Confiabilidad y Seguridad

POR MARIO ROSSAINZ LÓPEZ
INGENIERIA DE SOFTWARE II
OTOÑO 2022
NRC: 10838

Introducción

El término **"confiabilidad"** fue propuesto por Laprie (1995) para cubrir los atributos relacionados disponibilidad, fiabilidad, protección y seguridad en los sistemas de software.

Según Sommerville, dichas propiedades están vinculadas entre sí; así que tiene sentido disponer de un solo término para tratarlas a todas.

Introducción

Razones por las cuales, la confiabilidad de los sistemas es importante:

1. Las fallas del sistema afectan a un gran número de individuos. Muchos sistemas incluyen funcionalidad que se usa rara vez. Si esta funcionalidad se retirara del sistema, tan sólo un número menor de usuarios resultarían afectados. Las fallas del sistema, que afectan la disponibilidad de un sistema, gravitarían potencialmente a todos los usuarios del sistema. La falla significaría que es imposible continuar con la normalidad del negocio.

Introducción

Razones por las cuales, la confiabilidad de los sistemas es importante:

2. Los usuarios rechazan a menudo los sistemas que son poco fiables, carecen de protecciones o son inseguros. Si los usuarios descubren que un sistema es poco fiable o inseguro, lo rechazarán. Más aún, ellos también pueden negarse a adquirir o usar otros productos de la compañía que desarrolló el sistema que no es fiable, porque considerarán que dichos productos tienen la misma probabilidad de no ser fiables o seguros.

Introducción

Razones por las cuales, la confiabilidad de los sistemas es importante:

3. Los costos por las fallas del sistema suelen ser enormes. Para ciertas aplicaciones, como un sistema de control de reactor o un sistema de navegación de aeronaves, el costo por la falla del sistema es una orden de magnitud mayor que el costo del sistema de control.

Introducción

Razones por las cuales, la confiabilidad de los sistemas es importante:

4. Los sistemas no confiables pueden causar pérdida de información. Los datos son muy costosos de recolectar y mantener; por lo general, valen mucho más que el sistema de cómputo donde se procesan. El costo por recuperar datos perdidos o contaminados generalmente es muy alto.

Introducción

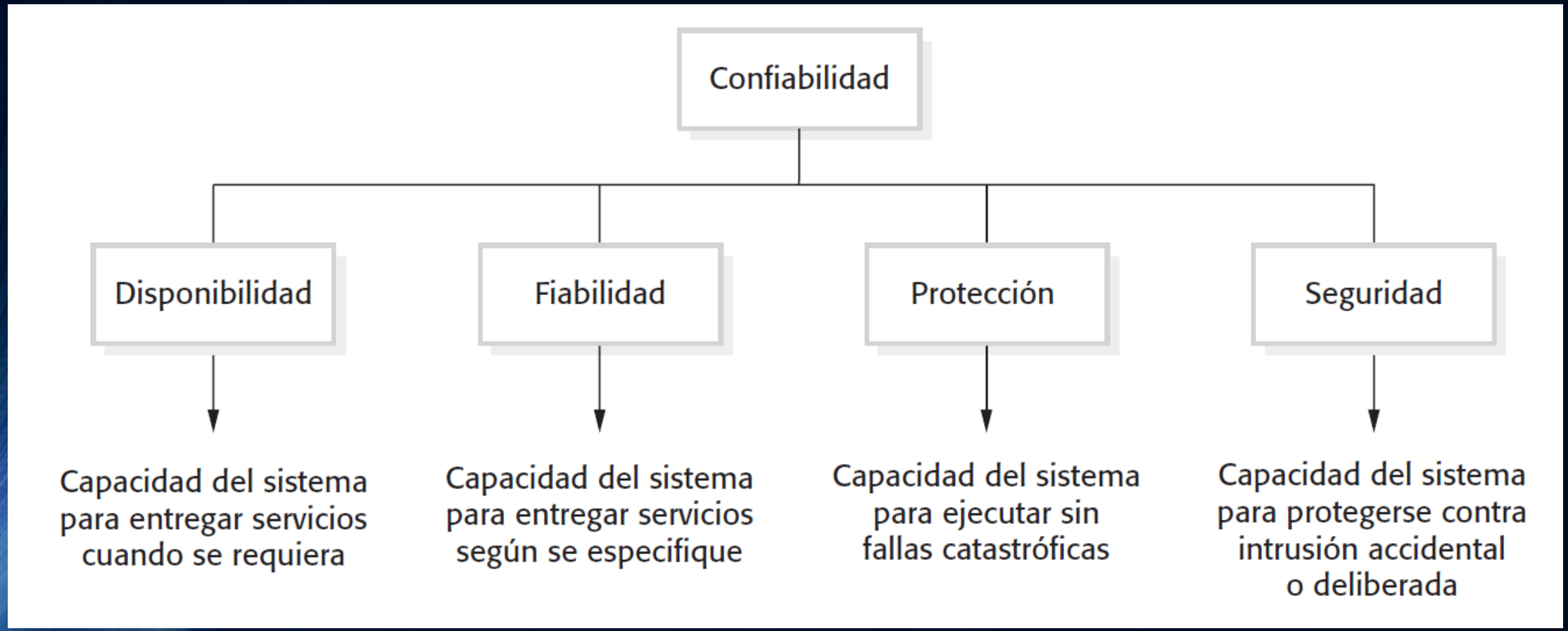
Consideraciones en el diseño de un software confiable:

- *Falla del hardware* El hardware del sistema puede fallar por errores en su diseño, componentes que se averían por errores de fabricación o porque los componentes llegaron al final de su vida operativa.
- *Falla en el desarrollo de software* El software del sistema puede fallar debido a errores en su especificación, diseño o implementación.
- *Falla de operación* Los usuarios fallan al usar o ejecutar el sistema correctamente. Conforme el hardware y el software se vuelven más confiables, las fallas en la operación son ahora, quizá, la principal causa individual de fallas del sistema.

Propiedades de Confiabilidad

- La confiabilidad de un sistema de cómputo es una propiedad del sistema que refleja su fiabilidad.
- La Fiabilidad es el grado de confianza que un usuario tiene que el sistema ejecutará como se espera, y que el sistema no “fallará” en su uso normal.

Dimensiones de la Confiabilidad



Dimensiones de la Confiabilidad

1. ***Disponibilidad.*** De manera informal, la disponibilidad de un sistema es la probabilidad de que en un momento dado éste funcionará, ejecutará y ofrecerá servicios útiles a los usuarios.

2. ***Fiabilidad.*** De manera informal, la fiabilidad de un sistema es la probabilidad, durante un tiempo determinado, de que el sistema brindará correctamente servicios como espera el usuario.

3. ***Protección.*** De modo no convencional, la protección de un sistema es un juicio de cuán probable es que el sistema causará daños a las personas o su ambiente.

4. ***Seguridad.*** Informalmente, la seguridad de un sistema es un juicio de cuán probable es que el sistema pueda resistir intrusiones accidentales o deliberadas.

Otras propiedades de confiabilidad

- ***Reparabilidad.*** Las fallas del sistema son inevitables; no obstante, el desajuste causado por la falla podría minimizarse siempre que el sistema logre repararse rápidamente. Para que ello ocurra, se puede diagnosticar el problema, acceder al componente que falló y hacer cambios para corregir dicho componente. La reparabilidad en software se mejora cuando la organización que usa el sistema tiene acceso al código fuente y cuenta con las habilidades para cambiarlo. El software de fuente abierta facilita esta labor, aunque la reutilización de componentes suele dificultarlo más.

