ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DESARROLLO DE SOFTWARE SEGURO

Unidad 1: Introducción y Estudio de Vulnerabilidades

Tema 2: Amenazas a la seguridad del software

Profesor: Geovanny Cudco/agcudco@espe.edu.ec

CONTENIDOS

Introducción

1.2. Amenazas a la seguridad del software

1.2.1. Amenazas a nivel de Hardware

1.2.2. Amenazas a nivel de Código

1.2.3. Amenazas a nivel de Diseño

1.2.4. Amenazas a nivel de Arquitectura

Introducción

Vulnerabilidad

- Fallo o debilidad de un sistema de información que pone en riesgo la seguridad de la misma.
- ***Agujero*** que puede ser producido por un error de configuración, una carencia de procedimientos o un fallo de diseño.
- **Los ciberdelincuentes aprovechan las vulnerabilidades de los sistemas informáticos para entrar en los mismos y realizar actividades ilegales, robar información sensible o interrumpir su funcionamiento.**
- Es una de las principales causas por las que una empresa puede sufrir un ataque informático contra sus sistemas

CONTENIDOS

Introducción

1.2. Amenazas a la seguridad del software

1.2.1. Amenazas a nivel de Hardware

1.2.2. Amenazas a nivel de Código

1.2.3. Amenazas a nivel de Diseño

1.2.4. Amenazas a nivel de Arquitectura

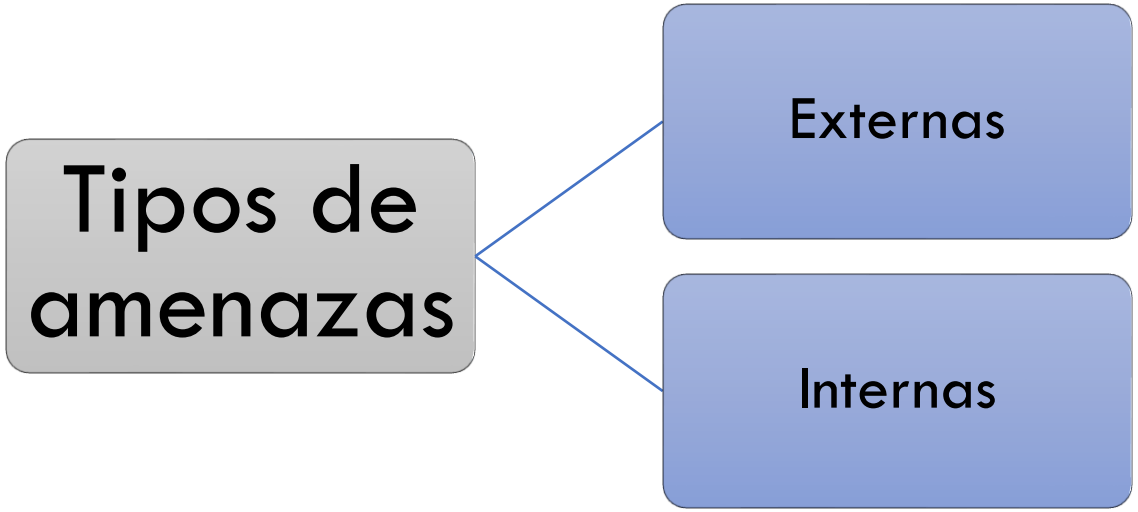
1.2. Amenazas a la seguridad del software

Amenazas

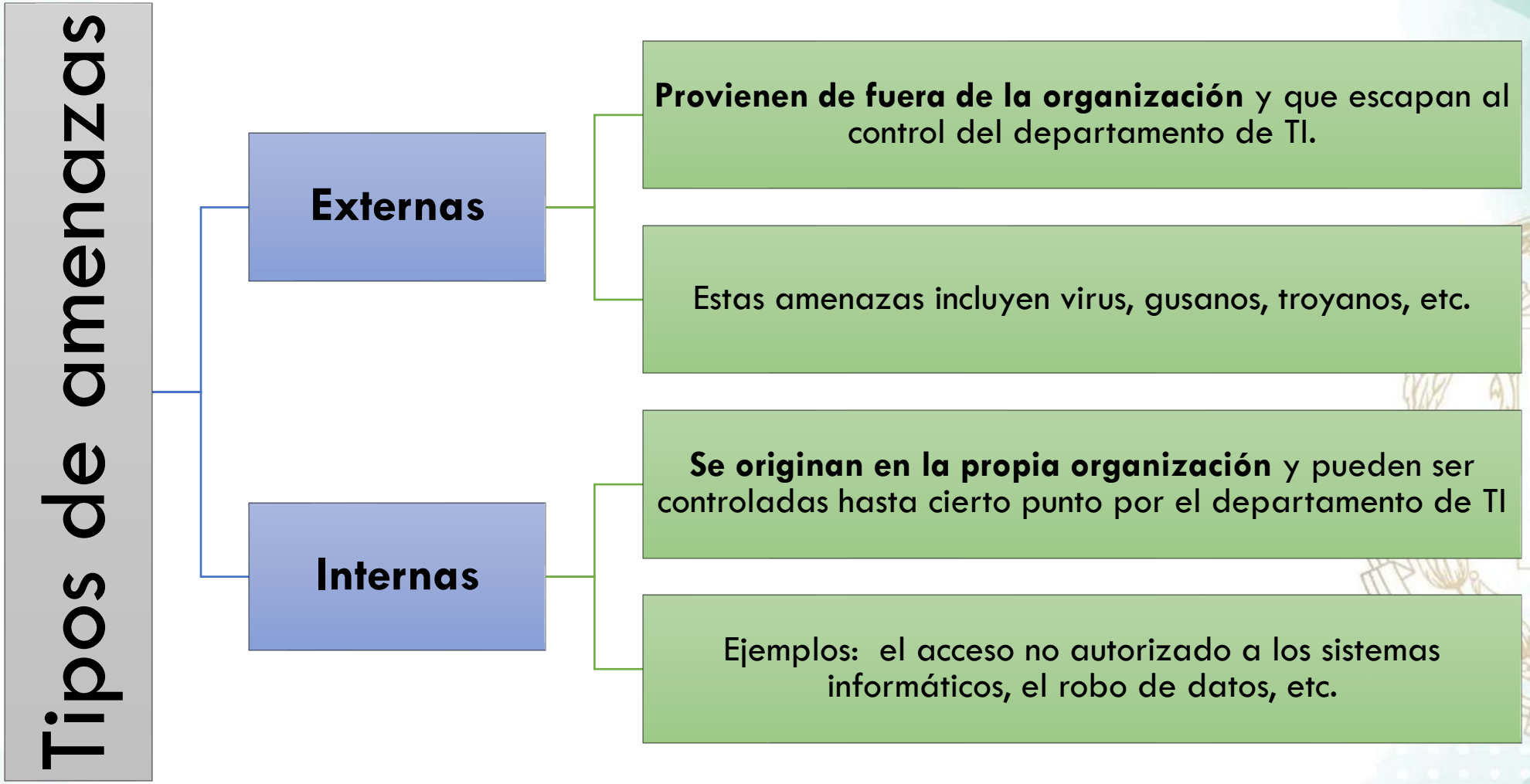
- Acción que aprovecha una vulnerabilidad para atacar o invadir un sistema informático
- Provienen en gran medida de ataques externos, aunque también existen amenazas internas
- **Explotación de una vulnerabilidades o fallos** que se utilizan para afectar la operatividad de un sistema
- Tienen intención de sacar algún provecho

1.2. Amenazas a la seguridad del software

Las amenazas informáticas pueden clasificarse, según su origen y a grandes rasgos, en dos categorías



