Definiendo requerimientos de seguridad sin morir en el intento

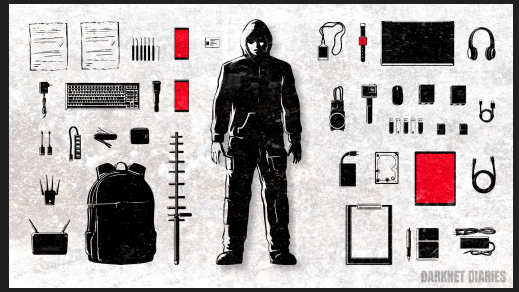
Arturo Sustaita
Dic_2024

@arturo_io

arturo-sustaita

$whoami

- ➔ Arquitecto de Ciberseguridad de Aplicaciones
- ➔ GWEB, CEH, CC.
- ➔ Developer (project side)
- ➔ Apasionado de la Ciberseguridad



Disclaimer

El contenido de esta plática parte de experiencias y aprendizajes profesionales. No refleja en ningún momento escenarios reales ni la opinión de mi actual o pasados empleadores. Cualquier parecido con la realidad es mera coincidencia.


REQUIRMENTY
SECURITY
LAUNCH DELAYED
SECURITY

- ¿Cómo definimos lineamientos de seguridad?
- ¿Cómo logramos la aceptación de requerimientos por Product Owners?
- ¿Cómo preparamos a nuestro equipo de seguridad ante estas situaciones?

Lineamientos, Frameworks y más...

Back to Basics

Triada de la información

- Confidencialidad
- Integridad
- Disponibilidad
- No repudio

Principio de mínimo privilegio

El privilegio más bajo para realizar sus operaciones

Necesidad de conocer

No todos los usuarios necesitan conocer el 100% de la información

Sin inventar el hilo negro…

Políticas internas
Lineamientos creados por la propia organización en materia de seguridad de la información

Compliance
- PCI-DSS
- LFDPP
- CUB

Frameworks
- OWASP
- CIS Controls

Application Security Verification Standard - ASVS
Top 10 OWASP
Cheat Sheets
Testing Guide

