UNIDAD 1: Introducción a la Seguridad

INFORMACIÓN: Grupo de datos procesados y ordenados que sirven para construir un mensaje que cambia el estado del receptor.

Características:

- **CRÍTICA:** Indispensable, genera un daño permanente y significatívo.
- **VALIOSA**: Activo valioso, puede no estar pero suma.
- **SENSITIVA**: Relación entre el dato y a quien pertenece. Solo debe ser conocida por personas autorizadas.

<< Grafico triangulo id >>

SEGURIDAD: "Psicosocialmente seguro" se refiere a sentirse mentalmente protegido y libre de peligro. La disciplina asociada se centra en analizar riesgos y amenazas, aplicar buenas prácticas y seguir normativas para asegurar la seguridad mental.

SEGURIDAD INFORMÁTICA: Se encarga de implementar la protección de la información mediante tecnologías como antivirus, firewalls, detección de intrusos, detección de anomalías y la gestión de incidentes. Su objetivo principal es asegurar la información valiosa de manera efectiva.

SEGURIDAD INFORMÁTICA	**SEGURIDAD DE LA INFORMACIÓN**
IMPLEMENTACIONES TECNICAS	**ORGANIZACION/ESTRATEGIAS DE LA INFORMACION**
Configuración segura.	Análisis de riesgo.
Firewalls y antivirus.	Mejores prácticas.
Administración de incidentes.	Políticas y procedimientos.
Auditoria de eventos.	Mejores normativas.
	Plan Director

Políticas de seguridad: Medidas que toma la organización en materia de seguridad respecto a sus sistemas en base al análisis de sus activos y sus riesgos.

Plan Director: Documento estratégico que establece los objetivos y las directrices clave para proteger la información y los sistemas de una organización contra amenazas cibernéticas con el fin de reducir riesgos a los que se expone la organización.

Análisis de riesgos: Proceso que comprende identificación de activos, vulnerabilidades y amenazas de estos

Incidentes de seguridad: Violaciones de seguridad que generan daño y/o alteración de datos personales. Puede ser de carácter accidental, interno o ciberataque.

Ciberataque: Intento no autorizado de acceder a un sistema informático utilizando técnicas y vulnerabilidades con fines maliciosos.

Intrusión: Acción generada por un atacante que se aprovecha de una vulnerabilidad para acceder a áreas o dispositivos sin autorización.

CAUSAS DE INSEGURIDAD

Activo: Se toma acción por desconocimiento.

Pasivo: No se toman medidas necesarias.

REQUISITOS FUNCIONALES PARA LA SEGURIDAD

AUDITORIA DE SEGURIDAD: Registro de actividad.

SOPORTE DE CIFRADO: Uso de criptografía.

GESTION DE SEGURIDAD: Gestión de perfiles de usuario y permisos.

PRIVACIDAD: Soporte del anonimato de los usuarios.

AUTODEFENSA: Controles para fallar de manera prevista.

CONTROL DE ACCESO: Manejo de cantidad, tiempo, concurrencia de las sesiones.

RUTAS O CANALES FIABLES: Mecanismos que dan confianza sobre los recursos accedidos.

FUNCION DE HASH: Genera un valor unívoco y no reversible que representa un dato, tiene longitud fija de salida y se usa para proteger datos y comprobar integridad de archivos.

SEGURIDAD LÓGICA: Conjunto de barreras y procedimientos que resguardan el acceso a datos.

- ***CONTROL DE ACCESO INTERNO***: Contraseñas, Listas de control de acceso, Limites sobre interfaz de Usuario, Etiquetas de Seguridad.
- ***CONTROL DE ACCESO EXTERNO***: Dispositivos de Control de Puertos, Firewalls o Puertas de Seguridad. (DMZ, IDS, IPS, DISPOSITIVOS UTM, WAF, NGFW), Acceso de personal contratado.
- ***IDENTIFICACIÓN***: Usuario se da a conocer en el sistema.
- ***AUTENTICACIÓN***: Verificación de la identificación. Puede ser algo que solo el individuo conoce, posee, es (huellas, voz) o es capaz de hacer.
- ***MODALIDAD DE ACCESO***: Lectura, escritura, ejecución, borrado (PERMISOS)
- ***ROLES***
- ***TRANSACCIONES***
- ***OTROS***... Firewalls, scanners, honeypots, vpn.

REFERENCIA: BYOD – Bring Your Own Device: Política empresarial que permite el uso de sus propios dispositivos personales a los empleados para el trabajo, aceptando una serie de requisitos por contrato.

RASTREO Y GESTION REMOTA DE DISPOSITIVOS: Tipo de software que permite hacer remotamente (sujeto a conectividad del dispositivo):

- Rastreo del dispositivo.
- Borrado de datos.
- Bloqueo del dispositivo.
- Grabar videos, audio, información por medio.

