y de los procedimientos se diga que se realizan determinadas tareas. Una prueba de seguridad de esas tareas puede ayudar a determinar la efectividad de las políticas y procedimientos de seguridad en la práctica. Por ejemplo, una política y un procedimiento en materia de contraseñas pueden parecer razonables y eficaces sobre el papel, pero cuando se utiliza una herramienta para descifrar contraseñas, el procedimiento puede resultar insuficiente para alcanzar sus objetivos.

Las políticas de seguridad y los procedimientos pueden ser una fuente de pruebas de seguridad; sin embargo, el probador de seguridad debe tener en cuenta que los ataques siempre están evolucionando. Surgen nuevos ataques y, al igual que en cualquier aplicación de software, pueden hacerse evidentes nuevos defectos, todo lo cual es motivo para realizar pruebas de seguridad desde la perspectiva de un atacante.

1.3 Auditoría de Seguridad y su Papel en la Prueba de Seguridad

La auditoría de seguridad es un examen y evaluación manual que identifica los puntos débiles de los procesos y la infraestructura de seguridad de una organización. Las auditorías de seguridad a nivel de procedimiento (por ejemplo, para revisar los controles internos) pueden realizarse manualmente. Las auditorías de seguridad a nivel de arquitectura suelen realizarse con herramientas de auditoría de seguridad, que pueden estar alineadas con una solución concreta de un proveedor para redes, arquitectura de servidores y estaciones de trabajo.

Al igual que las pruebas de seguridad, una auditoría de seguridad no garantiza que se encuentren todas las vulnerabilidades. Sin embargo, la auditoría es una actividad más en el proceso de seguridad para identificar las áreas problemáticas e indicar dónde es necesario corregirlas.

En algunos enfoques de auditoría de seguridad, las pruebas se realizan como parte del proceso de auditoría. Sin embargo, el alcance de la auditoría de seguridad es mucho mayor que el de las pruebas de seguridad. La auditoría de seguridad suele investigar áreas como los procedimientos, las políticas y los controles que son difíciles de probar de forma directa. Las pruebas de seguridad están más relacionadas con las tecnologías de apoyo a la seguridad, como la configuración del cortafuegos, la aplicación correcta de la autenticación y el cifrado, y la aplicación de los derechos de usuario.

La auditoría de seguridad cuenta con cinco pilares [Jackson, 2010]:

Evaluación - Las evaluaciones documentan e identifican las amenazas potenciales, los activos clave, las políticas y los procedimientos, y la tolerancia al riesgo de la dirección. Las evaluaciones no son eventos puntuales. Dado que el entorno y el negocio están en constante cambio, las evaluaciones deben realizarse de forma regular. Esto también ofrece la oportunidad de saber si las políticas de seguridad siguen siendo pertinentes y eficaces.

Prevención - Esto va más allá de la tecnología e incluye controles administrativos, operativos y técnicos. La prevención no se consigue sólo con la tecnología, sino también con las políticas, los procedimientos y la concienciación. Aunque la prevención de todos y cada uno de los ataques es poco realista, la combinación de defensas puede ayudar a dificultar mucho más el éxito de un atacante.

Detección - La detección es la forma en que se identifica una brecha de seguridad o una intrusión. Sin los mecanismos de detección adecuados, se corre el riesgo de no saber si la red ha sido comprometida. Los controles de detección ayudan a identificar las incidencias de seguridad y proporcionan visibilidad de las actividades en la red. La detección temprana de incidencias permite una reacción adecuada para recuperar los servicios rápidamente.

Reacción - El tiempo de reacción se reduce en gran medida con buenas defensas de seguridad y mecanismos de detección. Aunque las brechas de seguridad son una mala noticia, es importante saber si se ha producido una. Un tiempo de reacción rápido es fundamental para minimizar la exposición a la incidencia. Una reacción rápida requiere buenas defensas preventivas y mecanismos de detección que proporcionen los datos y el contexto necesarios para la respuesta. La rapidez y eficiencia de la respuesta a incidentes es un indicador clave de la efectividad de los esfuerzos de seguridad de una organización.

Recuperación - La recuperación comienza con la determinación de lo ocurrido para poder recuperar los sistemas sin recrear la misma vulnerabilidad o condición que causó la incidencia en primer lugar. La fase de recuperación no termina con la restauración del sistema. También está el análisis de la causa raíz que determina qué cambios se necesitan hacer en los procesos, procedimientos y tecnologías para reducir la probabilidad de que vuelva a presentarse el mismo tipo de vulnerabilidad en el futuro. Un auditor debe asegurar que la organización cuenta con un plan de recuperación que incluye formas de prevenir futuras incidencias similares.

1.3.1 Objetivo de una Auditoría de Seguridad

A continuación, se ofrece una lista de elementos que podrían detectarse en una auditoría de seguridad:

- Seguridad física inadecuada. Una política de seguridad podría exigir el cifrado de todos los datos de los clientes, tanto en el almacenamiento como en la transmisión. Por ejemplo, durante la auditoría, se descubre que una vez a la semana se envía un archivo de información de clientes mediante un informe físico a todos los gestores. Este informe se desecha cada semana, pero se descubre que algunos gestores tiran por descuido los informes físicos a la basura, donde pueden ser encontrados por cualquiera que quiera rebuscar en ella (es decir, "buscar entre la basura"[5]).
- Mantenimiento inadecuado de las contraseñas. Una política de seguridad puede exigir que cada usuario cambie su contraseña cada 30 días. Una auditoría de seguridad revela que las contraseñas se cambian, pero muchos usuarios simplemente alternan entre "ContraseñaA" y "ContraseñaB" cada mes. (El historial de contraseñas es una prestación habitual en las herramientas de auditoría de contraseñas).
- Controles inadecuados de los derechos de usuario y de los privilegios de uso compartido. Un ejemplo de hallazgo negativo podría ser cuando se han concedido a los usuarios más derechos de acceso a elementos de los que necesitan para realizar su trabajo. Otro ejemplo podría ser cuando los archivos de un usuario individual se comparten en la red cuando deberían ser privados. Este es un asunto de interés especial para los usuarios con ordenadores portátiles y, sobre todo, para los que acceden a la intranet a través de conexiones Wi-Fi en casa o en lugares públicos.
- Seguridad inadecuada a nivel de servidor. Las áreas de auditoría específicas incluyen:
  - Asignación de puertos y seguridad.
  - Protección de datos.
  - Protección de las cuentas de usuario (inicios de sesión y otra información sensible).

[5] buscar entre la basura es la traducción del término “dumpster diving”.

- Aplicación inadecuada de las actualizaciones de seguridad del proveedor.
- Mecanismos inadecuados de detección de intrusiones.
- Planes de respuesta inadecuados en caso de una brecha de seguridad.

1.3.2 Identificación, Evaluación y Mitigación del Riesgo

Una vez que la auditoría haya identificado las áreas problemáticas, hay que evaluar el riesgo y poner en marcha un plan de mejora. El informe del auditor puede incluir recomendaciones, así como otras áreas de riesgo. A partir de este punto, se pueden planificar las actividades de identificación, evaluación y mitigación del riesgo.

La identificación del riesgo es el proceso de documentar un riesgo o un área de riesgo. En el contexto de la seguridad informática, los riesgos están relacionados con la seguridad. La evaluación del riesgo es la actividad que asigna un valor a los riesgos identificados. Es importante entender que los modelos tradicionales de evaluación del riesgo de TI no son suficientes para abordar los riesgos de seguridad de TI. Cualquier modelo o enfoque de evaluación del riesgo de seguridad debe estar orientado específicamente a los perfiles de riesgo de seguridad de TI.

Los riesgos de seguridad suelen medirse en términos de exposición al riesgo. La exposición al riesgo se calcula multiplicando el impacto o la pérdida potencial por la probabilidad de que se produzca esa pérdida. Por ejemplo, si la información de la cuenta de un cliente se ve comprometida, ¿cuál sería el impacto? ¿Y si ese cliente tuviera 100 millones de dólares en activos depositados?

La probabilidad de que ocurra puede determinarse aplicando un modelo de evaluación del riesgo de seguridad como el que se encuentra en la publicación 800-30 del NIST, Guía para la realización de evaluaciones de riesgos [NIST 800-30]. Otra excelente guía para realizar evaluaciones del riesgo de seguridad es la metodología de clasificación de riesgos de OWASP [OWASP2]. La siguiente información se ha extraído de [NIST 800-30].

Los modelos de riesgo definen los factores de riesgo que deben evaluarse y las relaciones entre esos factores. Los factores de riesgo son características utilizadas en los modelos de riesgo como entradas para determinar los niveles de riesgo en las evaluaciones del riesgo. Los factores de riesgo también se utilizan ampliamente en las comunicaciones de riesgo para destacar lo que afecta en gran medida a los niveles de riesgo en situaciones, circunstancias o contextos particulares.

Los factores de riesgo típicos son la amenaza, la vulnerabilidad, el impacto, la probabilidad y la condición predisponente. Los factores de riesgo pueden descomponerse en características más detalladas (por ejemplo, las amenazas se descomponen en fuentes y eventos de amenaza). Estas definiciones son importantes para que las organizaciones se documenten antes de llevar a cabo las evaluaciones de riesgo, ya que éstas se basan en atributos bien definidos de las amenazas, las vulnerabilidades, el impacto y otros factores de riesgo para determinar efectivamente el riesgo.

Amenazas

Una amenaza es cualquier circunstancia o evento con el potencial de impactar negativamente en las operaciones y activos de la organización, en los individuos, en otras organizaciones o en un país a través de un sistema de información mediante el acceso no autorizado, la destrucción, la divulgación o la modificación de la información, y/o la denegación de servicio.

Los eventos de amenaza son causados por fuentes de amenaza. Una fuente de amenaza se caracteriza por:

- La intención y el método dirigidos a la explotación de una vulnerabilidad; o
- Una situación y un método que pueden explotar accidentalmente una vulnerabilidad.

Entre los tipos de fuentes de amenaza se encuentran, en general:

- Ataques hostiles, cibernéticos o físicos
- Errores humanos de omisión o comisión
- Fallos estructurales de los recursos controlados por la organización (por ejemplo, hardware, software, controles ambientales)
- Catástrofes naturales o provocadas por el hombre, accidentes y fallos que escapan al control de la organización.

Se han desarrollado varias taxonomías de fuentes de amenaza. Algunas taxonomías de fuentes de amenaza utilizan el tipo de impactos adversos como principio organizador. Múltiples fuentes de amenaza pueden iniciar o causar el mismo evento de amenaza -por ejemplo, un servidor de aprovisionamiento puede quedar fuera de servicio por un ataque de denegación de servicio, un acto deliberado de un administrador de sistemas malintencionado, un error administrativo, un fallo de hardware o un fallo de alimentación.

Vulnerabilidades y Condiciones Predisponentes

Una vulnerabilidad es una debilidad en un sistema de información, en los procedimientos de seguridad del sistema, en los controles internos o en la implementación que podría ser explotada por una fuente de amenazas.

La mayoría de las vulnerabilidades de los sistemas de información pueden asociarse a controles de seguridad que o bien no se han aplicado (intencionadamente o no), o bien se han aplicado pero conservan alguna debilidad. Sin embargo, también es importante tener en cuenta la posibilidad de que surjan vulnerabilidades emergentes que pueden ocurrir de forma natural a lo largo del tiempo, a medida que evolucionan las misiones/funciones de negocio de la organización, cambian los entornos de operación, proliferan las nuevas tecnologías y surgen nuevas amenazas. En el contexto de tales cambios, los controles de seguridad existentes pueden resultar inadecuados y puede ser necesario reevaluar su efectividad. La tendencia de los controles de seguridad a degradar potencialmente su efectividad a lo largo del tiempo refuerza la necesidad de mantener las evaluaciones de riesgo durante todo el ciclo de vida del software y también la importancia de los programas de monitorización continua para obtener un conocimiento permanente de la situación de la postura de seguridad de la organización.

Las vulnerabilidades no se identifican sólo dentro de los sistemas de información. Considerando los sistemas de información en un contexto más amplio, las vulnerabilidades pueden encontrarse en las estructuras de gobernanza de la organización (por ejemplo, la falta de estrategias eficaces de gestión del riesgo y de un encuadre adecuado del mismo, una comunicación intrainstitucional deficiente, decisiones incoherentes sobre las prioridades relativas de las misiones/funciones de negocio, o un desajuste de la arquitectura corporativa para apoyar las actividades de la misión/negocio).

Las vulnerabilidades también pueden encontrarse en las relaciones externas (por ejemplo, las dependencias de determinadas fuentes de energía, cadenas de suministro, tecnologías de la información

y proveedores de telecomunicaciones), en los procesos de la misión/negocio (por ejemplo, procesos mal definidos o que no son conscientes del riesgo) y en las arquitecturas de seguridad de la empresa/información (por ejemplo, decisiones arquitectónicas deficientes que dan lugar a la falta de diversidad o resiliencia de los sistemas de información de la organización).

Impacto

El nivel de impacto de un evento que representa una amenaza es la magnitud del daño que puede esperarse de las consecuencias de la divulgación no autorizada de información, la modificación no autorizada de información, la destrucción no autorizada de información o la pérdida de disponibilidad de la información o del sistema de información. Dicho daño puede ser experimentado por una variedad de implicados tanto de la organización como ajenos a ella, incluyendo:

- Responsables de organismos.
- Propietarios de misión y de negocio.
- Dueños/administradores de la información.
- Propietarios de misión / proceso de negocio.
- Propietarios de sistemas de información.
- Individuos/grupos de los sectores público o privado que dependen de la organización -en esencia, cualquier persona que tenga un interés en las operaciones, activos o individuos de la organización, incluyendo otras organizaciones en asociación con la organización, o un país.

La organización debería documentar explícitamente la siguiente información:

- El proceso utilizado para realizar las determinaciones de impacto.
- Los supuestos relacionados con las determinaciones del impacto.
- Las fuentes y los métodos para obtener la información sobre el impacto.
- La justificación de las conclusiones alcanzadas con respecto a las determinaciones del impacto.

Las organizaciones pueden definir explícitamente cómo las prioridades y los valores establecidos guían la identificación de los activos de alto valor y los posibles impactos adversos para los implicados de la organización. Si dicha información no está definida, las prioridades y los valores relacionados con la identificación de los objetivos de las fuentes de amenaza y los impactos organizativos asociados pueden obtenerse normalmente de la planificación y las políticas estratégicas. Por ejemplo, los niveles de categorización de la seguridad indican los impactos para la organización de comprometer diferentes tipos de información.

Probabilidad

La probabilidad de ocurrencia aborda la probabilidad (o posibilidad) de que el evento que representa una amenaza cause un impacto adverso, independientemente de la magnitud del daño que pueda esperarse. Se trata de un factor de riesgo ponderado que se basa en un análisis de la probabilidad de que una amenaza determinada sea capaz de explotar una vulnerabilidad determinada (o un conjunto de vulnerabilidades). El factor de riesgo de probabilidad combina una estimación de la probabilidad de que el

evento de amenaza se inicie con una estimación de la probabilidad de impacto (es decir, la probabilidad de que el evento de amenaza resulte en impactos adversos).

La evaluación de la probabilidad de que se produzca una amenaza adversa normalmente se basa en lo siguiente

- La intención del adversario.
- La capacidad del adversario.
- El objetivo del adversario.

Para otros eventos que no sean amenazas adversas, la probabilidad de ocurrencia se estima utilizando evidencias históricas, datos empíricos u otros factores. Tenga en cuenta que la probabilidad de que un evento de amenaza se inicie o se produzca se evalúa con respecto a un marco temporal específico (por ejemplo, los próximos seis meses, el próximo año o el periodo hasta que se alcance un hito especificado).

Si es casi seguro que un evento que representa una amenaza se inicie o se produzca en el marco temporal (especificado o implícito), la evaluación del riesgo puede tener en cuenta la frecuencia estimada del evento. La probabilidad de que se produzca una amenaza también puede basarse en el estado de la organización (incluyendo, por ejemplo, su misión/procesos de negocio principales, la arquitectura de la empresa, la arquitectura de seguridad de la información, los sistemas de información y los entornos en los que operan esos sistemas. También deben tenerse en cuenta las condiciones predisponentes y la presencia y efectividad de los controles de seguridad desplegados para proteger contra comportamientos no autorizados/indeseables, detectar y limitar los daños, y/o mantener o restaurar las capacidades de la misión/negocio.

Determinación del Nivel de Riesgo de Seguridad

La evaluación de la probabilidad de ocurrencia y la evaluación del impacto pueden combinarse para calcular una severidad general para el riesgo. Las puntuaciones específicas de la evaluación pueden utilizarse como base para completar la matriz de riesgos. En otros casos, pueden utilizarse estimaciones (bajas, medias o altas).

Los valores de la matriz de riesgo pueden basarse en una escala de 0 a 9, en la que los valores numéricos están determinados por criterios específicos. Por ejemplo, los criterios de probabilidad del riesgo podrían evaluarse para la privacidad de datos como:

0 - <3 (Bajo)	Los datos privados no se almacenan en los dispositivos locales y están cifrados cuando se almacenan en medios seguros.
3 - <6 (Medio)	Los datos privados pueden residir en dispositivos como ordenadores portátiles, pero están cifrados.
6 - 9 (Alto)	No se sabe exactamente si los datos privados residen en dispositivos locales. No se puede asegurar el cifrado.

Del mismo modo, los criterios de impacto del riesgo podrían evaluarse en la misma escala de 0 a 9, basándose en criterios específicos. Por ejemplo:

0 - <3	(Bajo)	Los datos privados comprometidos impactarían a menos de 200 personas.
3 - <6	(Medio)	Los datos privados comprometidos impactarían a entre 200 y 1.000 personas.
6 - 9	(Alto)	Los datos privados comprometidos impactarían a más de 1.000 personas.

Sea cual sea la forma en que el probador llegue a las estimaciones de probabilidad e impacto, éstas pueden combinarse en una calificación final de severidad para el elemento de riesgo. Si existe una buena información sobre el impacto en el negocio, ésta debería utilizarse en lugar de la información sobre el impacto técnico. Si no hay información sobre el negocio, el impacto técnico es la siguiente mejor opción.

A continuación, se muestra un ejemplo de matriz de riesgo que puede utilizarse para determinar la severidad de los riesgos individuales.

Severidad Global del Riesgo				
Impacto del Riesgo	**Alto**	Medio	Alto	Crítico
	Medio	Medio	Medio	Alto
	Bajo	Bajo	Bajo	Medio
		Bajo	**Medio**	**Alto**
		Probabilidad del Riesgo		

En la matriz de ejemplo anterior, si la probabilidad es media y el impacto es alto, la severidad global es alta.

Además, el informe de evaluación del riesgo debe identificar si el riesgo es continuo. Los riesgos continuos son indicadores de una mayor probabilidad de que se produzca una pérdida.

La severidad de un riesgo determina la importancia relativa de mitigar el riesgo. Cuanto mayor sea la severidad del riesgo, más inmediato será el requisito de la respuesta. El nivel de detalle proporcionado en cualquier evaluación del riesgo en particular es coherente con el propósito de la evaluación del riesgo y el tipo de entradas necesarias para apoyar las determinaciones de probabilidad e impacto posteriores.

1.3.3 Personas, Procesos y Tecnología

Las prácticas informáticas de una organización también tienen tres componentes: las personas, los procesos y la tecnología. Todos ellos tienen un impacto en la seguridad. Según Chris Jackson en su libro, Network Security Auditing [Jackson, 2010], "Todos los incidentes de seguridad, desde los robos hasta la pérdida de registros de clientes, suelen poder trazarse hasta una deficiencia que puede atribuirse a las personas, al proceso o a la tecnología".

Personas: Las personas pueden incluir a los usuarios finales, los administradores de sistemas, los propietarios de los datos y los directivos de la organización. Cada persona tiene distintos niveles de

competencia, actitudes y agendas, lo que influye en el modo en que la seguridad les afecta y en la efectividad de los controles de seguridad. Independientemente de la presencia de políticas de seguridad, procedimientos y controles, éstos serán ineficaces si las personas no los siguen. Si las personas no respetan las políticas de seguridad, es posible que se necesiten medidas correctoras, como la necesidad de una formación de concienciación en materia de seguridad o la imposición de sanciones por su incumplimiento. Las estructuras organizativas y las políticas de seguridad suelen ser impulsadas por las personas, tanto internas como externas a una organización.

Procesos: Los procesos definen cómo se prestan los servicios TIC, incluidos los relacionados con la seguridad. En un contexto de seguridad, los procesos incluyen los procedimientos y estándares que se ponen en marcha para proteger los activos que se consideran valiosos. Para ser efectivos, los procesos deben estar definidos, actualizados, ser consistentes y seguir las buenas prácticas de seguridad. Los procesos definen los roles y las responsabilidades, los controles, las herramientas y los pasos específicos que implica la realización de una tarea.

Tecnología: La tecnología engloba las instalaciones, el equipamiento, el hardware informático y el software que automatizan o apoyan un negocio. La tecnología permite a las personas realizar trabajos repetitivos más rápidamente y con menos errores que si se realizan manualmente sin ella. De hecho, algunas tareas, como la aplicación de contraseñas, serían imposibles sin las herramientas adecuadas. El riesgo es que la tecnología utilizada de forma incorrecta puede ayudar a las personas a cometer errores más rápidamente.

Estas tres áreas pueden considerarse como un "triángulo de hierro" que, en conjunto, forma una solución informática completa. Si se ignora alguna de las tres áreas, todo el esfuerzo de entrega y seguridad en el ámbito TIC se resiente.

Al evaluar los controles de seguridad, el auditor debe ver un sistema desde la perspectiva de un atacante y anticiparse a la forma en que las personas, los procesos o la tecnología podrían ser explotados para obtener un acceso no autorizado a activos considerados valiosos. La dirección de las organizaciones suele sorprenderse de que los mecanismos de seguridad que creían seguros, no lo son. La única manera de saber con certeza si una determinada defensa de seguridad funciona y es eficaz es probar el sistema desde la perspectiva de un atacante. Esto se conoce a menudo como hacking ético o prueba de penetración (pen).

Aquí es donde la relación entre la auditoría y las pruebas se vuelve más directa. La auditoría identifica las deficiencias y las áreas importantes que hay que probar. La prueba de seguridad es el medio por el que se prueba o se refuta que los controles de seguridad están realmente en su sitio y funcionan con eficacia.

Ejemplo de escenario:

La agencia tributaria de un país es objeto de una auditoría de seguridad. Uno de los hallazgos de la auditoría es que es posible que los delincuentes presenten una declaración de impuestos fraudulenta y obtengan las devoluciones de impuestos que corresponden al contribuyente defraudado. Este hallazgo de auditoría se confirma con pruebas de seguridad y el riesgo se califica de "crítico". La agencia tributaria reconoce la posibilidad de este riesgo de fraude, pero decide no actuar sobre el riesgo hasta el año siguiente.

Los contribuyentes defraudados, que han seguido todos los procedimientos de seguridad prescritos, pueden presentar una reclamación ante la agencia tributaria que tenía conocimiento del defecto en el proceso de declaración de impuestos. En este caso, la agencia tributaria sería responsable por el fraude.

2 Propósitos, Objetivos y Estrategias de la Prueba de Seguridad - 130 minutos

Duración: 105 minutos

Palabras Clave

secuencia de comandos de sitios cruzados (XSS)	(“cross-site scripting”)
ofuscación de datos	(“data obfuscation”)
denegación de servicio	(“denial of service”)
aseguramiento de la información	(“information assurance”)
política de seguridad	(“security policy”)
prueba de seguridad	(“security testing”)
vulnerabilidad de seguridad	(“security vulnerability”)
ciclo de vida del software	(“software lifecycle”)
estrategia de prueba	(“test strategy”)

Objetivos de Aprendizaje para “Propósitos, Objetivos y Estrategias de la Prueba de Seguridad”:

2.1 Introducción

No hay objetivos de aprendizaje para esta sección.

2.2 El propósito de la Prueba de Seguridad

AS-2.2.1	(K2)	Comprender por qué motivo se necesita la prueba de seguridad en una organización, incluyendo los beneficios para la organización, como la reducción de riesgos y niveles más altos de seguridad y confianza.

2.3 El Contexto de una Organización

AS-2.3.1	(K2)	Comprender cómo las realidades del proyecto, las restricciones del negocio, el ciclo de vida de desarrollo del software y otras consideraciones afectan la misión del equipo de pruebas de seguridad

2.4 Objetivos de la Prueba de Seguridad

AS-2.4.1	(K2)	Explicar por qué las metas y los objetivos de la prueba de seguridad deben alinearse con la política de seguridad de la organización y con otros objetivos de prueba de la organización.
AS-2.4.2	(K3)	Demostrar la capacidad de identificar los objetivos de la prueba de seguridad basados en la funcionalidad, los atributos tecnológicos y las vulnerabilidades conocidas, para un escenario de proyecto determinado.
AS-2.4.3	(K2)	Comprender la relación entre el aseguramiento de la información y la prueba de seguridad.

2.5 Alcance y Cobertura de los Objetivos de la Prueba de Seguridad

AS-2.5.1	(K3)	Demostrar la capacidad de definir la relación entre los objetivos de la prueba de seguridad y la necesidad de solidez de la integridad de los activos digitales y físicos sensibles para un proyecto determinado.

2.6 Enfoques para la Prueba de Seguridad

AS-2.6.1	(K4)	Analizar una situación dada y determinar qué enfoques de prueba de seguridad tienen más probabilidades de éxito.
AS-2.6.2	(K4)	Analizar una situación en la que falló un enfoque de prueba de seguridad determinado, identificando las causas probables del fallo.
AS-2.6.3	(K3)	Demostrar la capacidad de identificar a los distintos implicados en un escenario dado e ilustrar los beneficios de la prueba de seguridad para cada grupo de implicados

2.7 Mejorar las Prácticas de la Prueba de Seguridad

AS-2.7.1	(K4)	Analizar los KPI (indicadores clave de rendimiento) para identificar las prácticas de la prueba de seguridad que necesitan ser mejoradas y los elementos que no necesitan serlo.

