**UNIDAD 0**

***U0: ANEXO – SEGURIDAD***

Para proporcionar seguridad hemos de tener en cuenta múltiples factores.

**Características del Sistema:**

**a. Sistemas aislados:** no están conectados a ningún tipo de red.

**b. Sistemas interconectados:** cualquier ordenador que pertenezca a alguna red.

**Clasificación de Seguridad:**

**1. Seguridad física:** asuntos relacionados con la salvaguarda de los soportes físicos de la información. Ej.: medidas contra incendios y sobrecargas eléctricas, las políticas de copias de respaldo, etc.

**2. Seguridad de la información:** preservación de la información frente a observadores no autorizados. Emplea criptografía simétrica como asimétrica.

**3. Seguridad del canal de comunicación:** escapan a nuestro control, pertenecen a terceros.

4**. Problemas de autentificación:** se suele emplear criptografía asimétrica en conjunción con funciones resumen (hash).

**5. Problemas de suplantación:** se emplean mecanismos basados en contraseñas.

**6. No repudio:** es fundamental impedir que el emisor pueda repudiar un mensaje, es decir; negar su autoria sobre el.

**7. Anonimato:** es crucial garantizar el anonimato del ciudadano para poder preservar su intimidad y su libertad.

***U0: ANEXO - NIVELES DE SEGURIDAD INFORMATICA*** *(\* seguridad lógica)*

Describen diferentes tipos de seguridad del Sistema Operativo y se enumeran desde el mínimo grado de seguridad al máximo. Estos son:

**Nivel D, división simple:** sistemas que han sido evaluados y no cumplen con ninguna especificación de seguridad. Ej. MS-DOS y System 7.0 de Macintosh.

**Nivel C1-Proteccion Discrecional:** se requieren identificación de usuario. Se hace la distinción entre usuarios y administrador.

**Nivel C2-Proteccion de Acceso Controlado:** agrega características adicionales como auditoria de accesos e intentos fallidos de acceso a objetos. Restringe aún más el que los usuarios ejecuten ciertos comandos o tengan acceso a ciertos archivos, permitir o denegar datos a usuarios en concreto.

**Nivel B1-Seguridad Etiquetada:** a cada objeto del sistema (usuario, dato, etc.) se le asigna una etiqueta, con un nivel de seguridad jerárquico (alto secreto, secreto, reservado, etc.).

**Nivel B2-Proteccion Estructurada:** requiere que se etiquete cada objeto de nivel superior por ser padre de un objeto inferior. Es capaz de alertar a usuarios si sus condiciones de accesibilidad y seguridad son modificadas; y el administrador es el encargado de fijar los canales de almacenamiento y ancho de banda.

**Nivel B3-Dominio de Seguridad:** refuerza a los dominios con la instalación de hardware. Cada usuario tiene asignado los lugares y objetos a los que puede acceder.

**Nivel A-Protección Verificada:** incluye un proceso de diseño, control y verificación, mediante métodos formales (matemáticos) para asegurar todos los procesos que realiza un usuario sobre el sistema.

**UNIDAD 1**

***U1: INTRODUCCION A LA SEGURIDAD***

**Seguridad de la Información:** Disciplina que nos habla de los riesgos, de las amenazas, de los análisis de los escenarios, de las buenas prácticas y esquemas normativos, que nos exigen niveles de aseguramiento de procesos y tecnologías para elevar el nivel de confianza en la creación, uso, almacenamiento, transmisión, recuperación y disposición final de la información.

**Seguridad Informática:** se encarga de las implementaciones técnicas de la protección de información, el despliegue de las tecnologías antivirus, firewall, detección de intrusos, detección de anomalías, correlación de eventos, atención de incidentes, entre otros elementos.

**Causas de Inseguridad**

Estado de Inseguridad dos:

**- Activo:** falta de conocimiento del usuario acerca de las funciones del sistema. Ej.: desactivar el antivirus.

**- Pasivo:** Falta de conocimiento de las medidas de seguridad disponibles. Ej.: forzar la autenticación en el sistema.

**Triangulo ID:** Confidencialidad – Integridad – Disponibilidad

INTEGRIDAD: La información sólo puede ser borrada o modificada por usuarios autorizados.