Otros elementos comunes:

- Firewalls.
- Firewalls personales.
- Escáneres de vulnerabilidades.
- Honeypots, honeynets, Padded cells.
- Verificadores de integridad.
- IDS.
- IPS.
- Antivirus.
- WAF (Web Aplicattion Firewall).

VPN: Estructura de red que permite el tráfico de información sobre una red pública usando criptografía. PROTOCOLOS: IPSec, SSL/TLS, PPTP, L2TP.

SEGURIDAD FÍSICA: Protección física que protege los recursos del sistema de amenazas como:

- ***DESASTRES***: (Inundación, incendio, etc)
- ***Acciones Hostiles*** (robo, fraude, sabotaje)
- ***Control de accesos*** (Uso de guardias, detectores de metales, sistemas biométricos, VAF, seguridad con animales, protección electrónica.

Practicas: No dispositivos en zonas públicas, contenedores contra impactos y líquidos.

UNIDAD 2: AMENAZAS A LA SEGURIDAD

DAÑO: Perjuicio producido a causa de la falla de un sistema informatico.Debe ser cuantificable

RIESGO: Producto entre la magnitud de un daño y de la probabilidad de su causa.

AMENAZA: Situación de daño cuyo riesgo es significativo.

VULNERABILIDAD: Deficiencia de un sistema susceptible de producir un fallo en el mismo.

Pueden ser debido a:

- ***DISEÑO****: Origen y arquitectura. Por ej. elección de productos y herramientas.*
- ***IMPLEMENTACIÓN****: Error en el desarrollo.*
- ***MAL USO****: El uso humano. No se respetan ni utilizan correctamente los recursos.*

Exploit: Técnica que permite aprovechar una vulnerabilidad del sistema para generar algún daño en el mismo

LA NUBE

VENTAJAS	***DESVENTAJAS***
Facilidad de acceso a la información desde diferentes dispositivos.	Dependencia en infraestructura y servicios de terceros
Infraestructura flexible y escalable basada en servicios.	Dependencia en servicios externos para acceder a los sistemas y datos propios
Reducción de costos por infraestructura y servicios	Pérdida del control de la seguridad de la información (delegada al proveedor)
Centralización de administración y gestión de datos.	

CVE (Common Vulnerabilities and Exposures): Código asignado a una vulnerabilidad (detalles y forma de solucionarla).

CWE (Common Weakness Enumeration): Lista de debilidades del software y hardware (MALAS PRACTICAS --> Generan debilidades --> Generan vulnerabilidades).

TANTO CVE COMO CWE FUERON CREADOS POR MITRE CORPORATION.

NVD (National Vulnerability Database): Repositorio de EE.UU para la gestión de datos de vulnerabilidades basados en estándares.

CVSS (Common Vulnerability Scoring System): Conjunto de estándares que sirven para asignar puntajes de severidad a las vulnerabilidades.

PREVENCION DE VULNERABILIDADES:

Listas Bugtraq.

Sistemas automáticos por análisis: DAST (dinámico), SAST (estático), IAST, Redes Trampa.

CERT/CSIRT --> ***CERT***: Computer Emergency Response Team.

CSIRT: Centro de Respuesta a Incidentes de Seguridad Informática.

Son equipos reconocidos como responsables de gestionar incidentes de seguridad informática que le competen según su alcance y comunidad.

FUNCIONES:

- Ayudar al público objetivo a atenuar y prevenir incidentes de seguridad.
- Ayudar a proteger información valiosa.
- Coordinar de forma centralizada la seguridad de la información.
- Guardar evidencias.
- Apoyar y asistir a los usuarios para recuperar de las incidencias de seguridad
- Dirigir de forma centralizada, promover confianza de control sobre la situación.

Denegación de Servicio (DoS): Ataque que causa que un servicio o recurso sea inaccesible. Hacerlo DEFICIENTE y DEGRADADO, que deje de funcionar correctamente. Existen varios tipos:

POR VOLUMEN *(Volumed-Based DDoS attack):* Gran volumen de paquetes o conexiones de red, sobrecargando red de servidores o ancho de banda.

POR APLICACIÓN *(Application DDoS Attack):* A nivel aplicación (HTTP) se satura el servidor y/o servicio que este presta EJ: Aplicando filtro que trae muchos datos.

DE BAJA TASA *(Low-rate DDoS attack)*: Se utiliza una vulnerabilidad en el diseño o implementación de la aplicación.

FLOODING*:* Busca saturar un servicio o que entre en modo de espera generando múltiples solicitudes maliciosas, anulando o limitando su funcionamiento.