2.1 Introducción

Antes de aplicar técnicas de prueba de seguridad especializadas, es importante comprender el contexto más amplio de la prueba de seguridad y su función dentro de una organización concreta. Esta comprensión responde a las siguientes preguntas:

- ¿Por qué es necesaria la prueba de seguridad?
- ¿Cuál es el objetivo de la prueba de seguridad?
- ¿Cómo se integra la prueba de seguridad en la organización?

La prueba de seguridad difiere de otras formas de prueba funcional en dos áreas significativas [ISTQB_ATTA_SYL]:

1. Las técnicas estándar para seleccionar los datos de entrada para la prueba pueden pasar por alto importantes cuestiones de seguridad.
2. Los síntomas de los defectos de seguridad son muy diferentes de los encontrados con otros tipos de pruebas funcionales.

La prueba de seguridad evalúa la vulnerabilidad de un sistema a las amenazas intentando comprometer la política de seguridad del sistema. La siguiente es una lista de amenazas potenciales, que deben ser exploradas durante la prueba de seguridad [ISTQB_ATTA_SYL]:

- Copia no autorizada de aplicaciones o datos.
- Control de acceso no autorizado (por ejemplo, capacidad de realizar tareas para las que el usuario no tiene derechos). Los derechos, el acceso y los privilegios de los usuarios son el objetivo de estas pruebas. Esta información debe estar disponible en las especificaciones del sistema.
- Software que muestra efectos secundarios no deseados cuando realiza su función prevista. Por ejemplo, un reproductor multimedia que reproduce correctamente el audio, pero que lo hace escribiendo los archivos en un almacenamiento temporal no cifrado, presenta un efecto secundario que puede ser aprovechado por piratas informáticos.
- Código insertado en una página web que puede ser ejecutado por usuarios posteriores (secuencia de comandos de sitios cruzados (XSS)[6]). Este código puede ser malicioso.
- Desbordamiento de memoria intermedia[7] que puede ser causado por la introducción de cadenas de datos en un campo de entrada de la interfaz de usuario que son más largas de lo que el código puede tratar correctamente. Una vulnerabilidad de desbordamiento de memoria intermedia representa una oportunidad para ejecutar instrucciones de código malicioso.
- La denegación de servicio[8], que impide a los usuarios interactuar con una aplicación (por ejemplo, sobrecargando un servidor web con peticiones "molestas").

[6] secuencia de comandos de sitios cruzados (XSS) es la traducción del término "cross-site scripting or XSS".
[7] desbordamiento de memoria intermedia es la traducción del término "buffer overrun".
[8] denegación de servicio es la traducción del término "denial of service".

- La interceptación, imitación y/o alteración y posterior retransmisión de las comunicaciones (por ejemplo, las transacciones con tarjetas de crédito) por parte de un tercero, de forma que el usuario no sea consciente de la presencia de ese tercero (ataque de tercero interpuesto[9]).
- Vulnerar los códigos de cifrado utilizados para proteger los datos sensibles.
- Bombas lógicas [10](a veces llamadas Huevos de Pascua), que pueden ser insertadas maliciosamente en el código y que se activan sólo bajo ciertas condiciones (por ejemplo, en una fecha específica). Cuando las bombas lógicas se activan, pueden realizar actos maliciosos como el borrado de archivos o el formateo de discos.

La prueba de seguridad debe integrarse con todas las demás actividades de desarrollo y prueba. Esto requiere tener en cuenta las necesidades únicas de la organización, cualquier política de seguridad existente, los conjuntos de competencias actuales para la prueba de seguridad y cualquier estrategia de prueba existente.

2.2 El Propósito de la Prueba de Seguridad

Al igual que la prueba de software en general, la prueba de seguridad no puede garantizar que un sistema o una organización estén a salvo de ataques. Sin embargo, la prueba de seguridad puede ayudar a identificar los riesgos y a evaluar la efectividad de las defensas de seguridad existentes. Hay otras actividades que complementan la prueba de seguridad, como las auditorías y las revisiones de las prácticas de seguridad.

La prueba de seguridad también demuestra que se ha actuado con la debida diligencia en la protección de los activos digitales. En caso de que se produzca una brecha de seguridad, pueden resultar en acciones legales. Si una empresa puede demostrar que tomó medidas razonables para proteger los activos digitales, como probar las vulnerabilidades, puede haber una defensa en un tribunal. La prueba de seguridad también puede ser una garantía para los clientes y consumidores de que una organización toma las medidas adecuadas para proteger la información sensible.

2.3 Contexto Relativo a la Organización

La seguridad suele ser un tipo de prueba funcional que se realiza junto con otros tipos de prueba. Con sólo una cantidad determinada de tiempo disponible para la prueba, un Jefe de Prueba debe decidir la cantidad de pruebas que se pueden realizar, incluidas las de seguridad. No es raro que la prueba de seguridad se considere un rol de especialista y que, por tanto, se subcontrate a una organización especializada en pruebas de seguridad. El alcance de la prueba de seguridad viene determinado, en última instancia, por los riesgos de negocio o de la organización que se basan en la seguridad. Cuando en una organización los riesgos de seguridad son numerosos, se necesita una prueba de seguridad más amplia.

Al igual que la prueba de software, la seguridad de la información es una actividad del ciclo de vida. Las necesidades de seguridad deben definirse en los requisitos, expresarse en el diseño e implementarse en el código. Entonces, la prueba de seguridad puede verificar y validar la corrección y la eficacia de la implementación de la seguridad. La seguridad no se puede introducir eficazmente en el código como si

[9] ataque de tercero interpuesto es la traducción del término "man in the middle attack".
[10] bomba lógica es la traducción del término "logic bomb".

fuera un parche, ni se puede probar eficazmente en el código. Sólo cuando la seguridad se incorpora al software utilizando técnicas de codificación y diseño seguras, el software puede ser seguro.

Las realidades de tiempo, recursos y alcance limitados, junto con los niveles de riesgo, el conjunto de competencias para la prueba de seguridad y los enfoques del ciclo de vida impactan en gran medida en el éxito de un equipo de pruebas de seguridad en una organización.

2.4 Objetivos de la Prueba de Seguridad

2.4.1 Alineación de los Objetivos de la Prueba de Seguridad

La política de prueba de seguridad puede redactarse una vez que la política de seguridad de la organización haya sido aprobada por la alta dirección. Es importante que las metas y los objetivos de la prueba de seguridad, tal como se expresan en la política de prueba de seguridad, estén alineados con la política de seguridad general de la organización. De lo contrario, se realizarán pruebas de seguridad no autorizadas o las pruebas de seguridad no alcanzarán los objetivos deseados.

2.4.2 Identificación de los Objetivos de la Prueba de Seguridad

Los objetivos de la prueba de seguridad pueden considerarse del mismo modo que los objetivos de la prueba funcional, pero se centran en los objetivos de seguridad. Debe haber uno o varios objetivos de la prueba de seguridad para cada prestación de seguridad del sistema o aplicación.

Los objetivos de la prueba de seguridad también deben basarse en los atributos de la tecnología (por ejemplo, web, móvil, nube, LAN) y en las vulnerabilidades conocidas, tanto en la aplicación como en las vulnerabilidades genéricas. Por ejemplo, los objetivos de las pruebas de seguridad podrían incluir:

- Verificar que la autentificación de la contraseña aplica la regla correcta para la fortaleza de la contraseña.
- Verificar que todos los campos de entrada de datos son de entrada validada para evitar ataques de inyección SQL.
- Verificar que los archivos de datos de los clientes están encriptados con la fuerza correcta.

2.4.3 Diferencia entre el Aseguramiento de la Información y la Prueba de Seguridad

El Aseguramiento de la Información (AI) se define como: "Medidas que protegen y defienden la información y los sistemas de información asegurando su disponibilidad, integridad, autentificación, confidencialidad y no repudio. Estas medidas incluyen la provisión de restauración de los sistemas de información mediante la incorporación de capacidades de protección, detección y reacción." [NISTIR 7298]

La prueba de seguridad es "un proceso utilizado para determinar que las prestaciones de seguridad de un sistema se implementan tal como se han diseñado y que son adecuadas para un entorno de aplicación propuesto." [MDA1]

Cuando se comparan los términos Aseguramiento de la Información (AI) y Prueba de Seguridad, AI es un término más amplio e inclusivo. Esta relación es similar a la que existe entre el Aseguramiento de la Calidad (QA) y la Prueba de Software.

2.5 El Alcance y la Cobertura de los Objetivos de la Prueba de Seguridad

Cuanto mayor sea la necesidad de integridad para los activos digitales y físicos sensibles, mayor será la necesidad de cubrir los objetivos de la prueba de seguridad. Los objetivos de la prueba de seguridad describen esencialmente el alcance de la prueba de seguridad. Si el alcance es demasiado pequeño, no se logrará la confianza de que la seguridad es adecuada. Si el alcance es demasiado grande, los recursos pueden agotarse antes de que la prueba pueda completarse.

Los objetivos de la prueba de seguridad deben describir lo que se espera conseguir con las pruebas de seguridad en lo que respecta a la verificación y validación de las protecciones existentes para los activos digitales y físicos sensibles. Los objetivos de la prueba de seguridad deben relacionarse directamente con los activos específicos, las medidas de protección, los riesgos y la identificación de las vulnerabilidades de seguridad.

2.6 Enfoques de la Prueba de Seguridad

La estrategia de la prueba de seguridad se define para formalizar y comunicar la dirección general de una organización para la prueba de seguridad. A continuación se definen los enfoques que implementan la estrategia de prueba de seguridad.

2.6.1 Análisis de los Enfoques de la Prueba de Seguridad

Cada organización tiene asuntos de interés únicos para el negocio y la misión, que a su vez requieren estrategias y enfoques de pruebas de seguridad únicos para identificar y mitigar los riesgos de seguridad. Sin embargo, también hay algunos asuntos de interés para la seguridad que son comunes en muchas organizaciones.

El enfoque de las pruebas de seguridad se define a nivel de proyecto y debe ser coherente con la política y la estrategia de prueba de la organización. El enfoque de prueba de seguridad de un proyecto será una combinación única de técnicas, herramientas y competencias para abordar los objetivos de la prueba de seguridad para ese proyecto.

Al analizar una situación con el fin de definir un enfoque para la prueba de seguridad, se debe tener en cuenta lo siguiente:

- La procedencia de los sistemas o aplicaciones.
- Cualquier prueba de seguridad anterior.
- La política de seguridad.
- La política de prueba de seguridad.
- Cualquier evaluación de riesgos de seguridad que haya sido realizada en la organización.

- El entorno técnico en uso (por ejemplo, tipo y versión de software, marcos de trabajo, lenguajes de programación, sistemas operativos).
- Las competencias en materia de prueba de seguridad del equipo de prueba.
- Los riesgos de seguridad frecuentes.
- La estructura de organización de prueba.
- La estructura del equipo de proyecto.
- La experiencia del equipo de prueba con diversas herramientas de prueba de seguridad.
- Restricciones (por ejemplo, recursos limitados, tiempo limitado, falta de acceso a los entornos).
- Supuestos (por ejemplo, suposiciones sobre otras formas anteriores de pruebas de seguridad realizadas).

Los diferentes entornos técnicos y tipos de aplicaciones (por ejemplo, cliente/servidor, web, mainframe) suelen requerir diferentes enfoques y estrategias de prueba de seguridad. Por ejemplo, el desarrollo de software puede requerir la revisión del código para detectar vulnerabilidades de seguridad en el mismo, mientras que la prueba de software puede requerir la ofuscación de los datos de prueba. Las aplicaciones basadas en la web tienen vulnerabilidades diferentes a las de los sistemas mainframe y, por tanto, requieren diferentes tipos de pruebas de seguridad.

Algunas vulnerabilidades son comunes en múltiples tecnologías. Por ejemplo, las vulnerabilidades por desbordamiento de la memoria temporal pueden darse en aplicaciones cliente-servidor, web y móviles, con diferencias basadas en cómo se trata la gestión de la memoria en cada tecnología. El resultado es el mismo en todos los entornos, es decir, un comportamiento imprevisible del software que puede permitir a un atacante acceder a una aplicación y realizar tareas que normalmente no estarían permitidas.

La protección inadecuada de los datos puede ocurrir en cualquier tecnología o entorno. Sin embargo, la encriptación de datos en los entornos web y móvil es diferente a la del entorno mainframe. Los algoritmos de cifrado pueden ser los mismos (o similares), pero la diferencia es que los datos deben protegerse en tránsito por Internet en el caso de las aplicaciones web y móviles. En todas las tecnologías, los datos sensibles deben almacenarse en un formato cifrado. Se han producido incidentes en los que los datos sensibles del mainframe se enviaron físicamente (utilizando cinta) a otra parte en un formato no cifrado. "El Grupo Cattles, especializado en préstamos personales y recuperación de deudas, admitió haber perdido dos cintas de copia de seguridad que contenían información de unos 1,4 millones de clientes". [ComputerWeekly]

2.6.2 Análisis de Fallos en los Enfoques de la Prueba de Seguridad

Es necesario entender que hay grados de fallo. El hecho de que no se detecte y resuelva una vulnerabilidad de seguridad no significa necesariamente que el enfoque de la prueba de seguridad haya fracasado. Hay demasiadas vulnerabilidades de seguridad posibles, y cada día se descubren otras nuevas. Sin embargo, hay otros casos en los que los enfoques de la prueba de seguridad han sido inadecuados para identificar eficazmente los riesgos de seguridad, lo que ha llevado a que los datos sensibles y otros activos digitales se vean comprometidos.

El análisis de la causa raíz puede ayudar a identificar por qué puede haber fallado un enfoque de prueba de seguridad. Entre las posibles causas se encuentran:

- Falta de liderazgo ejecutivo en el establecimiento de la prueba de seguridad.
- Falta de provisión ejecutiva de los recursos necesarios para implementar la estrategia de prueba de seguridad (como falta de financiación, falta de tiempo, falta de recursos).
- Falta de implementación efectiva del enfoque de la prueba de seguridad (como la falta de competencias necesarias para realizar las tareas requeridas).
- Falta de comprensión y apoyo de la organización al enfoque de prueba de seguridad.
- Falta de comprensión y apoyo por parte de los implicados respecto al enfoque de prueba de seguridad.
- Falta de comprensión de los riesgos de seguridad.
- Falta de alineación entre el enfoque de prueba y la política de seguridad de la organización.
- Falta de alineación entre el enfoque de la prueba y la política y estrategia de prueba de seguridad de la organización.
- Falta de comprensión del objetivo del sistema.
- Falta de información técnica sobre el sistema (que induce a hacer suposiciones erróneas).
- Falta de herramientas eficaces para la prueba de seguridad.
- Falta de competencias en materia de prueba de seguridad.

2.6.3 Identificación de los Implicados

Para que un esfuerzo en materia de prueba de seguridad sea eficaz, debe presentarse a la dirección un caso de negocio para ello. Este caso de negocio debe definir claramente los riesgos de los fallos de seguridad y los beneficios de contar con un enfoque eficaz de prueba de seguridad para un proyecto concreto.

Diferentes implicados verán diferentes beneficios de un enfoque de prueba de seguridad:

- La dirección ejecutiva verá la protección del negocio como un beneficio.
- La alta dirección puede ver la diligencia debida.
- Los clientes del negocio pueden ver la protección contra el fraude.
- Los agentes responsables del cumplimiento (para las políticas de seguridad internas de la empresa) pueden percibir la posibilidad de asegurar que la organización cumple con los términos de las obligaciones legales.
- Los agentes reguladores (para las leyes de seguridad externas) pueden ver el beneficio de que se cumplan las normas de seguridad.
- Los agentes de la privacidad pueden ver el beneficio de que los datos privados se mantienen seguros y se ha mostrado la debida diligencia en la protección de los activos digitales.

2.7 Mejora de las Prácticas en la Prueba de Seguridad

Para mejorar las prácticas relativas a la prueba de seguridad, primero es necesario realizar una evaluación de las prácticas existentes. Debe haber una forma objetiva de evaluar las prácticas en materia de prueba de seguridad. Éstas se basan en métricas clave para los objetivos de seguridad de la prueba, a partir de las cuales es posible identificar el grado de éxito de los elementos clave de la estrategia.

Estas prácticas deben ser evaluadas de la siguiente manera:

- Desde una perspectiva a corto y largo plazo.
- Considerando el proceso y la organización.
- Considerando las personas, las herramientas, los sistemas y las técnicas.

Las métricas clave incluyen, entre otras, las siguientes:

- Niveles de cobertura de los riesgos de seguridad mediante pruebas.
- Niveles de cobertura de las políticas y prácticas de seguridad mediante pruebas.
- Niveles de cobertura de los requisitos de seguridad mediante pruebas.

Niveles de eficacia del esfuerzo en materia de prueba de seguridad en el pasado, en función de cuándo y dónde se identificaron las vulnerabilidades de seguridad. Esto incluye tanto las vulnerabilidades de seguridad previas como las posteriores a la entrega.

3 Procesos de la Prueba de Seguridad

Duración: 105 minutos

Palabras Clave

cosecha de cuentas	(“account harvesting”)
reventado de contraseña	(“password cracking”)
ingeniería social	(“social engineering”)
enfoque de prueba	(“test approach”)
plan de prueba	(“test plan”)
proceso de prueba	(“test process”)

Objetivos de Aprendizaje para Procesos de la Prueba de Seguridad:

3.1 Definición del Proceso de Prueba de Seguridad

AS-3.1.1 (K3) Demostrar la capacidad de definir los elementos de un proceso de prueba de seguridad eficaz para un proyecto determinado.

3.2 Planificación de la Prueba de Seguridad

AS-3.2.1 (K4) Analizar un plan de pruebas de seguridad dado, aportando retroalimentación sobre los puntos fuertes y débiles del plan.

3.3 Diseño de la Prueba de Seguridad

AS-1.3.1 (K3) Implementar pruebas de seguridad conceptuales (abstractas), basadas en un enfoque de prueba de seguridad determinado, junto con los riesgos de seguridad funcionales y estructurales identificados para un proyecto determinado.

AS-1.3.2 (K3) Implementar casos de prueba para validar políticas y procedimientos de seguridad.

3.4 Ejecución de la Prueba de Seguridad

AS-3.4.1 (K2) Entender los elementos clave y las características de un entorno de prueba de seguridad eficaz.

AS-3.4.2 (K2) Entender la importancia de planificar y obtener las aprobaciones antes de realizar cualquier prueba de seguridad.

3.5 Evaluación de la Prueba de Seguridad

AS-3.5.1 (K4) Analizar los resultados de una prueba de seguridad para determinar lo siguiente

- Naturaleza de la vulnerabilidad de seguridad.
- Extensión de la vulnerabilidad de seguridad.
- Impacto potencial de la vulnerabilidad de seguridad.
- Remedio sugerido.
- Métodos óptimos de información de la prueba.

3.6 Mantenimiento de la Prueba de Seguridad

AS-3.6.1 (K2) Comprender la importancia de mantener los procesos de prueba de seguridad dada la naturaleza evolutiva de la tecnología y las amenazas

3.1 Definición del Proceso de Prueba de Seguridad

Al igual que en la prueba de software en general, la prueba de seguridad es también una actividad del ciclo de vida. Si no se implementan y prueban las defensas de seguridad a lo largo de un proyecto, se pueden producir graves defectos de seguridad que quizá nunca se resuelvan del todo. El proceso de prueba de seguridad debe estar alineado con el proceso de desarrollo para que se realicen las actividades de prueba adecuadas cuando sea necesario.

Los riesgos y necesidades de las pruebas de seguridad de cada organización serán únicos debido a la naturaleza de la misma, los entornos técnicos, el proceso de desarrollo de software y los riesgos de negocio. Por lo tanto, el proceso de prueba de seguridad debe definirse en el contexto de estos factores.

3.1.1 Proceso de Prueba de Seguridad de ISTQB

La tabla 3.1 muestra la relación entre el proceso de prueba general de ISTQB®, tal y como se describe en los programas de estudio de nivel básico y avanzado de ISTQB, y el proceso de prueba de seguridad de ISTQB. Se muestran ejemplos de tareas de pruebas de seguridad para cada paso del proceso.

Proceso de Prueba de ISTQB	Proceso de Prueba de Seguridad de ISTQB	Ejemplo de Tareas de la Prueba de Seguridad
Planificación y Control de la Prueba	Planificación y Control de la Prueba de Seguridad - El objetivo es definir un alcance adecuado de la prueba que corresponda a los riesgos de seguridad.	• Definir los objetivos de la prueba de seguridad. • Definir el alcance de la prueba de seguridad. • Identificar los recursos para la prueba de seguridad. • Definir las estimaciones y calendarios de la prueba de seguridad. • Definir las métricas, los criterios de entrada y de salida de la prueba de seguridad. • Monitorizar el avance y los resultados de la prueba de seguridad. • Tomar las medidas necesarias en respuesta a la información obtenida durante otras actividades de la prueba de seguridad.

Proceso de Prueba de ISTQB	Proceso de Prueba de Seguridad de ISTQB	Ejemplo de Tareas de la Prueba de Seguridad
Análisis y Diseño de la Prueba	Análisis y Diseño de la Prueba de Seguridad - El objetivo es obtener una comprensión de las amenazas y riesgos de seguridad específicos basados en evaluaciones de seguridad, auditorías y fuentes estándar de vulnerabilidades conocidas.	• Revisar los elementos que sirven de base a la prueba de seguridad, como las evaluaciones del riesgo de seguridad, los requisitos de seguridad y las políticas de seguridad. • Definir las condiciones de prueba de seguridad en función de: o Objetivos de prueba. o Riesgos de seguridad. o Estándares de seguridad y vulnerabilidades conocidas. o Defensas implementadas para asegurar el sistema y sus datos. o Alcance de la prueba de seguridad. o Capacidad de aplicación de las herramientas de prueba de seguridad.
Implementación y Ejecución de la Prueba	Implementación y Ejecución de la Prueba de Seguridad - El objetivo es traducir las pruebas conceptuales en pruebas que puedan ser ejecutadas manualmente o con herramientas. Además, el objetivo es realizar estas pruebas utilizando una variedad de perspectivas de pruebas de seguridad - usuario interno, usuario externo, usuario malicioso, etc.	• Crear casos de prueba de seguridad, escenarios de prueba, guiones de prueba u otras especificaciones de prueba. • Realizar pruebas funcionales de seguridad basadas en especificaciones de prueba de seguridad definidas. • Realizar pruebas de seguridad funcionales y de penetración basadas en los conocimientos e intuición del probador. • Realizar la prueba de seguridad basándose en el modelo de un sistema. • Configurar o preparar un entorno de prueba para realizar la prueba de seguridad.

Proceso de Prueba de ISTQB	Proceso de Prueba de Seguridad de ISTQB	Ejemplo de Tareas de la Prueba de Seguridad
Evaluación de los Criterios de Salida y Suministro de Información	Evaluación y Suministro de Información de los Resultados de la Prueba de Seguridad - Esto se realiza a menudo junto con la ejecución de prueba para evaluar las pruebas individuales e informar de las nuevas amenazas tan pronto como sea posible.	• Determinar las vulnerabilidades de seguridad específicas basándose en los resultados de la prueba. • Evaluar los niveles de riesgo para la seguridad basándose en los resultados de las pruebas de seguridad realizadas. • Informar de los resultados provisionales y finales de las pruebas de seguridad a la dirección y a otras partes autorizadas.
Cierre de la Prueba	Cierre de la Prueba - El objetivo es llevar las actividades de prueba de seguridad a un punto de cierre para que las pruebas puedan ser mantenidas y llevadas a cabo de forma regular para apoyar cualquier nuevo requisito de seguridad y/o detectar cualquier nueva amenaza. Además, todos los productos de prueba de seguridad y los resultados se almacenan de forma segura, a la vez que están disponibles para su uso si se necesitan en futuras pruebas de seguridad.	• Asegurar que se han realizado todas las pruebas de seguridad planificadas. • Determinar si se han entregado los entregables de la prueba de seguridad (informes). • Archivar los resultados de la prueba, los datos de la prueba y otra información sensible en lugares seguros. • Analizar los resultados de la prueba de seguridad para mejorar el desarrollo de sistemas y aplicaciones en términos de seguridad.

Es importante entender que el Proceso de Prueba de Seguridad de ISTQB no es necesariamente de naturaleza secuencial. El proceso de prueba de seguridad debe alinearse con el proceso del ciclo de vida del software de la organización. Una de las principales implicaciones del proceso descrito en esta sección es que las actividades de la prueba de seguridad se realizan junto con otras actividades y pruebas del ciclo de vida del proyecto y durante las mismas.

Además, las tareas de la prueba de seguridad que se muestran en la tabla 3.1 pretenden ser ejemplos, no requisitos prescriptivos para las tareas de la prueba de seguridad. Las tareas de la prueba de seguridad exactas para una organización dependen de la estrategia de prueba de seguridad y del enfoque adoptado por la organización, como se muestra a continuación, en la figura 3.1.

