PROYECTO DE SOFTWARE

TEMARIO

- Problemas de seguridad: SQLi

PROBLEMAS DE SEGURIDAD

¿QUÉ ES SQLI?

INYECCIÓN SQL

- Una SQL Injection (**SQLi**) suele ocurrir cuando se arma en forma descuidada una consulta a la base de datos a partir de los **datos ingresados por el usuario**.
- Dentro de estos parámetros pueden venir el código malicioso.
- El atacante logra que los parámetros que ingresa se transformen en comandos SQL en lugar de usarse como datos para la consulta que es lo que originalmente pensó el desarrollador.
- Top 10 de Open Web Application Security Project (**OWASP**) => **https://owasp.org/www-project-top-ten/**

INYECCIÓN SQL

Obtener acceso a una aplicación:

- Suponiendo que la consulta de autenticación de una página que pide email y password es:

```
SELECT * FROM users AS u WHERE
u.email = '"+ email +"' AND u.password = '"+ password +"'
```

- Suponiendo **email='admin'** y **password='admin'** el sql quedaría:

```
SELECT * FROM users AS u WHERE
u.email = 'admin' AND u.password = '"admin'
```

INYECCIÓN SQL

¿Qué sucede si usamos **email** == **pass** => **1' or '1'='1** ?

```
SELECT * FROM users AS u WHERE
u.email = '"+ "1' or '1'='1" +"' AND u.password = '"+ "1' or
'1'='1" +"'
```

Lo que se se resuelve en:

```
SELECT * FROM users AS u WHERE
u.email = '1' or '1'='1' AND u.password = '1' or '1'='1'
```

(Cualquier cosa) OR TRUE es siempre **TRUE**

- Veamos como funciona... **http://localhost:5000/iniciar_sesion_sqli**

INYECCIÓN SQL

Para obtener acceso a una aplicación web, dependiendo del motor de base de datos, otras estructuras que se pueden usar son:

- ' or 1=1--
- " or 1=1--
- or 1=1--
- ' or 'a'='a
- " or "a"="a
- ') or ('a'='a

PARAMETRIZACIÓN: EVITANDO SQLI

- Python soporta múltiples maneras de **parametrizar** las consultas SQL para evitar formar consultas erróneas.

qmark: Símbolo de pregunta.

```
cursor.execute("SELECT first_name FROM users WHERE email = ?",
(email))
```

numeric: Numérico o posicional.

```
cursor.execute("SELECT first_name FROM users WHERE email = :1",
(email))
```

named: Nombrado.

```
cursor.execute("SELECT first_name FROM users WHERE email =
:mail", {'mail': email})
```

PARAMETRIZACIÓN: EVITANDO SQLI

- Python Enhancement Proposals: **https://www.python.org/dev/peps/pep-0249/#paramstyle**

format: Formato ANSI C printf.

```
cursor.execute("SELECT first_name FROM users WHERE email = %s",
(email))
```

pyformat: Formato de Python extendido.

```
cursor.execute("SELECT first_name FROM users WHERE email = %
(mail)s",{'mail': email})
```

SEGUIMOS LA PRÓXIMA ...

Speaker notes