File Upload Cheat Sheet

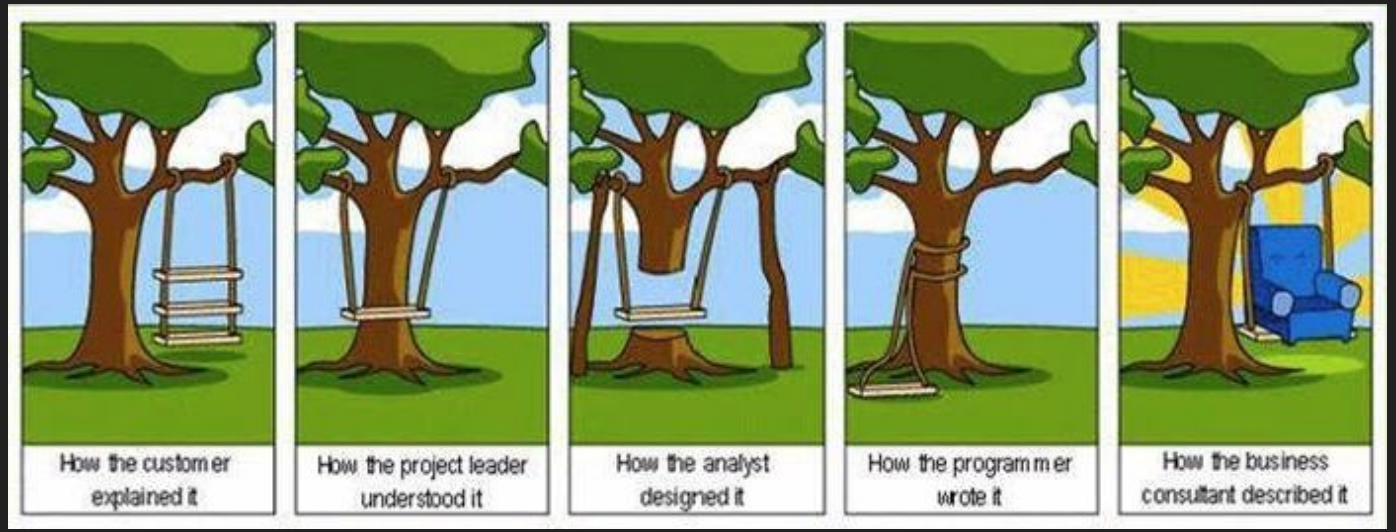
Introduction

File upload is becoming a more and more essential part of any application, where the user is able to upload their photo, their CV, or a video showcasing a project they are working on. The application should be able to fend off bogus and malicious files in a way to keep the application and the users safe.

In short, the following principles should be followed to reach a secure file upload implementation:

- **List allowed extensions. Only allow safe and critical extensions for business functionality**
  - **Ensure that input validation is applied before validating the extensions.**
- **Validate the file type, don't trust the Content-Type header as it can be spoofed**
- **Change the filename to something generated by the application**
- **Set a filename length limit. Restrict the allowed characters if possible**
- **Set a file size limit**
- **Only allow authorized users to upload files**
- **Store the files on a different server. If that's not possible, store them outside of the webroot**
  - **In the case of public access to the files, use a handler that gets mapped to filenames inside the application (someid -> file.ext)**
- **Run the file through an antivirus or a sandbox if available to validate that it doesn't contain malicious data**
- **Ensure that any libraries used are securely configured and kept up to date**
- **Protect the file upload from CSRF attacks**

Entendiendo a los usuarios



Retos principales...

Quiero vs Necesito

Las verdaderas necesidades de los usuarios pueden venir disfrazadas

Quiero, que mi sesión nunca caduque
Necesito, acceso rápido a mis aplicaciones.

Traducir el lenguaje técnico

Implementar doble factor de autenticación, autorización, cifrado.

Fácil vs Adecuado

Lo fácil para el usuario suele ser inseguro

Fácil: Descargar información.
Adecuado: Consulta a través de aplicaciones.

Threat Modeling

“Threat modeling is an effective technique to help secure your systems, applications, networks, and services. It helps you identify potential threats and risk reduction strategies earlier in the development lifecycle.”

https://learn.microsoft.com/en-us/training/modules/tm-introduction-to-threat-modeling/1-introduction

1. **Spoofing:** Making false identity claims -> *Suplantación de credenciales*
2. **Tampering.** Unauthorized data modification -> *Alterar los datos en tránsito entre un cliente y un servidor*
3. **Repudiation:** Performing actions and then denying that you did -> *Un usuario niega haber realizado una compra sin un registro confiable*

1. **Information Disclosure:** Leaking sensitive data to unauthorized parties -> *Fuga de datos sensibles.*
2. **Denial of Service:** Crashing or overloading a system to impact its availability -> *Ataques DDoS*
3. **Elevation of Privilege:** Manipulating a system to gain unauthorized privileges -> *Un atacante logra acceso como administrador en un sistema*

Definiendo requerimientos...