1.2. Amenazas a la seguridad del software



1.2. Amenazas a la seguridad del software

IMPACTOS

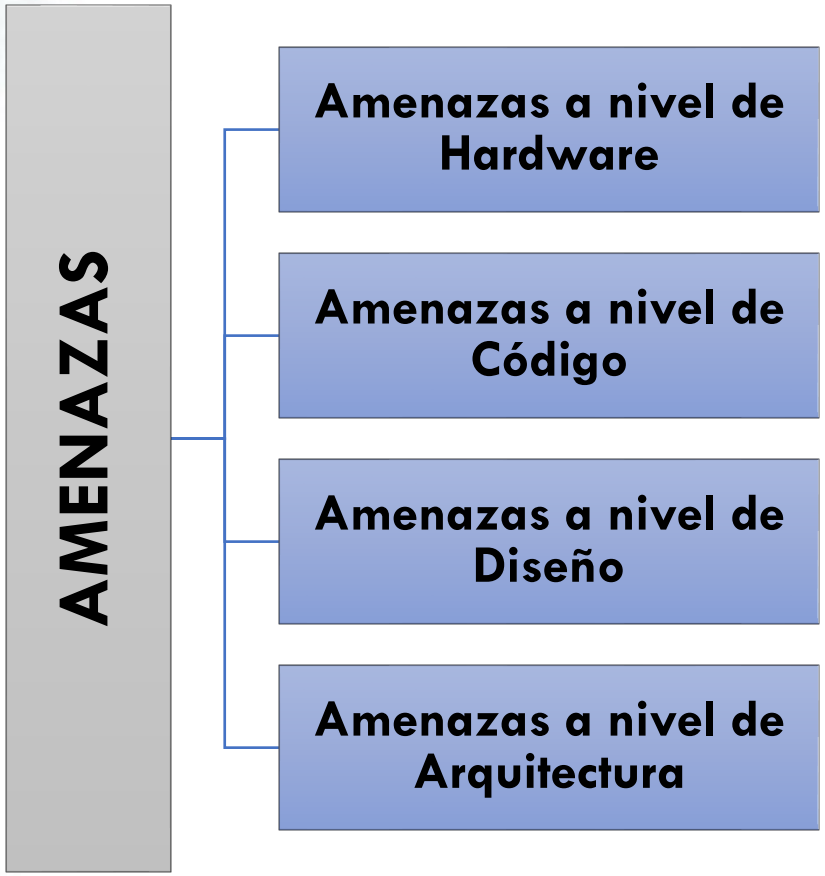
- Pueden conducir a la pérdida de datos, al tiempo de inactividad de los sistemas críticos, a pérdidas financieras, etc.
- Es importante que tener una estrategia de seguridad informática eficaz para protegerse de estas amenazas.

1.2. Amenazas a la seguridad del software

MECANISMOS DE PREVENCIÓN

- Recurrir a acciones de mantenimiento de equipos.
- Configurar de manera adecuada la seguridad de los equipos y software.
- Añadir validaciones en sitios web.
- Crear contraseñas seguras.
- No abrir documentos anexos en correos electrónicos pertenecientes a remitentes desconocidos.
- No proporcionar datos personales o sensibles a terceros porque te pueden robar información.
- Mantener actualizado el software de los equipos y dispositivos, entre otras.