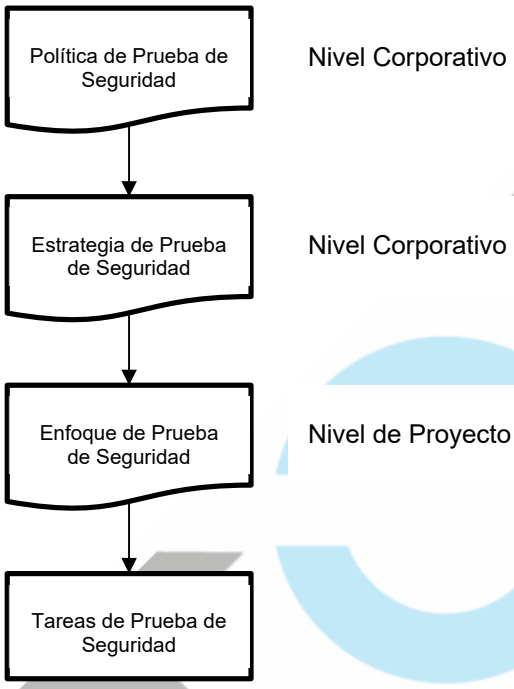

Figura 3.1 - Jerarquía de la Planificación de la Prueba de Seguridad

3.1.2 Alineación del Proceso de Prueba de Seguridad con un Modelo del Ciclo de Vida de una Aplicación Particular

Cada uno de los siguientes tipos de procesos del ciclo de vida tiene asuntos de interés relativos a la seguridad. Es importante alinear la prueba de seguridad para que concuerde con el ciclo de vida.ç

Ciclos de Vida Secuenciales

En estos proyectos, el probador de seguridad debe tener en cuenta los siguientes aspectos:

- Las necesidades y los riesgos de seguridad se definen al principio del proyecto y deben documentarse en las especificaciones de requisitos software.
- Las necesidades de seguridad pueden cambiar durante el proyecto, pero pueden no reflejarse en los requisitos software actualizados. Las pruebas de seguridad pueden parecer muy específicas y completas, pero pueden no estar completas o actualizadas debido a los riesgos de proyecto tardíos.
- Las pruebas de seguridad pueden realizarse en cualquier momento, pero es habitual que estas pruebas se realicen en una fase tardía del proyecto.
- Puede ser difícil abordar los resultados de la prueba de seguridad al final de un proyecto de ciclo de vida secuencial.

Ciclos de Vida Iterativos/Incrementales

Los proyectos incrementales se caracterizan por realizar entregas de una aplicación pequeñas y frecuentes. Los métodos ágiles son un ejemplo de este enfoque. En estos proyectos, el probador de seguridad debe tener en cuenta los siguientes aspectos:

- Las necesidades y los riesgos de seguridad surgen a lo largo del proyecto (normalmente en el contexto de una iteración o esprint) y pueden definirse en especificaciones de requisitos, historias de usuario, modelos, criterios de aceptación y/o prototipos.
- Las necesidades y los riesgos de seguridad pueden cambiar durante el proyecto y pueden (deben) abordarse en la iteración en la que se identifican.
- Las pruebas de seguridad pueden realizarse de forma continua a lo largo del proyecto.
- Dependiendo de la naturaleza del riesgo de seguridad, puede que no sea posible mitigarlo y probarlo completamente durante un ciclo de entrega corto.

Software Comercial de Distribución Masiva[11] (o COTS por su acrónimo en inglés)

Estos proyectos suelen ser de naturaleza "caja negra" y pueden o no ser personalizados. A menudo contienen vulnerabilidades de seguridad hasta el punto de que se requieren frecuentes actualizaciones y parches de seguridad. No hay acceso al código, por lo que no es posible realizar análisis y/o prueba estructural.

Software de Código Abierto

Se trata de una variante de Software Comercial de Distribución Masiva, pero con una distinción importante: el código está disponible para que todo el mundo lo vea. Estos productos también tienen vulnerabilidades de seguridad, por lo que es de vital importancia que los parches de seguridad se mantengan actualizados. Una vez que se ha hecho pública una vulnerabilidad de seguridad, los usuarios de esa versión concreta del software (y anteriores) corren el riesgo de sufrir un ataque.

Ejemplo - Proceso de Prueba de Seguridad en un Ciclo de Vida Secuencial

Es importante señalar que la prueba de seguridad no necesita limitarse a una fase o actividad de un proyecto. Es especialmente importante evitar la situación en la que la prueba de seguridad (y otras pruebas) no se realiza hasta la fase de aceptación del proyecto. Al final del proyecto, resulta especialmente costoso y arriesgado hacer frente a cualquier defecto descubierto. A continuación se muestran las tareas de pruebas de seguridad adecuadas que se deberían llevar a cabo en cada fase del ciclo de vida secuencial:

- **Requisitos** - Los requisitos de seguridad se definen y revisan como parte del esfuerzo general de requisitos para expresar las necesidades de la organización. Aquí es también donde se pueden escribir los casos de uso. Es en este punto donde debe desarrollarse un enfoque de prueba de seguridad.
- **Análisis y Diseño** - Normalmente, alguien en el rol de Analista de Negocio examinará el enunciado inicial de los requisitos y los perfeccionará para llenar los vacíos. A continuación, un

[11] **Software Comercial de Distribución Masiva es la traducción del término "Commercial Off-the-Shelf software (COTS)"**

analizador y/o arquitecto de sistemas analizará los requisitos para proponer la forma más óptima de ofrecer una solución que satisfaga las necesidades de usuario. En este caso, la seguridad sería una de las necesidades funcionales y no funcionales, junto con otras como la usabilidad y la eficiencia. En este punto, los diseñadores de la prueba de seguridad pueden hacerse una idea de la arquitectura y de lo que se necesita probar desde el punto de vista de la seguridad estructural y funcional. En este momento deben definirse los principales objetivos de la prueba de seguridad.

- **Diseño Detallado** - En este punto se diseñan las interfaces de usuario y las bases de datos. Se refinan las reglas funcionales y el diseño de la prueba de seguridad se vuelve más detallado. Las primeras pruebas de seguridad pueden realizarse basándose en modelos.

- **Codificación/implementación** - Es cuando las especificaciones del diseño se implementan como código. Esta es la primera oportunidad de probar la estructura de la aplicación, incluyendo la prueba de vulnerabilidades de seguridad como los defectos de desbordamiento de memoria intermedia y las ediciones a nivel de campo que podrían permitir la inyección SQL. El análisis estático y la revisión del código son muy valiosos en esta fase y deben incluir el examen del código desde la perspectiva de la seguridad. La prueba de componente es también una actividad clave para verificar que el código funciona como se ha especificado. La prueba de integración entre componentes también puede comenzar a medida que los componentes que se interconectan entre sí estén disponibles para ser probados en conjuntos de tamaño reducido.

- **Prueba de Sistema** - Es la prueba de sistemas y subsistemas. La prueba del sistema incluye el software, el hardware, los datos, los procedimientos y la forma en que las personas interactúan con el sistema. Estas pruebas suelen ser de naturaleza transaccional para probar los procesos de negocio. La base para las pruebas de sistema pueden ser los requisitos, los modelos de diseño, los casos de uso y cualquier otra especificación que transmita la perspectiva del sistema. Además, puede ser necesario realizar pruebas de integración de sistemas para probar cómo se comunican e intercambian datos varios (sub) sistemas. La prueba de seguridad en esta fase adquiere una visión más amplia porque intervienen el hardware y los intercambios de datos. Se puede probar la seguridad de las transacciones, que incluye la autenticación, el almacenamiento de datos, la implementación de cortafuegos, así como los controles de seguridad de los procedimientos.

- **Prueba de Aceptación de Usuario** - Es cuando la prueba valida que un sistema soporta los procesos de negocio del mundo real y puede abarcar múltiples sistemas en múltiples organizaciones. El objetivo de esta fase no es tanto encontrar defectos como validar que el sistema satisface las necesidades de usuario en condiciones reales. Esto incluye asegurar que los requisitos de seguridad se han implementado y cumplido correctamente. En esta fase, la prueba de seguridad ya se debería haber realizado en gran medida, pero todavía hay oportunidades para probar los escenarios de seguridad que se producen en el nivel del proceso de negocio.

- **Despliegue** - Es el momento en que el sistema terminado y probado se despliega a los usuarios. Hay muchas formas en que esto puede ocurrir, como en despliegues piloto a grupos seleccionados, o un despliegue masivo a todos los usuarios. Otro enfoque es un despliegue paralelo en el que un sistema antiguo y un sistema nuevo están en funcionamiento al mismo tiempo durante un tiempo limitado. Gran parte de la decisión de una implementación directa depende del riesgo de desplegar a todos los usuarios y de la confianza obtenida durante las pruebas de aceptación. La seguridad es un asunto de interés durante el despliegue del sistema, ya que todos los componentes del sistema deben desplegarse de forma que no se introduzcan nuevas vulnerabilidades. Esto puede ocurrir si las configuraciones de seguridad no son correctas en el entorno de destino. Un ejemplo de ello sería si los derechos de acceso a la base de datos no son correctos en el entorno real.

- **Mantenimiento** - A medida que surgen nuevas necesidades o se descubren defectos después del despliegue, se realiza el mantenimiento. La prueba adquiere una dimensión diferente, ya que se concentra en probar los cambios y en realizar pruebas de regresión. También se deberían realizar pruebas de seguridad para asegurar que no se introducen nuevas vulnerabilidades durante los cambios. Parte del proceso de mantenimiento consiste en mantener actualizados los cortafuegos y otras tecnologías de seguridad. La monitorización continua del sistema puede detectar actividades sospechosas que pueden necesitar ser tratadas inmediatamente.

Ejemplo - Proceso de Prueba de Seguridad en un Ciclo de Vida Iterativo/Incremental

Hay una variedad de metodologías que se han introducido en los últimos 20 años para definir la construcción de software en incrementos más pequeños o iteraciones. En este ejemplo, las entregas del software se realizan cada cuatro semanas. La base del trabajo (y de las pruebas) son las historias de usuario, cada una con criterios de aceptación definidos.

La selección de las prestaciones que se van a construir y entregar se basa en una lista de trabajo acumulado priorizada. Las prestaciones seleccionadas deben reflejar los elementos que aportan el mayor valor y que son realizables en el plazo del esprint. El probador de seguridad trabaja con negocio y/o el propietario de producto para tener los requisitos de seguridad adecuados y correctos.

En este ejemplo, se seleccionan cuatro funciones principales de seguridad para la primera iteración porque se necesitarán para desarrollar muchas de las demás prestaciones. Las prestaciones son:

- Inicio de sesión del usuario.
- Habilitación de SSL (Secure Socket Layer).
- Restablecimiento de la contraseña perdida.
- Bloqueo de la cuenta después de tres intentos fallidos.

Cada una de estas prestaciones se describe en forma de historias de usuario y se perfecciona en requisitos más detallados, cada uno de ellos con criterios de aceptación.

Desde la perspectiva de la prueba de seguridad, el probador de seguridad trabaja con el desarrollador para asegurar que las políticas y los protocolos correctos se reflejan en el código. El probador de seguridad también trabajará junto al desarrollador para probar las funciones a medida que se desarrollan.

En este ejemplo, la primera entrega puede ser sólo la página de inicio de sesión y las funciones asociadas a ésta, como el restablecimiento de una contraseña perdida y el control de bloqueo. En la siguiente iteración, se desarrollarán otras funciones, según la prioridad de los implicados. En cada iteración, el probador de seguridad probará que los controles de seguridad funcionan correctamente y que no se han introducido nuevas vulnerabilidades de seguridad. Las iteraciones continuarán hasta que se hayan completado las tareas de la lista de trabajo acumulado.

En ambos ejemplos (iterativo/incremental y secuencial), los pasos del proceso de la prueba de seguridad pueden verse como tareas integrales para asegurar una aplicación segura.

3.2 Planificación de la Prueba de Seguridad

3.2.1 Objetivos de la Planificación de la Prueba de Seguridad

En general, la prueba de seguridad debe concentrarse en dos aspectos:

- Verificar que las defensas de seguridad diseñadas se implementan y funcionan como se han diseñado.
- Verificar que no se introducen vulnerabilidades durante el desarrollo de la aplicación.

Como se ha mencionado anteriormente en este programa de estudio, todas las defensas de seguridad que se implementen deben basarse en un análisis del riesgo. Esto proporciona un punto de partida a la hora de planificar la prueba de seguridad de un proyecto.

Muchas de las vulnerabilidades desarrolladas involuntariamente pueden evitarse utilizando actividades de aseguramiento de la calidad y buenas prácticas durante las actividades de arquitectura, diseño y codificación. Probar si se introducen vulnerabilidades comienza con una evaluación de las prácticas utilizadas por el equipo de desarrollo. En función del resultado, puede ser necesario seleccionar e introducir pruebas de seguridad adicionales.

3.2.2 Elementos Clave del Plan de Prueba de Seguridad

A continuación, se enumeran los elementos clave de un plan de prueba de seguridad. Cada uno de estos elementos se puede determinar formulando las preguntas especificadas para un proyecto determinado.

- Identificación del alcance de la prueba de seguridad:
  - ¿Qué está incluido y qué está excluido del alcance?
  - ¿Qué se puede conseguir teniendo en cuenta los recursos del proyecto, los riesgos de seguridad y las limitaciones de tiempo?
- Identificar quién debe realizar la prueba de seguridad:
  - ¿La organización cuenta con personas con las competencias adecuadas para realizar la prueba de seguridad?
  - ¿La organización se siente cómoda con la externalización de la prueba de seguridad?
  - En el caso del software comercial y del desarrollado por el proveedor, ¿qué pruebas de seguridad son responsabilidad del proveedor y cuáles del cliente?
  - ¿Los probadores de seguridad necesitan formación en el uso de herramientas de prueba de la seguridad específicas?
- Determinar un calendario adecuado para probar la seguridad teniendo en cuenta otros requisitos relativos al calendario para la prueba del proyecto:
  - ¿Qué elementos relacionados con la seguridad necesitan ser implementados y probados antes de que se realicen otras pruebas? (por ejemplo, los derechos de acceso y los inicios de sesión).

- ¿Cuándo estarán disponibles las prestaciones de seguridad para ser probadas?
- ¿Cuánto tiempo se tardará en probar la seguridad teniendo en cuenta los recursos y el alcance previstos?

- Definir las tareas a realizar y el tiempo necesario para cada una:
  - ¿Cuánto tiempo se necesita para diseñar las pruebas de seguridad adecuadas en función de los recursos y el alcance previstos?
  - ¿Cuánto tiempo se necesita para evaluar e informar los resultados de las pruebas de seguridad?
  - ¿Cuánto tiempo se necesita para realizar las pruebas de regresión relacionadas con la seguridad?
  - ¿Cuánto tiempo se necesita para establecer el entorno de prueba de seguridad?
- Definición del o de los entornos de prueba de seguridad:
  - ¿Qué amplitud tiene el entorno? (plataforma, tecnología, tamaño, ubicación).
  - ¿Se trata de un entorno nuevo?
  - ¿Qué herramientas de prueba de seguridad y otras herramientas de prueba necesitan instalarse en el entorno?
- Obtención de autorizaciones y aprobaciones para las actividades en materia de prueba de seguridad:
  - ¿Quién necesita autorizar y aprobar las pruebas de seguridad?
  - ¿Cuándo se necesita esa autorización?
  - ¿Son suficientes el presupuesto y los fondos?

Como cualquier entregable de proyecto, el plan de prueba de seguridad debe ser revisado para evaluar su completitud y corrección. Dado que las pruebas de seguridad suelen ser de naturaleza técnica, una sesión de revisión técnica puede ser el método más adecuado. Sin embargo, las revisiones guiadas y las inspecciones también pueden ser adecuadas.

Una lista de comprobación estándar puede ayudar a formar la base de lo que se cubre en una sesión de revisión. Al igual que cualquier otra revisión, la retroalimentación debe ser constructiva y no estar dirigida a quien ha elaborado el plan de prueba de seguridad. El equipo de revisión debe incluir a personas con conocimientos de todas las áreas afectadas por los aspectos de seguridad tratados en el plan de prueba de seguridad. Los miembros del equipo de revisión no tienen por qué ser necesariamente probadores de seguridad o poseer conocimientos de seguridad. Por ejemplo, el director de una unidad de negocio puede tener información sobre los riesgos de seguridad que deberían registrarse en el plan de pruebas de seguridad. Los auditores de TI y los administradores de seguridad son especialmente útiles en las revisiones del plan de prueba de seguridad debido a su conocimiento de las políticas y los procedimientos de seguridad.

3.3 Diseño de la Prueba de Seguridad

Hay varias formas de empezar el diseño de una prueba de seguridad. Por ejemplo, se podría comenzar:

- Basándose en un análisis del riesgo realizado.
- Basado en un modelo de amenaza disponible.
- Basado en una categorización de origen ad hoc de los riesgos de seguridad (véase [ISTQB_ATTA_SYL]) Cualquiera de ellas puede constituir una base viable.

Dependiendo del tipo de proyecto, es importante asegurar que hay pruebas de seguridad en cada fase de desarrollo aplicable.

3.3.1 Diseño de la Prueba de Seguridad

Las pruebas de seguridad detalladas se basan en los riesgos de seguridad, en una estrategia de pruebas de seguridad y en otras fuentes como los modelos de amenazas. Las pruebas de seguridad también pueden considerarse de naturaleza funcional y estructural. Por ejemplo, en el caso de las pruebas de seguridad de un sitio web de comercio electrónico, los riesgos de seguridad funcionales pueden ser la inyección SQL, la cosecha de cuentas y el reventado de contraseñas. Un ejemplo de riesgo de seguridad estructural sería una condición de desbordamiento de memoria intermedia que permitiera a un atacante acceder a través de un fallo de memoria.

Los siguientes son atributos esenciales de las pruebas de seguridad detalladas:

- Priorizados por los riesgos de seguridad identificados y los modelos de amenaza.
- Trazados a requisitos de seguridad definidos.
- Definidos en función del público al que van dirigidos (desarrolladores, probadores funcionales, probadores de seguridad).
- Definidos en función de perfiles de defectos de seguridad.
- Diseñados para ser automatizados, si fuera aplicable.

El flujo básico del diseño de una prueba de seguridad puede verse como:

1. El enfoque de prueba de seguridad (nivel de proyecto).
2. Los riesgos de seguridad de las pruebas, los modelos de amenazas y los requisitos (nivel de proyecto).
3. Técnicas de diseño de pruebas de seguridad (basadas en los riesgos, los requisitos y la aplicación).
4. Casos y escenarios de prueba de seguridad.

En el resto de este capítulo, se presentan los riesgos y vulnerabilidades de seguridad más comunes junto con la técnica de diseño de prueba de seguridad asociada. Los nuevos riesgos de seguridad y las vulnerabilidades surgen con rapidez, por lo que se aconseja que los planificadores de las pruebas de

seguridad se mantengan al día con los estándares de seguridad y las listas de amenazas, como se indica en el capítulo 9.

Un principio clave es que un proceso de diseño de una prueba de seguridad debe ser capaz de crear e implementar pruebas basadas en cualquier riesgo, requisito o amenaza de seguridad identificados.

Controles de Seguridad Funcional (por ejemplo, controles de transacción)

Estas pruebas están diseñadas para verificar y validar que los controles se encuentran en su lugar, funcionan correctamente y son eficaces para detectar y prevenir acciones no autorizadas.

Ejemplo: El empleado de un banco no puede autorizar una retirada de efectivo que supere una determinada cantidad de dinero sin la autorización del responsable de los empleados del banco introducida en el sistema.

Controles de Acceso Funcional (por ejemplo, inicios de sesión, contraseñas, testigos[12])

Posiblemente, la mayoría de las personas piense inmediatamente en estas pruebas de seguridad. Las pruebas incluyen:

- La aplicación correcta de las políticas de nombres de usuario y contraseñas.
- El nivel de control de acceso es adecuado para el riesgo.
- Los controles de acceso son resistentes al software de reventado de contraseñas.

Ejemplo: La cosecha de cuentas es la práctica de identificar un nombre de usuario. Una vez adivinado o identificado el nombre de usuario, la contraseña es la pieza restante que se necesita para obtener acceso al sistema. Una prueba común es verificar que cuando se introduce un nombre de usuario correcto con una contraseña incorrecta, el mensaje de error no indica cuál de los elementos es incorrecto.

Controles de Acceso Estructural (por ejemplo, derechos de acceso de los usuarios, niveles de cifrado, autenticación)

Las pruebas de estos controles se basan en cómo se han establecido los derechos de usuario para el acceso a los datos, el acceso funcional y los niveles de privacidad. Los controles de acceso estructural suelen ser aplicados por un administrador de sistemas, un administrador de seguridad o un administrador de bases de datos. En algunos casos, los derechos de acceso son una opción de configuración en una aplicación. En otros casos, los derechos de acceso se aplican a nivel de la infraestructura del sistema.

Las pruebas de los controles de acceso estructural incluyen la creación de cuentas de usuario de prueba para cada nivel de acceso de seguridad y la verificación de que cada nivel de acceso no tiene derechos de acceso restringidos para ese nivel. Por ejemplo, se crearían cuentas de usuario para un nivel de acceso mínimo, para un nivel de acceso de gerente y para un nivel de acceso de administrador. Deben realizarse pruebas para asegurar que un usuario con acceso mínimo no pueda realizar actividades de acceso a nivel de gestor.

Prácticas de Codificación Segura

[12] **"testigo" es la traducción del término "token".**

Se trata principalmente de un método de prueba estática para determinar si los desarrolladores de software y sistemas siguen los métodos de seguridad establecidos durante la creación de las aplicaciones.

Un principio clave es que muchos ataques contra la seguridad se logran a través de la explotación de defectos de software para hacer que un sistema se comporte de una manera inesperada.

Una lista muy breve de prácticas de código seguro incluye:

- Se utilizan algoritmos y controles de gestión de sesiones probados para crear identificadores de sesión aleatorios.
- Las decisiones de autorización son tomadas sólo por objetos del sistema de confianza bajo el control de la organización que proporciona la autorización (por ejemplo, la autorización debería ocurrir en el lado del servidor).
- La información segura no debe aparecer en los mensajes de error. Esta información puede incluir detalles del sistema, identificadores de sesión e información de la cuenta.
- Los errores de la aplicación deberían gestionarse dentro de la misma, en lugar de depender de la configuración del servidor.
- Las solicitudes HTTP GET no deben incluir información sensible.
- El tratamiento de errores no debe mostrar el trazado de la pila u otra información propio de depuración.
- Todos los fallos de validación de entrada de datos deben registrarse.
- Cualquier información sensible que pueda almacenarse temporalmente en el servidor debe estar protegida (por ejemplo, debe utilizarse el cifrado). Esta información segura temporal debería borrarse cuando ya no se necesite.
- Una aplicación no debería poder emitir órdenes directamente al sistema operativo. En su lugar, deberían utilizarse las IPA ("Application Programming Interface - API") incorporadas para llevar a cabo las tareas del sistema operativo.
- Las contraseñas, las cadenas de conexión u otra información sensible no deben almacenarse en texto claro en los equipos cliente (por ejemplo, en las cookies). Debe prohibirse la incrustación de dicha información en formatos no seguros como Adobe flash, código compilado y MS viewstate.
- Debe utilizarse el cifrado para la transmisión de toda la información sensible. La seguridad de la capa de transporte (TLS) es una forma de proteger los datos en tránsito cuando se utilizan conexiones HTTP. En el caso de las conexiones que no sean HTTP, debe utilizarse el cifrado para la transmisión de información sensible.
- Los datos suministrados por el usuario no se deben directamente a ninguna función dinámica de "inclusión".
- Todos los datos proporcionados por el usuario deben ser saneados y validados adecuadamente antes de ser utilizados por la aplicación.
- Las variables deben estar fuertemente tipadas en lenguajes que soporten la comprobación de tipos. Es decir, las variables deberían tener un tipo de entrada definido. Por ejemplo, un campo

numérico no debería aceptar caracteres alfa. Esta restricción se definiría en la definición del tipo de la variable, así como en la base de datos. Es posible escribir código seguro en JavaScript (Node JS) o en otros lenguajes que no admiten la comprobación de tipos reforzada por el compilador.

- En lugar de utilizar código nuevo no gestionado para tareas comunes, utilice código probado, de confianza y aprobado que esté bajo gestión de la configuración.
- Ejecute los servicios con los menores privilegios posibles (nunca bajo root) y cada servicio debe tener su propia cuenta de usuario en el sistema operativo.

Se puede encontrar una lista de prácticas de codificación seguras en la Guía de referencia rápida de prácticas de codificación seguras de OWASP [OWASP1] y en Top 10 Secure Coding Practices [CERT1]. Además, SANS recopila una lista de los 25 errores de software más peligrosos en [SANS1].

Se pueden realizar pruebas dinámicas para determinar si los desarrolladores han seguido prácticas como la validación de datos y el envío de mensajes de error. Además, una de las vulnerabilidades de seguridad más comunes, el desbordamiento de memoria intermedia, puede identificarse con herramientas de prueba de memoria dinámica.

Acceso al Sistema Operativo

Una vez que se obtiene acceso al sistema operativo, un atacante puede controlar los datos, el acceso a la red e introducir software malicioso. Las pruebas para esto pueden incluir la comprobación de la capacidad de plantar rootkits y otros códigos maliciosos en un sistema.

Vulnerabilidades del Lenguaje (por ejemplo, Java)

Según los investigadores de seguridad de WhiteHat Security, un proveedor de seguridad de aplicaciones, en general no hubo diferencias significativas entre los idiomas cuando se trata de vulnerabilidades de seguridad. [WhiteHat Security, 2014] En abril de 2014, WhiteHat security publicó un informe de estadísticas de seguridad de sitios web basado en evaluaciones de vulnerabilidad realizadas contra 30.000 sitios web de clientes utilizando un escáner propio y los resultados indicaron diferencias insignificantes en la seguridad relativa de lenguajes como .NET, Java, PHP, ASP, ColdFusion y Perl. Esos seis lenguajes compartían un número medio de vulnerabilidades relativamente similar, y problemas como la inyección SQL y las vulnerabilidades de scripting entre sitios seguían siendo generalizados. [WhiteHat Security, 2014] Es importante reconocer que se puede conseguir un código seguro con muchos lenguajes, al igual que un código no seguro. El factor clave es cómo se codifica (implementa) una aplicación en cualquier lenguaje que se utilice.