Confidencialidad

PCI-DSS

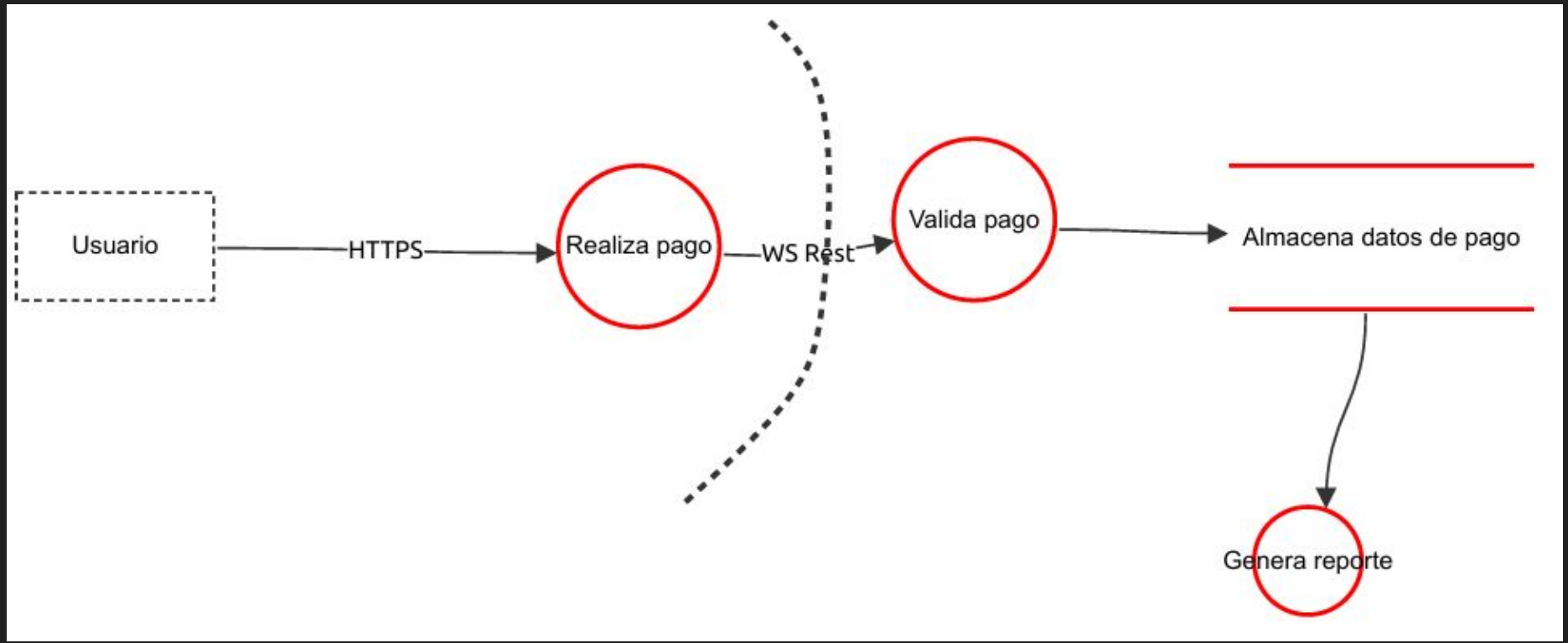
Como analista financiero, quiero analizar los pagos realizados en la plataforma utilizando los números de tarjeta como identificadores, además de generar reportes diarios con los datos de los pagos, para realizar un seguimiento de las transacciones

¿Qué se necesita realmente?

¿Qué tipos de datos se requieren conocer?

Como analista financiero, quiero **analizar los pagos** realizados en la plataforma utilizando identificadores tokenizados en lugar de números de tarjeta, además de **generar reportes diarios** que cumplan con PCI DSS y protejan los datos sensibles, **para realizar un seguimiento** seguro de las transacciones

- Tokenización
- Encripción
- Control de acceso
- Reportes mínimos datos
- Acciones auditables


Usuario
HTTPS
Realiza pago
WS Rest
Valida pago
Almacena datos de pago
Genera reporte

Valida pago (Process)