CONFIDENCIALIDAD: el acceso al sistema está limitado a usuarios autorizados.

DISPONIBILIDAD: El acceso a la información se da en un tiempo razonable a los usuarios autorizados.

**Requisitos funcionales para la seguridad**

* Auditoria de Seguridad
* Soporte de cifrado
* Privacidad
* Autodefensa
* Control de acceso
* Rutas o canales fiables

**SEGURIDAD LOGICA**

Aplicación de barreras y procedimientos que resguarden el acceso a los datos y solo permita acceder a ellos a las personas autorizadas.

**Control de Acceso**

**Interno**

* Palabras Claves (password)
* Encriptación
* Listas de Control de Accesos
* Limites sobre la Interface de Usuario
* Etiquetas de Seguridad

**Externo:**

* Dispositivos de Control de Puertos
* Firewalls
* Acceso de Personal Contratado
* Accesos Públicos

**Prácticas de Seguridad Lógica en móviles**

* Usar contraseñas robustas y bloqueo automático.
* Realizar copias de seguridad periódicas.
* Instalar software solo de fuentes oficiales.
* Utilizar software solo con acceso legal a sus funcionalidades.
* Considerar el uso de software de seguimiento, borrado de datos y/o bloqueo remoto.
* Evitar o restringir conexiones a redes públicas o no confiables.
* Deshabilitar sistemas de Bluetooth, NFC y otras tecnologías inalámbricas cuando no se requiera.
* En dispositivos con conexión de datos móviles tener el pin activado y su PUK e IMEI memorizado.

**Rastreo y gestión remota de dispositivos**

Tipo de software que permite:

* Rastreo del dispositivo
* Borrado de datos
* Bloqueo del dispositivo
* Obtención de información del medio.
* Etc.

**Niveles de Seguridad:** *(ver anexo)* ***\****

**Otros elementos comunes de seguridad lógica de sistemas:**

* Firewalls
* Firewalls personales
* Escaners de vulnerabilidades
* Honeypots, Honeynets, Padded cells
* Verificadores de integridad
* IDS (Intrusion Detection System)
* IPS (Intrusion Protection System)
* Antivirus

**Función de Hash**: función o método no reversible para generar un valor que represente de manera univoca un dato.

**VPN**: permite el tráfico de información privada sobre una infraestructura de red pública mediante el uso de criptografía. Protocolos: IPSec, SSL, PPTP, L2TP.

**SEGURIDAD FISICA**

Mecanismos destinados a proteger físicamente cualquier recurso del sistema de amenazas producidas tanto por el hombre como por la naturaleza.

De Acciones Hostiles: -Robo; -Fraude; -Sabotaje.

**Prácticas de Seguridad Física en móviles:**

* Evitar o restringir la manipulación del dispositivo en zonas públicas.
* No transportar el dispositivo en contenedores que puedan ser visibles a terceros.
* Utilizar contenedores de transporte que reduzcan la fuerza ante impactos.
* Utilizar contenedores de transporte que protejan al dispositivo del contacto con liquidos.

***U1: ANEXO - Seguridad Logica y Seguridad Fisica***

**SEGURIDAD LOGICA**

**Modalidad de Acceso:** se refiere al modo de acceso que se permite al usuario sobre los recursos y a la información.

* Lectura
* Escritura
* Ejecución
* Borrado
* Todas las anteriores
* Accesos especiales: Creación; Búsqueda

**Control de Acceso Externo**

**Firewall:** Dispositivo que genera una separación entre redes públicas (externas - no confiables - ‘área no controlada’) y redes privadas (confiables - internas ‘área controlada’) mediante el análisis del tráfico de red.

Características:

- Dispositivo de defensa perimetral que separa redes

- Filtra trafico dependiendo de reglas predefinidas

- No protege de ataques internos

- No protege de accesos no autorizados

- No protege de todos los ataques dañinos

Clasificación**:** Por software o por hardware (Appliances).