La División CERT del Instituto de Ingeniería del Software ofrece publicaciones [CERT2] y herramientas [CERT3] que abordan problemas de seguridad específicos de cada lenguaje. Además, la base de datos de notas de vulnerabilidad [CERT4] proporciona información puntual sobre las vulnerabilidades del software. Las notas de vulnerabilidad incluyen resúmenes, detalles técnicos, información de remediación y listas de proveedores afectados.

Vulnerabilidades de Plataforma (por ejemplo, Windows, Linux, Mac OS, iOS, Android)

Cada plataforma informática tiene su propio conjunto de vulnerabilidades de seguridad. El asunto de interés para el probador de seguridad es asegurar que las actualizaciones de seguridad de la plataforma se aplican con prontitud y en todos los dispositivos que se ejecutan en la plataforma afectada.

Amenazas Externas

Las amenazas externas a la seguridad son las que la mayoría de las personas consideran cuando piensan en ciberataques. Algunas amenazas externas, como la explotación de las vulnerabilidades de las aplicaciones o del lenguaje, pueden ser detectadas, probadas y prevenidas.

La denegación de servicio (DdS)[13] es otro tipo de amenaza externa. En general, estos ataques se basan en la sobrecarga de los recursos del sistema o de la aplicación, de forma que el sistema o la aplicación se vuelven inaccesibles para los usuarios legítimos. Los ataques DdS pueden dirigirse al ancho de banda de la red, a la conectividad del sistema o de la aplicación o a servicios o funciones específicas.

Un ataque de denegación de servicio distribuido (DdSD) es un tipo de DdS en el que el ataque se lanza indirectamente utilizando otros recursos informáticos. Las técnicas posibles son la amplificación o el uso de botnets[14], que es un gran número de ordenadores comprometidos anteriormente bajo el control o el mando de un atacante. Un atacante puede obtener el control simplemente originando infecciones de virus o el uso de troyanos. Los ordenadores infectados pueden ser utilizados como agentes, cada uno de los cuales envía tráfico a una víctima específica (red) como objetivo del atacante.

Cuando se utilizan ataques de amplificación o reflexión, el atacante se sirve de una vulnerabilidad (o incluso de una funcionalidad deseada) en protocolos específicos (por ejemplo, DNS o NTP). El atacante envía una gran cantidad de tráfico a direcciones IP de difusión (múltiples hosts) que contienen la dirección de origen falsificada de la víctima. El resultado es que el servicio de difusión se hace eco de este tráfico hacia la dirección de la víctima y multiplica la cantidad original de tráfico con el número de hosts. Cuando un atacante envía este tipo de solicitudes varias veces por segundo, la víctima se ve repentinamente confrontada con el elevado número de respuestas que tiene que enviar.

Ejemplo: El atacante A envía una solicitud al sistema B para obtener una lista completa de todos los registros DNS conocidos mientras se hace pasar por la víctima C, a menudo con una dirección IP falsa. El sistema B enviará entonces la lista completa a la Víctima C, que inundará el servidor de la Víctima C con una cantidad amplificada de datos.

Otra forma de ataques DdS son los ataques de agotamiento de recursos. Este tipo de ataques abusa de una funcionalidad deseada consumiendo los recursos informáticos (CPU, memoria, almacenamiento en disco, etc.) que se necesitan para proporcionar la funcionalidad.

Ejemplo: Una de las funcionalidades del protocolo SSL es la opción de generar nuevas claves en una sesión existente si el cliente o el servidor sospechan que la sesión está comprometida. La generación de claves es un proceso que consume muchos recursos. Cuando un atacante envía una solicitud para generar nuevas claves varias veces por segundo, un sistema mal configurado o desprotegido puede acabar en una situación en la que sólo genera nuevas claves y no le quedan recursos para hacer otras cosas.

Por último, están los llamados ataques DdS lógicos, en los que un atacante puede abusar de la funcionalidad prevista para impedir que otros usuarios accedan al sistema.

Ejemplo: Una aplicación utiliza nombres de usuario predecibles y bloquea a un usuario de forma permanente después de tres intentos fallidos de inicio de sesión. Un atacante puede adivinar los nombres de usuario y bloquear muchas cuentas en el sistema, provocando que muchos usuarios no puedan acceder a él (e indirectamente haciendo un DdS al servicio de asistencia).

[13] **"denegación de servicio (DdS)" es la traducción del término "Denial of Service (DoS)".**
[14] **"botnet" término pendiente de traducción.**

Hay cuatro niveles de prueba para el DdSD.

1. Prueba para asegurar que los ordenadores no están infectados con software malicioso conocido.
2. Probar la capacidad de los sistemas de detección de intrusos para identificar rápidamente múltiples solicitudes de un solo ordenador en un corto período de tiempo.
3. Identificar las configuraciones que permiten funcionalidades de las que puede abusar un atacante (por ejemplo, SSL, servidor web, DNS).
4. Identificar los defectos lógicos que pueden permitir una DdS.

Las intrusiones son otra forma de ataque externo. Hay muchas formas de lograr una intrusión externa en un sistema. Estos ataques se basan en que alguien "irrumpa" en un sistema para obtener información. Algunos de los métodos se mencionan en la siguiente lista:

- Ingeniería social.
- Ataques de inyección (SQL, código malicioso).
- Compromiso de cuentas (cosecha de cuentas, restablecimiento de contraseñas).
- Explotación de vulnerabilidades conocidas (cortafuegos, sistema operativo, marco de trabajo, aplicación).
- Ataques de software malicioso.
- Ataques de configuración insegura.
- Defectos de autorización.
- Ataques a la lógica de la aplicación (aprovechamiento de los defectos de la aplicación, especialmente en las aplicaciones basadas en la web, para hacer un uso indebido de la función; por ejemplo, realizar pasos fuera de orden en una aplicación de compra de comercio electrónico para conseguir un descuento o un crédito).

Interceptar una transmisión de red enviada desde dentro de una organización a alguien de otra organización no se tiene en cuenta como un ataque de intrusión, sino como una brecha interna.

Amenazas Internas

Las mayores amenazas pueden ser internas. Se deben tener en cuenta las siguientes fuentes de ataques internos:

- Espionaje corporativo en el que un empleado de confianza puede vender información corporativa, incluyendo información de cuentas de clientes, secretos comerciales, información de acceso de los empleados, etc.
- Información obtenida por desarrolladores, probadores y otro personal subcontratado (como los representantes del servicio de atención al cliente). A veces, las personas dejan el empleo de una empresa subcontratada y se llevan la información en la cabeza.
- El robo de discos duros y otros dispositivos de almacenamiento físico.

- Empleados descontentos que tratan de perjudicar a la empresa filtrando información confidencial, o cometiendo actos de robo pagándose a sí mismos dinero bajo la apariencia de facturas legítimas (pero fraguadas).

Formato y Estructura de la Prueba de Seguridad

Cada organización que realiza una prueba de seguridad tendrá su propia manera de dar formato a las pruebas detalladas. A menudo es posible utilizar el mismo formato para el diseño de pruebas de seguridad que el utilizado para otros tipos de prueba, con la única diferencia del objetivo de la prueba y el entorno de la misma.

Incluso si una organización sigue estándares como el IEEE 829-2008 y el ISO 29119 [ISO/IEC/IEEE 29119-3], el uso de ese estándar debe adaptarse a las necesidades de la organización. Sin embargo, estos estándares conforman un entendimiento estándar de lo que deben contener los distintos documentos de planificación de la prueba. En muchos casos, los casos de prueba y los procedimientos de prueba (guiones) pueden definirse e implementarse en una herramienta de gestión de la prueba que, a menudo, proporciona una estructura de formato.

Los casos de prueba son la forma más autónoma de descripción de una prueba. No requieren una ejecución secuencial. Si se necesita una ejecución secuencial para lograr un objetivo de prueba concreto, los casos de prueba se combinan en una secuencia expresada en un procedimiento o guion de prueba. Los casos de prueba suelen utilizarse para probar condiciones únicas. Por ejemplo, en la prueba de seguridad, la comprobación de la función de inicio de sesión podría consistir en casos de prueba diseñados para validar que los requisitos de formato de la contraseña se cumplen correctamente.

Durante la implementación de la prueba, los casos de prueba se desarrollan, priorizan y organizan en la especificación de procedimiento de prueba. El procedimiento de prueba especifica la secuencia de ejecución de los casos de prueba. Si las pruebas se realizan con una herramienta de ejecución de la prueba, la secuencia de acciones se especifica en un guion de prueba (que es un procedimiento de prueba automatizado). Los procedimientos de prueba se utilizan cuando la secuencia es importante. Por ejemplo, un procedimiento de prueba sería útil para probar el proceso de "recuperación de la contraseña perdida".

Cuando se necesitan pruebas basadas en la experiencia, como las pruebas exploratorias, las condiciones de la prueba y los resultados esperados no se definen de antemano, sino que las condiciones probadas y los resultados reales deben ser registrados por el probador de seguridad para el suministro de información (por ejemplo, a través de informes).

3.3.2 Diseño de la Prueba de Seguridad Basada en Políticas y Procedimientos

Cuando se diseñan pruebas para validar las políticas y los procedimientos de seguridad, estos elementos se convierten en la base de la prueba. Desde esta perspectiva, las pruebas de seguridad son casi un medio de auditoría de seguridad.

Las políticas y los procedimientos de seguridad no deben ser la única base de la prueba porque se necesitan otras perspectivas para la prueba de la seguridad. Los objetivos del diseño de pruebas para validar las políticas y procedimientos de seguridad son:

- Comprender el propósito y el alcance de la política o el procedimiento.
- Evaluar la capacidad de ser probado de la política/procedimiento.
- Crear pruebas relacionadas directamente con la política/procedimiento.

Por ejemplo, puede haber un procedimiento que diga *"Todos los sistemas informáticos de XYZ limitan a tres el número de intentos fallidos de inicio de sesión. Después de tres intentos fallidos de inicio de sesión se producirá un periodo de bloqueo especificado. Las personas que no dispongan de la información de la cuenta de usuario local adecuada no podrán acceder a nuestro sistema informático y deberán ponerse en contacto con los servicios de apoyo informático para verificar su identidad y obtener una contraseña temporal."*

Se trata de un procedimiento susceptible de ser probado, que requeriría los siguientes pasos:

1. Intente iniciar sesión en una aplicación tres veces sin éxito. Al tercer intento fallido debería aparecer un mensaje de bloqueo. Cualquier otro intento de acceder a la cuenta recibirá el mensaje de bloqueo.
2. Póngase en contacto con los servicios de apoyo de TI y verifique su identidad. Se emitirá una contraseña temporal a una dirección de correo electrónico conocida.
3. Inicie sesión con la contraseña temporal. El acceso debería ser concedido.
4. Cree una nueva contraseña que se ajuste a la política de contraseñas. La nueva contraseña debe ser aceptada.
5. Cierre la sesión.
6. Inicie sesión con la contraseña recién creada. Debería concederse el acceso. Tenga en cuenta que el PASO 4 ofrece la oportunidad de probar también la política de contraseñas.

No todas las políticas de seguridad pueden ser probadas de este modo. Por ejemplo, *"El contenido de los registros de auditoría de XYZ, Inc. contiene todos los eventos auditados con sello de fecha/hora y son trazables a individuos específicos. Los registros específicos del fabricante que proporcionen información suficiente para cumplir estos requisitos se considerarán adecuados a efectos de auditoría".*

Aunque no es imposible de probar, se necesitaría definir y realizar una prueba para cubrir todos los eventos auditados. Habría que realizar acciones para desencadenar un conjunto de eventos de muestra que se registraran en los registros de auditoría, y habría que verificar que la exactitud de la información registrada fuera correcta, como la identificación del usuario y el sello de fecha y hora.

3.4 Ejecución de la Prueba de Seguridad

3.4.1 Elementos y Características Clave de un Entorno de Prueba de Seguridad Efectivo

Aunque muchas formas de prueba pueden utilizar un entorno de prueba situado en el mismo servidor y red con otros sistemas, la prueba de seguridad tiene riesgos únicos que requieren un enfoque de construcción del entorno de prueba segregado. Esto es especialmente cierto cuando se prueban aplicaciones que no son de confianza (como las de un proveedor de terceros o de código abierto).

Algunas pruebas de seguridad, como la comprobación de los controles funcionales y la gestión de la sesión, pueden realizarse en un entorno de prueba integrado típico sin que suponga un riesgo elevado. Sin embargo, cuando se prueban códigos desconocidos y no fiables, la posibilidad de que un software

malicioso corrompa un servidor y/o una red hace aconsejable probarlo en un entorno de prueba aislado o virtual.

Los principales atributos de un entorno de prueba de seguridad son los siguientes

1. Aislado - de otros sistemas (dependiendo del nivel de riesgo de software malicioso).
2. Completo - el entorno total necesitará reflejar el entorno de destino (producción) en términos de:
    - Sistemas y aplicaciones sujetos a prueba.
    - Sistemas operativos (versión y configuración exactas).
    - Redes.
    - Middleware.
    - Ordenadores de sobremesa (marca de hardware, procesador, memoria).
    - Dispositivos móviles (fabricante, procesador, memoria, gestión de la energía).
    - Bases de datos.
    - Derechos de acceso.
    - Navegadores y complementos.
    - Aplicaciones coexistentes.
    - Datos (datos de prueba de ingeniería o datos de producción que hayan sido ofuscados).
3. Recuperable: para repetir las pruebas según se necesite y para recuperarse de la corrupción en caso de que se produjera.

3.4.2 La Importancia de la Planificación y las Aprobaciones en la Prueba de Seguridad

Hay varias razones por las que un probador de seguridad debe contar con la aprobación antes de ejecutar pruebas de seguridad:

- En casi todos los países, es contrario a la ley (intentar) acceder a los sistemas de datos y a su información. En algunos países es incluso contrario a la ley tener acceso a las herramientas de prueba de la seguridad. Esto significa que en la mayoría de las actividades de prueba de seguridad una persona estaría infringiendo una o más leyes al hacerlo. La única forma posible de realizar la prueba es obtener una exención del propietario de los datos o del sistema y la aprobación de su dirección.
- Las pruebas de seguridad pueden desencadenar alertas de detección de intrusos y el probador puede parecer un infiltrado malicioso. Las pruebas de penetración son un caso específico en el que dicha autorización es especialmente necesaria.

- Las pruebas de seguridad pueden provocar fallos importantes en el sistema y cortes de servicio. Hay que conocer el riesgo y tomar posibles precauciones.

Sin una autorización previa y específica para las pruebas de seguridad, un probador puede estar violando las políticas y procedimientos de seguridad. Esto puede hacer que el probador esté sujeto a un despido o a un proceso judicial.

Un formulario de autorización para la prueba de seguridad debería contener la siguiente información:

- Nombre de la entidad que da la autorización.
- Nombres del personal y/o de la entidad que realiza la prueba.
- Especificación del trabajo/servicio[15].
- Fechas de autorización (desde/hasta).
- Otros detalles relevantes, como direcciones IP de origen, cuentas de usuario, etc.
- Certificados[16]:
    - El cliente es el propietario del sistema que se va a probar.
    - El cliente tiene autoridad para autorizar la prueba de seguridad.
    - El cliente ha realizado una copia de seguridad de todos los sistemas y datos.
    - El cliente ha probado que el sistema puede restaurarse a partir de las copias de seguridad si fuera necesario.
    - El cliente entiende los riesgos asociados a las pruebas de seguridad.
- Una cláusula de "exención de responsabilidad" para la entidad encargada de la prueba.
- Firmas del representante del cliente autorizado a celebrar dichos acuerdos.

Se puede encontrar un ejemplo de formulario en [OWASP3].

3.5 Evaluación de la Prueba de Seguridad

Al igual que gran parte de la prueba, la evaluación de la prueba de seguridad se realiza durante la ejecución de la prueba a medida que se realizan las pruebas individuales. La evaluación de la prueba de seguridad es la evaluación del resultado de una prueba de seguridad. Cuando se identifican defectos de seguridad (vulnerabilidades), debe presentarse un informe de incidencia en el que se indique, como mínimo, lo siguiente:

- Nombre del probador que observó la vulnerabilidad.

[15] "especificación del trabajo/servicio" es la traducción del término "statement of work".
[16] "certificado" es la traducción del término "attestation".

- Entorno de prueba donde se observó la vulnerabilidad.
- Pasos de la prueba realizados (para facilitar la recreación de los resultados de la prueba).
- Naturaleza de la vulnerabilidad de seguridad.
- Alcance de la vulnerabilidad de seguridad.
- Impacto potencial de la vulnerabilidad de seguridad.
- Sugerencia de curso de acción correctiva.

Los informes de incidencias de la prueba de seguridad pueden archivarse utilizando el mismo sistema de gestión de incidencias que otras formas de prueba. A los informes de prueba de seguridad se les debe asignar una categoría especial y es posible que necesiten ser protegidos para prohibir su visualización por personal no autorizado. Estas situaciones pueden darse cuando:

- La prueba de seguridad está siendo realizada por una organización independiente y los incidentes se informan en una herramienta que tiene pocas restricciones para ver los informes de incidencias.
- Se pueden identificar vulnerabilidades de seguridad, pero no se resuelven inmediatamente.
- El personal interno puede ser considerado una amenaza potencial para aprovecharse de las vulnerabilidades de seguridad.

El auditor de TI debe ser capaz de tomar la decisión de restringir o no el acceso a los resultados de las pruebas de seguridad.

Al concluir un esfuerzo importante en materia de prueba de seguridad, como por ejemplo al concluir la prueba de sistema, se puede emitir un informe final de prueba de seguridad. Este informe también puede necesitar ser tenido en cuenta como confidencial, dependiendo del estado de la resolución de la vulnerabilidad.

3.6 Mantenimiento de la Prueba de Seguridad

En muchos casos, modificar el proceso de prueba de seguridad puede consistir únicamente en añadir nuevos tipos de pruebas en respuesta a nuevos tipos de amenazas. Sin embargo, una cosa es cierta. Los objetivos de la prueba de seguridad y las amenazas cambian a diario, por lo que el proceso de prueba de seguridad necesita estar diseñado para cambiar fácilmente.

También aparecen en el mercado nuevas herramientas que ayudan a realizar las pruebas de seguridad. Los probadores de seguridad deben mantenerse al día con estos avances y evaluar qué herramientas podrían añadir potencia y flexibilidad a la prueba de seguridad.

4 La Prueba de Seguridad a lo Largo del Ciclo de Vida del Software

Duración: 225 minutos

Palabras Clave

caso de abuso ("abuse case")

prueba borrosa ("fuzz testing")

Objetivos de Aprendizaje para La Prueba de Seguridad a lo Largo del Ciclo de Vida del Software:

4.1 Rol de la Prueba de Seguridad en el Ciclo de Vida del Software

AS-4.1.1 (K2) Explicar por qué se logra una mejor seguridad dentro de un proceso del ciclo de vida.

AS-4.1.2 (K3) Implementar las actividades adecuadas relacionadas con la seguridad para un determinado ciclo de vida del software (por ejemplo, iterativo, secuencial).

4.2 Rol de la Prueba de Seguridad en Requisitos

AS-4.2.1 (K4) Analizar un conjunto de requisitos dado desde la perspectiva de la seguridad para identificar las deficiencias.

4.3 El Rol de la Prueba de Seguridad en Diseño

AS-4.3.1 (K4) Analizar un documento de diseño dado desde la perspectiva de la seguridad para identificar las deficiencias.

4.4 Rol de la Prueba de Seguridad en las Actividades de Implementación

AS-4.4.1 (K2) Comprender el rol de la prueba de seguridad durante la prueba de componente.

AS-4.4.2 (K3) Implementar pruebas de seguridad a nivel de componente (abstracto) dada una especificación de código definida.

AS-4.4.3 (K4) Analizar los resultados de una prueba a nivel de componente dada para determinar la adecuación del código desde el punto de vista de la seguridad

AS-4.4.4 (K2) Comprender el rol de la prueba de seguridad durante la prueba de integración de componentes.

AS-4.4.5 (K3) Implementar pruebas de seguridad de integración de componentes (resumen) dada una especificación de sistema definida

4.5 Rol de la Prueba de Seguridad en las Actividades de Prueba de Sistema y Aceptación

AS-4.5.1 (K3) Implementar un escenario de prueba, extremo a extremo, para la prueba de seguridad que verifique uno o más requisitos de seguridad dados y pruebe un proceso funcional descrito.

AS-4.5.2 (K3) Demostrar la capacidad de definir un conjunto de criterios de aceptación para los aspectos de seguridad de una prueba de aceptación dada.

4.6 Rol de la Prueba de Seguridad en el Mantenimiento

AS-4.6.1 (K3) Implementar un enfoque de prueba de seguridad extremo a extremo/regresión basado en un escenario dado.

4.1 Rol de la Prueba de Seguridad en el Ciclo de Vida del Software

La seguridad no se prueba ni se aplica como parche en una aplicación ya construida. Más bien, se consigue mediante un diseño orientado a la seguridad y una verificación a lo largo del proceso de construcción. Al igual que las pruebas de software en general, las pruebas de seguridad también son un proceso que debe tener lugar dentro del ciclo de vida de desarrollo.

4.1.1 Vista del Ciclo de Vida de la Prueba de Seguridad

Un proceso del ciclo de vida del software proporciona un marco para realizar ciertas actividades en momentos que se alinean con otras actividades. Por ejemplo, las necesidades del usuario deben obtenerse antes de que se produzca el diseño de la aplicación. La selección del ciclo de vida del software depende de la naturaleza de la organización, del proyecto y de factores similares [IEEE 12207]. A efectos de este programa de estudio y de la prueba de seguridad, los conceptos y las técnicas pueden aplicarse a cualquier proceso del ciclo de vida, ya sea secuencial o iterativo.

En el capítulo 3 de este programa de estudio, se describe un proceso de prueba de seguridad que se ajusta a un ejemplo de ciclo de vida del software genérico. Las razones para integrar la prueba de seguridad en el ciclo de vida del software se abordan en las siguientes secciones.

Proporcionar un marco temporal determinado del ciclo de vida en el que se deberían realizar las actividades relacionadas con la seguridad.

Por ejemplo, al capturar y definir las necesidades de los usuarios, el analista de negocio o de sistema debería plantearse preguntas como las siguientes:

- ¿Qué niveles de seguridad de acceso son necesarios?
- ¿Hay activos digitales o físicos que requieran defensas de seguridad especiales?
- ¿Qué grado de "apertura" se pretende que tenga la aplicación?
- ¿Cuáles son los riesgos de seguridad?

Otro ejemplo sería durante la codificación. En este momento, el desarrollador tiene la mejor oportunidad de aplicar prácticas de codificación seguras para evitar ataques como la inyección SQL y los desbordamientos de memoria intermedia. Encontrar este tipo de vulnerabilidades durante las fases posteriores del proyecto es difícil y costoso porque es posible que muchos otros componentes del software también necesiten ser tratados y corregidos de forma similar.

Proporcionar puntos de control para revisión.

Por ejemplo, deben revisarse los requisitos de seguridad o las historias de usuario para asegurar que los aspectos relacionados con la seguridad de las necesidades del usuario se han investigado y documentado adecuadamente. También deben revisarse los cambios de código para detectar la presencia de codificación maliciosa por parte de empleados internos o contratistas.

Proporcionar puntos de control para probar.

Por ejemplo, en el desarrollo, deben documentarse y realizarse pruebas de componentes para verificar que se han seguido y aplicado con éxito las prácticas de codificación segura.

Proporcionar criterios de entrada y salida a lo largo del proyecto.

Un ejemplo de esta práctica sería que ningún componente puede ser aceptado en un entorno de pruebas integrado hasta que se pueda demostrar que todas las actividades relacionadas con la seguridad (tanto el desarrollo como la prueba) se han completado con éxito. Esto es especialmente importante en las últimas fases del proyecto, en las que una vulnerabilidad de seguridad podría causar un riesgo de seguridad que afectara a todo el sistema o la aplicación.

4.1.2 Actividades Relacionadas con la Seguridad en el Ciclo de Vida del Software

Las siguientes actividades relacionadas con la seguridad se llevan a cabo junto con otras actividades del proyecto, en lugar de realizarse en su propio ciclo de vida independiente.

Requisitos: Los requisitos se recopilan y se definen de diversas maneras en función del ciclo de vida del software que se utilice. Hay que reconocer que los requisitos pueden ir más allá de las necesidades de los usuarios y de los implicados. Por ejemplo, puede haber requisitos normativos, requisitos técnicos y requisitos de negocio, entre otros.

Los objetivos de los requisitos incluyen:

- Comprender e identificar las necesidades de seguridad desde todas las perspectivas dentro y fuera de la organización. Por ejemplo, el cliente de un negocio no forma parte de la organización, pero necesita que su información privada permanezca segura.
- Documentar las necesidades de seguridad de forma detallada e inequívoca. Esto permite que la implementación y la prueba sean trazables a los requisitos, permitiendo que los requisitos sean verificados y validados.

Las actividades de requisitos incluyen:

- Definir todas las personas impactadas y con conocimientos que puedan contribuir con entradas a los requisitos.
- Utilizar diversos métodos (entrevistas, talleres, etc.) para recoger las necesidades de seguridad expresadas por cada grupo. Esto también puede realizarse durante la educción de otros requisitos.
- Documentar los requisitos de forma que puedan ser revisados y trazados.
- Revisar los requisitos para comprobar su corrección, completitud, capacidad de ser entendido y no ambigüedad.

Diseño: El sistema o la aplicación se diseñan en función de las necesidades expuestas en los requisitos. Los requisitos expresan las necesidades de seguridad, mientras que el diseño traduce las necesidades en un enfoque de solución viable.

Los objetivos del diseño incluyen:

- Crear un diseño de sistema o aplicación que cumpla los requisitos de seguridad establecidos.

Las actividades de diseño incluyen:

- Analizar los requisitos documentados.