1.2. Amenazas a la seguridad del software



CONTENIDOS

Introducción

1.2. Amenazas a la seguridad del software

1.2.1. Amenazas a nivel de Hardware

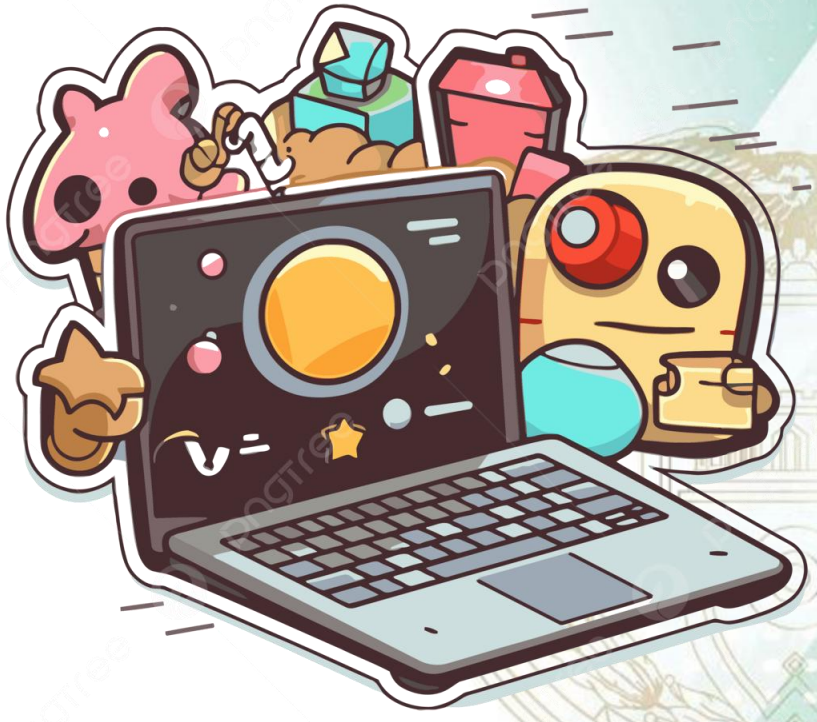
1.2.2. Amenazas a nivel de Código

1.2.3. Amenazas a nivel de Diseño

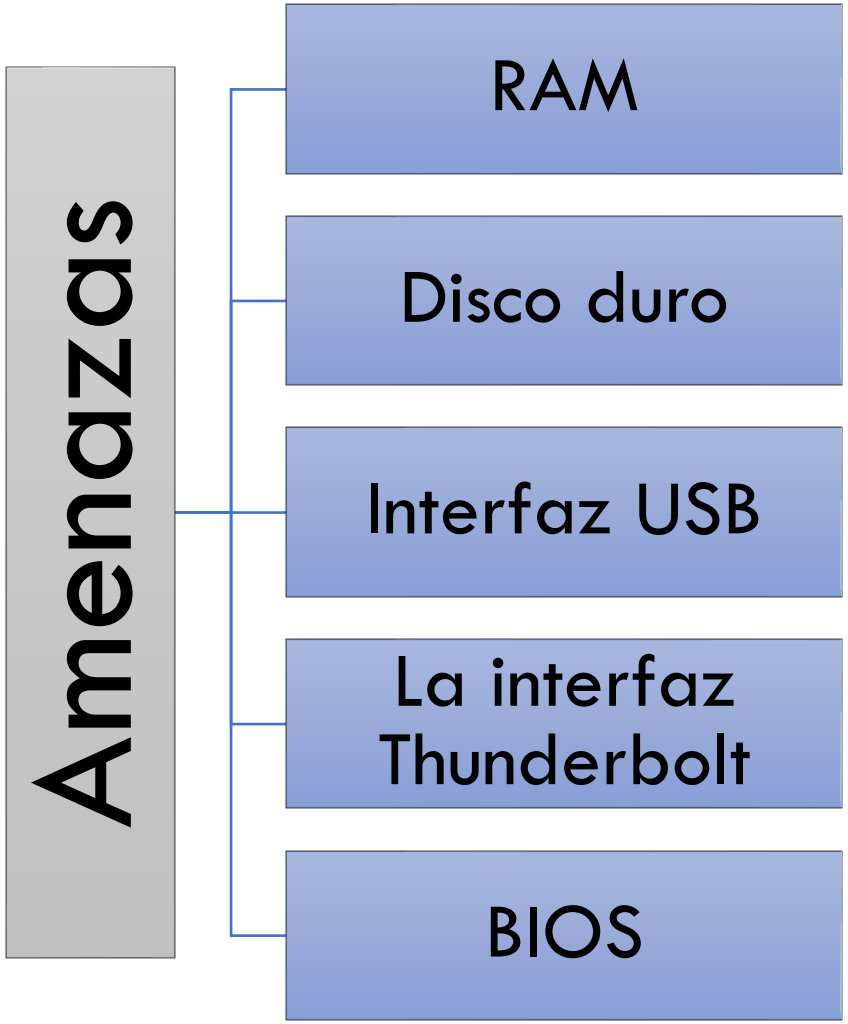
1.2.4. Amenazas a nivel de Arquitectura

1.2.1. Amenazas a nivel de Hardware

- La mayoría asocia malware, virus y vulnerabilidades solo con programas y apps. Pocos saben que también afectan al hardware.
- El hardware parece seguro y limpio. Pero la realidad es distinta.
- La complejidad creciente de los firmwares genera nuevas vulnerabilidades en el hardware.
- Muchas amenazas son indetectables por las soluciones de seguridad actuales. En algunos casos, un equipo infectado no se puede reparar.



1.2.1. Amenazas a nivel de Hardware



1.2.1. Amenazas a nivel de Hardware

RAM

- La mayor amenaza para el hardware es la seguridad de la DDR DRAM, que no se puede solucionar mediante ningún parche de software.

- Una vulnerabilidad para este tipo de memoria RAM se denomina Rowhammer, y radica en que como los elementos del hardware soldados en el chip se encuentran colocados cada vez más cerca, comienzan a interferir unos con otros.

1.2.1. Amenazas a nivel de Hardware

DISCO DURO

- El firmware que controla los discos duros contiene elementos que se pueden piratear, dañando el dispositivo de tal manera que no haya reparación posible y que la forma más fiable de deshacerse del malware sea destruyendo el disco duro.