Funcionalidades accesorias:

- Punto de conexión de VPNs

- VPN host a red

- VPN red a red

- Escaneo de Virus

- Filtrado de Contenidos / Anti Spam

- Balanceo de carga de Firewalls (BCFW)

- Alta Disponibilidad (AD)

**DMZ - Zona Desmilitarizada:** Alojar zona de conexión de servidores lo que interactúe con red externa.

**Tipos de Firewall**

**1. Packet Filters:** monitorea las direcciones IP de origen y destino de la conexión verificando puertos, pero no el contenido de la comunicación.

**2. Circuit Level Gateways:** opera a nivel de la capa de sesión. Monitorea la conexión con una verificación de contenido limitada.

**3. Application Level Gateways:** conocidos como Proxies, es similar al anterior solo que son específicos para cada aplicación/protocolo.

**4. State-Full Multilayer Insperction:** combina los tres tipos previos.

**Consideraciones sobre el uso de firewalls**

* El mejor firewall no es un producto, es una combinación de factores.
* Un firewall es tan bueno como las políticas que se implementen.
* Se justifica su existencia en la reducción del impacto y/o probabilidad de amenazas que reduzcan el riesgo.
* Debe ser administrado pro-activamente, revisado y actualizado periódicamente.
* Se puede implementar una combinación de firewalls.
* Los firewalls ayudan a proteger los recursos, pero son parte de un plan de seguridad general.

**Firewall Personales:** Dispositivo lógico, software que se instala en la terminal.

**Intrusión**: conjunto de acciones que intentan comprometer la integridad, confidencialidad o disponibilidad de un recurso.

**IDS - Sistema de Detención de Intrusiones**: elemento que detecta identifica y responde a actividades no autorizadas.

Consta de tres fases:

1. **Recolección de datos**: Registros de auditoria, sistemas de archivos, paquetes de red, etc.;
2. **Análisis:** de usos indebidos (ataques conocidos-añadir reglas), de anomalías (ataques no conocidos- actualizar perfil sistema);
3. **Respuesta:** -Activa (cierre de la sesión de usuario, bloqueo de la conexión). -Pasiva (generación de eventos, envío de alerta).

**IPS - Sistema de Prevención de Intrusos:** identifica el curso de un ataque y lo bloquea antes que suceda. Es el resultado de combinar un IDS + un Firewall.

**Dispositivos UTM: s**on firewall de red que manejan diferentes servicios en un mismo equipo (Antivirus, VPN, Antispam, Antipishing).

Posee dos modos:

**-Modo Proxy:** hacen uso de proxys para redirigir el tráfico interno. Intermediario indirecto entre la comunicación desde y hacia internet.

**-Modo Transparente:** No redirigen paquetes, procesan y analizan en tiempo real los paquetes.

Conceptos Básicos

**IP: -** Únicamente identifica un host: permite filtrar tráfico; - Es jerárquico y permite que en una red un firewall controle el trafico entrante y saliente.

**TCP:** - Corre sobre IP;

- Un paquete contiene un numero de puerto origen y destino;

- Contiene un numero de secuencia y un FLAG;

- Relación servicio-puerto.

**UDP**: - Corre sobre IP; - Contiene el número de puerto; - No es orientado a la conexión.

**NAT:** Función traducir direcciones privadas en una o algunas direcciones publicas

**Escáner de Vulnerabilidades**: realizan comprobaciones de seguridad o ataques contra sistemas para encontrar fallos.

**Honeypot (Sistema Trampa) y Honeynet (Red Trampa)**: recurso que simula ser un objetivo real, el cual se espera que sea atacado o comprometido, los principales objetivos son el distraer al atacante y obtener información sobre el ataque.

**Padded Cell** (Célula de Asilamiento): sistema que redirige el tráfico no deseado a una zona asilada

**Comprobadores de integridad**: herramientas que aplican algoritmos de hash sobre archivos para detectar cambios en los mismos

**WAF - Web Application Firewall**: dispositivo lógico o físico que analiza el trafico web. Trata de proteger de ataques dirigidos al servidor web que los IDS/IPS no nos pueden defender. Tiene un proceso de aprendizaje donde verifica como trabaja la aplicación en condiciones normales (Learning Mode).