- Lograr el enfoque más factible para desarrollar la aplicación de forma segura.
- Documentar el diseño utilizando las técnicas adecuadas de acuerdo con el ciclo de vida del software. Por ejemplo, en un enfoque iterativo, las sesiones de diseño pueden llevarse a cabo en una pizarra, mientras que en otros procesos el diseño puede expresarse en modelos.

Implementación: Es lo que comúnmente se conoce como la actividad de codificación.

Los objetivos de la implementación incluyen:

- Traducir los requisitos y el diseño en un código seguro que satisfaga las necesidades funcionales expuestas en los requisitos.
- Implementar cualquier otro procedimiento o tecnología necesarios (cortafuegos, testigos, etc.) para satisfacer las necesidades de seguridad.

Las actividades de implementación incluyen:

- Crear un código que cumpla los requisitos de seguridad.
- Realizar la prueba de componentes para verificar la corrección, la eficiencia y la seguridad de la implementación.
- Realizar revisiones de componentes para inspeccionar visualmente la corrección, eficiencia y seguridad de la implementación.

Prueba de Sistema:

Se debe tener en cuenta que algunos modelos de ciclo de vida de software, como los enfoques de entrega iterativos, añaden nuevos componentes o perfeccionan los existentes en un periodo de tiempo más corto y pueden realizar la prueba de sistema con mucha más frecuencia que otros enfoques más secuenciales.

Los objetivos de la prueba de sistema incluyen:

- Realizar una prueba de extremo a extremo para observar el funcionamiento general y el rendimiento de todo el sistema (hardware, software, datos, personas y procedimientos) después de que los distintos componentes del sistema se hayan implementado e integrado en un sistema completo.
- Probar que los requisitos de seguridad se han implementado correctamente desde la perspectiva de sistema.

Las actividades de prueba de sistema incluyen:

- Llevar a cabo la prueba de seguridad en alguna aproximación al entorno final de destino, lo que requiere una transición desde el entorno de desarrollo en el que han tenido lugar las actividades previas de implementación e integración.

Prueba de Aceptación:

Se trata del último nivel de la prueba durante el cual los usuarios o los representantes de los usuarios del sistema desarrollan confianza en que el sistema ofrecerá las capacidades necesarias en el entorno objetivo.

Los objetivos de la prueba de aceptación incluyen:

- Que los usuarios, o los agentes que actúan en nombre de los usuarios, realicen la prueba de seguridad en función de los criterios de aceptación relacionados con la seguridad establecidos para el sistema. Muchas veces, los criterios de aceptación relacionados con la seguridad se concentran en los controles y procesos de seguridad funcionales.

Las actividades de la prueba de aceptación incluyen:

- Instalar el sistema en su entorno de operaciones.
- Realizar pruebas de seguridad basadas en los criterios de aceptación.
- Determinar la aceptación basándose en los resultados de las pruebas.

Cabe señalar que tanto la prueba de aceptación como la de sistema son esencialmente pruebas de "caja negra" o de estímulo-respuesta sin tener en cuenta la estructura interna o el comportamiento de los componentes dentro del sistema global. Las pruebas de integración y de componente anteriores proporcionan evaluaciones complementarias al tener en cuenta y explotar la arquitectura interna de los componentes y sus interacciones dentro del sistema.

Mantenimiento:

Tras la puesta en servicio de un sistema, puede ser necesario un esfuerzo de desarrollo adicional para corregir defectos en la versión entregada (mantenimiento correctivo), para ajustarse a otros cambios en el entorno operativo (mantenimiento adaptativo) o para ampliar o mejorar las prestaciones (mantenimiento perfectivo).

La perspectiva de la prueba de seguridad para el mantenimiento del sistema se concentra en probar los cambios realizados para corregir defectos (prueba de confirmación) y la funcionalidad principal (prueba de regresión) para:

- Asegurar que no se han introducido nuevas vulnerabilidades en el sistema por las actividades de mantenimiento.
- Verificar que las defensas de seguridad existentes siguen siendo efectivas tras un cambio.

En este contexto, el mantenimiento puede incluir actualizaciones (por ejemplo, del sistema operativo, de las bases de datos), cambios de código, conversiones de datos y migraciones de plataformas.

En esencia, cualquier actividad de mantenimiento debe tratarse con el mismo cuidado y atención que el desarrollo original. De lo contrario, el riesgo de introducir nuevas vulnerabilidades puede poner en grave peligro la seguridad del sistema operativo.

4.2 Rol de la Prueba de Seguridad en Requisitos

Es necesario entender las siguientes consideraciones sobre los requisitos en general:

- Muchas organizaciones se enfrentan al reto de sólo escribir requisitos básicos para el usuario que sean claros, inequívocos, completos, correctos y que puedan ser probados.

- Los requisitos están muy expuestos a cambios a lo largo de un proyecto, por lo que su mantenimiento puede ser un reto.
- Se necesitan competencias especiales para entender las necesidades del usuario y otras necesidades, como el cumplimiento y las necesidades técnicas, antes de poder redactarlas en documentos o introducirlas en herramientas de gestión de requisitos.
- Los requisitos pueden contener lagunas y errores. Por ello, se necesitan tanto la verificación como la validación.
- Los requisitos deben contener necesidades de características de calidad como la seguridad, el rendimiento, la usabilidad, etc. Sin embargo, estos atributos suelen pasarse por alto en favor de la funcionalidad únicamente.

El reto consiste en conseguir que la perspectiva de la seguridad se entienda y se exprese en el conjunto de requisitos de un proyecto. A la hora de evaluar los requisitos, una técnica eficaz es utilizar una lista de comprobación como guía. La lista de comprobación puede contener muchos elementos para cubrir una variedad de temas. Para los atributos relacionados con la seguridad, lo siguiente es un buen punto de partida para la evaluación:

Necesidades Relativas a la Privacidad

- ¿Se han identificado y documentado todos los grupos de usuarios y sus correspondientes necesidades de privacidad de datos?
- ¿Se han identificado todos los tipos de datos afectados por este requisito y se han definido las necesidades de privacidad relacionadas?
- ¿Se han identificado y definido los derechos de acceso de los usuarios?

Necesidades de Cumplimiento (de las Políticas de Seguridad)

- ¿Se han identificado y documentado todas las políticas de seguridad relevantes?
- ¿Se han identificado y documentado las excepciones a las políticas de seguridad?

Vulnerabilidades Comunes - Estas cambiarán con el tiempo a medida que cambien los ataques contra la seguridad, pero deberían definirse como riesgos en el momento de definir los requisitos. También se convierten en la base de las pruebas de seguridad.

- ¿Se han identificado todas las vulnerabilidades de seguridad comunes y conocidas para la prestación que se está documentando como riesgos conocidos?

Capacidad de Ser Probado

- ¿El requisito está redactado de forma que puedan redactarse pruebas de seguridad y otras pruebas basadas en el documento?
- ¿Se han identificado y aclarado los términos ambiguos como "el proceso debe ser seguro" y "el acceso sólo se concede al personal autorizado" para que sean específicos y puedan ser probados?

Usabilidad - Hay compromisos entre la seguridad y la usabilidad. Por ejemplo, el inicio de sesión de un usuario en un sitio web puede ser tan confuso y difícil que los clientes abandonen y se vayan a otra parte.

- ¿Los requisitos reflejan un nivel adecuado del proceso de seguridad en relación con la función que se especifica?
- ¿Los procedimientos de seguridad son claros y comprensibles?
- ¿Se especifican soluciones para los usuarios legítimos que puedan tener problemas para acceder a la información?

Rendimiento - Hay un compromiso entre la seguridad y el rendimiento. Por ejemplo, es posible que un alto nivel de cifrado reduzca el rendimiento.

- ¿Los requisitos reflejan un nivel adecuado de eficiencia en materia de seguridad en relación con la función que se especifica?

4.3 Rol de la Prueba de Seguridad en Diseño

Hay que identificar y evitar las prácticas de diseño que degraden la seguridad. Las actividades relacionadas con las pruebas contribuyen a reconocer los diseños de los sistemas de software que probablemente sean vulnerables a los compromisos y a dirigir el diseño de los sistemas de software que presenten propiedades de seguridad sólidas e identificables.

IEEE Computer Society Center for Secure Design [IEEE1] recomienda estos enfoques clave de diseño:

- Ganar o conceder, pero nunca asumir, la confianza.
- Utilizar un mecanismo de autenticación que no pueda ser eludido o manipulado.
- Autorizar después de autenticar.
- Separar, de forma estricta, los datos y las instrucciones de control, y no procesar nunca las instrucciones de control recibidas de fuentes no fiables.
- Definir un enfoque que asegure la validación explícita de todos los datos.
- Utilizar la criptografía de forma correcta.
- Identificar los datos sensibles y cómo deben tratarse.
- Siempre tener en cuenta a los usuarios.
- Comprender cómo la integración de componentes externos cambia su superficie de ataque.
- Ser flexible a la hora de tener en cuenta futuros cambios en los objetos y actores.

4.4 Rol de la Prueba de Seguridad en las Actividades de Implementación

La prueba de seguridad, como en otros tipos de prueba, comienza en el nivel más bajo de implementación, ejecutando componentes de software separados que se ensamblarán en el sistema global. Después de la evaluación estática de estos componentes la prueba proporciona un nivel adicional de valoración que examina el comportamiento dinámico en respuesta a entradas válidas e inválidas.

4.4.1 Prueba de Seguridad Durante la Prueba de Componente

4.4.1.1 Consideraciones sobre la Prueba de Caja Blanca/Caja de Cristal

Ya se ha señalado que la prueba estática implica toda la gama de actividades de inspección, revisión guiada, auditoría y revisión técnica.

La denominada prueba de caja blanca y/o de caja de cristal (estructural) se refiere a las pruebas diseñadas sobre la base de la visibilidad del diseño o la implementación del software. En cambio, las pruebas de caja negra (funcionales y no funcionales) no se basan en el acceso a ninguna información estructural de este tipo y son simplemente pruebas de estímulo-respuesta.

La prueba de caja blanca puede dirigirse a controles específicos implementados en el módulo y determinar su efectividad. La visibilidad de la estructura de los componentes también permite medir la cobertura de las pruebas, en términos de porcentaje de sentencias ejecutables practicadas, porcentaje de resultados de decisión practicados o porcentaje de caminos lógicos recorridos.

Las pruebas estructurales de seguridad pueden realizarse mediante herramientas de análisis estático automatizadas y herramientas de escaneo de seguridad. La prueba borrosa es una técnica de prueba de seguridad utilizada para descubrir vulnerabilidades de seguridad mediante la entrada de cantidades masivas de datos aleatorios, llamados "fuzz", al componente o sistema sujeto a prueba. La prueba de caja blanca (en pequeños bloques de software, funciones, clases) puede obtener resultados utilizables en mucho menos tiempo que lo que haría una herramienta de prueba borrosa de caja negra.

Las herramientas de prueba borrosa de caja blanca son capaces de detectar la corrupción de memoria, el desbordamiento de memoria intermedia, etc., instrumentalizando el código que se está probando.

Las siguientes vulnerabilidades de seguridad pueden ser identificadas y corregidas durante las pruebas estructurales:

- Desbordamientos de memoria intermedia
- Código malicioso insertado por un empleado interno o un contratista
- "Acceso de puerta trasera" (acceso a través de una interfaz no documentada que sólo conoce el desarrollador y que se ha implementado intencionadamente para eludir los controles de seguridad normales).

4.4.1.2 Consideraciones sobre la Prueba de Seguridad Funcional

La adecuación de las pruebas de seguridad a cualquier nivel debe determinarse confirmando el cumplimiento de los requisitos de seguridad especificados. Esto se añade a la observación de las respuestas a las tensiones no especificadas explícitamente en los requisitos de seguridad, las evaluaciones del riesgo de seguridad y otros documentos similares. La creatividad es necesaria para probar los puntos débiles de la seguridad porque los probadores están investigando aquello que los desarrolladores de software han pasado por alto.

4.4.2 Diseño de la Prueba de Seguridad a Nivel de Componente

Un ejemplo de conjunto de buenas prácticas de codificación de alto nivel se puede encontrar en el artículo "Top 10 Secure Coding Practices" [CERT1] que establece:

"Las pruebas para cualquier componente deben incluir la evaluación de posibles incumplimientos de estas prácticas:

- Validar la entrada.
- Prestar atención a las advertencias [17] del compilador.
- Desarrollar la arquitectura y el diseño de las políticas de seguridad.
- Mantener la sencillez.
- Denegar por defecto.
- Adherirse al principio de mínimo privilegio.
- Sanear los datos enviados a otros sistemas.
- Practicar la defensa en profundidad.
- Utilizar técnicas eficaces de aseguramiento de la calidad.
- Adoptar un estándar de codificación seguro."

Las pruebas realizadas con estas listas de buenas prácticas deberían incluir evaluaciones de posibles incumplimientos de estas prácticas sobre la base de un análisis del riesgo bien documentado que incorpore un modelo de amenaza realista. En otras palabras, hay que concentrarse en los requisitos más cruciales en cuanto a la probabilidad de ataque y las consecuencias del compromiso (derivadas de un peligro).

4.4.3 Análisis de las Pruebas de Seguridad a Nivel de Componente

Una medida clave de la adecuación implica la evaluación de la cobertura de la prueba. De la naturaleza de las pruebas realizadas se obtienen diversas medidas de cobertura.

Las pruebas basadas en los requisitos practican el sistema para proporcionar la confianza de que satisface sus requisitos según lo especificado. Sin tener en cuenta la implementación (caja negra), la cobertura puede medirse como cualquiera de las siguientes:

- Porcentaje del número total de requisitos probados.
- Porcentaje de casos de uso/abuso especificados probados.
- Porcentaje de funciones, escenarios o hilos de misión críticos probados.

La prueba guiada por datos utiliza el sistema para asegurar su comportamiento a través de un rango y una combinación de datos de entrada, intentando elegir el menor número posible de valores de prueba dividiendo el espacio de datos en clases de equivalencia y seleccionando un representante de cada clase (con la expectativa de que los elementos de esta clase sean equivalentes en términos de su capacidad para detectar fallos). Los criterios de cobertura por pares y por N son formas típicas de criterios de cobertura de datos.

[17] **"advertencia" es la traducción del término "warning".**

La prueba basada en modelos permite asegurar la cobertura en términos de una notación de modelado elegida. Cuando el modelo utiliza una notación pre-post, los criterios pueden incluir la cobertura causa-efecto y la cobertura de todos los disyuntos en la postcondición. Para las notaciones de modelado algebraicas, la cobertura de los axiomas es un criterio de cobertura típico.

Para los modelos basados en transiciones, que utilizan gráficos explícitos que contienen nodos y arcos, los criterios de cobertura de los gráficos incluyen el porcentaje de nodos (estados), el porcentaje de transiciones, el porcentaje de pares de transiciones y el porcentaje de ciclos.

La prueba estructural permite asegurar la visibilidad y el análisis de la implementación real. Por simple enumeración, la cobertura de las pruebas suele informarse como el porcentaje de los paquetes, clases, métodos, decisiones o líneas de código ejecutable de la aplicación que son ejecutados por las pruebas. Esto último se denomina cobertura de sentencia.

La complejidad ciclomática es una medida de cuántos caminos independientes diferentes existen a través de un elemento y puede visualizarse en términos de un grafo del flujo de control con nodos (puntos de decisión) y arcos (caminos). El más fuerte de los criterios basados en el flujo de control es la cobertura de caminos, que mide contra todos los caminos de entrada y salida en el grafo del flujo de control. Dado que la prueba exhaustiva de caminos no suele ser practicable debido a los bucles, otros criterios menos estrictos pueden expresarse en términos de caminos lógicos seleccionados considerados críticos (cobertura de caminos críticos) o del porcentaje de resultados de decisión practicados (cobertura de ramas).

4.4.4 Prueba de Seguridad durante la Prueba de Integración de Componentes

A medida que los componentes de nivel inferior se integran en los subsistemas y, finalmente, en el sistema objetivo completo, las posibilidades de brechas de seguridad no son simplemente la suma de las vulnerabilidades de cada uno de los componentes considerados por separado. En su lugar, se hacen posibles nuevos vectores de ataque debido a las interacciones entre componentes y con elementos más amplios del sistema y de la organización.

Por otra parte, algunas interacciones entre componentes pueden mitigar o bloquear posibles secuencias que conduzcan a brechas de seguridad. Una vez más, los probadores de seguridad necesitan ser creativos para buscar lo que, de otro modo, han pasado por alto los desarrolladores.

La prueba de integración puede demostrar la complejidad del diseño de un sistema y la estabilidad de su comportamiento. El enfoque de prueba de integración (por ejemplo, descendente o ascendente) puede afectar a la cronología de la revelación de asuntos de interés a la seguridad o a la necesidad de pruebas adicionales específicas de seguridad.

4.4.5 Diseño de la Prueba de Seguridad en el Nivel de Integración de Componentes

Al igual que con la prueba de componentes, las pruebas de integración deben diseñarse sobre la base de un análisis del riesgo bien documentado que incorpore un modelo de amenaza realista. A medida que los componentes separados se integran entre sí, es necesario tener en cuenta que el andamiaje (en forma de stubs y controladores) puede ser necesario para probar caminos menos completos a través de un sistema durante la integración. A medida que se añaden más componentes implementados al sistema, este andamiaje se elimina de forma incremental, lo que permite una evaluación más completa de la funcionalidad, así como de las nuevas vías de acceso a las vulnerabilidades que podrían explotarse.

4.5 Rol de la Prueba de Seguridad en las Actividades de Prueba de Sistema y Aceptación

4.5.1 Rol de la Prueba de Seguridad en la Prueba de Sistema

La prueba de sistema es el primer ejercicio de extremo a extremo de los componentes totalmente integrados. Aunque suele realizarse en un entorno de desarrollo, debe revelar propiedades emergentes del sistema que no se habrían observado antes de que se completara la integración. Los requisitos de seguridad suelen considerarse junto con uno o varios requisitos funcionales.

Por ejemplo, "En el proceso de hacer x el sistema no debe permitir que ocurra y". A medida que se realicen las pruebas funcionales, el probador debe buscar formas en las que se puedan violar las restricciones de seguridad.

Los requisitos funcionales, incluidos los de seguridad, suelen abordar los imperativos. Otras especificaciones, como los casos de uso, los casos de abuso, los modelos de proceso y los modelos de transición de estado describen procedimientos que pueden utilizarse para definir escenarios de prueba de extremo a extremo para la prueba de seguridad.

4.5.2 Rol de la Prueba de Seguridad en la Prueba de Aceptación

La prueba de aceptación se distingue de la prueba de sistema en que se lleva a cabo en un entorno operativo realista, si no en el escenario real en el que el sistema entrará en operación. Dichas pruebas permiten una evaluación razonable del rendimiento y otros comportamientos basados en las interacciones a través de interfaces externas. También sitúa finalmente al sistema en el escenario en el que los agentes de amenazas externas buscarían encontrar debilidades en el día a día.

Lo ideal es que la prueba de aceptación valide que se han cumplido los objetivos iniciales del proyecto. Esto se consigue diseñando y realizando pruebas para validar que se cumplen los criterios de aceptación. Las necesidades de seguridad deben documentarse en los criterios de aceptación.

El mejor momento para definir y documentar los criterios de aceptación es antes del desarrollo o la compra del sistema. De este modo, se puede llegar a un entendimiento inicial entre el proveedor y el cliente, incluso si ambos se encuentran en la misma organización. También es común que los criterios de aceptación cambien o surjan durante un proyecto, por lo que estos criterios deben ser analizados por su impacto en la prueba de seguridad.

En el contexto de la prueba de seguridad, los criterios de aceptación pueden ser de naturaleza global. Por ejemplo, podría haber puntos de criterios de aceptación que especifiquen lo que es aceptable en términos de seguridad global del sistema. Esto incluiría criterios que se aplican a todas las funciones del sistema, como la autenticación de los usuarios, los derechos de los usuarios, los niveles de encriptación, los registros de auditoría, etc. En otros casos, pueden ser necesarios criterios específicos de aceptación de la seguridad. Por ejemplo, algunas funciones, como la emisión de pagos que superen una determinada cantidad, podrían requerir que dos personas aprobaran el pago.

4.6 Rol de la Prueba de Seguridad en el Mantenimiento

La prueba de regresión pretende confirmar que todos los comportamientos previamente aceptables del sistema permanecen intactos con posterioridad a determinadas modificaciones. En los aspectos negativos de la prueba de seguridad, dicha confirmación consistiría en comprobar que el sistema sigue resistiendo

con éxito los intentos de anular los controles de seguridad establecidos. Las mejoras en la usabilidad o la eficacia son especialmente propensas a sacrificar los controles de seguridad.

Las pruebas de regresión de la seguridad deben centrarse en confirmar la satisfacción de todos los requisitos de seguridad, así como en probar las nuevas vulnerabilidades que puedan haberse introducido durante las actividades de mantenimiento.

A menudo, la prueba de regresión se aplica con una colección de casos de prueba que se basan en la comprobación de funciones individuales. Sin embargo, para la prueba de seguridad, esto suele ser insuficiente para detectar defectos de regresión con impacto en la seguridad. Los escenarios de pruebas de regresión de extremo a extremo son más robustos y proporcionan un mayor nivel de confianza en cuanto a la posibilidad de realizar transacciones completas de forma segura.

Para este tipo de prueba de regresión, debe definirse un conjunto de escenarios de pruebas de seguridad y probarse cada vez que se realice un cambio en el sistema. Es necesario tener en cuenta que los cambios en el sistema pueden incluir el hardware, los archivos de configuración, los sistemas operativos, los SGBD, las redes y el software, así como cualquier otro componente del sistema. Los defectos de regresión pueden aparecer por cambios en cualquiera de ellos. Algunos de los defectos de regresión pueden tener impacto en la seguridad.

Ejemplos de escenarios:

Los usuarios pueden acceder a un sitio web y completar una compra de forma segura sin comprometer su información personal.

Los usuarios sólo pueden realizar las acciones definidas en sus derechos y privilegios de usuario. (Un usuario que trabaja en el departamento de nóminas puede ser capaz de añadir un nuevo empleado, pero no tener acceso a su información bancaria).

5 Prueba de Mecanismos de Seguridad

Duración: 240 minutos

Palabras Clave

"pharming"	("pharming")
amenaza interna	("insider threat")
autenticación	("authentication")
autorización	("authorization")
cifrado	("encryption")
cortafuegos	("firewall")
escaneo de software malicioso	("malware scanning")
escáner de vulnerabilidad	("vulnerability scanner")
fortificación del sistema	("system hardening")
función resumen (o función de extractado)	("hashing")
medida para evitar software malicioso	("anti-malware")
salado	("salting")
sistema de detección de intrusos	("intrusion detection system")
software malicioso	("malware")
suplantación de identidad	("phishing")
zona de red	("network zone")
zona desmilitarizada	("demilitarized zone")

Objetivos de Aprendizaje para Prueba de Mecanismos de Seguridad:

5.1 Fortificación de Sistema

AS-5.1.1	(K2)	Comprender el concepto de fortificación de un sistema y su rol en la mejora de la seguridad
AS-5.1.2	(K3)	Demostrar cómo probar la efectividad de los mecanismos comunes de fortificación de sistema

5.2 Autenticación y Autorización

AS-5.2.1	(K2)	Comprender la relación entre autenticación y autorización y cómo se aplican para asegurar (en términos de protección) los sistemas de información.
AS-5.2.2	(K3)	Demostrar cómo probar la efectividad de los mecanismos comunes de autenticación y autorización.

5.3 Cifrado

AS-5.3.1	(K2)	Comprender el concepto de cifrado y cómo se aplica para asegurar (en términos de protección) los sistemas de información.
AS-5.3.2	(K3)	Demostrar cómo probar la efectividad de los mecanismos comunes de cifrado.

5.4 Cortafuegos y Zonas de Red

AS-5.4.1	(K2)	Comprender el concepto de cortafuegos y el uso de zonas de red y cómo se aplican para asegurar (en términos de protección) los sistemas de información.
AS-5.4.2	(K3)	Demostrar cómo probar la efectividad de las implementaciones de cortafuegos y zonas de red existentes.

5.5 Detección de Intrusiones

AS-5.5.1	(K2)	Comprender el concepto de herramientas de detección de intrusiones y cómo se aplican para asegurar (en términos de protección) los sistemas de información.
AS-5.5.2	(K3)	Demostrar cómo probar la efectividad de las implementaciones de herramientas de detección de intrusiones existentes.

5.6 Escaneo de Software Malicioso

AS-5.6.1	(K2)	Comprender el concepto de herramientas de escaneo de software malicioso y cómo se aplican para asegurar (en términos de protección) los sistemas de información.
AS-5.6.2	(K3)	Demostrar cómo probar la efectividad de las implementaciones de herramientas de escaneo de software malicioso existentes.

5.7 Ofuscación de Datos

AS-5.7.1	(K2)	Comprender el concepto de herramientas de ofuscación de datos y cómo se aplican para asegurar los sistemas de información.
AS-5.7.2	(K3)	Demostrar cómo probar la efectividad de los enfoques de ofuscación de datos

5.8 Formación

AS-5.8.1	(K2)	Comprender el concepto de formación en seguridad como una actividad del ciclo de vida del software y por qué es necesaria para asegurar los sistemas de información
AS-5.8.2	(K3)	Demostrar cómo probar la efectividad de la formación en seguridad

5.1 Fortificación de Sistema

A lo largo de los años, han surgido diversos mecanismos de seguridad como prácticas clave para asegurar los activos digitales y físicos. Cada uno de estos mecanismos puede aplicarse de diversas maneras: algunos a través de herramientas e infraestructuras, otros mediante un esfuerzo manual. Ninguno de estos mecanismos por sí solo es suficiente en la mayoría de los casos para asegurar la información. Cada mecanismo tiene sus propias ventajas y desventajas.