- No obstante, este tipo de ataques resultan caros y complicados

1.2.1. Amenazas a nivel de Hardware

La interfaz USB

- BadUSB es una vulnerabilidad crítica que permite inyectar un código malicioso en el controlador del USB y ningún antivirus es capaz de detectarlo.

- Algunos expertos, incluso, aconsejan dejar de usar los puertos USB para minimizar los riesgos.

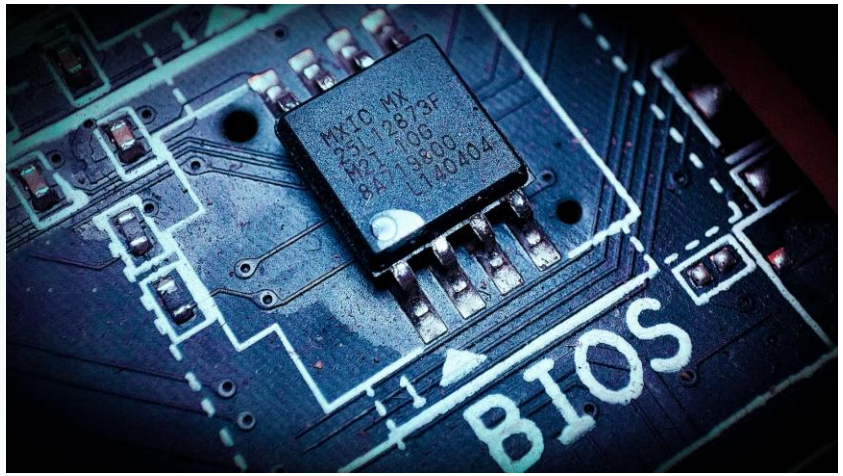
1.2.1. Amenazas a nivel de Hardware

La interfaz Thunderbolt

Existe una vulnerabilidad que afecta a la conexión del cable Thunderbolt y un PoC que permite explotar los módulos auxiliares de inicio desde dispositivos externos conectados a través de ese cable y hacerse con el equipo infectado.

1.2.1. Amenazas a nivel de Hardware

BIOS



La última vulnerabilidad de UEFI **(Interfaz Extensible de Firmware)** permite sobrescribir sobre el BIOS sin que se pueda hacer nada al respecto.

CONTENIDOS

Introducción

1.2. Amenazas a la seguridad del software

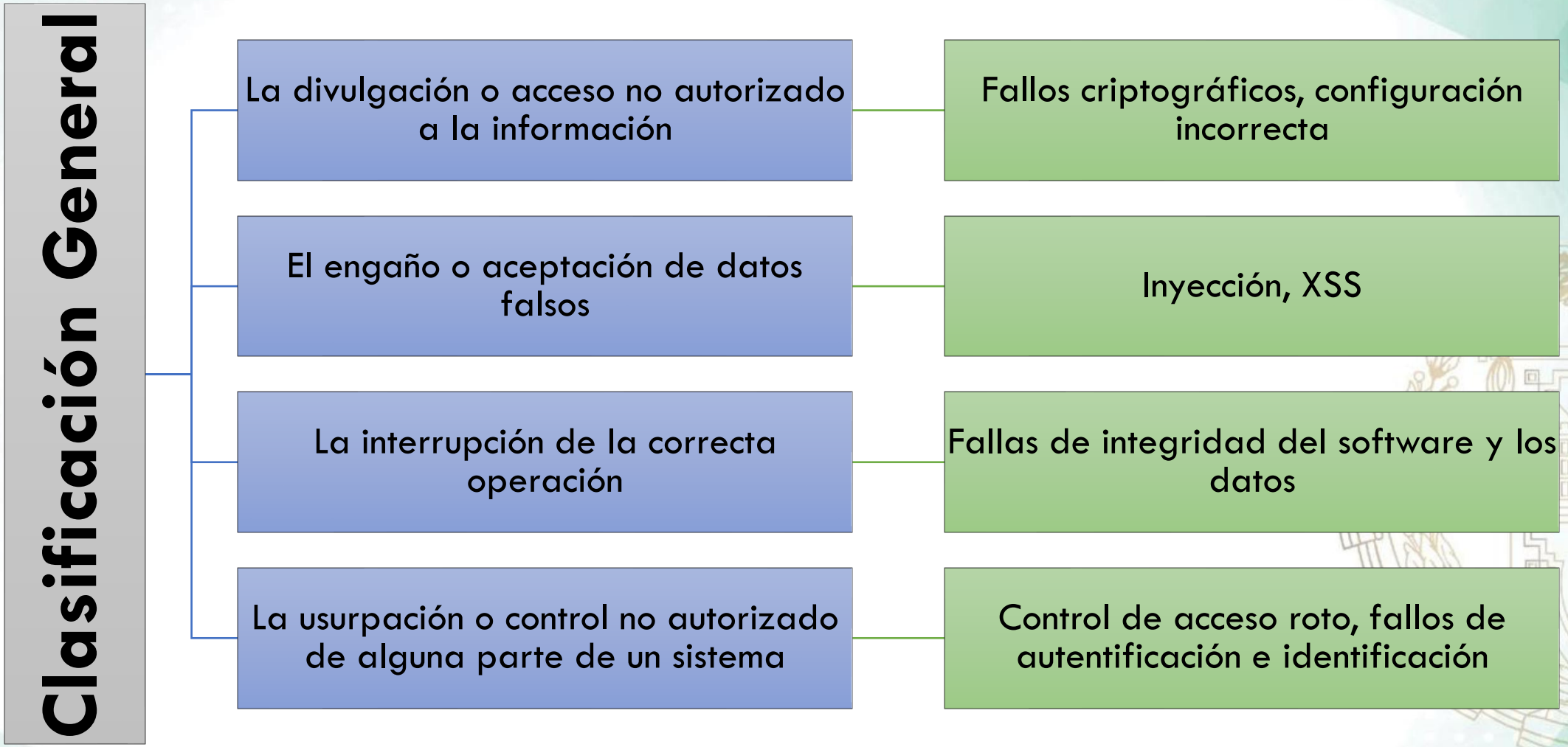
1.2.1. Amenazas a nivel de Hardware

1.2.2. Amenazas a nivel de Código

1.2.3. Amenazas a nivel de Diseño

1.2.4. Amenazas a nivel de Arquitectura

1.2.2. Amenazas a nivel de Código



1.2.2. Amenazas a nivel de Código

Malware:

- Abarca las vulnerabilidades que van desde virus informático hasta adware que pueden infectar tanto los ordenadores como las páginas web.

