Tema 1.1: Principios básicos de seguridad

Garantía y Seguridad de la Información.

Motivación

- Hay muchas listas de principios de seguridad
- Las vulnerabilidades suelen estar asociadas a violaciones de estos principios.
- Las buenas soluciones de seguridad o las buenas contramedidas se ajustan a estos principios.
- No son completamente independientes:
  - Algunos se solapan
  - Algunos se oponen a otros
- Los principios se pueden aplicar a diferentes niveles:
  - Código fuente de una aplicación
  - Entre aplicaciones de una máquina, a nivel Sistema Operativo
  - Nivel de red
  - Dentro de una organización
  - Entre organizaciones

- Las listas de comprobación son muy útiles en ciberseguridad

Principios de Seguridad: nuestra propuesta

- EN RESUMEN:
  - Los principios de seguridad nos **ayudan a lograr los objetivos de la seguridad** y a **analizar los sistemas** con respecto a su seguridad.
- Elegimos unos principios:
  - En 1975, Saltzer y Schroeder analizan los **conceptos básicos sobre protección de información en sistemas informáticos**.
  - Son el origen y la base de casi todas las listas posteriores

Los 12 principios (Saltzer & Schöreder)

1. Simplificar.
2. Abrir el diseño (open).
3. Compartimentar.
4. Exponer lo mínimo.
5. Menor privilegio posible.
6. Confianza mínima, Fiabilidad máxima.
7. Modo seguro y tolerante a fallos, por defecto.
8. Intermediar completamente los objetos.
9. Evitar los puntos de fallo únicos.
10. Registrar lo que pasa en el sistema.
11. Generar secretos impredecibles.
12. Hacer un sistema de seguridad usable.

Los doce principios de la seguridad

- Estos principios no son independientes, de modo que a veces se solapan y otras veces entran en conflicto.
- Esta lista de criterios debe utilizarse como un checklist en las fases de diseño e implementación del modelo de seguridad.
  - indicando qué principio se logra
  - cual se contraviene, y en este caso habrá que explicar el porqué.

1. Simplicidad

- ***Los sistemas simples son más fáciles de entender y por ello más sencillos de analizar y mantener.***
  - También son más fáciles de probar.
  - Al ser más fáciles de analizar y revisar, es más fácil establecer su confiabilidad.

2. Abrir el diseño

- ***La seguridad de un sistema no debe depender del secreto de sus mecanismos de protección.***
  - El conocimiento público del mecanismo de seguridad es beneficioso, pues queda abierto al escrutinio público
  - Es un principio clave de los cripto-sistemas modernos (Kerchoff):
    - Un criptosistema basa toda su seguridad en la posesión de claves secretas.

- Actividad:
  - Kerchoff? Busca información sobre los principios de Kerchoff y su relación con los de Saltzer & Schröder

3. Compartimentar

- ***Hay que organizar los recursos en grupos aislados con necesidades similares.***
  - Los compartimentos intercambian información sólo de modo controlado y limitado, mediante protocolos.
    - La compartimentación se utiliza a menudo en las redes. Dispositivos de filtrado, como los firewalls se emplean para dividir una red en zonas separadas, y la comunicación entre zonas se rige por una política.

3. Compartimentar: EJEMPLOS (I)

- **Separación de** …
  - Computadoras físicamente separadas, para minimizar la posibilidad de daño físico.
  - Contenedores de componentes separados, para aislar plataformas de ejecución de componentes.
  - Máquinas virtuales que separan plataformas dentro de una misma máquina.
  - Redes, con cableados distintos, redes virtuales, cortafuegos, ...
  - Lenguajes que permiten la encapsulación y la modularización.

3. Compartimentar: EJEMPLOS (I)

- **Separación de datos y código**.
  - La mezcla de datos y códigos en pilas lleva a desbordamientos de búfer.
  - Mezclar datos y códigos en páginas web lleva a cross-site scripting.
  - Mezclar datos y códigos en sentencias SQL conduce a inyecciones de SQL.

4. Exposición mínima

- **Minimizar la superficie atacable que un sistema presenta al adversario**.
  - Reduce la posibilidad de ataque de un enemigo.
  - Actividades y Principios de Diseño / Configuración:
    - Reducir las interfaces externas a la mínima expresión.
    - Limitar la cantidad de información proporcionada.
    - Minimizar la ventana de oportunidad de un adversario reduciendo el tiempo disponible para un ataque.
    - Desactivar todas las funcionalidades innecesarias.

4. Exposición mínima: EJEMPLOS

- Recortando funciones (ej. servicios de red, infrarrojos, WLAN o Bluetooth).
- Incluyendo CAPTCHAs (Completely Automated Public Turing test to tell Computers and Humans Apart).
- Bloqueando la pantalla automáticamente.
- Disminuyendo los posibles intentos de login.
- Disminuyendo los protocolos manejados por una aplicación.

5. Menor privilegio posible

- ***Cualquier componente (y usuario) de un sistema debe operar utilizando el menor conjunto de privilegios necesarios para completar su trabajo***.
  - Las ventajas son muy importantes:
    - Ayuda a minimizar las consecuencias negativas de errores
    - Reduce los efectos negativos de los ataques desde dentro.