Los probadores de seguridad necesitan comprender los matices de cada línea de defensa, de modo que puedan diseñarse pruebas adecuadas para verificar y validar su efectividad. Los probadores de seguridad de nivel avanzado necesitan comprender las implicaciones de cada uno de los mecanismos descritos en este capítulo para diseñar una arquitectura de prueba que proporcione un marco para la prueba continua de la seguridad.

5.1.1 Comprender el Concepto de Fortificación de Sistema

Los sistemas modernos son cada vez más complejos, por lo que su superficie de ataque crece continuamente. Las vulnerabilidades provienen de errores de diseño (por vulnerabilidades de diseño), de defectos en el código fuente (por vulnerabilidades de construcción) o de la falta de rigor en la configuración de estos sistemas (por vulnerabilidades de configuración).

Fortificar el sistema es el proceso, paso a paso, de reducir la superficie de ataque aplicando una política de seguridad y diferentes capas de protección. El objetivo principal es asegurar el sistema y reducir los riesgos de que la seguridad se vea comprometida.

Dependiendo del contexto, se puede fortificar a diferentes niveles:

- Fortificar un componente de software o hardware.
- Fortificar un producto/aplicación.
- Fortificar un sistema.
- Fortificar un sistema de sistemas.

Las defensas de seguridad técnicas y relativas a la organización que deberían aplicarse incluyen:

- Eliminar el software innecesario (puede contener defectos).
- Eliminar librerías y herramientas de desarrollo innecesarias (pueden contener defectos).
- Eliminar las cuentas/inicios de sesión innecesarios (vectores de ataque).
- Eliminar las aplicaciones (pueden contener defectos) y servicios de red (vectores de ataque) innecesarios.
- Eliminar los periféricos y puertos de hardware innecesarios (por ejemplo, puertos USB, lectores de tarjetas).
- Instalar de forma oportuna parches y actualizaciones en los sistemas (por ejemplo, activar las actualizaciones automáticas).

- Actualizar las configuraciones.
- Seguir las reglas de codificación (evitar las vulnerabilidades "por construcción").
- Configurar el servidor de registro remoto (por ejemplo, rsyslog) para que, en caso de verse comprometida la seguridad, el atacante sólo pueda eliminar los archivos de registro en la máquina comprometida, pero no en el servidor de registro remoto.

Se deberían utilizar los siguientes mecanismos de seguridad:

- Autenticación fuerte y gestión de autorización eficiente (dar sólo los derechos necesarios para realizar acciones a los roles dedicados).
- Cifrado (comunicación y almacenamiento local alojado).
- Cortafuegos (personales, de sistema o de aplicación web) y zonas de seguridad definidas (por ejemplo, ejecución en un entorno aislado[18]).
- Sistema de detección de intrusos[19].
- Medidas contra software malicioso[20]/ software espía[21].
- Ofuscación de datos y aplicaciones (por ejemplo, protección contra la ingeniería inversa)

Fortificar el sistema es vital para proteger los activos sensibles de una organización, pero las normas de seguridad deben aplicarse al nivel adecuado y en equilibrio con la usabilidad del sistema. En el extremo de este compromiso, las protecciones se desactivan porque bloquean la productividad de la organización.

5.1.2 Prueba de la Efectividad de los Mecanismos de Fortificación de Sistemas

Se puede probar la efectividad de los mecanismos para fortificar el sistema de varias maneras. Las pruebas dependerán de la naturaleza del sistema o de la aplicación que se esté fortificando, de la sensibilidad de los activos protegidos y de las amenazas identificadas. Al fortificar el sistema se restringe el acceso al mismo a los roles adecuados, se abren sólo los servicios necesarios y se monitorizan las actualizaciones de las aplicaciones. Por lo tanto, para probar la efectividad de fortificar el sistema, se deben diseñar pruebas para saber si los esfuerzos de fortificación están funcionando, se aplican en los lugares correctos y se aplican de la manera adecuada. También es importante probar si las protecciones de fortificación del sistema son demasiado restrictivas y podrían ser excesivas en vista de los riesgos de seguridad.

Algunas pruebas de fortificación del sistema pueden estar basadas en revisiones o auditorías, mientras que otras pruebas pueden basarse en la capacidad de ciertos grupos de usuarios para realizar determinadas acciones o acceder a ciertos datos.

Las pruebas podrían incluir:

[18] "entorno aislado" o "aislamiento de procesos" son posibles traducciones del término "sand box".
[19] "sistema de detección de intrusos" es la traducción del término "intrusion detection system".
[20] "medida contra software malicioso" es la traducción del término "anti-malware".
[21] "medida contra software espía" es la traducción del término "anti-spyware". "software espía" es la traducción del término "spyware".

- La auditoría de la configuración de los servidores de bases de datos y aplicaciones para verificar que se han modificado las contraseñas por defecto.
- La auditoría de la configuración de los servidores de bases de datos y aplicaciones para verificar que se han modificado las contraseñas por defecto.
- La auditoría de la configuración del sistema para identificar los servicios y puertos de red innecesarios.
- La verificación de las versiones de componentes, librerías y aplicaciones para comprobar que no están obsoletas ni son vulnerables.

Se puede ejecutar un escaneo de vulnerabilidades para facilitar las tareas de evaluación de las mismas, especialmente si el sistema es complejo (por ejemplo, un entorno multi emplazamiento). Se pueden utilizar herramientas de análisis estático para detectar incumplimientos de las reglas de codificación que puedan introducir vulnerabilidades de construcción. Los analizadores orientados a la seguridad pueden ser especialmente útiles para detectar vulnerabilidades.

5.2 Autenticación y Autorización

5.2.1 La Relación entre Autenticación y Autorización

Los activos sensibles de una organización (por ejemplo, los números de las cuentas bancarias de una lista de clientes, el diseño de un nuevo producto) necesitan ser protegidos y deben ser accesibles sólo por una persona autorizada.

La autenticación se basa en la verificación de un identificador de usuario y un testigo[22] para responder a las preguntas:

- Inicio de sesión[23]: ¿quién es el usuario?
- Contraseña[24]: ¿el usuario es realmente quien pretende ser?

Se pueden utilizar diferentes implementaciones de mecanismos de autenticación en función de la necesidad de protegerse contra ataques para apropiarse de la autenticación o robar una contraseña. Entre ellos se encuentran la detección de contraseñas débiles, el empleo de contraseñas de un solo uso (CdUSU)[25], la toma de huellas dactilares, los certificados de software, los certificados en testigos físicos[26] y otros medios de autenticación similares.

Dependiendo de la arquitectura del sistema, del contexto aplicativo y de las necesidades de una organización (facilidad para gestionar el inicio de sesión/contraseña), los mecanismos de autenticación

[22] **"testigo" es la traducción del término "token".**
[23] **"inicio de sesión" es la traducción del término "login".**
[24] **"contraseña" es la traducción del término "password".**
[25] **"contraseña de un solo uso (CdUSU)" es la traducción del término "one-time password (OTP)".**
[26] **"testigo físico" es la traducción del término "hard token".**

pueden incluir la autenticación local, la autenticación de servidor, la autenticación de red, el Inicio de Sesión Único (IdSU)[27] y otros medios similares.

La autorización se utiliza para los siguientes fines:

- Para verificar si el usuario autenticado tiene los derechos para realizar una acción (por ejemplo, el usuario puede iniciar sesión en un servidor pero no puede modificar sus datos, o un usuario está autorizado a utilizar un servidor FTP pero sólo en su espacio dedicado).
- Determinar qué nivel de acceso a los recursos del sistema se debería permitir.

Existe un fuerte vínculo entre la autenticación y la autorización, basado en el principio de que un usuario no autenticado no tiene derechos o tiene derechos restringidos en el sistema (no está autorizado a manipular datos sensibles). Por ejemplo, en el contexto de la página web de un comerciante, un usuario no autorizado puede ver la lista de productos pero, antes de comprar el artículo elegido, debe crear una cuenta de usuario. El usuario autenticado puede comprar un elemento, pero no puede realizar funciones administrativas.

5.2.2 Prueba de la Efectividad de los Mecanismos de Autenticación y Autorización

El objetivo de los atacantes es robar contraseñas o burlar los sistemas para ejecutar acciones no autorizadas. Normalmente, aprovechan diferentes tipos de debilidades: errores de código (falta de filtrado de entrada), versiones antiguas y vulnerables de las librerías, errores de configuración del sistema (mantenimiento de contraseñas por defecto, derechos por defecto) y contraseñas débiles (por ejemplo, la contraseña más utilizada es "123456").

Una organización puede tener un conjunto de reglas relativas a las contraseñas que deben seguirse, pero si el usuario no es diligente a la hora de mantener la seguridad de la contraseña, las reglas de la contraseña no supondrán ninguna diferencia. Además, las reglas de contraseña deben reflejar las buenas prácticas actuales en la definición de contraseñas. Dichas prácticas pueden encontrarse en las Directrices de Construcción de Contraseñas de SANS Institute [SANS2].

Las pruebas para los mecanismos de autenticación y autorización podrían incluir:

- Ataques de fuerza bruta y de diccionario para intentar descubrir las contraseñas de los usuarios. Los primeros pasos podrían ser intentar "123456", "111111", la fecha de nacimiento, el nombre de la mascota, etc.
- Aprovechar la falta de filtrado en las entradas, por ejemplo, para inyectar peticiones SQL con el fin de autenticarse sin ningún inicio de sesión/contraseña conocido.
- Introducir una URI no autorizada (../../ en una cuenta FTP) o una URL (dirección del sitio/admin) para intentar acceder a datos sensibles.

Otro ejemplo podría ser aprovechar una vulnerabilidad en el sistema objetivo (tal vez porque no ha sido actualizado) para provocar un comportamiento no deseado y que suele dar como resultado la obtención del control del sistema y permitir la escalada de privilegios.

[27] **"Inicio de Sesión Único (IdSU)" es la traducción del término "Single Sign-On (SSO)".**

5.3 Cifrado

5.3.1 Comprender el Concepto de Cifrado

Para evitar la divulgación de datos sensibles, aunque se pueda acceder a ellos cuando se almacenan o intercambian entre el cliente y el servidor, se puede utilizar un mecanismo de cifrado. Las funciones resumen (o función de extraído) y el salado son métodos que se utilizan durante el cifrado.

El cifrado es un proceso de codificación de datos (texto plano) en datos cifrados (texto cifrado), utilizando un algoritmo criptográfico y secretos, de tal manera que sólo las personas autorizadas tienen derecho a acceder utilizando un mecanismo de descifrado. El secreto es compartido y sólo lo conocen los usuarios autorizados. El objetivo es que el cifrado sea lo suficientemente fuerte como para impedir que un atacante, que haya conseguido robar los datos cifrados, recupere el texto sin formato. El uso de algoritmos criptográficos ayuda a asegurar la confidencialidad, la integridad, la disponibilidad de los activos sensibles y el repudio de su manipulación.

Los protocolos criptográficos pueden utilizarse para proteger información:

- Almacenada en un sistema, por ejemplo, contraseñas cifradas en una base de datos, disco lógico cifrado, disco duro completo cifrado.
- Durante la comunicación, por ejemplo, correo electrónico cifrado, protocolo de comunicación cifrada (SSL, TLS).

Los protocolos criptográficos, principales y ampliamente conocidos, que se utilizan son:

- Cifrado simétrico: uso de una clave secreta compartida.

Cifrado asimétrico: uso de clave privada y pública.

5.3.2 Prueba de la Efectividad de los Mecanismos de Cifrado Comunes

Se sabe que algunos mecanismos criptográficos son débiles, especialmente debido al corto tamaño de las claves secretas, o a las claves estáticas. Otros mecanismos son vulnerables porque, o bien no se implementan con las buenas prácticas, o bien incorporan defectos de codificación (como el desbordamiento de memoria intermedia).

Las pruebas de los mecanismos de cifrado deberían incluir:

- Pruebas para detectar vulnerabilidades de diseño:
  - Evaluación de que se utilizan los modos correctos en el cifrado simétrico.
  - Verificación de que el tamaño de las claves criptográficas no es demasiado pequeño (por ejemplo, a partir de 2015 una clave RSA inferior a 2048 bits se considera insegura).
  - Validación de la validez de los certificados y capacidad de emitir una alerta si el certificado está autofirmado (se puede utilizar SSL-trip para evitar ataques de tercero interpuesto[28]).

[28] **"ataques de tercero interpuesto" es la traducción del término "Man In The Middle attack".**

- Repetición de ataque (por ejemplo, ataques contra los protocolos de Privacidad Equivalente por Cable (WEP)).
- Ataques contra protocolos criptográficos para verificar su nivel de fortaleza [Bittau].

- Pruebas de vulnerabilidades de construcción:
  - Revisiones de código (por ejemplo, para verificar que la función “random()” por defecto no se utiliza para generar números aleatorios (semilla) porque el algoritmo aleatorio es relativamente fácil de descifrar).
  - Prueba aleatoria[29] para aprovechar comportamientos inesperados (para atacar la seguridad).
  - Ataques temporales[30] (analizando el tiempo de ejecución de los algoritmos criptográficos).
  - Análisis de potencia (utilizado para los dispositivos de hardware cifrados).
- Pruebas de vulnerabilidades de configuración:
  - Evaluación de la configuración de los protocolos criptográficos (por ejemplo, la configuración del lado servidor de la Seguridad de la Capa de Transporte (TLS), los protocolos autorizados del lado cliente, basándose en la guía de configuración de TLS para administradores).
  - Orden de cifrado TLS en el lado servidor, para ver si existe algún medio para reducir de prestaciones o renegociar el cifrado que se está utilizando.
- Pruebas de envejecimiento[31] para verificar los mecanismos de cifrado que puedan haberse vuelto débiles y propensos a ser descifrados[32].

5.4 Cortafuegos y Zonas de Red

5.4.1 Comprender los Cortafuegos

Según [Chapman 2000], "un cortafuegos es un componente o un conjunto de componentes que restringe el acceso entre una red protegida e Internet, o entre otros conjuntos de redes". Un cortafuegos implementa y aplica una política de seguridad basada en la definición de las comunicaciones autorizadas y prohibidas. Un cortafuegos puede estar basado en un dispositivo anfitrión[33] (software que se ejecuta en un único anfitrión y que monitoriza las entradas/salidas de las aplicaciones) o en una red (software que monitoriza el tráfico entre redes).

[29] “prueba aleatoria” es la traducción de los términos “fuzzing” o “fuzz testing”.
[30] “ataque temporal” es la traducción del término “timing attack”.
[31] “prueba de envejecimiento” es la traducción del término “test for aging”.
[32] “descifrado” es la traducción del término “cracked”.
[33] "dispositivo anfitrión” o “anfitrión” son posibles traducciones del término “host”.

La principal tarea de un cortafuegos es controlar el tráfico entre las diferentes zonas de la red de confianza filtrando los datos que fluyen por la red. De este modo, se detecta y bloquea el tráfico malicioso procedente de una zona no fiable.

Una zona de red es una subred identificada con un nivel de confianza definido:

- Internet/zona pública que se tiene en cuenta como no fiable.
- Varias zonas de seguridad denominadas zonas desmilitarizadas o ZDM, con diferentes niveles de confianza.
- Una o varias redes privadas/internas que se tienen en cuenta como las más fiables.

Las zonas de red forman parte de la configuración del cortafuegos: sirven para definir los flujos autorizados entre las distintas redes. Todo el tráfico prohibido se bloquea.

Normalmente, un cortafuegos filtra la comunicación basándose en:

- Las direcciones y protocolos de origen y destino (direcciones Ethernet o IP, puertos TCP/UDP, etc.).
- Opciones de protocolo (fragmentación, TTL, etc.).
- Tamaño de los datos.

Los Cortafuegos de Aplicaciones Web (CAW)[34] también filtran la comunicación basándose en:

- Las conexiones de usuarios.
- Filtrado de datos (por ejemplo, utilizando descripciones de patrones).

5.4.2 Prueba de la Efectividad del Cortafuegos

Debido a la cantidad de protocolos, sus diferentes opciones y la complejidad de las redes a proteger, es difícil configurar un cortafuegos de forma eficiente. Las pruebas de efectividad de los cortafuegos deberían incluir:

- Escaneo de puertos para verificar si la política de seguridad está bien implementada.
- Uso de paquetes de red malformados y pruebas aleatorias de red para aprovechar un comportamiento inesperado (por ejemplo, una Denegación de Servicio).
- Ataques de fragmentación para eludir las prestaciones de filtrado con el objetivo de llevar a cabo el ataque detrás del cortafuegos

Otro ejemplo de pruebas, dirigidas al CAW, consiste en codificar y comprimir datos u ofuscarlos para ocultar la información maliciosa que transmite el ataque.

[34] **"Cortafuegos de Aplicaciones Web (CAW)" es la traducción del término "Web Application Firewall (WAF)".**

5.5 Detección de Intrusiones

5.5.1 Comprender las Herramientas de Detección de Intrusión

Cada año aumenta el número de ataques. Las técnicas de intrusión evolucionan rápidamente y ningún sistema es 100% seguro.

Un Sistema de Detección de Intrusión (SDI) es un sistema (dispositivo o aplicación independiente) que monitoriza las actividades a diferentes niveles (desde la red a la aplicación, 7 capas del modelo OSI) para detectar violaciones de la política de seguridad. Si se detectan desviaciones del comportamiento normal, se lanzan alertas que pueden ser analizadas para emprender otras acciones (por ejemplo, bloqueo de tráfico, aplicación de parches virtuales).

En cuanto a la estandarización de los SDI, el "Internet Engineering Task Force Working Group Intrusion Detection Exchange Format" describe un modelo de diseño para un SDI basado en dos modelos de seguridad:

- Modelo de seguridad negativo (detección basada en firmas o detección de listas negras): la regla es "todo lo que no está explícitamente prohibido está permitido". La detección de intrusiones se basa en una lista de ataques o patrones conocidos.
- Modelo de seguridad positiva (detección basada en el comportamiento o detección de lista blanca): la regla es "todo lo que no está explícitamente permitido es rechazado". La detección de intrusiones se basa en la especificación del comportamiento del sistema a proteger, por ejemplo, las características de una entrada en forma descrita como expresión regular. La intrusión se detecta si el comportamiento se desvía del comportamiento normal o esperado del sistema. El tráfico de confianza puede utilizarse para generar la especificación.

Un SDI se diferencia de un cortafuegos en que un cortafuegos observa el tráfico exterior para detener las intrusiones, mientras que el SDI analiza las intrusiones sospechosas y emite una alerta si se confirman.

5.5.2 Prueba de la Efectividad de las Herramientas de Detección de Intrusiones

La detección basada en escenarios es fácil de eludir porque sólo se detectan los ataques conocidos. Las pruebas podrían incluir las siguientes técnicas de evasión:

- Codificación de caracteres o modificación de datos (por ejemplo, añadir espacios en blanco, finales de línea, etc.)
- Fragmentación IP, segmentación TCP
- Cifrado, ofuscación
- Codificación de URL

La detección basada en el comportamiento genera un gran número de resultados falsos positivos y falsos negativos. Un resultado de falso negativo es cualquier alerta que debería haber informado pero no lo hizo. Los falsos negativos pueden producirse cuando se desarrolla un nuevo ataque del que un SDI no tiene constancia, o tal vez una regla puede estar escrita de tal manera que detecte algunos ataques pero pase por alto otros. También hay que tener en cuenta la exactitud de este método de detección. Es posible que un atacante desvíe el comportamiento del SDI de su comportamiento normal, dando como resultado una

nueva especificación que contenga un comportamiento intrusivo. Así, este nuevo tráfico no se tiene en cuenta como anómalo. Las pruebas complementarias deben utilizar tráfico malicioso para añadir nuevas especificaciones intrusivas consideradas como tráfico autorizado.

Algunas entradas pueden utilizarse para definir un conjunto de pruebas para los SDI, como el "Sistema de Detección de Intrusión de Protección de Perfiles" [SDI-PP][35] y los "Criterios de Evaluación de Cortafuegos de Aplicaciones Web" [CECAW][36].

5.6 Escaneo de Software Malicioso

5.6.1 Comprender las Herramientas de Escaneo de Software Malicioso

El código malicioso puede afectar a los servidores y a los ordenadores de los usuarios finales, proporcionando a sus creadores los privilegios esperados y los datos sensibles que se pretenden obtener. El código malicioso se coloca en el objetivo utilizando diferentes medios como el correo electrónico con archivos adjuntos maliciosos, direcciones web (Localizador de Recurso Uniforme - LRU[37]) falsas, ejecución de código del lado del cliente, etc.

Una aplicación contra software malicioso es un software utilizado para analizar, detectar y eliminar el código malicioso recibido de diferentes fuentes, con diferentes objetivos de detección: software malicioso, suplantación de identidad y “pharming”.

La principal prestación de detección utilizada por productos contra software malicioso es una estrategia basada en firmas. El principio consiste en buscar en una base de datos patrones de datos conocidos que describan un fragmento de código sospechoso. Sin embargo, los nuevos programas maliciosos o el software malicioso cuya firma no esté presente en la base de datos no serán detectados y podrían infectar a su víctima. A menudo se incorpora un mecanismo heurístico en el anti-patrón para identificar ligeras variaciones de patrones maliciosos conocidos para ayudar a combatir este problema.

5.6.2 Prueba de la Efectividad de las Herramientas de Escaneo de Software Malicioso

Los desarrolladores de software malicioso y puertas traseras utilizan diferentes técnicas para proteger su código contra la ingeniería inversa y la detección por parte del producto contra el software malicioso. Algunas de estas técnicas incluyen:

- Aprovechamiento de las funciones de la librería del sistema utilizadas por el software malicioso (por ejemplo, FindWindow, que puede utilizarse para cerrar una aplicación contra software malicioso).
- Ofuscación de cadenas para impedir la comprensión del comportamiento del código malicioso (por ejemplo, utilizando el cifrado). Un ejemplo podría ser almacenar guiones Java en un documento PDF. Otro es utilizar la compresión como Ultimate Packer para ejecutables (UPX).

[35] "Sistema de Detección de Intrusión de Protección de Perfiles [SDI-PP]” es la traducción del término "Profile Protection Intrusion Detection System [PP-IDS]”.

[36] “Criterios de Evaluación de Cortafuegos de Aplicaciones Web [CECAW]” es la traducción del término “Web Application Firewall Evaluation Criteria [WAFEC]”.

[37] “Localizador de Recurso Uniforme – LRU” es la traducción del término “Uniform Resource Locator (URL)”.

- Carga dinámica de funciones y librerías (por ejemplo, para limitar el análisis del código malicioso).
- Actualización automática de aplicaciones (por ejemplo, el Troyano de Skype).

El software malicioso también puede utilizar otros recursos de hardware, como la Unidad de Procesamiento Gráfico (UPG), para descomprimir el código malicioso y almacenarlo en la memoria para que sea ejecutado por el procesador. En este caso, el software malicioso no puede ser analizado antes de su ejecución.

Desde el punto de vista de las pruebas funcionales, una herramienta como "Eicar" [EICAR] (archivo de prueba contra el software malicioso) podría utilizarse para probar la efectividad del producto contra el software malicioso sin desarrollar piezas de código malicioso reales.

Una consideración importante a la hora de implementar una nueva aplicación contra software malicioso, o de actualizar una aplicación contra software malicioso existente, es probar la implementación en una plataforma representativa antes de desplegarla en toda la organización. Se han dado casos en los que el programa contra software malicioso identificó erróneamente archivos legítimos del sistema operativo como software malicioso y los puso en cuarentena, con lo que se inhabilitó toda la capacidad informática de la organización.

5.7 Ofuscación de Datos

5.7.1 Comprender la Ofuscación de Datos

La ofuscación (a veces llamada enmascaramiento de datos) es un mecanismo para hacer que los datos y el código fuente no sean comprensibles para una persona.

Esta técnica se utiliza principalmente para proteger los datos sensibles contra:

- La copia, para eludir los mecanismos de protección de las licencias.
- Ingeniería inversa, para entender el código con el fin de aprovechar las vulnerabilidades.

La ofuscación de datos también puede utilizarse para permitir a los empleados de una empresa (personal de apoyo, probadores funcionales, etc.) trabajar con datos no sensibles, al tiempo que se ocultan los datos sensibles a la vista. Algunos pueden referirse a la ofuscación de datos como "de datos", en el sentido de que mantiene anónimos los datos personales de un individuo.

La ofuscación también puede utilizarse para proteger el código fuente contra el simple copiar y pegar (por ejemplo, para proteger un nuevo algoritmo innovador) y su futura reutilización después de haber realizado ingeniería inversa para entenderlo.

A veces los desarrolladores necesitan optimizar su código para hacerlo más eficiente. Esto puede dar como resultado un código fuente ofuscado (por ejemplo, codificando algunas partes en lenguaje ensamblador). Algunos ataques a nivel de aplicación web consisten en la inyección de guiones. Para tener éxito, los atacantes necesitan conocer la estructura del sitio web y las páginas HTML. La ofuscación puede ayudar a proteger las páginas HTML sensibles y críticas (por ejemplo, las de conexión y administración).

Se pueden utilizar varias técnicas de ofuscación, como la codificación en base64, el XOR, el renombramiento aleatorio de funciones, la anulación de métodos, la eliminación del espacio de retorno de las pestañas, el barajado, etc. El cifrado también es una técnica de ofuscación, pero con problemas porque los datos cifrados seguirán siendo visibles para quienes tengan claves válidas.

Nota: La ofuscación de datos suele ser utilizada por los atacantes para ocultar sus códigos maliciosos y sus ataques.

5.7.2 Prueba de la Efectividad de los Enfoques de Ofuscación de Datos

Se necesita un estricto control de la configuración entre los datos ofuscados y las claves utilizadas para la ofuscación para asegurar que se utilizan las versiones correctas de las claves. De lo contrario, no se podrá retirar la ofuscación de los datos para su uso.