Sql injection:

- La Inyección SQL es un tipo de ciberataque que involucra declaraciones SQL maliciosas o códigos de aplicación que se inyectan en los campos de entrada del usuario. Este proceso permite a los atacantes obtener acceso al backend de la web o al contenido corrupto de la base de datos.

1.2.2. Amenazas a nivel de Código

Interception:

- Uhacker captura datos que los usuarios envían a una web, y luego los utiliza para su propio beneficio. Puede ser información de contacto o datos sensibles como la tarjeta de crédito.

Cross-Site Scripting (xss):

- Líneas de código JavaScript malicioso se inyectan en una página para dirigirse a los usuarios de dicha web, manipulando scripts del lado del cliente.
- Estos scripts secuestran las sesiones de los usuarios a través de la barra de búsqueda de una página web o comentarios (a través del backend).

1.2.2. Amenazas a nivel de Código

Ataques de contraseñas

- Algunos hackers adivinan contraseñas o usan herramientas y programas de diccionario para probar diferentes combinaciones hasta que las encuentran.

Desbordamiento de buffer:

- Se produce cuando el software que escribe datos en un búfer desborda la capacidad del búfer, lo que provoca que se sobrescriban las ubicaciones de memoria adyacentes.
- En otras palabras, se pasa demasiada información a un contenedor que no cuenta con el espacio suficiente, y esa información acaba sustituyendo a los datos de los contenedores adyacentes.

1.2.2. Amenazas a nivel de Código

- Desbordamiento de buffer

C++

```cpp
// example1.cpp
// stack-buffer-overflow error
#include <string.h>

int main(int argc, char **argv) {
    char x[10];
    memset(x, 0, 10);
    int res = x[argc * 10];

    return res;
}
```

```c
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    char nombre[5]="";
    char apellido[10]="";

    printf("Ingrese su nombre: ");
    scanf("%s",&nombre);

    printf("\nSu nombre es: %s\n",nombre);
    printf("\nSu apellido es: %s\n",apellido);
    return 0;
}
```

```c
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    char nombre[5]="";
    char apellido[10]="";

    printf("Ingrese su nombre: ");
    scanf("%s",&nombre);

    printf("\nSu nombre es: %s\n",nombre);
    printf("\nSu apellido es: %s\n",apellido);
    return 0;
}
```

CONTENIDOS

Introducción

1.2. Amenazas a la seguridad del software

1.2.1. Amenazas a nivel de Hardware

1.2.2. Amenazas a nivel de Código

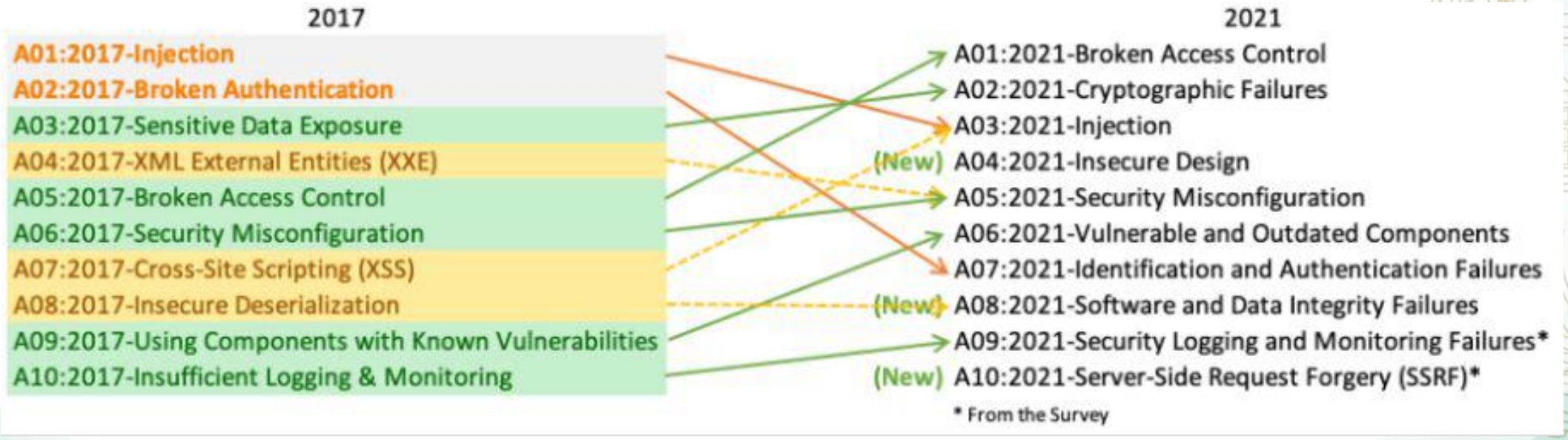
1.2.3. Amenazas a nivel de Diseño

1.2.4. Amenazas a nivel de Arquitectura

1.2.3. Amenazas a nivel de Diseño

Diseño Inseguro

La OWASP Top 10 2021 incluye el Diseño Inseguro como una nueva categoría, clasificada como la cuarta preocupación crítica de seguridad.