Modelo de seguridad:

**- Positiva:** Por defecto deniega todo y solo acepta las que reglas que se cargaron como válidas.

**- Negativa:** Por defecto acepta todas las transacciones menos las reglas que se cargaron como posibles amenazas o ataques

**UNIDAD 2**

**Daño**: definimos como daño a el perjuicio que se produce cuando un sistema informático falla. Dicho perjuicio tiene que ser cuantificable.

**Riesgo**: es la suma o el producto del alcance/magnitud de un daño y la probabilidad de que este se pueda cumplir.

**Amenaza:** situación de daño cuyo riesgo de producirse es significativo.

**Vulnerabilidad**: es una deficiencia en un sistema susceptible a producir un fallo en el sistema.

**Exploit:** técnica que se aprovecha de una vulnerabilidad para producir un daño en el mismo.

**Tipos de vulnerabilidades**

**-** Debidas a la implementación

- Debidas al diseño

- Debidas al uso.

**Bugtraq**: lista de notificaciones sobre vulnerabilidades encontradas en software o hardware.

**CERT/CSIRT**: buscan problemas para compartirlos, recopila amenazas de distintas regiones (colaboración entre regiones) soporte legislaciones diferentes / CERTS del ámbito privado.

**CVE**: código asignado a una vulnerabilidad para poder identificarla.

**DoS - Denegación de Servicio**: deja un servicio inaccesible para los usuarios, perdida de conexión de la red ancho de banda o sobre carga de sus ordenadores o red ej.: saturar la subida.

Tipos:

* **Volume-based:** atacante inunda la red con gran volumen de paquetes.
* **Application:** atacante opera a nivel de aplicación usualmente por HTTP intentando saturar el servicio que este presta.
* **Low-rate:** atacante utiliza una vulnerabilidad en el diseño o implementación de la aplicación.

Ejemplos de DoS:

* Inundacion SYN (SYN Flood)
* Inundacion ICMP (ICMP Flood)
* SMURF (ICMP Flood)
* Inundacion UDP (UDP Flood)
* Peer-to-peer
* Utilizacion de recursos
* A nivel de aplicación
* Degradacion de servicio
* Slowloris (HTTP request parciales. Low-rate)

**Flooding o inundación**: esta técnica busca generar solicitudes maliciosas a un servicio con la finalidad de hacer que el mismo se sature o entre en modo de espera, de esta forma anula o limita su comportamiento.

**BotNet**: terminales infectadas que mediante un software se permite su control total o parcial desde ubicaciones remotas se denominan terminales bots o zombies.

**Sniffer:** implementación de la técnica de sniffing (escuchar paquetes), mediante un software se capturan las tramas de red para distintas finalidades.

*Atacando a los navegadores clientes*

**Tampering o Data Diddling:** modificación no autorizada de la información.

**Ataques mediante JavaScript**: explotan fallas de seguridad de navegadores y servidores de correo.

**Ataques drive –by download:** infectan de forma masiva a los usuarios entrando en un sitio web

*Otras tecnologías generadoras de riesgo*:

JavaScript, ActiveX, Shockwave, Java Applets, PDF.

*Otros ataques a clientes*

**Hijackers**: programas que alteran el funcionamiento o configuración del cliente para que el atacante pueda secuestrar información (EJ: sesiones).

**Rootkis**: son programas que permiten que una aplicación maliciosa permanezca oculta en el sistema operativo o que la misma no pueda ser eliminada del sistema (Ej. procesos fantasmas)

**Backdoors:** programas que habilitan un acceso alternativo al sistema que no es el principal.

**Stealers**: recupera información almacenada para brindársela al atacante (Ej.: contraseñas recordadas en el navegador).

**Keyloggers:** son programas o dispositivos físicos que registran la actividad de los dispositivos de entrada.

**Ransomware**: cifran información y solicitan generalmente un pago por la recuperación de la misma.

**Phishing**: obtener información de forma fraudulenta.

**OWASP – Proyecto Abierto de Seguridad en Aplicaciones Web**

Es una comunidad abierta dedicada a facultar a las organizaciones a desarrollar, adquirir y mantener aplicaciones que pueden ser confiables.

Objetivo: crear conciencia acerca de la seguridad en aplicaciones mediante la identificación de algunos de los riesgos más críticos que enfrentan las organizaciones.