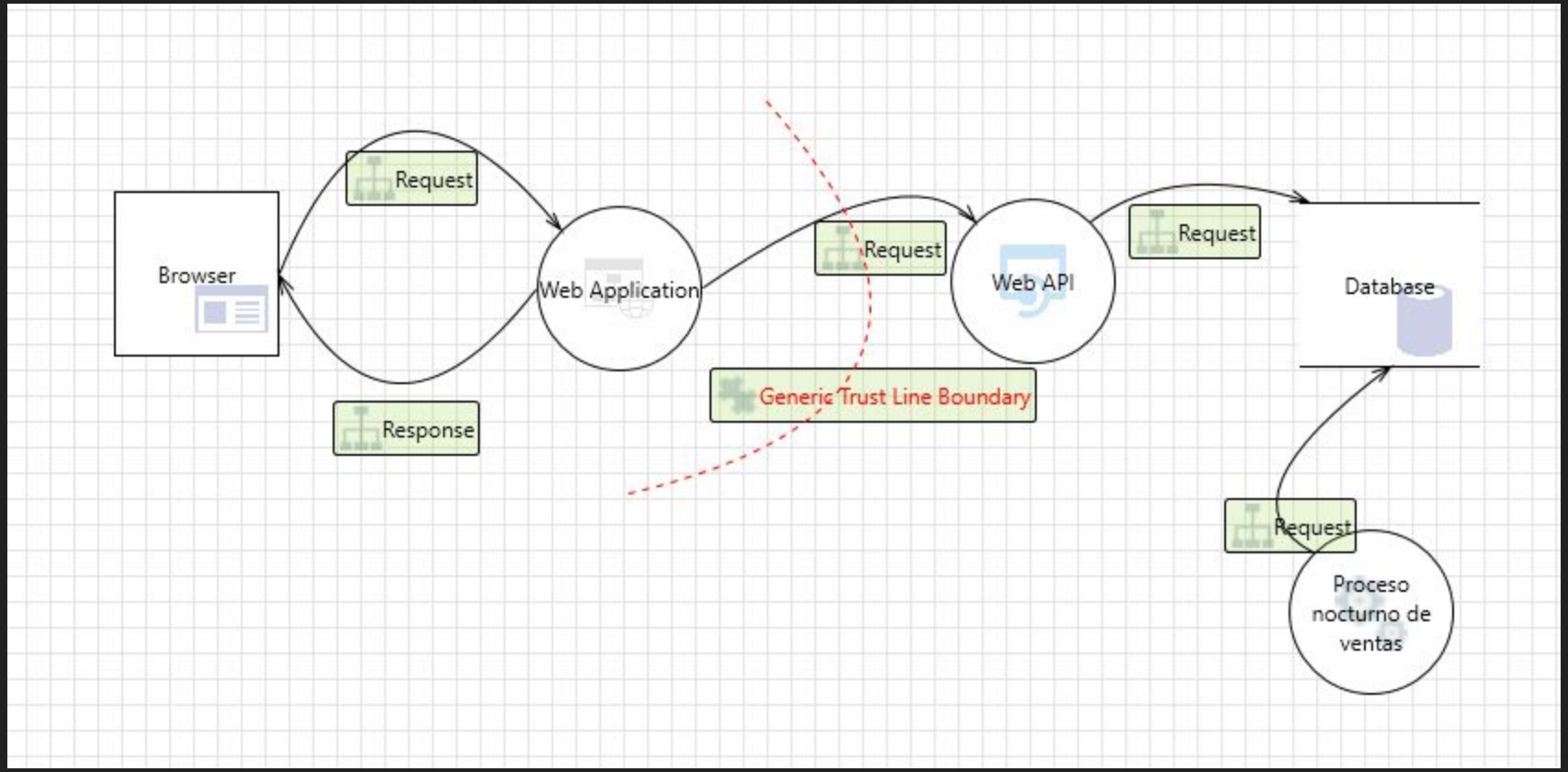
Reglas de negocio

Number	Title	Type	Priority	Status	Score	Description	Mitigations
12	Sin validación de datos de entrada en APIs	Spoofing	Medium	Open		Explotación de vulnerabilidades de tipo de inyección por falta de validación de datos de entrada	Validación de acuerdo al tipo de datos
14	Sin validación de confirmación de pago	Tampering	Medium	Open		Ausencia de validación de aprobación de pago exitoso por banco	No procesar la venta hasta que se confirme el pago por parte del banco

Genera reporte (Process)

Proceso todos los días 23:00

Number	Title	Type	Priority	Status	Score	Description	Mitigations
7	Exposición de datos sensibles	Information disclosure	Medium	Open		Reportes con datos sensibles podrían fugarze al no tener controles para su transporte y almacenamiento	Provide remediation for this threat or a reason if status is N/A