1.2.3. Amenazas a nivel de Diseño

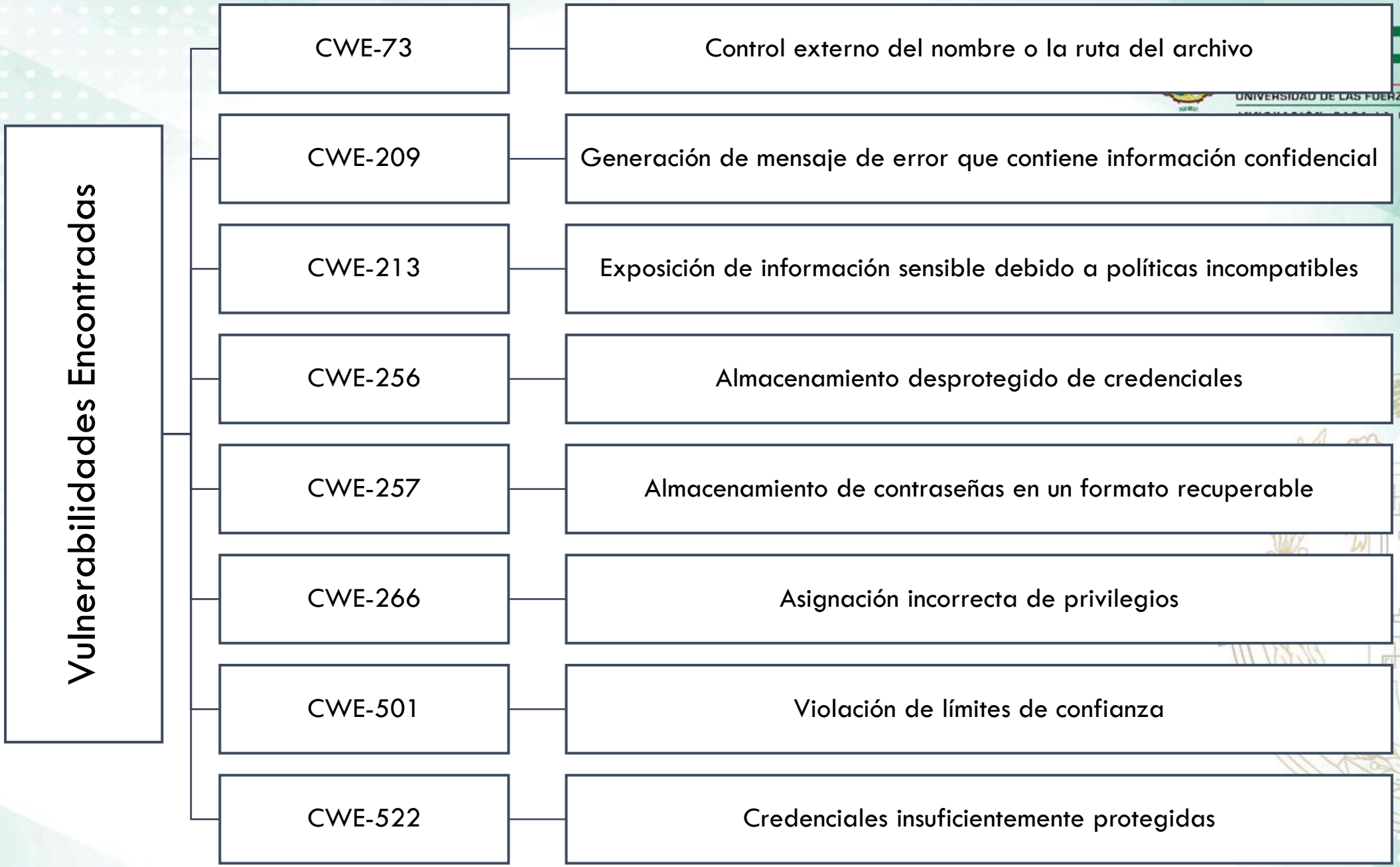
Diseño Inseguro

- El Diseño Inseguro es una nueva categoría con un enfoque en los riesgos relacionados con fallas de diseño.
- Si realmente queremos "avanzar" como industria, se necesita más modelados de amenazas, patrones y principios de diseño seguros, así como arquitecturas seguras de referencia.
- Un diseño inseguro no puede solucionarse con una implementación perfecta ya que, por definición, los controles de seguridad necesarios nunca se crearon para defenderse de ataques específicos

1.2.3. Amenazas a nivel de Diseño

¿Cuando se generan vulnerabilidades en el diseño?

El diseño inseguro involucra los riesgos relacionados con el diseño y las fallas arquitectónicas que se incorporan desde el comienzo del desarrollo del software.


Vulnerabilidades Encontradas
CWE-73
Control externo del nombre o la ruta del archivo
CWE-209
Generación de mensaje de error que contiene información confidencial
CWE-213
Exposición de información sensible debido a políticas incompatibles
CWE-256
Almacenamiento desprotegido de credenciales
CWE-257
Almacenamiento de contraseñas en un formato recuperable
CWE-266
Asignación incorrecta de privilegios
CWE-501
Violación de límites de confianza
CWE-522
Credenciales insuficientemente protegidas

Métodos de Prevención

Establecer y utilizar una biblioteca de patrones de diseño seguro

Los patrones de diseño son una solución general, reutilizable y aplicable a diferentes problemas de diseño de software.

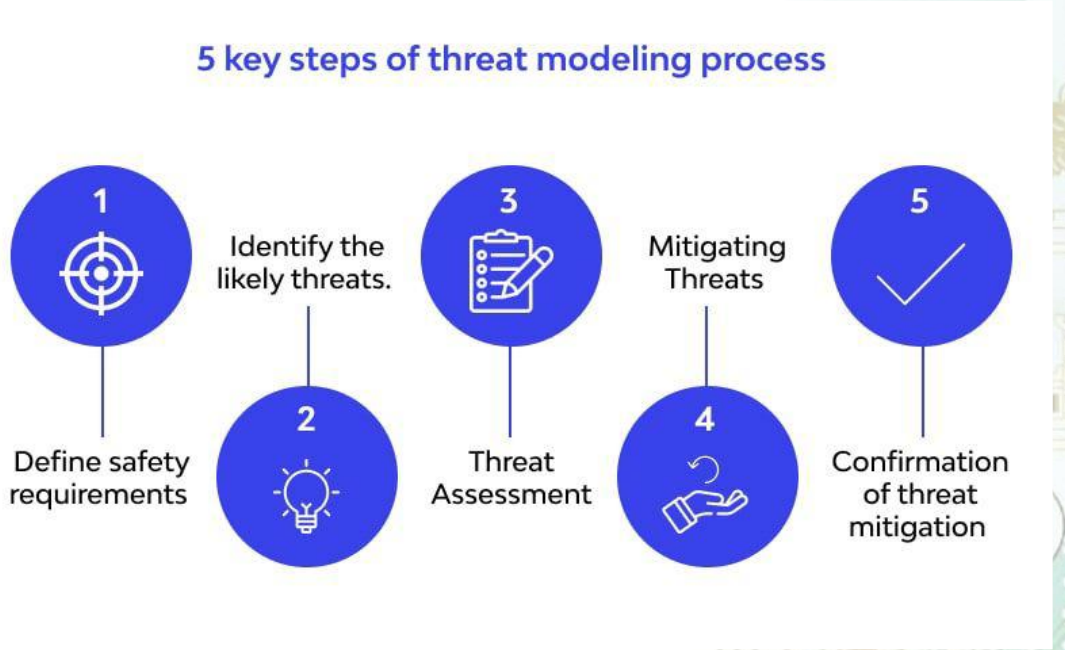
- **Factory Method:** una fábrica de software produce objetos.
- **Singleton:** se utiliza para limitar la creación de una clase a un solo objeto
- **Observer:** cuando un objeto cambia de estado, se notifica a todos sus dependientes.
- **Strategy:** agrupar algoritmos relacionados bajo una abstracción.
- **Adapter:** esto permite que las clases incompatibles trabajen juntas al convertir la interfaz de una clase en otra
- **Builder:** patrón de construcción para construir objetos
- **State:** Encapsula los diversos estados en los que puede estar una máquina

Métodos de Prevención

Utilizar el modelado de amenazas para la autenticación crítica, el control de acceso, la lógica empresarial y los flujos clave (Threat Modeling Process)

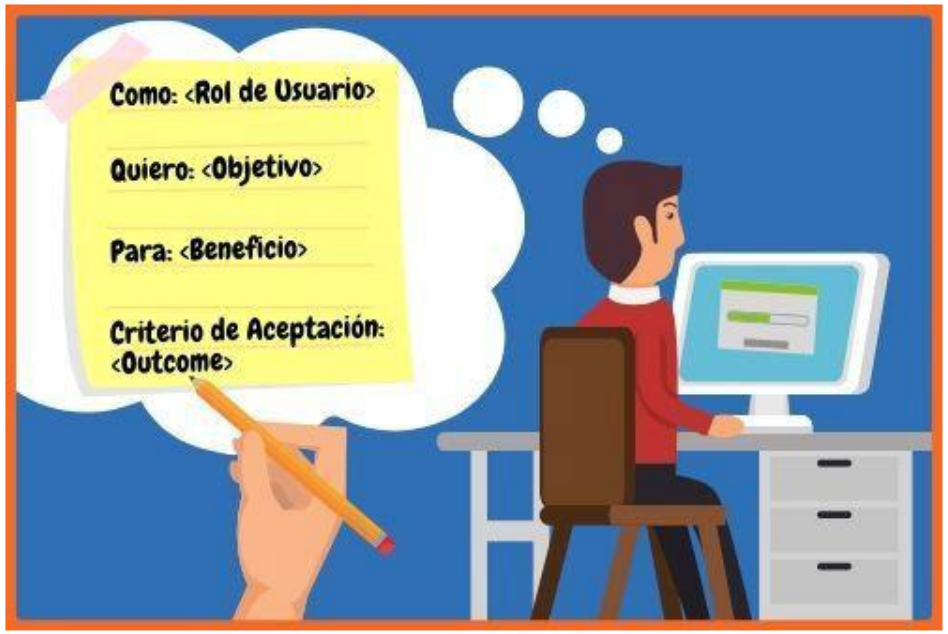
Modelado de amenazas de aplicaciones que le permite identificar, cuantificar y abordar los riesgos de seguridad asociados a una aplicación.