Esquema de calificaciones OWASP

-Agente de amenaza; -Vectores de ataque; -Debilidades de Seguridad (Prevalencia de Debilidades, Detectabilidad de Debilidades); -Impacto Técnico; -Impacto al Negocio.

|  |  |
| --- | --- |
| **Mobile Top Ten 2016**  **M1:** Uso inapropiado de la plataforma  **M2:** Almacenamiento inseguro de datos  **M3:** Comunicación insegura  **M4:** Autenticación insegura  **M5:** Insuficiente criptografía  **M6:** Autorización insegura  **M7:** Calidad del código del cliente  **M8:** Tampering de código  **M9:** Ingeniería reversa  **M10:** Funcionalidades extrañas | **OWASP TOP 10 Controles Pro-activos**  Es una lista de técnicas de seguridad que deberías incluir en todo proyecto de desarrollo de software.  Esta ordenado por importancia, siendo el control numero 1 el más importante.  **C1:** Verifica la seguridad temprano y con frecuencia.  **C2:** Parametrizar consultas.  **C3:** Encodear datos.  **C4:** Validar todas las entradas.  **C5:** Implementar controles de Identidad y Autenticación.  **C6:** Implementar controles de acceso adecuados.  **C7:** Proteger los datos.  **C8:** Implementar detección de intrusiones y registro de  actividades.  **C9:** Hacer uso de librerías y marcos de trabajo de seguridad.  **C10:** Manejar errores y excepciones. |

**TOP TEN 10 OWASP 2013**

**A1- INYECCION**: Datos no confiables son enviados a un intérprete como parte de un comando o consulta, los datos hostiles del atacante pueden engañar al interprete y ejecutar comandos o acceder a información. Ej.: SQL, OS, LDAP.

Medidas de prevención:

- Uso de APIs con manejo parametrizado de intérpretes;

- Codificación de los caracteres especiales en función del interprete a utilizar;

- Validación de entradas positiva o de "lista blanca".

**A2 –Perdida de autenticación y gestión de sesiones:** Las funciones de la aplicación relacionadas a autenticación y gestión de sesiones son frecuentemente implementadas incorrectamente, permitiendo a los atacantes comprometer contraseñas y demás. (Ej. de la tarjeta de crédito de la reserva de vuelos).

Medidas de prevención:

- Disponer de un único y robusto conjunto de controles de autenticación y gestión de sesiones;

- Realizar un gran esfuerzo para evitar vulnerabilidades de XSS que pueden dar espacio al robo de IDs de sesión.

**A3-Secuencia de comandos en sitios cruzados (XSS)**: Se denominan fallas XSS Y ocurren cada vez que una aplicación toma datos no confiables y los envía al navegador web sin una validación y codificación apropiada, sin escapar caracteres o considerar que se podría enviar código que pueda interpretar el navegador de la víctima y realizar un perjuicio.

Medidas de prevención:

- Codificar los datos no confiables basados en el contexto HTML (cuerpo, atributo, JavaScript, CSS o URL) donde serán ubicados;

- Validación de entrada positiva o de "Lista blanca";

- Para formato enriquecido considere utilizar APIs de autosanitizacion (ejemplo AntiSamy);

- Considerar el uso de políticas de seguridad de contenido CSP.

**A4-Referencia Directa Insegura a Objetos**: Una referencia directa a objeto ocurre cuando un desarrollador expone una referencia a un objeto de implementación interna, los atacantes pueden manipular estas referencias para acceder a contenido no autorizado.

Medidas de prevención:

- Utilizar referencias indirectas por usuario o sesión;

- Comprobar el nivel de acceso al objeto.

**A5-Defectuosa configuración de seguridad**: La aplicación consta con la configuración de seguridad deficiente y los atacantes hacen uso de estas vulnerabilidades para cometer un daño. Ej.: sistema operando sin actualización aplicada permitiendo por el momento fallos de XSS.

Medidas de prevención:

- Disponer de un proceso rápido, fácil y repetible de fortalecimiento para obtener un entorno apropiadamente asegurado;

- Un proceso para mantener y desplegar las actualizaciones y parches en cada entorno;

- Una arquitectura de aplicación que proporcione separación segura y efectiva entre los componentes;

- Ejecutar escaneo y realizar auditorías regularmente para identificar fallos de configuración o parches omitidos.