Browser
Request
Response
Web Application
Request
Web API
Request
Database
Generic Trust Line Boundary
Request
Proceso nocturno de ventas

Threat List

ID	Diagram	Changed By	Last Modified	State	Title	STRIDE Categ	Description	Justification	Interaction	Po
21	Diagram 1		Generated	Not Started	An adversary may gain unauthorized access to Web API due to poor access control checks	Elevation of Privi	An adversary may gain unauthorized access to Web API due to poor access control checks		Request	Im
22	Diagram 1		Generated	Not Started	An adversary can gain access to sensitive information from an API through error messages	Information Disc	An adversary can gain access to sensitive data such as the following, through verbose error messages		Request	En
23	**Diagram 1**		**Generated**	**Not Started**	**An adversary can gain access to sensitive data by sniffing traffic to Web API**	**Information Dis**	**An adversary can gain access to sensitive data by sniffing traffic to Web API**		**Request**	**Fo**
24	**Diagram 1**		**Generated**	**Not Started**	**An adversary can gain access to sensitive data stored in Web API's config files**	**Information Dis**	**An adversary can gain access to the config files. and if sensitive data is stored in it, it would be co**		**Request**	**En**
25	**Diagram 1**		**Generated**	**Not Started**	**Attacker can deny a malicious act on an API leading to repudiation issues**	**Repudiation**	**Attacker can deny a malicious act on an API leading to repudiation issues**		**Request**	**En**
26	**Diagram 1**		**Generated**	**Not Started**	**An adversary may spoof Web Application and gain access to Web API**	**Spoofing**	**If proper authentication is not in place, an adversary can spoof a source process or external entit**		**Request**	**En**
27	**Diagram 1**		**Generated**	**Not Started**	**An adversary may inject malicious inputs into an API and affect downstream processes**	**Tampering**	**An adversary may inject malicious inputs into an API and affect downstream processes**		**Request**	**En**

Export Csv | Clear Filters | 15 Threats Displayed, 43 Total

Threat Properties

ID: 26 Diagram: Diagram 1 Status: Not Started Last Modified: Ger

Title: An adversary may spoof Web Application and gain access to Web API

STRIDE Category: Spoofing

Description: If proper authentication is not in place, an adversary can spoof a source process or external entity and gain unauthorized access to the Web Application

Justification:

Interaction: Request

Possible Mitigation(s): Ensure that standard authentication techniques are used to secure Web APIs. Refer: https://aka.ms/tmtauthn#authn-secure-api

Severity: High

Threat Properties | Notes - 1 entry

Confidencialidad

Necesidad de conocer

Como gerente de equipo, quiero acceder a la información de los miembros de mi equipo (números de seguro social, historial salarial y evaluaciones de desempeño) para facilitar la gestión de recursos humanos