Otras propiedades de confiabilidad

- ***Mantenibilidad.*** Mientras se usan los sistemas, surgen nuevos requerimientos y es importante mantener la utilidad de un sistema al cambiarlo para acomodar estos nuevos requerimientos. El software mantenible es aquel que económicamente se adapta para lidiar con los nuevos requerimientos, y donde existe una baja probabilidad de que los cambios insertarán nuevos errores en el sistema.

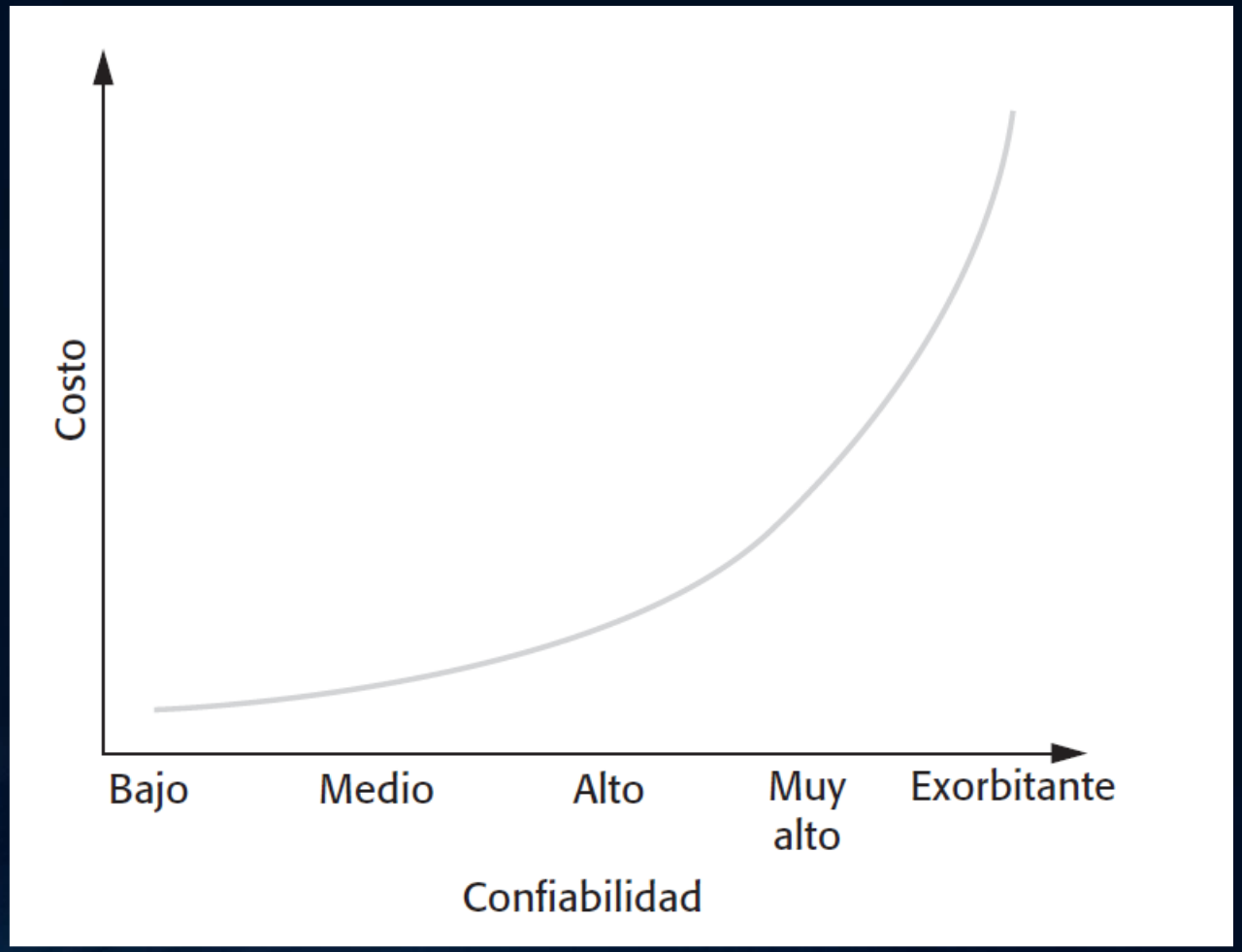
Otras propiedades de confiabilidad

- ***Supervivencia.*** Un atributo muy importante para sistemas basados en Internet es la supervivencia que se define como la habilidad de un sistema para continuar entregando servicio en tanto está bajo ataque y mientras que, potencialmente, parte del sistema se deshabilita. El trabajo sobre supervivencia se enfoca en la identificación de los componentes clave del sistema y en la garantía de que ellos puedan entregar un servicio mínimo. Para mejorar la supervivencia se usan tres estrategias: resistencia al ataque, reconocimiento del ataque y recuperación del daño causado por un ataque.

Otras propiedades de confiabilidad

- ***Tolerancia para el error.*** Esta propiedad se considera como parte de la usabilidad y refleja la medida en que el sistema se diseñó de modo que se eviten y toleren los errores de entrada de usuario. Cuando ocurren errores de usuario, el sistema debe, en la medida de lo posible, detectar dichos errores y corregirlos automáticamente o solicitar al usuario que reintroduzca sus datos.

Relación costo-confabilidad



Disponibilidad y Fiabilidad

- ***Disponibilidad.*** La probabilidad de que un sistema, en un momento en el tiempo, sea operativo y brinde los servicios solicitados.

- ***Fiabilidad.*** La probabilidad de operación libre de falla durante cierto tiempo, en un entorno dado, para un propósito específico.

Disponibilidad y Fiabilidad

EJEMPLO:

- Si, en promedio, 2 entradas de cada 1,000 causan fallas, entonces la fiabilidad, que se expresa como una tasa de ocurrencia de falla es 0.002.
- Si la disponibilidad es 0.999, esto significa que, durante cierto tiempo, el sistema está disponible para el 99.9% de ese tiempo.

Disponibilidad y Fiabilidad

- EJEMPLO: Un conmutador telefónico que enruta llamadas telefónicas es un ejemplo de un sistema donde la disponibilidad es más importante que la fiabilidad.