Dado que en algunas pruebas podrían estar implicados datos privados, la ofuscación de datos puede utilizarse con fines de prueba para hacer que los datos de producción utilizados en un entorno de prueba del sistema sean anónimos. Los datos sensibles, como la información de los usuarios utilizada por un sistema de información sanitaria, no deben ser divulgados a los probadores. Las pruebas podrían incluir:

- Ataques de fuerza bruta o de diccionario para intentar obtener datos simples a partir de datos ofuscados.

Las pruebas para verificar la ofuscación del código podrían incluir:

- Ingeniería inversa del bytecode Java (por ejemplo, regenerar el código fuente Java utilizando el Descompilador Java) o de los programas .Net (por ejemplo, recuperar el código fuente .Net con el Reflector .NET).
- Ataques de fuerza bruta, porque algunos mecanismos de ofuscación son vulnerables (por ejemplo, utilizando unXOR [Chopitea]).

En teoría, el código no puede protegerse contra la desofuscación, porque siempre se puede recurrir a la depuración. Aunque existen herramientas para proteger el código contra la desobstrucción, sigue habiendo riesgos y limitaciones a la hora de proteger la información propietaria representada por el código.

5.8 Formación

5.8.1 La importancia de la Formación en Seguridad

Los seres humanos suelen ser el eslabón más débil en el panorama general de la seguridad. Por lo tanto, se necesita una formación consistente y continua para recordar a las personas la importancia de seguir las políticas de seguridad establecidas y para subrayar por qué son necesarias las políticas. Esta formación debe realizarse a lo largo del proceso del ciclo de vida del software y actualizarse a medida que se añaden nuevas políticas y surgen nuevas amenazas. La formación debe cubrir la identificación de los ataques de ingeniería social y las amenazas internas.

5.8.2 Cómo Probar la Efectividad de la Formación en Seguridad

Por ejemplo, un programa de formación en seguridad podría abordar la importancia de contar con contraseñas de usuario sólidas y confidenciales.

Las pruebas podrían incluir:

- Ingeniería social para intentar que un usuario revele su contraseña durante una conversación telefónica con un falso miembro del servicio de soporte técnico.

- Buscar en los escritorios notas adhesivas con contraseñas (especialmente debajo de los teclados).
- Ejecutar herramientas de auditoría de contraseñas para identificar las contraseñas débiles. Uno de los riesgos de este tipo de herramientas es que las contraseñas pueden ser visibles para la persona que ejecuta la prueba.

Otro ejemplo sería que un desarrollador fallara al colocar una edición a nivel de campo para evitar la entrada de comandos SQL en un campo de entrada de datos. Debido a esta equivocación, un probador de seguridad es capaz de inyectar un comando SQL y ver el contenido de la base de datos de un cliente. Esto indicaría que el desarrollador necesita formación adicional en prácticas de codificación segura. También sería bueno examinar las prácticas de codificación de otros desarrolladores para ver si esta práctica está extendida y se necesita una iniciativa general de mejora del proceso.

Un tercer ejemplo sería cuando un probador intenta acceder físicamente sin autorización a una oficina y ver documentos que se han dejado al descubierto.

6 El Factor Humano en la Prueba de Seguridad

Duración: 105 minutos

Palabras Clave

atacante	(“attacker”)
red de robots	(“botnet”)
informática forense	(“computer forensics”)
pirata informático	(“hacker”)
reconocimiento	(“reconnaissance”)
niño de guion	(“script kiddie”)

Objetivos de Aprendizaje para “El Factor Humano en la Prueba de Seguridad”:

6.1 Entender a los Atacantes

AS-6.1.1	(K2)	Explicar cómo el comportamiento humano puede conducir a riesgos de seguridad y cómo impacta en la efectividad de la prueba de seguridad.
AS-6.1.2	(K3)	Demostrar la capacidad de identificar las formas en las que un atacante podría descubrir información clave sobre un objetivo y aplicar medidas para proteger el entorno en un escenario determinado.
AS-6.1.3	(K2)	Explicar las motivaciones y las fuentes comunes para realizar ataques a sistemas informáticos.
AS-6.1.4	(K4)	Analizar un escenario de ataque (ataque realizado y descubierto) e identificar las posibles fuentes y motivaciones del ataque.

6.2 Ingeniería Social

AS-6.2.1	(K2)	Explicar cómo las defensas de seguridad pueden verse comprometidas por la ingeniería social

6.3 Concienciación sobre la Seguridad

AS-6.3.1	(K2)	Comprender la importancia de la concienciación sobre la seguridad en toda la organización.
AS-6.3.2	(K3)	Dados ciertos resultados de la prueba, aplicar las acciones adecuadas para aumentar la conciencia sobre la seguridad.

6.1 Entender a los Atacantes

En el contexto de la seguridad de la información, el ser humano es tanto la mayor amenaza como el punto más débil de la defensa.

Los ataques contra la seguridad son realizados por personas con una gran variedad de competencias y motivaciones. Además, los seres humanos son los (mayores) facilitadores de la mayoría de los ataques contra la seguridad. El mero hecho de entender la tecnología de la seguridad y su implementación no es suficiente para defenderse de los ataques. También es importante comprender la mentalidad, las motivaciones y los métodos de los atacantes maliciosos y ser consciente de las debilidades humanas en la línea de defensa.

6.1.1 El impacto del Comportamiento Humano en los Riesgos de Seguridad

La fase clave de cualquier ataque es la de recopilación de información (reconocimiento), en la que el atacante trata de encontrar y recopilar información sobre el objetivo. Toda la información que se publique, a veces sin saberlo, sobre una organización, los sistemas en uso, etc., y que se almacene en Internet será encontrada y puede ser utilizada o se utilizará en un ataque. No es una cuestión de "si ocurre ..." sino de "cuándo". Además de la información publicada oficialmente por la organización, los empleados también publican información sobre la empresa en sus redes sociales. La cantidad y el contenido de esta información cambian continuamente, lo que a menudo supone una información clave para los atacantes.

Los atacantes no utilizan una política de seguridad ni procedimientos predefinidos cuando atacan un sistema. Basándose en la información que pueden recoger, deciden su estrategia. Actualizarán su base de conocimientos para cada ataque realizando búsquedas selectivas y "visitando" direcciones IP ya conocidas.

Cuando se formula la política de seguridad de una empresa, suele basarse en la situación y en los datos disponibles. A veces eso no incluye toda la información accesible al público y, aunque lo hiciera, es probable que esa información cambie. Las pruebas de seguridad que eran válidas cuando se crearon pueden no proporcionar una cobertura adecuada cuando la información publicada cambie.

6.1.2 Comprender la Mentalidad del Atacante

Durante la actividad de reconocimiento o rastreo de información, el atacante tratará de encontrar todo tipo de información sobre el objetivo utilizando medios pasivos o activos. La mayoría de los equipos informáticos que se enfrentan a redes públicas dejan una huella en dichas redes. Estas huellas pueden ser y serán encontradas. Google (incluidos Google Earth y Street View) u otros motores de búsqueda, Shodan [Web-5], Facebook, LinkedIn y otras redes sociales son las primeras fuentes utilizadas para encontrar información sobre el objetivo. Las direcciones IP, las páginas web, los números de teléfono, los nombres y las estructuras de las direcciones de correo electrónico, los sistemas operativos y las aplicaciones pueden proporcionar información útil para el atacante.

Un posible uso del motor de búsqueda de Google es encontrar información específica sobre un objetivo. En la base de datos de Google Hacking [Web-4] se pueden encontrar cientos de consultas. Shodan [Web-5] es otra herramienta que se utiliza para encontrar información específica, por ejemplo, qué empresas de una zona concreta están ejecutando un servidor Apache con una versión vulnerable.

La mayor parte de esta información puede encontrarse de forma pasiva sin necesidad de conectarse realmente al sistema objetivo. Otras herramientas utilizadas son:

- Whois [Web-13]
- Base de datos Ripe (European IP Networks) [Web-12]
- Búsquedas DNS [Web-25]

Con las técnicas de reconocimiento activo, el atacante utiliza herramientas para detectar hosts, puertos abiertos, sistemas operativos y aplicaciones tocando el sistema. Entre los métodos y herramientas utilizados se encuentran:

- Pinging - Fping [Web-15], Hping [Web-19]
- TCP/UDP scan - Nmap [Web-20], Zenmap [Web-21]
- OS detection - Nmap [Web-20], Xprobe2 [Web-22]
- Service fingerprinting - Servicio de huella digital (Nmap tiene capacidades para determinar también el tipo y la versión del servicio que se ejecuta en el puerto abierto descubierto. Esto se hace comparando la "huella digital" del servicio descubierto con la propia base de datos de huellas digitales de Nmap).

Como piratear un sistema está prohibido por la ley en la mayoría de los países, si no en todos, el pirata informático intentará destruir después todas las pruebas del pirateo. Otras razones para destruir las pruebas son alargar la estancia, continuar con el uso del sistema en el futuro y utilizar el sistema comprometido o la red de sistemas (red de robots) para atacar otros sistemas. El atacante puede desplegar herramientas como NetCat [Web-14] para ello o utilizar sitios web como IP TRacer [Web-7], así como tunelar o alterar los archivos de registro.

Otros métodos y herramientas utilizados para ocultar pruebas son las herramientas de ocultación [Web-16], los encubridores[38] y la transmisión de archivos. Todas, o la mayoría, de las herramientas mencionadas aquí son accesibles a través de Internet. La descarga de la última versión de Kali Linux [Web-17] y una búsqueda en el sitio de OWASP [OWASP1] darán acceso a muchas de esas herramientas.

6.1.3 Motivaciones y Fuentes Comunes de los Ataques a los Sistemas Informáticos

Muchos ataques y vulneraciones a los sistemas de información provienen del interior de la organización. Los usuarios malintencionados de los sistemas (piratas informáticos internos o amenazas internas) intentarán comprometer los sistemas siendo un usuario autorizado de la red. La mayoría de las veces la motivación es la venganza, pero las tendencias recientes muestran un aumento del espionaje económico o el robo.

Los piratas informáticos o hacker externos son responsables de la minoría de los ataques. La curiosidad por la información fue uno de los primeros motivadores para hackear los sistemas de información, y lo sigue siendo. Disponer de alguna información de empresas u organizaciones importantes y saber que otros no la tienen es otro motivador (prestigio). Otros motivadores son la notoriedad o la fama, el desafío, el aburrimiento y la venganza, donde esta última se considera la forma más peligrosa (motivación más alta).

[38] **"encubridor" es la traducción del término "rootkit".**

Los atacantes se suelen clasificar por su motivación y sus habilidades. En el extremo inferior del espectro de atacantes se encuentran los "script kiddies" que simplemente ejecutan los ataques que otros crean, mientras que en el extremo superior se encuentran las organizaciones e individuos profesionales (gubernamentales, hacktivismo). El hacktivismo es el ataque a sistemas basado en motivos principalmente políticos, pero también económicos u otros demográficos.

La capacidad de hackeo varía desde los individuos que tienen algunos conocimientos sobre sistemas y redes y que trabajan con un simple ordenador doméstico hasta los profesionales altamente capacitados y formados que tienen acceso a laboratorios, redes proxy y todo el resto del equipo técnico necesario. Tener una idea sobre los posibles atacantes ayudará a una organización a implementar la protección necesaria y da una pauta para la estrategia de pruebas de seguridad.

6.1.4 Comprender los Escenarios de Ataque y las Motivaciones

Una incidencia de seguridad se define como un evento del sistema relevante para la seguridad en el que se desobedece o se incumple la política de seguridad del sistema. [RFC2828]

Averiguar qué ocurrió y quién fue el responsable del incidente relacionado con la seguridad es un objetivo de la disciplina de la informática forense [Web-8], en la que uno se concentra en encontrar evidencias digitales del ataque.

El proceso de recuperación de evidencias se basa en tres fases:

1. Adquirir y Autenticar
2. Analizar
3. Informar

6.1.4.1 Adquirir y Autenticar

El proceso de gestión de incidencias de la organización debe restaurar el sistema a su estado original (antes del ataque) después de que se recojan y almacenen las pruebas. Comienza cuando el administrador del sistema es alertado por el IDS u otros medios de monitorización. Otros síntomas típicos de los incidentes de seguridad son:

- Entradas de registro sospechosas.
- Cuentas de usuario inexplicables.
- Archivos/carpetas modificados.
- Servicios inusuales en ejecución.
- Comportamiento inusual del sistema.
- Intentos infructuosos de inicio de sesión.

Tras recibir la alerta, el proceso a realizar es el siguiente:

1. Realizar una instantánea o copia del sistema investigado para reunir todas las pruebas necesarias.

2. Después de autenticar las evidencias (que sea una copia auténtica y completa), realice una copia y guárdela en un lugar seguro.
3. Analizar las evidencias.
4. Una vez finalizado el proceso forense, eliminar la causa de la incidencia (erradicación).
5. Devolver el sistema a su estado normal (recuperación).

Durante estos pasos, se elimina cualquier vulnerabilidad con parches o instalando nuevo software. Al informar de los resultados, debe describirse el proceso seguido junto con las herramientas utilizadas durante este proceso.

6.1.4.2 Analizar

Tras los intentos de hackeo, puede ser posible encontrar el origen de los ataques examinando los archivos de registro del sistema y las conexiones de red activas. Es importante hacer copias de todos los archivos de registro y capturar la información del estado del proceso. Durante un ataque activo, puede tener sentido reunir información del sistema relacionada con el atacante o atacantes antes de bloquearlos.

Cualquier ataque a través de Internet puede ser rastreado hasta la dirección IP de origen, tanto si ha utilizado el correo electrónico como las conexiones a Internet. Es sólo una cuestión de tiempo, dinero y esfuerzo, y una evaluación de los costes implicados. La mayoría de los atacantes utilizan proxies o cadenas de proxies, la red Tor [Web-9] u otras opciones anónimas gratuitas para cubrir su dirección IP real. Cuantos más proxies utilicen los atacantes, más tiempo se necesitará para trazar la dirección de origen. Las leyes locales basadas en la ubicación física de los proxies también pueden obstaculizar esta investigación.

Descubrir a los intrusos y trazar una dirección IP hasta su origen puede hacerse con herramientas como Netstat (Windows) [Web-10], Tracert [Web-11] y el sitio web IP Trazador [Web-7]. Netstat muestra las conexiones a una máquina, los puertos y los servicios en ejecución. Esta herramienta puede utilizarse para buscar cualquier dirección IP o número de puerto extraño o desconocido. Nota: También existe una utilidad tracert en el sistema operativo Microsoft Windows (en Linux y OS/X, es "traceroute"), pero los servicios basados en la web mencionados anteriormente son independientes de estas utilidades O/S.

En la cabecera de un correo electrónico que contenga virus, puede aparecer la dirección IP del ISP que envió el correo. Sin embargo, para la mayoría de los correos electrónicos basados en la web (Gmail, Yahoo mail, Outlook.com), ésta es la dirección IP del proveedor. Para encontrar la verdadera dirección IP hay que buscar el valor X-Originating-IP. Utilizando las bases de datos Whois [Web-13] se obtendrán los detalles que se pueden utilizar para contactar con la organización del ISP para continuar la investigación. Hay que tener en cuenta que el correo electrónico puede originarse en servidores privados y en servidores de correo de retransmisión abierta. En ese caso, puede ser muy difícil identificar el origen real de un mensaje de correo electrónico.

Investigar los ataques que utilizan una red de bots[39] es difícil. No es necesario que el atacante tenga una conectividad en línea con el servidor de bots[40] o los clientes bot[41], por lo que trazarlo es muy difícil o casi imposible. En este caso, la investigación de los clientes puede conducir al servidor, pero hay que tener

[39] **"red de bots" es la traducción del término "botnet".**
[40] **"servidor de bots" es la traducción del término "botserver".**
[41] **"cliente bot" es la traducción del término "botclient".**

acceso al servidor para investigar el verdadero origen del ataque. Los propietarios de estos servidores pueden no ser conscientes de que sus máquinas forman parte de una red de bots.

6.1.4.3 Informar

El suministro de información sobre las vulnerabilidades de seguridad se describe en el capítulo 7.

6.2 Ingeniería Social

Se pueden implementar todas las defensas técnicas que se puedan imaginar para proteger los activos digitales del mundo exterior, pero al final todo se reduce al hecho de que los empleados (usuarios y administradores) necesitan tener acceso a estos activos para realizar su trabajo. Es posible que necesiten utilizar la autenticación para acceder desde sus ordenadores de sobremesa, portátiles, teléfonos inteligentes, tabletas u otros medios. Cualquier defensa de seguridad física para proteger el acceso a la oficina y a los equipos de la misma carece de sentido si la seguridad de la zona de trabajo del responsable de TI en su casa puede verse fácilmente comprometida.

El ser humano y su comportamiento son la mayor amenaza para la seguridad. Si la gente es descuidada con la información sensible, esto deja demasiadas huellas en los lugares seguros y difunde esta información en voz alta (tanto oral como electrónicamente) en lugares públicos.

La ingeniería social es el arte de aprovechar al ser humano utilizando su comportamiento general como vector de ataque. Como seres sociales, las personas están dispuestas a confiar y ayudar a los extraños. Esto crea una vulnerabilidad a los ataques. Al manipular, influir y persuadir a las personas serviciales, un atacante intentará obtener acceso, detalles de autorización u otro tipo de información sensible.

El aprovechamiento de las oportunidades (contra la seguridad) puede realizarse mediante la interacción humana directa o mediante el uso de equipos informáticos/de red. La interacción humana directa puede realizarse en persona, incluyendo:

- Parasitar[42] ("tailgating") o "piggybacking"[43] (alguien que carece de la autenticidad adecuada sigue a un empleado a una zona restringida).
- Escuchar secretamente[44] (escuchar las conversaciones privadas de otra persona sin su conocimiento).
- Espiar por encima del hombro[45] (mirar por encima del hombro de alguien sin su conocimiento mientras realiza tareas en el ordenador o por escrito).
- Utilizar el teléfono (por ejemplo, obtener la contraseña de un usuario desprevenido haciéndose pasar por otra persona, como un gerente o una persona de soporte técnico).

[42] **Mejorar la traducción.**
[43] **Traducir.**
[44] **"escuchar secretamente" es la traducción del término "eavesdropping". Mejorar la traducción.**
[45] **"espiar por encima del hombro" es la traducción del término "shoulder surfing". Mejorar la traducción.**

Se puede llevar a cabo ingeniería social basada en el ordenador mediante:

- Enviar correos electrónicos infectados con software malicioso.
- Utilizar aplicaciones de chat o de mensajería instantánea. Mediante el uso de aplicaciones de chat y mensajería instantánea, cualquier persona anónima puede mantener una conversación con otra en cualquier parte del mundo, sin conocer la verdadera identidad de la otra persona. Además, los datos a través de los mensajeros instantáneos pueden ser fácilmente olfateados[46].
- Utilizar pantallas emergentes[47]. Por ejemplo, puede aparecer una ventana en la pantalla del ordenador de un usuario con un mensaje en el que se le indica que se ha perdido la conexión a la red. En ese momento, se pide al usuario que vuelva a introducir su nombre de usuario y su contraseña. Un programa previamente instalado por el intruso puede entonces transmitir la información a un sitio remoto.
- Envío de correo electrónico no deseado[48]. Los correos electrónicos no deseados están plagados de ofertas y enlaces fraudulentos. Al hacer clic en estos enlaces se puede instalar software malicioso que puede dejar al descubierto toda una red.
- Persuadir a las personas para que visiten sitios web infectados (manipulados). Estos intentos de suplantación de identidad pueden ser enviados de forma generalizada o pueden ser muy individualizados (suplantación de identidad dirigida).

No existe una única defensa contra la ingeniería social. Se pueden implementar defensas para controlar el daño (por ejemplo, proporcionar el menor nivel de privilegio que aún permita a alguien realizar su trabajo, separación de funciones, rotación de tareas), pero la principal defensa es la educación y la concienciación en todos los niveles de la organización.

6.3 Concienciación sobre la Seguridad

6.3.1 Importancia de la Concienciación sobre la Seguridad

El modelo de amenazas cambia constantemente, como se ha mencionado en otros capítulos de este programa de estudio. Las redes evolucionan, se introducen nuevas aplicaciones, se hacen operativas nuevas interfaces y se introducen y descubren nuevas vulnerabilidades.

Además de estos aspectos técnicos, está el factor humano. Los riesgos que en su día se identificaron pero no se convirtieron en problemas se olvidan fácilmente y se retiran las protecciones. Esto ofrece mayores oportunidades para los ataques de hacking y de ingeniería social. Se necesita una formación regular de concienciación sobre la seguridad para mantener a los administradores de seguridad y a todos los empleados alerta e informados sobre los cambios en el modelo de amenazas. La formación de concienciación sobre la seguridad puede concentrarse en diferentes grupos de usuarios: desarrolladores, operaciones, personal de gestión y usuarios en general.

[46] **"olfateado" es la traducción del término "sniffed".**
[47] **"pantalla emergente" es la traducción del término "pop-up screen".**
[48] **"correo electrónico no deseado" es la traducción del término "spam email".**

6.3.2 Incrementar la Concienciación sobre la Seguridad

Es importante mantener una mentalidad "consciente de la seguridad". Además de la información general sobre las defensas de seguridad en la empresa, la formación debería contener estudios de casos reales, descubiertos durante la prueba de seguridad o en incidentes reales. Sobre la base de estos casos, debería ser más fácil debatir sobre las defensas o los cambios que deben implementarse en la organización.

Un esquema para esta sección en la formación de concienciación debería incluir respuestas a las siguientes preguntas:

- ¿Cómo lo hicieron (hicimos)?
- ¿Cuáles fueron las consecuencias para el negocio?
- ¿Cuáles fueron los costes de investigar y procesar la incidencia?
- ¿Cuáles fueron los costes para reparar el problema?
- ¿Cómo se podría haber evitado la incidencia?
- ¿Qué cambios se van a implementar?

7 Evaluación de la Prueba de Seguridad y Suministro de Información

Duración: 70 minutos

Palabras Clave

criterios de aceptación	(“acceptance criterio”)
criterios de salida	(“exit criteria”)
panel de control	(“dashboard”)
vector de ataque	(“attack vector”)

Objetivos de Aprendizaje para “Evaluación de la Prueba de Seguridad y Suministro de Información”:

7.1 Evaluación de la Prueba de Seguridad

AS-7.1.1	(K2)	Comprender la necesidad de revisar las expectativas de seguridad y los criterios de aceptación a medida que evolucionan el alcance y los objetivos de un proyecto.

7.2 Suministro de Información sobre la Prueba de Seguridad

AS-7.2.1	(K2)	Comprender la importancia de mantener la confidencialidad y la seguridad de los resultados de la prueba de seguridad.
AS-7.2.2	(K2)	Comprender la necesidad de crear controles y mecanismos de recopilación de datos adecuados para proporcionar los datos de origen para los informes de estado de las pruebas de seguridad de manera oportuna, exacta y precisa (por ejemplo, un panel de control de las pruebas de seguridad)
AS-7.2.3	(K4)	Analizar un determinado informe de estado de la prueba de seguridad provisional para determinar el nivel de exactitud, la capacidad de ser entendido y la adecuación de los implicados.

7.1 Evaluación de la Prueba de Seguridad

Es necesario medir los resultados de las pruebas de seguridad y evaluar el estado con respecto a las expectativas de seguridad, los criterios de salida y/o los criterios de aceptación para determinar la compleción de la prueba.

Es difícil conocer todos los riesgos de seguridad al inicio de un proyecto. Además, las expectativas de los implicados y de los usuarios a veces cambian respecto al nivel de seguridad que se necesita. Por ejemplo, el conocimiento de una nueva amenaza puede hacer que los implicados requieran niveles de seguridad más altos de lo que se pensaba inicialmente. Esta es una de las razones por las que las evaluaciones del riesgo de seguridad deben revisarse a lo largo del proyecto y los resultados deben incorporarse a la planificación de la prueba de seguridad y a su ejecución.

7.2 Suministro de Información[49] sobre la Prueba de Seguridad

7.2.1 Confidencialidad de los Resultados de la Prueba de Seguridad

Es cierto que el probador medio sabe más sobre el objeto de prueba una vez finalizada la prueba que la mayoría de los desarrolladores o diseñadores. Al probar en profundidad se pueden encontrar los puntos débiles y fuertes más importantes del sistema. Lo mismo ocurre con las pruebas de seguridad.

Al probar la implementación de seguridad se pueden encontrar agujeros ocultos y vulnerabilidades de seguridad. La diferencia radica en el posible impacto negativo de comunicar estas vulnerabilidades a personas distintas de los implicados directos. Una buena práctica general es que la información se ponga a disposición sólo de aquellos que necesiten conocerla. Esto se aplica específicamente a los resultados de pruebas de seguridad; ser conservador a la hora de compartir este tipo de información se considera una buena práctica.

7.2.2 Creación de Controles Adecuados y Mecanismos de Recogida de Datos para Informar sobre el Estado de la Prueba de Seguridad

El impacto y el efecto de una vulnerabilidad de seguridad se juzga normalmente como de mayor sensibilidad en comparación con los defectos "normales". Esto lleva a la necesidad de ser más preciso y exacto a la hora de informar sobre la naturaleza del defecto y los riesgos implícitos. En la mayoría de los proyectos, los defectos de seguridad se clasifican con una severidad mayor que los defectos funcionales comparables.

Esto último implica que la dirección se concentra más en los defectos de seguridad, sus riesgos y sus posibles resoluciones. El suministro de información sobre los defectos de seguridad debe evaluar cuidadosamente el impacto de un problema descubierto, la exactitud de los resultados de la prueba, y debe estar disponible de una manera bien definida y oportuna. Es una buena práctica discutir con la dirección cómo y cuándo les gustaría tener acceso a los informes de defectos de seguridad.