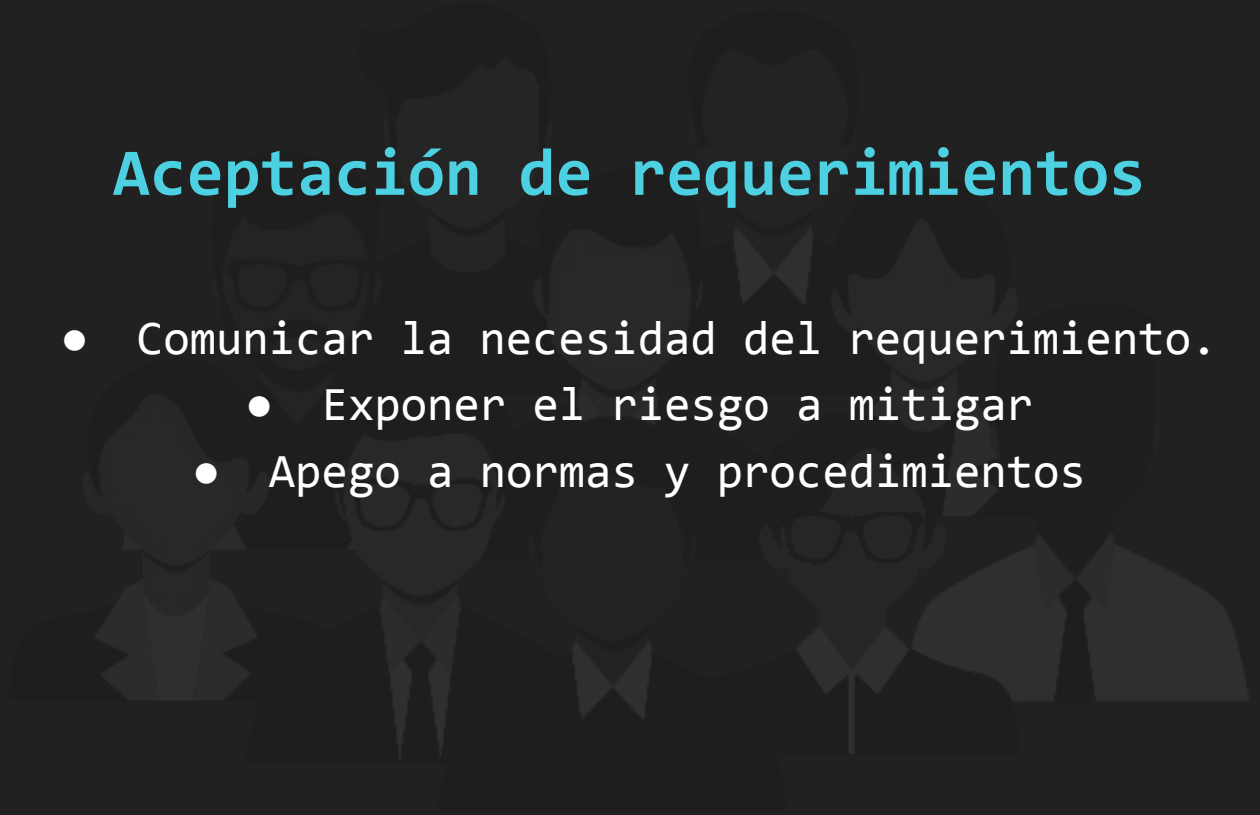
¿Para que se necesita la información?

¿De qué manera se va a proteger la información?

Como gerente de equipo, quiero acceder de manera segura a la información relevante de los miembros de mi equipo (por ejemplo, historial salarial y evaluaciones de desempeño) cumpliendo con los principios de minimización de datos, control de acceso y protección de información sensible, para facilitar la gestión de recursos humanos

- Control de acceso
- Encripción de datos
- Datos mínimos
- Bitácoras de acceso
- Manejo de sesiones

Aceptación de requerimientos



- Comunicar la necesidad del requerimiento.
- Exponer el riesgo a mitigar
- Apego a normas y procedimientos

Concluyendo...

When designing a new application, ***creating a secure architecture prevents vulnerabilities*** before they even become part of the application. This prevents costly repairs and repudiation problems

C4: Address Security from the Start - OWASP Top 10 Proactive Controls

1. **Define cuanto antes, define procesos.** Sesiones de requerimientos.
2. **Entiende el problema a fondo.** involucrate en el proceso.
3. **Pregunta todo lo necesario.** No lo sabemos todo.
4. **Prepárate para cambios inesperados.**. Cualquier cosa puede pasar.
5. **Evaluar el riesgo.** Seguridad vs Negocio
6. **Be Nice.** Please!

Consejos finales

Referencias

1. https://www.microsoft.com/en-us/securityengineering/sdl/practices
2. https://owasp.org/www-project-application-security-verification-standard/
3. https://cheatsheetseries.owasp.org/
4. https://owasp.org/www-project-mobile-app-security/
5. https://top10proactive.owasp.org/archive/2024/
6. https://learn.microsoft.com/en-us/training/modules/tm-introduction-to-threat-modeling/

Gracias

POSADEY

Arturo Sustaita
@arturo_io
arturo-sustaita