5. Menor privilegio posible: EJEMPLOS

- Los empleados de una empresa rara vez necesitan acceder a los archivos de datos de los compañeros, salvo que ocupen rangos de supervisión.
- Rara vez los administrativos necesitan instalar software, o administrar partes de un sistema. Es posible restringir sus permisos sin mayores problemas.
- Un firewall bien configurado restringe el acceso a la red a computadoras concretas, en función de puertos, direcciones y protocolos.
- Una regla de oro de configuración de firewalls consiste en partir de una política sin permisos e ir añadiendo reglas positivas caso por caso.

6. Confianza mínima, Fiabilidad máxima

- ***Fiarse lo menos posible y ser muy de fiar***. (zero-trust)
  - Implica
    - reducir al mínimo las expectativas
    - reducir al mínimo la confianza depositada en un sistema
    - maximizar la confiabilidad que ofrecemos.

7. Modo seguro y tolerante a fallos, por defecto

- ***El sistema debe comenzar y retornar a un estado seguro en caso de un fallo***.
  - Los estados seguros de reinicio son imprescindibles para recuperase de una caída.
  - Los procedimientos de relanzamiento deben utilizar mecanismos de política restrictiva.

7. Modo seguro y tolerante a fallos, por defecto: EJEMPLOS

- En el control de acceso, el estado predeterminado y de seguridad debe evitar cualquier acceso.
  - Identificar las condiciones bajo las cuales se otorga el acceso.
  - Si no se identifican las condiciones, se debe denegar el acceso (por defecto).

7. Modo seguro y tolerante a fallos, por defecto: EJEMPLOS

- Firewalls y antivirus habilitados por defecto
  - se reactivan después de un bloqueo o un reinicio manual.
- Conjuntos de reglas de firewall con enfoque de la lista blanca.
  - La regla predeterminada es denegar el acceso a cualquier paquete de red.

8. Intermediar completamente los objetos

- ***El acceso a cualquier objeto debe ser monitorizado y controlado***
  - El mecanismo de control de acceso debe interpenetrar e intermediar todos los objetos del sistema y funcionar bajo cualquier condición.
    - En operación normal, apagado, mantenimiento y fallo.
    - Este mecanismo no debe poder puentearse.

9. Evitar los puntos de fallos únicos

- ***Construir mecanismos de seguridad redundantes cuando sea factible***.
  - Si un mecanismo falla, hay otros en su lugar que todavía pueden impedir el fallo, así que no hay ningún punto único de fallo.
  - Su viabilidad depende de un análisis coste-riesgo-beneficio.

9. Evitar los puntos de fallos únicos: EJEMPLOS

- Uso de secretos de dos componentes:
  - relaja las condiciones de custodia de cada secreto por separado (cajas de seguridad, tarjetas de crédito por correo, ...)
- Introducción de cortafuegos redundantes
- Configuración de diversos mecanismos alternativos antivirus
- Redundancia de mecanismos de comunicación y almacenamiento

10. Registrar lo que pasa en el sistema (Trazabilidad)

- ***Registrar eventos del sistema relevantes para la seguridad.***
  - Los logs pueden servir para:
    - detectar errores de funcionamiento y ataques,
    - identificar el método de los adversarios,
    - analizar la extensión y el origen de un ataque,
    - deshacer los efectos.

11. Generar secretos impredecibles

- ***Maximizar la entropía de los secretos***
  - Previene los ataques por fuerza bruta, de diccionario y por suposición informada.
    - Es necesario que las claves tengan la longitud adecuada
    - Un buen mecanismo pseudoaleatorio de generación de claves.
    - Para las claves de usuario es preciso imponer reglas que dificulten la adivinación de claves

12. Usabilidad del Sistema de Seguridad

- ***Diseñar mecanismos de seguridad utilizables.***
  - Un sistema demasiado burocrático o complejo hace que los usuarios utilicen otros cauces alternativos inseguros.

12. Usabilidad del Sistema de Seguridad: EJEMPLOS

- Un sistema de filtrado de e-mail demasiado riguroso produce que se vuelva al fax.
- Un sistema de llaves poco usable hace que se hagan duplicados.
- Un sistema de encriptación dificultoso produce que los usuarios guarden versiones antiguas.
- O que cambien de claves poco a menudo, o que se apunten en un papel.

Lecturas y Actividades

- Artículo original de Saltzer y Schröder de 1975
  - Capítulo 1 de libro de Basin (Base prácticas)

- Actividades:
  - Analizar los principios de seguridad para Cloud del National Cybersecurity Centre y alinearlos con los de Saltzer
  - Localiza otras listas de principios de seguridad y checklists:
    - OWASP Security Principles
    - OWASP 10 Security by Design
  - Gestión de Crisis (CCN-CERT, ver materiales CV)
    - Las listas de comprobación y los principios ayudan …

Fases de Gestión de Crisis (Institut Cerdà)