Paso 1: Descomponer la aplicación
Paso 2: Determinar y clasificar las amenazas
Paso 3: Determinar las contramedidas y la mitigación



Métodos de Prevención

Integrar el lenguaje y los controles de seguridad en las historias de los usuarios



- Escribir pruebas unitarias y de integración para validar que todos los flujos críticos son resistentes al modelo de amenazas.
- Compilar casos de uso y casos de mal uso para cada nivel de su aplicación.

CONTENIDOS

Introducción

1.2. Amenazas a la seguridad del software

1.2.1. Amenazas a nivel de Hardware

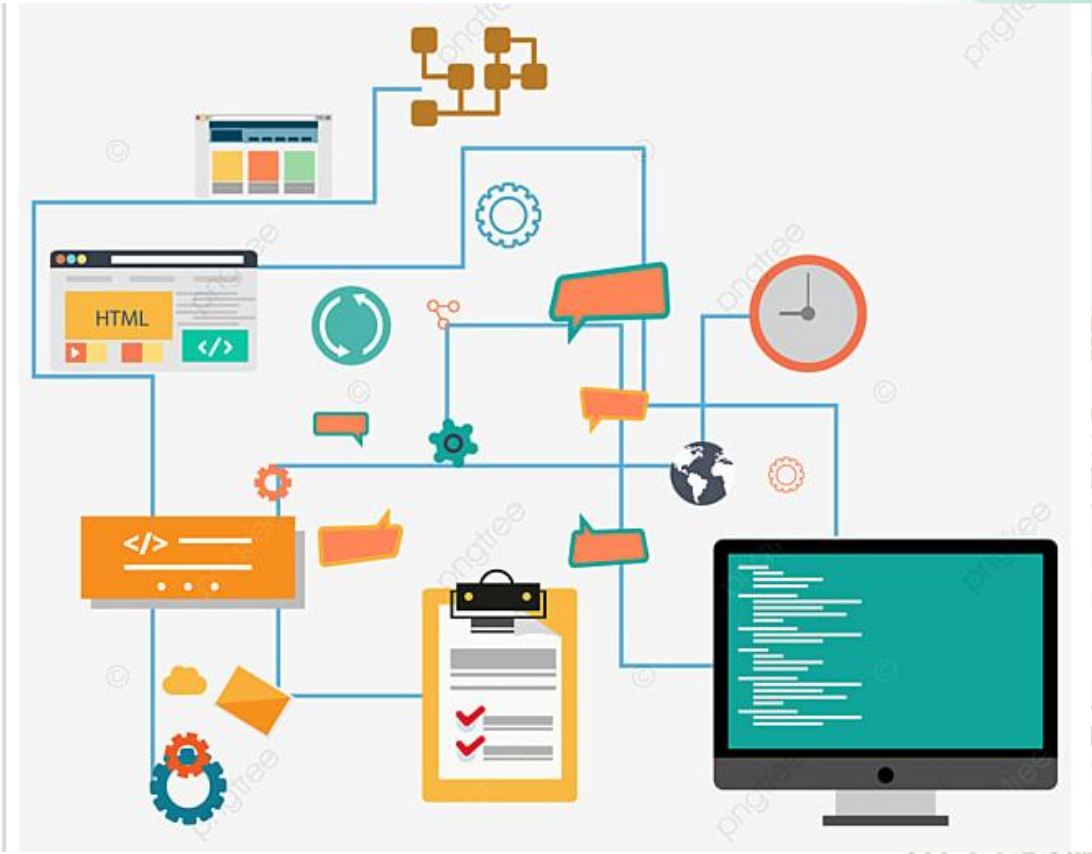
1.2.2. Amenazas a nivel de Código

1.2.3. Amenazas a nivel de Diseño

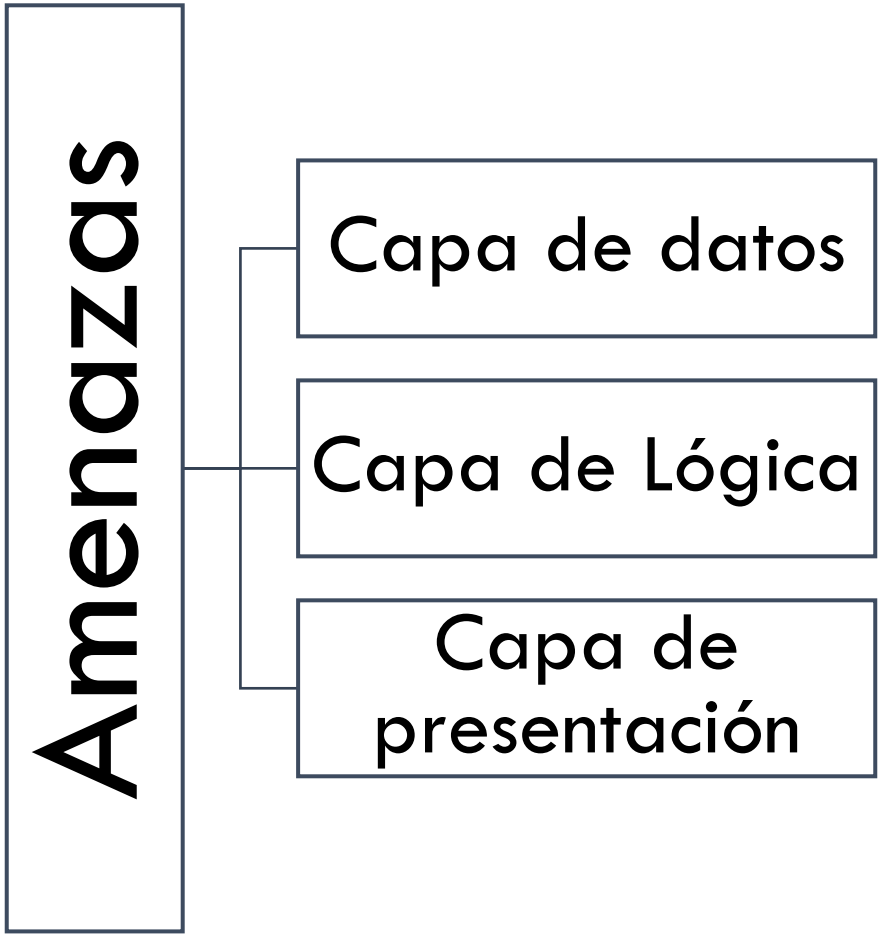
1.2.4. Amenazas a nivel de Arquitectura

1.2.4. Amenazas a nivel de Arquitectura

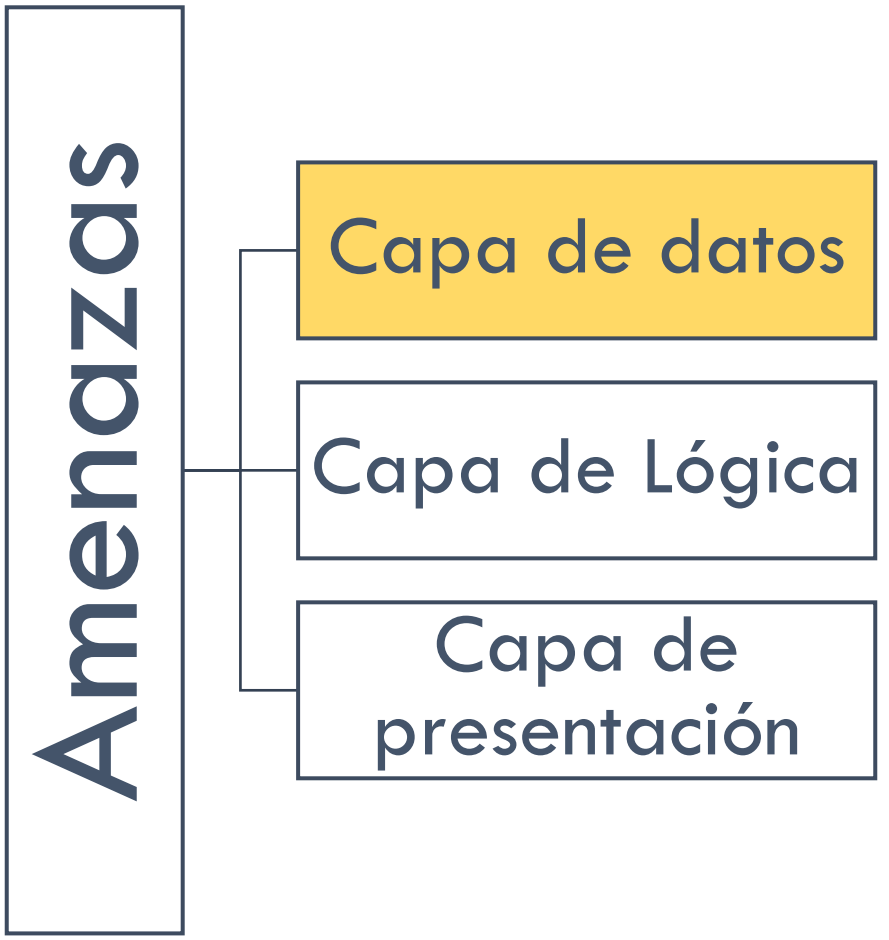
De acuerdo al Software Engineering Institute (SEI), la Arquitectura de Software se refiere a "**las estructuras de un sistema, compuestas de elementos con propiedades visibles de forma externa y las relaciones que existen entre ellos.**"