BOTNET*:* Conjunto de terminales que ejecutan software, permitiendo control desde ubicaciones remotas.

SNIFFERS*: Programa que captura tramas de red/paquetes de red.*

ATAQUE A LOS NAVEGADORES (CLIENTES)

Tampering o Data Diddling: Alteración de información no autorizada. Ej: cambio contenido de pags. Web

Ataque mediante JavaScript: Es utilizado para explotar fallas de navegadores web y servidores de correo

Ataques drive-by download: Infectan masivamente con el ingreso a un determinado sitio web e inyectan código malicioso.

TECNOLOGÍAS: Java Applets, JavaScript, Shockwave, ActiveX, PDF, Plugins de navegador, Microsoft Silverlight.

Hijackers: Programa que altera el funcionamiento o configuración del cliente con el fin de secuestra información de interés.

Rootkits: Programa que permite que una app maliciosa permanezca oculta en el sistema operativo o que no pueda ser eliminada.

Backdoors: Programas que permiten un acceso alternativo al sistema, evitando la autenticación.

Stealers: Programas que acceden a la información del equipo. Su objetivo son contraseñas o correos de email.

Keyloggers: Programas o dispositivos físicos que registran la actividad de los dispositivos, el teclado es el más común.

Ransomware: Programas que retienen el control del equipo o cifran información del mismo, a veces solicitando un pago por su desactivación. Ej: “WannaCry”, “Cryptolocker”, etc.

OWASP – Riesgos de Seguridad en Aplicaciones.

Colecta datos de proveedores. Tasa de incidencia = porcentaje de aplicaciones con vulnerabilidades.

Factores de datos:

CWEs mapeadas

Tasa de incidencia

Cobertura (de pruebas)

Explotabilidad ponderada

Impacto ponderado

Total, de ocurrencias

Total, de CVEs

Top Ten Web Web Application Security Risks 2021

A1 – Pérdida del Control de Acceso

Se refiere a cuando se pierde el control de acceso o seguimiento del usuario desde el servidor, provocando que estos actúen fuera de los permisos asignados.

Causas:

- URL INTERCAMBIABLE. Sin validar el cambio y los permisos a acceder a esa URL
- REFERENCIA DIRECTA INSERGURA A OBJETOS. No se validan las consultas de referencia.

Medidas de Prevención:

- **Utilizar referencia indirecta.**
- **Validar siempre el nivel de acceso.**
- **Validar ingreso de datos y peticiones (request).**
- **Implementar controles no solo en la capa de visualización.**

A2 – Fallas Criptográficas

Se refiere a fallas en la encriptación de la información en su almacenamiento o tránsito.

Medidas de Prevención:

- **Cifrar datos sensibles.**
- **No almacenar datos sensibles innecesariamente.**
- **Almacenar claves con algoritmos (como PBKDF2).**
- **Algoritmos de cifrado en claves robustas y gestionadas de forma segura.**
- **Deshabilitar autocompletar y cache.**

A3- Inyección (SQL- ORM – OS – LDAP), área del servidor

Ocurre cuando se envían datos no confiables a un intérprete como comando o consulta para que este los ejecute y obtener datos o generar movimientos. EJ:

String query = "SELECT * FROM accounts WHERE custID='" + request.getParameter("id") +"'";

Medidas de prevención:

- Uso de APIs.
- Codificación de caracteres especiales.
- Validación de datos de entrada o lista blanca.
- Uso de LIMIT y otros controles SQL.

CSRF: Ataque que no usa interprete, se ejecuta un comando usando HTTP. EJ: ponen en un src de img. Explota petición HTTP/s en HTML.

Medidas de prevención:

- Uso de tokens únicos.
- Pedir nueva autenticación antes de ejecutar el pedido.

Secuencia de comandos en sitios cruzados (XSS-Cross Site Scripting) : Esta falla ocurre cuando la app toma datos no confiables y los envía al navegador sin validación previa, permitiendo a atacantes ejecutar secuencias de comandos con fines maliciosos como robar sesiones de usuario, entre otros. Existen tres formas de XSS:

- **XSS Reflejado -> Usa datos dados por el usuario sin validación o codificación (CSP).**
- **XSS Almacenado -> Almacena datos del usuario y lo entrega a un "administrador".**
- **XSS Basados en DOM -> Se basa en el envío de datos controlables a través de APIs no seguras.**

Medidas de prevención:

- Codificar datos no confiables basados en HTML.
- Validación de entrada positiva – lista blanca.
- Uso de APIs de autosanitización.
- Uso de políticas de seguridad CSP.

A4 - Diseño Inseguro

Se centra en fallas de diseño y arquitectura.