- Los usuarios esperan un tono de marcado cuando levantan el auricular, así que el sistema tiene altos requerimientos de disponibilidad.
- Si ocurre una falla del sistema mientras se establece una conexión, esto con frecuencia es rápidamente recuperable.
- Por consiguiente, la disponibilidad más que fiabilidad es el principal requerimiento de confiabilidad para este tipo de sistema.

Disponibilidad y Fiabilidad

- EJEMPLO:
- Si el sistema A falla una vez al año y el sistema B falla una vez al mes, entonces a todas luces A es más fiable que B.
- Sin embargo, suponga que el sistema A tarda tres días en reiniciarse después de una falla, mientras que el sistema B tarda sólo 10 minutos en reiniciar. La disponibilidad del sistema B durante el año (120 minutos de tiempo muerto) es mucho mejor que la del sistema A (4,320 minutos de tiempo muerto).

Disponibilidad y Fiabilidad

Cuando se discute la fiabilidad, es útil usar terminología precisa y distinguir entre los términos:

- “falla en el desarrollo”,
- “error del sistema” y
- “caída del sistema”

Disponibilidad y Fiabilidad

Término	Descripción
Error o equivocación humano	El comportamiento humano que resulta en la introducción de fallas en el desarrollo en un sistema. Por ejemplo, en la estación meteorológica a campo abierto, un programador podría decidir que la forma de calcular la hora para la siguiente transmisión es agregar una hora a la hora actual. Esto funciona salvo cuando la hora de transmisión es entre 23:00 y medianoche (medianoche es 00:00 en el reloj de 24 horas).
Falla en el desarrollo del sistema	Una característica de un sistema de software que puede conducir a un error del sistema. La falla en el desarrollo es la inclusión del código para agregar una hora a la hora de la última transmisión, sin una verificación para saber si la hora es mayor o igual a 23:00.
Error del sistema	Un estado erróneo del sistema que puede conducir a un comportamiento de sistema que es inesperado para los usuarios del mismo. El valor de la hora de transmisión se establece de manera incorrecta (a 24.XX en vez de 00.XX), cuando se ejecuta el código defectuoso.
Caída del sistema	Un evento que ocurre en algún punto del tiempo, cuando el sistema no entrega un servicio como espera su usuario. No se transmiten datos meteorológicos porque la hora es inválida.