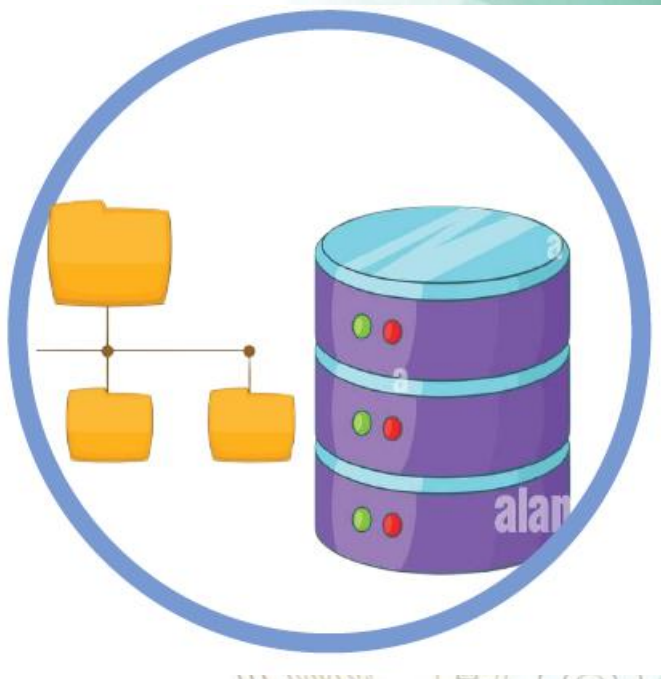


1.2.4. Amenazas a nivel de Arquitectura

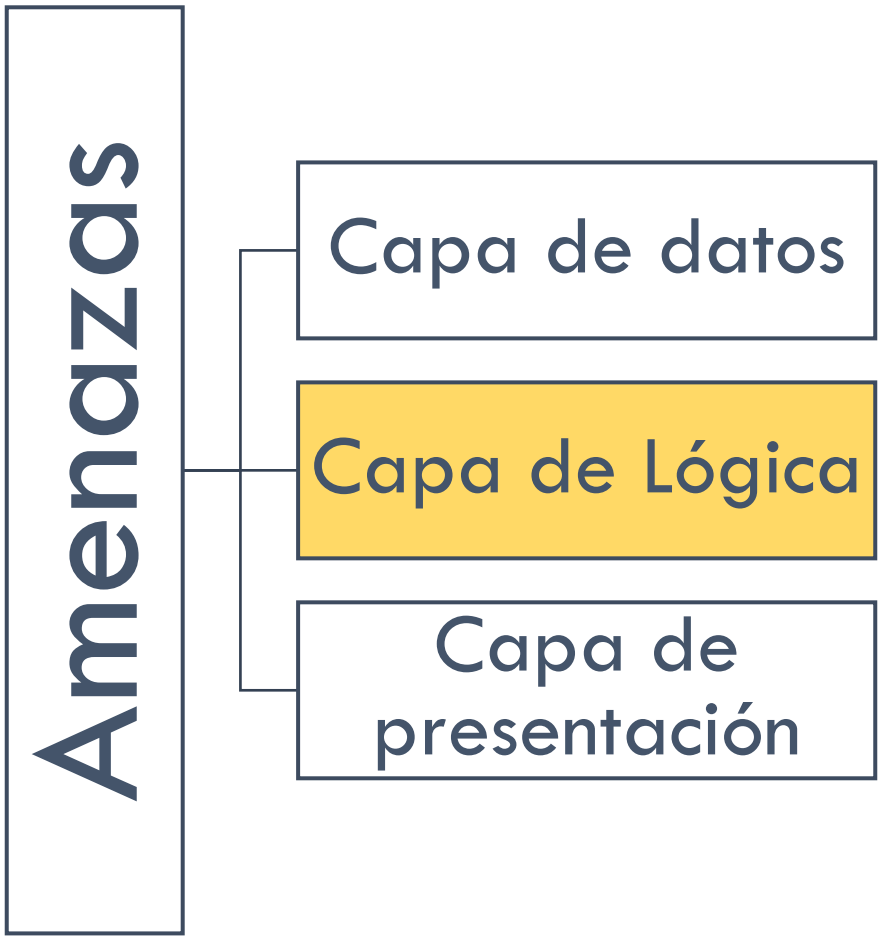


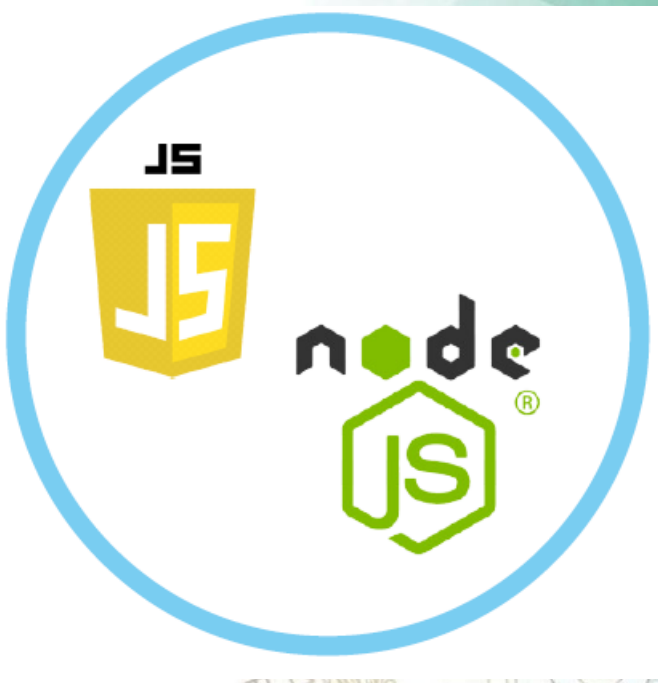
1.2.4. Amenazas a nivel de Arquitectura





1.2.4. Amenazas a nivel de Arquitectura





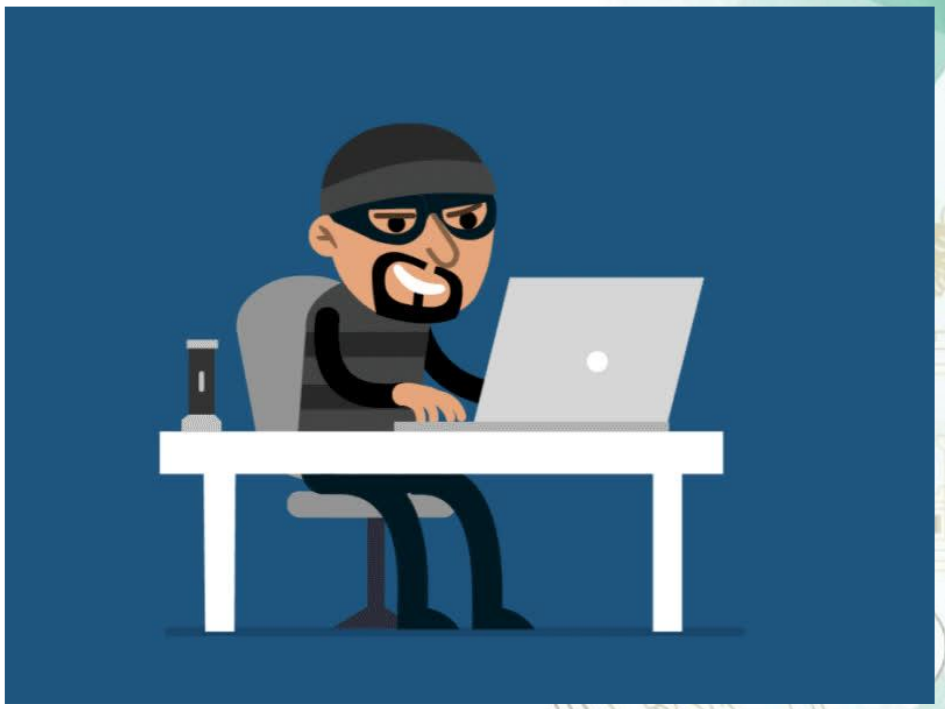
1.2.4. Amenazas a nivel de Arquitectura

1.2.4. Amenazas a nivel de Arquitectura

- **Ejemplos**
  - **Cracker**
  - **Criminal Informático**
  - **Terrorist**
  - **Industrial espionaje**
  - **Insiders**

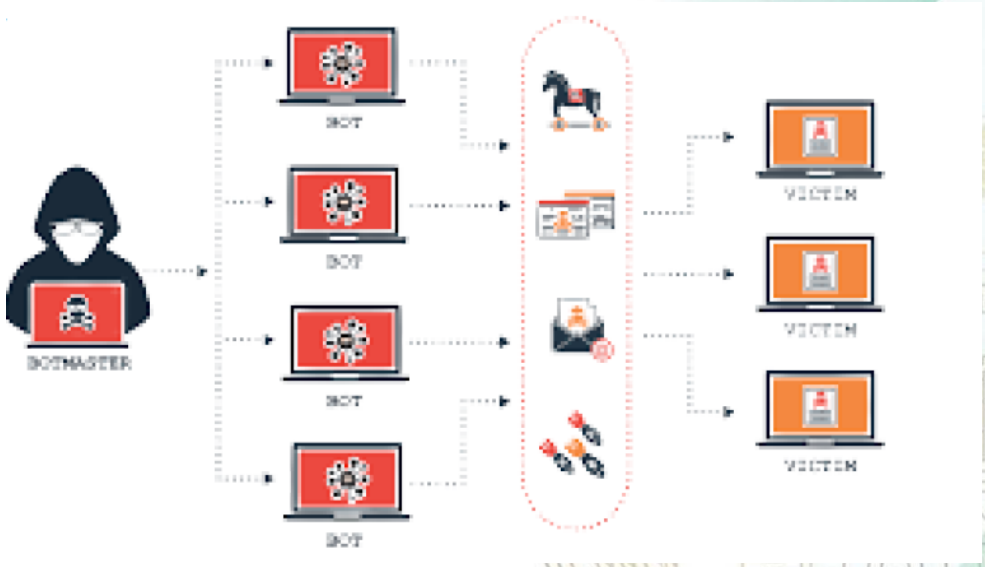
1.2.4. Amenazas a nivel de Arquitectura

- Ejemplos
  - Cracker
  - Criminal Informático
  - Terrorist
  - Industrial espionaje
  - Insiders



1.2.4. Amenazas a nivel de Arquitectura

- Ejemplos
  - Cracker
  - Criminal Informático
  - Terrorist
  - Industrial espionaje
  - Insiders



1.2.4. Amenazas a nivel de Arquitectura

- Ejemplos
  - Cracker
  - Criminal Informático
  - Terrorist
  - Industrial espionaje
  - Insiders

1.2.4. Amenazas a nivel de Arquitectura

- Ejemplos
  - Cracker
  - Criminal Informático
  - Terrorist
  - Industrial espionaje
  - Insiders

1.2.4. Amenazas a nivel de Arquitectura

Mitigación de amenazas

Solución	Tecnología
Prevención de fuga de información	DLP
Protección de datos en nubes públicas	Office365/Gsuite Protection
Doble factor de autenticación	Tokens
Cifrado de portales Web	HTTPS
Cifrado de datos	Encryption Disk

1.2.4. Amenazas a nivel de Arquitectura

Mitigación de amenazas

Solución	Tecnología
Protección de servicios con acceso desde internet	NGIPS/IPS
Interconexión segura entre oficinas	VPN
Mitigación de ataques de DDoS	Anti-DDoS
Protección contra sitios web maliciosos	URL-Filter

1.2.4. Amenazas a nivel de Arquitectura

Mitigación de amenazas

Solución	Tecnología
Protección contra sitios web maliciosos	Secure DNS
Red de invitados segura	Antimalware
Protección de dispositivos móviles	MDM
Protección contra correos electrónicos maliciosos	E-Mail Protection
Protección contra dispositivos de almacenamiento removibles	Antimalware

1.2.4. Amenazas a nivel de Arquitectura

Mitigación de amenazas

Solución	Tecnología
Protección de aplicaciones web	Web Application Firewall
Parcheo virtual de servidores	Virtual Patch
Protección de bases de datos	Data Base Firewall