Medidas de prevención:

- Uso de aplicaciones de seguridad para un ciclo de vida de desarrollo seguro.
- Utilizar bibliotecas de patrones de diseño seguros o "Paved Road".
- Uso de modelado de amenazas.
- Integración del lenguaje y controles de seguridad en "historias del usuario".
- Verificación de admisión en todos sus niveles.
- Escritura de pruebas de integración y pruebas unitarias.
- Compilar casos de uso.
- Separación de capas en niveles.
- Limitar consumo de recursos por usuario o servicio.

A5 - Configuración de Seguridad Incorrecta

Se debe a la no implementación de una configuración segura, siendo definida, implementada y mantenida con software actualizado.

Medidas de prevención:

- Usar formatos como JSON y evitar serialización de datos confidenciales.
- Actualizar procesadores y bibliotecas XML.
- Usar validadores de dependencias y SOAP.
- Implementar validación de “lista blanca”.
- Arquitectura de separación segura y efectiva.

A6 – Componentes Vulnerables y Desactualizados

Componentes (Librerías, Frameworks...) son vulnerables y explotados, dando lugar a ataques que facilitan la perdida de datos o comprometen al servidor.

Medidas de prevención:

- Identificar componentes y sus versiones.
- Revisar la seguridad de los componentes.
- Establecer políticas de seguridad que regulen el uso de componentes.
- Agregar capas de seguridad alrededor del componente.

A7 – Fallas de identificación y autenticación

Apunta hacia funciones de autenticación y gestión de sesiones incorrectamente implementadas, comprometiendo contraseñas, claves, token de sesiones, etc.)

Medidas de prevención:

- Disponer de un único y robusto conjunto de controles de autenticación.
- Esforzarse en evitar vulnerabilidades de XSS.
- Implementar autenticación multifactorial.
- Implementar un control contra contraseñas débiles.
- Limitar o aumentar tiempo de respuesta en cada intento fallido de inicio de sesión.
- Alinear políticas de largo, complejidad, rotación de contraseñas.

A8 – Fallas en el Software y en la Integridad de Datos

Fallas relacionadas al código e infraestructura no protegidos contra alteraciones (**Integridad**) Ej: CDN, dependencia en plugins, módulos de fuentes no confiables, actualizaciones no firmadas (routers, decodificadores, etc. sin firma en actualización) etc.

Deserialización Insegura: Ocurre cuando una aplicación recibe objeto serializados hostiles, lo que conduce a la ejecución del código contenido.

EJ 1: Un foro PHP usa serialización de objetos PHP para almacenar una “super” cookie, conteniendo el id, rol, hash de la contraseña y otros estados del usuario.

a:4:{i:0;i:…**"user”... (El atacante modifica el objeto y se da privilegios de admin) ->**

a:4:{i:0;i:…**"admin”...**

Medidas de prevención:

- No aceptar objetos serializados de fuentes no confiables o aceptar serialización de datos primitivos.
- Implementar verificaciones de integridad como firmas digitales en objetos serializados.
- Cumplimiento estricto de verificaciones durante la serialización y antes de crear el objeto.
- Restringir o monitorear conexiones de red entrantes o salientes vinculadas a la serialización.

A9 – Fallas en el Registro y Monitoreo

Se debe al registro y monitoreo insuficiente más falta o integración inefectiva de respuesta de incidentes, le permite más tiempo y libertad al atacante.

Medidas de prevención:

- Registrar: inicio de sesión fallido, control de acceso y validación de entradas de dato
- Las transacciones de alto impacto deben tener una pista de auditoria con controles de integridad.
- Implementar monitorización y alertas efectivas.

A10 - Falsificación de Solicitudes de Lado del Servidor (SSRF)

Ocurre cuando una app web obtiene un recurso remoto sin validad la URL proporcionada por el usuario.

Medidas de prevención:

- Sanitizar y validar todos los datos de entrada.
- Hacer cumplir el esquema de URL, puerto y destino.
- No enviar respuestas en formato “crudo” a los clientes.
- Deshabilitar redirecciones HTTP.
- Tener presente coherencia en la URL y condiciones.

CONTROLES PRO-ACTIVOS 2018

- C1: Definir Requisitos de Seguridad.

- C2: Hacer Uso De Librerías y Marcos de Trabajo de Seguridad.
- C3: Acceso Seguro a Bases de Datos.
- C4: Encodear y Escapar Datos.
- C5: Validar todas las Entradas de Datos.
- C6: Implementar Identidad Digital.
- C7: Implementar Controles de Acceso Adecuados.
- C8: Proteger los Datos.
- C9: Implementar el Registro y Monitoreo de Seguridad.
- C10: Manejar Errores y Excepciones.