[49] "suministro de información" es la traducción del término "reporting". En algunos programas de estudio, el término "reporting" se tradujo como "generación de informes" o "informes". Estas últimas dos traducciones son muy restrictivas. En cualquier caso, se debe tener en cuenta que un informe es un soporte muy habitual para suministrar información.

7.2.3 Análisis de Informes Provisionales del Estado de la Prueba de Seguridad

Los informes de prueba de seguridad pueden elaborarse a lo largo de todo el proceso de prueba de seguridad o sólo al final de las pruebas de seguridad (como al final de las pruebas de seguridad del sistema o al final de las pruebas de seguridad realizadas como parte de las pruebas de aceptación). Se fomenta el suministro de información de la prueba de seguridad de forma temprana porque permite disponer de más tiempo para remediar las vulnerabilidades de seguridad. Si el proceso de prueba de seguridad sigue el descrito en este programa de estudio, el equipo de prueba puede descubrir vulnerabilidades y documentar las observaciones durante todas las actividades de prueba.

La estructura de un informe de prueba de seguridad debería contener las siguientes secciones:

1) Identificador del informe.
2) Resumen.
    a) Resumen ejecutivo.
    b) Hallazgos clave.
3) Desviaciones.
    a) Proceso de prueba seguido.
    b) Cualquier desviación respecto del proceso de prueba planificado.
    c) Métodos y herramientas (configuraciones, políticas) utilizados.
4) Evaluación general.
    a) Evaluación de la cobertura de la prueba basada en los criterios indicados en el plan de prueba.
    b) Explicación de los elementos o prestaciones que no se hayan probado.
5) Resumen de los resultados.
    a) Resumen de los resultados de la prueba de seguridad.
    b) Lista de todas las vulnerabilidades de seguridad resueltas y sus resoluciones.
    c) Lista de todas las vulnerabilidades no resueltas.
6) Evaluación.
    a) Evaluación de los resultados de prueba observados y su estado en función de los criterios de salida.
    b) Riesgos identificados (clasificaciones) e impacto de las vulnerabilidades de seguridad no resueltas.
7) Resumen de actividades.
8) Aprobaciones

La efectividad del suministro de información (a través de informes) sobre la prueba de seguridad depende de lo siguiente:

- La cronología o frecuencia del informe.
- El contenido del informe.
- Los destinatarios del informe.
- La adaptación del contenido para que se ajuste a la necesidad de información de los destinatarios.

Es posible que se necesiten diversos informes para satisfacer las necesidades de los implicados. Por ejemplo, el contenido de un informe para la dirección ejecutiva no será el mismo que para un arquitecto de sistemas.

8 Herramientas de Prueba de Seguridad

Duración: 55 minutos

Palabras Clave

Ninguna

Objetivos de Aprendizaje para "Herramientas de Prueba de Seguridad":

8.1 Tipos y Objetivos de las Herramientas de Prueba de Seguridad

AS-8.1.1	(K2)	Explicar el papel de las herramientas de análisis estático y dinámico en la prueba de seguridad.

8.2 Selección de Herramientas

AS-8.2.1	(K4)	Analizar y documentar las necesidades de las pruebas de seguridad que deben ser abordadas por una o más herramientas
AS-8.2.2	(K2)	Comprender los problemas de las herramientas de código abierto.
AS-8.2.3	(K2)	Comprender la necesidad de evaluar la capacidad del proveedor de actualizar las herramientas con frecuencia para mantenerse al día con las amenazas a la seguridad.

8.1 Tipos y Objetivos de las Herramientas de Prueba de Seguridad

Las formas de aprovechar oportunidades ideadas por la comunidad de piratas informáticos han impulsado el desarrollo de herramientas de prueba de la seguridad para defenderse de estas amenazas. Incluso desde las primeras actividades de hacking (como el reventado de contraseña) se inventaron herramientas sencillas, creadas y mejoradas por quienes las utilizaban. Las herramientas que demostraron su efectividad se compartieron en la comunidad de piratas informáticos y se siguieron mejorando y perfeccionando. Al principio, estas herramientas se desarrollaron para tareas y entornos específicos. La usabilidad no era un problema, ya que casi todos los usuarios tenían conocimientos técnicos. Con el tiempo, algunas de las herramientas de los piratas informáticos se convirtieron en la base de las herramientas de prueba de seguridad legítimas utilizadas por los administradores y probadores de la seguridad de la información.

Como ejemplo, "John the Ripper" fue una de las primeras herramientas de reventado de contraseña de código abierto, utilizada originalmente por los piratas informáticos para adivinar (crackear) contraseñas y obtener acceso a redes o aplicaciones Unix. En la actualidad, esta herramienta se ha perfeccionado y se utiliza con fines legítimos para detectar contraseñas débiles de Unix. [Web-26]

A medida que los principales proveedores de herramientas de prueba y desarrollo de software y los proveedores de herramientas especializadas comenzaron a desarrollar herramientas de prueba de la seguridad, muchas de estas herramientas alcanzaron capacidades funcionales más amplias y mejoraron la usabilidad. Sin embargo, esta amplia funcionalidad condujo a configuraciones de herramientas más complejas y a asuntos de interés para su implementación.

Al mismo tiempo que surgían las primeras herramientas de seguridad, se desarrollaron las primeras versiones de frameworks como Nessus, Metasploit y otros como herramientas de código abierto que ofrecían una funcionalidad mejorada y ampliada y, en algunos casos, también una IGU ("Graphic User Interface - GUI") fácil de aprender.

Hoy en día, se dispone de un gran número de herramientas de prueba de la seguridad. Para casi cualquier entorno o tarea se puede encontrar una herramienta de prueba dedicada, ya sea de código abierto o con licencia. El reto de todas estas herramientas es que la mayoría de ellas son sistemas inteligentes que despliegan pruebas no estandarizadas. Todos los desarrolladores de estos sistemas están más o menos de acuerdo en cómo probar las defensas de seguridad o probar las vulnerabilidades. Sin embargo, estas herramientas pueden utilizar diferentes datos de prueba, diferentes implementaciones de prueba y diferentes interpretaciones de los resultados.

Las herramientas de prueba de seguridad pueden utilizarse para automatizar la evaluación de las defensas de la seguridad. Las herramientas de prueba de seguridad también pueden utilizarse para detectar tipos conocidos de vulnerabilidades. Teniendo en cuenta que el mismo tipo de defensa de seguridad o vulnerabilidad puede implementarse de diferentes maneras, la selección y el uso de herramientas de prueba de seguridad es un reto para el probador de seguridad porque las herramientas difieren en la forma de encontrar vulnerabilidades y validar las defensas.

Los sitios web del Web Application Security Consortium [Web-18] y del OWASP [OWASP1] ofrecen listas de herramientas clasificadas. El marco de pruebas de penetración Backtrack [Web-23] (o Kali Linux [Web-17]) presenta otras formas de clasificar las herramientas de prueba de la seguridad.

El número de herramientas de seguridad comerciales es bastante limitado en comparación con el número de herramientas de código abierto. En el momento en que se elaboró este programa de estudio (2016) sólo pudimos encontrar un número limitado de recursos que presentaban una visión general más o menos completa de las herramientas de seguridad de código abierto fiables y de confianza. Una lista de herramientas de seguridad puede encontrarse en https://sectools.org [Web-24]. Se espera que el probador de seguridad avanzado mantenga su propia lista de herramientas disponibles y que la actualice a medida que el mercado de herramientas cambia.

Tanto las herramientas de análisis estático como dinámico son útiles en la prueba de seguridad. La ventaja de las pruebas estáticas es que pueden realizarse en una fase muy temprana del ciclo de vida de desarrollo. Las herramientas de análisis estático están disponibles para la mayoría de los lenguajes de software y suelen tener la capacidad de informar sobre los aspectos de seguridad.

La diferencia entre las herramientas de prueba dinámica y estática en el contexto de las pruebas de seguridad es a veces un poco confusa en comparación con otros tipos de pruebas. La definición de prueba estática está relacionada con la realización de actividades de prueba mientras el sistema u objeto sometido a prueba no está en modo operativo. No es raro que las herramientas de prueba dinámica de seguridad prueben el sistema en lugar de la aplicación sometida a prueba. Desde esta perspectiva, estas herramientas de pruebas dinámicas se utilizan como un tipo de herramientas de pruebas estáticas. Por ejemplo, una herramienta de prueba de seguridad dinámica puede realizar un análisis estático de una base de datos. Por supuesto, si se tiene en cuenta todo el sistema como objeto de prueba, entonces las herramientas son realmente herramientas de prueba dinámicas.

8.2 Selección de Herramientas

8.2.1 Analizar y Documentar las Necesidades de la Prueba de Seguridad

Los siguientes documentos, entre otros, pueden constituir una base de prueba para la prueba de seguridad:

- Política de seguridad de la organización.
- Resultados del análisis de amenazas y riesgos para el sistema/proyecto real.
- Requisitos y otras especificaciones del sistema.
- Arquitectura y diseño del sistema.
- Estrategia de seguridad (prueba).
- El sistema o la aplicación que se está probando.
- Amenazas, medios para aprovechar oportunidades (para atacar la seguridad) y vulnerabilidades de seguridad conocidas.
- Perfiles de usuario.

Todo esto y más puede proporcionar información sobre las amenazas y sobre las vulnerabilidades que podrían aprovecharse. Los documentos de requisitos y diseño deben indicar cómo se protegen los datos o la información. Así se obtendrá una visión general de:

- Las interfaces que hay que probar (incluida la IGU - Interfaz Gráfica de Usuario).
- Protocolos y estándares que se deben verificar.
- Directrices de codificación web que promuevan las prácticas de codificación seguras que se deben utilizar.
- Configuraciones de los componentes del sistema que deben ser verificadas (fortificadas).

Es necesario determinar si la prueba de seguridad será una actividad de desarrollo o de mantenimiento/operación. Toda esta información conducirá a los requisitos del conjunto de herramientas de prueba de seguridad.

8.2.2 Problemas con las Herramientas de Código Abierto

Véase [ISTQB_ATM_SYL] para una discusión completa de los problemas que se pueden encontrar con las herramientas de código abierto.

Como ya se ha mencionado, muchas herramientas de prueba de la seguridad se encuentran en el dominio del código abierto. Estas herramientas se distribuyen y pueden utilizarse bajo una amplia variedad de licencias que permiten tanto el uso como la modificación libre del código fuente. No todas las empresas o proyectos pueden tener en cuenta el uso de herramientas de código abierto en sus procesos de desarrollo. Por problemas de cumplimiento normativo, las organizaciones pueden verse obligadas a utilizar únicamente herramientas comerciales o certificadas.

Hay muchas ventajas y desventajas relacionadas con las herramientas bajo estas licencias. En muchos casos, las herramientas de código abierto pueden obtenerse de forma gratuita, pero la organización puede necesitar disponer de capacidad técnica para el soporte y la configuración específica. Si se carece de esta capacidad, se puede incurrir en un coste para obtenerla del desarrollador del software. Los manuales de administración y de usuario, si los hay, se escriben en su mayoría pensando en un público específico (técnico) y lo más frecuente es que no describan o cubran toda la funcionalidad de la herramienta. Los canales de medios como YouTube son últimamente una fuente adicional de información sobre el uso de estas herramientas.

Entre los aspectos que hay que tener en cuenta a la hora de establecer el cálculo del retorno de la inversión de cualquier herramienta de código abierto se encuentran:

- El alcance limitado de la herramienta (en la mayoría de los casos no se ofrece ninguna otra funcionalidad).
- El tiempo necesario para aprender a administrar, configurar y utilizar la herramienta.
- El tiempo que hay que invertir en los foros y grupos de usuarios durante el ciclo de vida.
- El tiempo que se necesita para actualizar y mejorar (y la política interna de actualizaciones).
- La evolución futura de la herramienta (algunas herramientas pueden desaparecer o volverse comerciales).
- El nivel de respuesta de la comunidad de apoyo a la herramienta.

Para la mayoría de los negocios o proyectos, el número de licencias necesarias para las herramientas de prueba de seguridad se limita a una o a unas pocas. Sólo las grandes empresas tendrán en cuenta más

licencias. El número de licencias se basará principalmente en la suma total de las áreas de funcionalidad proporcionadas por la herramienta (por ejemplo, aplicación web, servicios web, análisis de código, otros) y la frecuencia supuesta, el tiempo de uso de estos servicios y el número de probadores de seguridad que utilizan la herramienta.

8.2.3 Evaluación de las Capacidades de un Proveedor de Herramientas

Si se adquiere una herramienta de un proveedor, éste debería ofrecer una serie de servicios para permitir que el servicio de prueba de seguridad comience y crezca hasta alcanzar el nivel de apoyo interno necesario.

Los siguientes atributos pueden utilizarse para evaluar las capacidades del proveedor:

- Tipos de licencias ofrecidas (fijas/de escritorio/flotantes/de testigo).
- Opciones de escalabilidad de las licencias (por área funcional, número de licencias).
- Servicio de asistencia/ayuda (horas de asistencia).
- Foro/comunidad de usuarios.
- Frecuencia de actualización.
- Manuales de administración y de usuario.
- Contratos de asistencia y mantenimiento.

9 Estándares y Tendencias en la Industria

Duración: 40 minutos

Palabras Clave

estándar basado en el consenso ("consensus-based standard")

Objetivos de Aprendizaje para "Estándares y Tendencias Industriales":

9.1 Comprender los Estándares de Prueba de Seguridad

AS-9.1.1	(K2)	Comprender las ventajas de utilizar estándares de prueba de seguridad y dónde encontrarlos.
AS-9.1.2	(K2)	Comprender la diferencia en la aplicabilidad de las normas en situaciones reglamentarias y contractuales

9.2 Aplicación de Estándares de Seguridad

AS-9.2.1	(K2)	Comprender la diferencia entre las cláusulas obligatorias (normativas) y opcionales (informativas) dentro de cualquier estándar.

9.3 Tendencias en la Industria

AS-9.3.1	(K2)	Comprender dónde informarse sobre las tendencias de la industria en materia de seguridad de la información.

9.1 Comprender los Estándares de Prueba de Seguridad

Los estándares de los distintos tipos proporcionan visibilidad al consenso profesional o a las obligaciones normativas. Una norma basada en el consenso representa la opinión ponderada de un cuerpo de expertos bien informados y se pone a disposición para su uso voluntario (en su totalidad o en parte) en los acuerdos contractuales entre proveedores y clientes. Hay otros tipos de estándares menores que surgen de grupos más informales o autoidentificados y que pueden ser específicos de un proveedor.

En las industrias reguladas (incluidos los sectores médico, financiero, del transporte y de la energía) los organismos gubernamentales pueden exigir el cumplimiento de sus propias normas o sus interpretaciones de normas que, de otro modo, serían voluntarias.

9.1.1 Ventajas del Uso de los Estándares de Prueba de Seguridad

Los estándares, en general, proporcionan orientación y consistencia en la realización de una tarea. Normalmente, los estándares son desarrollados por expertos en la materia basados en el consenso de las prácticas efectivas. A continuación, se enumeran las ventajas de utilizar estándares de prueba de seguridad:

- Definen un marco para la prueba de seguridad eliminando la necesidad de empezar todo desde cero.
- Describen las defensas efectivas y la forma de probar los ataques de seguridad más comunes.
- Los estándares pueden adaptarse a las necesidades del proyecto o de la organización.
- Se puede demostrar la debida diligencia en la prueba de seguridad cumpliendo con los estándares de prueba de seguridad reconocidos.

9.1.2 Aplicabilidad de Estándares en Situaciones Reguladas Frente a Contractuales

En las actividades reguladas, todas las partes deben ser conscientes de la obligación de cumplir las normas impuestas, ya que su incumplimiento puede retrasar o impedir la aprobación del producto en desarrollo y, en casos extremos, dar lugar a sanciones financieras o penales.

En situaciones contractuales, las normas proporcionan una base razonable y conveniente para negociar un acuerdo sobre los requisitos del proyecto y del producto; proporcionan un punto de partida en lugar de que las partes comiencen sin nada. Las normas basadas en el consenso permiten comunicar las mejores prácticas y adoptarlas o adaptarlas a la situación concreta.

A menos que sean impuestas unilateralmente por un regulador o en un contrato no negociable, los estándares pueden utilizarse como marco básico de un acuerdo negociado o autoimpuesto en la realización del propio trabajo. Si un contrato se adjudica sobre la base de una demanda o acuerdo de cumplimiento de normas específicas, la entidad tiene la obligación de seguir estrictamente dichas normas y documentar cualquier divergencia.

9.1.3 Selección de Estándares de Seguridad

Ciertamente, no todas las normas de seguridad se aplican a todas las situaciones. Es responsabilidad de una organización investigar el estándar o los estándares más adecuados para sus sistemas, aplicaciones, activos digitales sensibles, nivel de riesgo y requisitos de cumplimiento. También es importante comprender que muchas normas pueden adaptarse para satisfacer los requisitos específicos de una organización.

En el capítulo 10 se puede encontrar una lista de los estándares de seguridad más comunes.

9.2 Aplicación de Estándares de Seguridad

Se debe tener en cuenta el uso preciso del lenguaje dentro de cualquier norma: La palabra deberá identifica los requisitos obligatorios que deben seguirse para ajustarse a la norma, mientras que las palabras “debería” y “puede” indican tareas opcionales que no son necesarias para reclamar la conformidad con la norma. Un error típico es confundir esta distinción exigiendo un elemento opcional o tratando un elemento obligatorio como opcional.

Las situaciones específicas de la organización o del proyecto pueden hacer que se desvíe del sentido estricto de una norma en uso. La justificación de las omisiones, modificaciones o adiciones al contenido de la norma debe ser documentada y acordada por todas las partes.

9.3 Tendencias en la Industria

9.3.1 Dónde Informarse de las Tendencias de la Industria en Seguridad de la Información

Tanto los servicios de noticias de propósito general como los específicos del sector (publicaciones, sitios web, distribuciones por correo electrónico) y los eventos (conferencias, ferias comerciales, reuniones de sociedades profesionales) ofrecen información y debate sobre preocupaciones nuevas o crecientes. Pertenecer a una sociedad profesional o a una comunidad de práctica centrada probablemente proporcionará actualizaciones oportunas y específicas. Con la velocidad a la que surgen nuevas formas de aprovechar o explotar debilidades[50], las alertas electrónicas pueden ofrecer las respuestas más inmediatas.

La publicación periódica de formas de aprovechar oportunidades más frecuentes o dañinas puede identificar tendencias generalizadas, pero se debe prestar especial atención a los problemas más específicos de la industria, el área de aplicación o los productos con los que se trabaja. Es más probable que estos temas se comuniquen en publicaciones especializadas y servicios de noticias o en conferencias técnicas y eventos profesionales.

9.3.2 Evaluación de las Prácticas de Prueba de Seguridad para su Mejora

A medida que se introducen nuevas tecnologías o usos novedosos de la tecnología existente, suele haber una ventana de oportunidad para el mal uso o la explotación de la tecnología hasta que se comprendan mejor sus riesgos y limitaciones.

Por ejemplo, consideremos los dispositivos móviles con servicios de localización. A cambio de comodidad u otros incentivos, las personas parecen estar dispuestas a permitir el seguimiento minuto a minuto de sus movimientos y actividades.

Está surgiendo una mayor gama de motivaciones y mayores recursos por parte de agentes criminales, hacktivistas, económicos y políticos. Los esquemas de extorsión y protección han pasado de las amenazas físicas a los dominios digitales.

[50] “aprovechar una oportunidad” o “explotar una oportunidad” son dos posibles traducciones del término “to exploit”. En el contexto de este programa de estudio se entiende que la oportunidad es algún tipo de debilidad que permitiría llevar a cabo un ataque a la seguridad.

Las grandes redes ad hoc de individuos con motivaciones ideológicas pueden dirigirse en muy poco tiempo contra los objetivos de su ira. El espionaje empresarial suele estar bien financiado y motivado. Los Estados-nación que buscan una ventaja económica y militar disponen de abundantes recursos y pueden creerse inmunes a las sanciones o represalias.

Dado que las amenazas cambian y evolucionan constantemente, los probadores de seguridad deben estar siempre preparados para hacer frente a la siguiente amenaza. El conocimiento del sector, el seguimiento de las tendencias de seguridad y la adquisición de las herramientas más adecuadas constituyen la mejor defensa para una organización.


SSTQB
Spanish Software Testing Qualifications Board

10 Referencias

10.1 Documentos del ISTQB

- [ISTQB_FL_SYL] ISTQB Foundation Syllabus, 2011 [ISTQB_ATM_SYL] ISTQB Advanced Test Manager Syllabus, 2012
- [ISTQB_ATTA_SYL] ISTQB Advanced Technical Test Analyst Syllabus, 2012

10.2 Estándares

- [ISO/IEC/IEEE 29119-3] - Software and systems engineering -- Software testing -- Part 3: Test documentation
- [IEEE 12207] - ISO/IEC/IEEE Standard for Systems and Software Engineering - Software Life Cycle Processes
- COBIT - http://www.isaca.org
- ISO27001 - Information Security Management - http://www.iso.org/iso/home/standards/management-standards/iso27001.htm
- PCI - Payment Card Industry Standard - https://www.pcisecuritystandards.org/

10.3 Libros

- Chapman, 2000] Chapman, Cooper, Zwicky, Building Internet Firewalls, O'Reilly & Associates, 2000.
- [Jackson, 2010] Jackson, Christopher; Network Security Auditing, 2010.

10.4 Artículos

- [ComputerWeekly] http://www.computerweekly.com/news/2240113549/Cattles-lost-backup-tapeshighlight-risk-of-unencrypted-data-storage
- [Northcutt, 2014] Northcutt, Stephen; Security Controls, SANS Institute.
- [Washington Post, 2007] http://www.washingtonpost.com/wpdyn/content/article/2007/05/04/AR2007050402152.html

10.5 Guías

- [Bittau] Cryptographic protection of TCP Streams (tcpcrypt) https://tools.ietf.org/html/draft-bittau-tcp-crypt-04
- [CERT1] Top 10 Secure Coding Practices https://www.securecoding.cert.org/confluence/display/seccode/Top+10+Secure+Coding+Practices
- [CERT2] http://www.cert.org/secure-coding/publications/index.cfm
- [CERT3] http://www.cert.org/secure-coding/tools/index.cfm
- [IEEE1] Avoiding the Top 10 Security Flaws http://cybersecurity.ieee.org/center-for-secure-design/avoiding-the-top-10-security-flaws.html
- [MDA1] MDA Glossary, DoD Missile Defense Agency, www.mda.mil
- [NIST 800-30] NIST Special Publication 800-30, Rev 1, Guide for Conducting Risk Assessments (2012)
- [NISTIR 7298] Glossary of Key Information - Security Terms, Revision 2 (2013)
- [OWASP1] OWASP Secure Coding Practices Quick Reference Guide https://www.owasp.org/index.php/OWASP_Secure_Coding_Practices_-_Quick_Reference_Guide
- [OWASP2] OWASP Risk Rating Methodology https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology
- [OWASP3] OWASP Sample Authorization Form https://www.owasp.org/index.php?title=Authorization_form
- [PP-IDS] US Government Protection Profile Intrustion Detection System for basic robustness environments, version 1.7, 25 July 2007.
- [SANS1] 25 Most Dangerous Software Errors – http://www.sans.org
- [SANS2] Password Construction Guidelines - https://www.sans.org/securityresources/policies/general/pdf/password-construction-guidelines

- [WAFEC] Web Application Firewall Evaluation Criteria, wasc-wafec-v1.0.pdf, 2006.

10.6 Informes

- [WhiteHat Security, 2014] https://www.whitehatsec.com

10.7 Web

- [CERT4] Vulnerability Notes Database - http://www.kb.cert.org/vuls/
- [Chopitea] tomchop.me/2012/12/yo-dawg-i-heard-you-like-xoring/
- [EICAR] www.eicar.org
- [RFC2828] Internet Security Glossary - http://www.rfc-archive.org/getrfc.php?rfc=2828
- [Web-1] Top 20 Critical Security Controls - http://sans.org
- [Web-2] National Vulnerability Database - https://web.nvd.nist.gov/view/ncp/repository
- [Web-3] Website Security Statistics Report - https://www.whitehatsec.com/resource/stats.html
- [Web-4] The Google Hacking Database – http://hackersforcharity.org/ghdb
- [Web-5] Shodan - shodanhq.com
- [Web-6] NetCat - http://sectools.org/tool/netcat/
- [Web-7] IP Tracer - http://www.ip-adress.com/ip_tracer
- [Web-8] Computer Forensics, Cybercrime and Steganography Resources – http://www.forensics.nl
- [Web-9] Tor Project - https://www.torproject.org/
- [Web-10] Netstat - https://technet.microsoft.com/en-us/library/Bb490947.aspx
- [Web-11] Tracert – http://www.tracert.com
- [Web-12] RIPE Scan - https://www.ripe.net
- [Web-13] Whois - https://www.whois.net/
- [Web-14] NetCat – http://netcat.sourceforge.net/
- [Web-15] Fping – fping.org
- [Web-16] Hidetools – http://hidetools.com/
- [Web-17] Kali Linux – https://www.kali.org/

- [Web-18] Web Application Security Consortium – http://www.webappsec.org/
- [Web-19] Hping - http://www.hping.org/
- [Web-20] Nmap - https://nmap.org/
- [Web-21] Zenmap - https://nmap.org/zenmap/
- [Web-22] Xprobe2 - http://null-byte.wonderhowto.com/how-to/hack-like-pro-conduct-os-fingerprintingwith-
- xprobe2-0148439/
- [Web-23] BackTrack - http://www.backtrack-linux.org/
- [Web-24] Top 125 Network Security Tools - at https://sectools.org
- [Web-25] DNS Lookup - https://who.is/dns/
- [Web-26] John the Ripper - http://www.openwall.com/john/