Disponibilidad y Fiabilidad

Enfoques para mejorar la fiabilidad de un sistema:

- ***Prevención de fallas de desarrollo.*** Se usan técnicas de desarrollo que minimizan la posibilidad de los errores humanos y/o captan las equivocaciones antes de que resulten en fallas de desarrollo del sistema. Los ejemplos de estas técnicas incluyen evitar códigos del lenguaje de programación proclives al error, como punteros y el uso de análisis estático para descubrir anomalías del programa.

Disponibilidad y Fiabilidad

Enfoques para mejorar la fiabilidad de un sistema:

- ***Detección y eliminación de fallas en el desarrollo.*** El uso de técnicas de verificación y validación que incrementan las oportunidades de que se detecten y eliminen las fallas en el desarrollo antes de que se use el sistema. Las pruebas y la depuración sistemáticas son un ejemplo de una técnica de detección de este tipo de fallas.

Disponibilidad y Fiabilidad

Enfoques para mejorar la fiabilidad de un sistema:

- ***Tolerancia a fallas en el desarrollo.*** Se refiere a las técnicas que aseguran que las fallas en el desarrollo de un sistema no deriven en errores del sistema o que los errores del sistema no deriven en caídas del sistema. La incorporación de mecanismos de autocomprobación en un sistema y el uso de módulos de sistema redundantes son ejemplos de técnicas de tolerancia a la fallas en el desarrollo.

Protección

- Los sistemas críticos para la protección son aquellos en los que resulta esencial que la operación del sistema sea segura en todo momento; esto es, el sistema nunca debe dañar a las personas o a su entorno, incluso cuando falle.

Protección

- SOFTWARE CRÍTICO PARA PROTECCIÓN:
- ***Software primario crítico para la protección.*** Éste es software embebido que sirve como controlador en un sistema. El mal funcionamiento de tal software puede repetirse en el hardware, lo cual derivaría en una lesión humana o daño ambiental.
- ***Software secundario crítico para la protección.*** Es un software que podría repercutir indirectamente en una lesión. Un ejemplo de dicho software consiste en un sistema de diseño de ingeniería auxiliado por computadora, cuyo mal funcionamiento ocasionaría un error de diseño en el objeto por desarrollar. Esta equivocación quizá cauce una lesión a los individuos, si el sistema diseñado funciona mal.

Protección

Un Sistema Fiable no necesariamente es seguro:

1. Nunca se puede tener una total certeza de que un sistema de software esté libre de fallas en el desarrollo y sea tolerante a los mismos. Las fallas en el desarrollo no detectadas pueden estar inactivas durante mucho tiempo y las fallas en la operación del software pueden ocurrir después de muchos años de operación infalible.
2. La especificación podría estar incompleta en cuanto a que no describe el comportamiento requerido del sistema en algunas situaciones críticas.
3. El mal funcionamiento del hardware origina que el sistema se comporte de forma impredecible, y presente al software con un entorno no anticipado
4. Los operadores del sistema pueden generar entradas que no son individualmente incorrectas, pero que, en ciertas situaciones, conducirían a un mal funcionamiento del sistema.