**A6-Exposicion de datos sensibles:** La falta de protección para los datos sensibles, los atacantes podrán modificar u obtener esta información realizando un perjuicio. Ej.: robo de una cookie porque no utiliza SSL y con es cookie se roba la sesión y puede operar como si fuera cliente.

Medidas de prevención:

- Cifrar los datos sensibles o en tránsito de manera de defenderse de las amenazas;

- No almacene datos sensibles innecesariamente;

- Aplicar algoritmos de cifrado fuertes y estándar al igual que claves robustas y gestionadas de forma segura;

- Almacenar claves con algoritmos diseñadas para tal fin (bcrypt, scrypt, PBKDF2);

- Deshabilitar el auto-completar en los formularios y cache de páginas que manejan datos sensibles.

**A7-Pérdida del control del nivel de funcionalidades:** Generalmente las aplicaciones web verifican el nivel de acceso a las funciones justo antes de mostrar las mismas, también es necesario realizar esta verificación del lado del servidor de lo contrario los atacantes podrán forzar peticiones con la finalidad de acceder a la funcionalidad sin la autorización apropiada. (Ej. ocultar un botón desde la pantalla sin quitar la funcionalidad, ej. sin una sesión corriendo en la pantalla general).

Medidas de prevención:

- El proceso para gestión de accesos y permisos debe ser actualizable y auditable fácilmente;

- La implementación debería negar todo acceso por defecto, requiriendo el establecimiento explícito de permisos a roles específicos para acceder a cada funcionalidad;

- Si la funcionalidad forma parte de un Workflow, se debe verificar y asegurar las condiciones del flujo para permitir el acceso;

- NO implementar controles en la capa de visualización.

**A8-Falsificacion de Peticiones de Sitios Cruzados (CSRF)**: Este ataque obliga al navegador de la víctima autenticada a enviar una petición HTTP falsificado, incluyendo la sesión del usuario y cualquier otra información de autenticación, esto permite al atacante hacer que la víctima ejecute en código que el preparo para realizar un daño a la víctima. Ej. del código JavaScript en el servidor que ejecuta la víctima.

Medidas de prevención:

- Utilizar un token único y no predecible, este se debería incluir en un campo oculto;

- Requerir una nueva autenticación del usuario o pruebas de que el usuario es legítimo.

**A9 – Uso de componentes con vulnerabilidades conocidas:** Utilizar aplicaciones que utilizan componentes con vulnerabilidades conocidas deben determinar las defensas de la aplicación para el caso y habilitar un rango de posibles ataque e impacto en los mismos. Ej.: un sistema que sirvió de apoyo para la creación de una aplicación de creación de imágenes en 4k, el sistema como bien su función es de generar imágenes gratuitas sin la marca de agua para ser utilizadas por el personal, la misma se planea descontinuar, dado que el sitio ya está en producción, como no se tuvieron en cuenta las vulnerabilidades de la aplicación de apoyo esta al estar colgada en la nube puede ser accedida y robar material de manera gratuita.

Medidas de prevención:

- Identificar todos los componentes y la versión que están ocupando, incluyendo dependencias;

- Revisar la seguridad del componente en bases de datos públicas, lista de correos del proyecto, y lista de correo de seguridad, y mantenerlos actualizados;

- Establecer políticas de seguridad que regulen el uso de componentes, como requerir ciertas prácticas en el desarrollo de software, pasar test de seguridad, y licencias aceptables;

- Sería apropiado, considerar agregar capas de seguridad alrededor del componente para deshabilitar funcionalidades no utilizadas y/o asegurar.

**A10- Redirección y reenvíos no validos**: las redirecciones falsas en busca de que se redirija a páginas que hacen phishing y roban información. Ej. tener una página que se llama redirect.jsp que recibe url y las redirecciona si el atacante modifica esta página puede hacer un perjuicio.

Medidas de prevención:

- No utilizar redirecciones y reenvíos;

- Si se utiliza, no involucrar parámetros manipulables por el usuario para definir el destino;

- Si se requiere que los parámetros del usuario definan el destino se debe verificar que el mismo es válido y autorizado.