Protección [Terminología de Seguridad]

Término	Definición
Accidente (o contratiempo)	Evento no planeado o secuencia de eventos que derivan en muerte o lesión de un individuo, o daño a la propiedad o al ambiente. Una sobredosis de insulina es un ejemplo de accidente.
Peligro	Condición con el potencial para causar o contribuir a un accidente. Un ejemplo de riesgo es una falla del sensor que mide la glucosa sanguínea.
Daño	Una medida de la pérdida que resulta de un contratiempo. El daño puede variar desde el hecho de que muchas personas mueran como resultado de un accidente, hasta una lesión o daño menor a la propiedad. El daño, producto de una sobredosis de insulina, sería una lesión grave o la muerte del usuario que utiliza la bomba de insulina.
Severidad del peligro	Una valoración del peor daño posible que resultaría de un peligro en particular. La severidad del peligro puede ser desde catastrófico, cuando muchas personas mueren, hasta menor, cuando sólo ocurre un daño mínimo. Cuando la muerte de un individuo es una posibilidad, la valoración razonable de la severidad del peligro es "muy alta".
Probabilidad del peligro	La probabilidad de que ocurran eventos que causen peligro. Los valores de probabilidad tienden a ser arbitrarios, pero varían de "probable" (por ejemplo, 1/100 de posibilidad de que ocurra un peligro) a "improbable" (no son probables situaciones concebibles en que pudiera ocurrir el peligro). Es baja la probabilidad de que la falla en un sensor de la bomba de insulina dé como resultado una sobredosis.
Riesgo	Ésta es una medida de probabilidad de que el sistema causará un accidente. El riesgo se valora al considerar la posibilidad, severidad y verosimilitud de que el peligro conduzca a un accidente. El riesgo de una sobredosis de insulina es quizá de medio a bajo.

Seguridad

- La seguridad es un atributo del sistema que refleja la habilidad de éste para protegerse a sí mismo de ataques externos, que podrían ser accidentales o deliberados.
- Estos ataques externos son posibles puesto que la mayoría de las computadoras de propósito general ahora están en red y, en consecuencia, son accesibles a personas externas.
- Ejemplos de Ataques: Virus, acceso sin autorización, modificación no aprobada de datos
- Los sistemas militares, los de comercio electrónico y los que requieren procesamiento e intercambio de información confidencial deben diseñarse de modo que logren un alto nivel de seguridad.

Seguridad

Amenazas a la seguridad en Sistemas en Red:

- ***Amenazas a la confidencialidad del sistema y sus datos.*** Pueden difundir información a individuos o programas que no están autorizados a tener acceso a dicha información.
- ***Amenazas a la integridad del sistema y sus datos.*** Tales amenazas pueden dañar o corromper el software o sus datos.
- ***Amenazas a la disponibilidad del sistema y sus datos.*** Dichas amenazas pueden restringir el acceso al software o sus datos a usuarios autorizados.

Seguridad

Controles que mejoran la seguridad de un sistema:

- ***Evitar la vulnerabilidad.*** Controles cuya intención sea garantizar que los ataques no tengan éxito. Aquí, la estrategia es diseñar el sistema de modo que se eviten los problemas de seguridad. Por ejemplo, los sistemas militares sensibles no están conectados a redes públicas, de forma que es imposible el acceso externo. También hay que pensar en la encriptación como un control basado en la evitación. Cualquier acceso no autorizado a datos encriptados significa que el atacante no puede leerlos. En la práctica, es muy costoso y consume mucho tiempo romper una encriptación fuerte.

Seguridad

Controles que mejoran la seguridad de un sistema:

- ***Detectar y neutralizar ataques.*** Controles cuya intención sea detectar y repeler ataques. Dichos controles implican la inclusión de funcionalidad en un sistema que monitoriza su operación y verifica patrones de actividad inusuales. Si se detectan, entonces se toman acciones, como desactivar partes del sistema, restringir el acceso a ciertos usuarios, etcétera.

Seguridad

Controles que mejoran la seguridad de un sistema:

- ***Limitar la exposición y recuperación.*** Controles que soportan la recuperación de los problemas. Éstos varían desde estrategias de respaldo automatizadas y "réplica" de la información, hasta pólizas de seguros que cubran los costos asociados con un ataque exitoso al sistema.