4TO. CONGRESO INTERNACIONAL

SEGURIDAD INFORMÁTICA

28 - 30 de Noviembre - La Paz, Bolivia

SEGURIDAD DE APLICACIONES WEB

#LA_CULPA_ES_DEL_PROGRAMADOR

#CIDSI 2022

ACERCA DE...

Saúl Mamani M.

Ingeniero Informático

Facultad Nacional de Ingeniería

Oruro – Bolivia

@kanito777

mail: luas0_1@yahoo.es

cel: +591 76137269

https://saulmamani.github.io

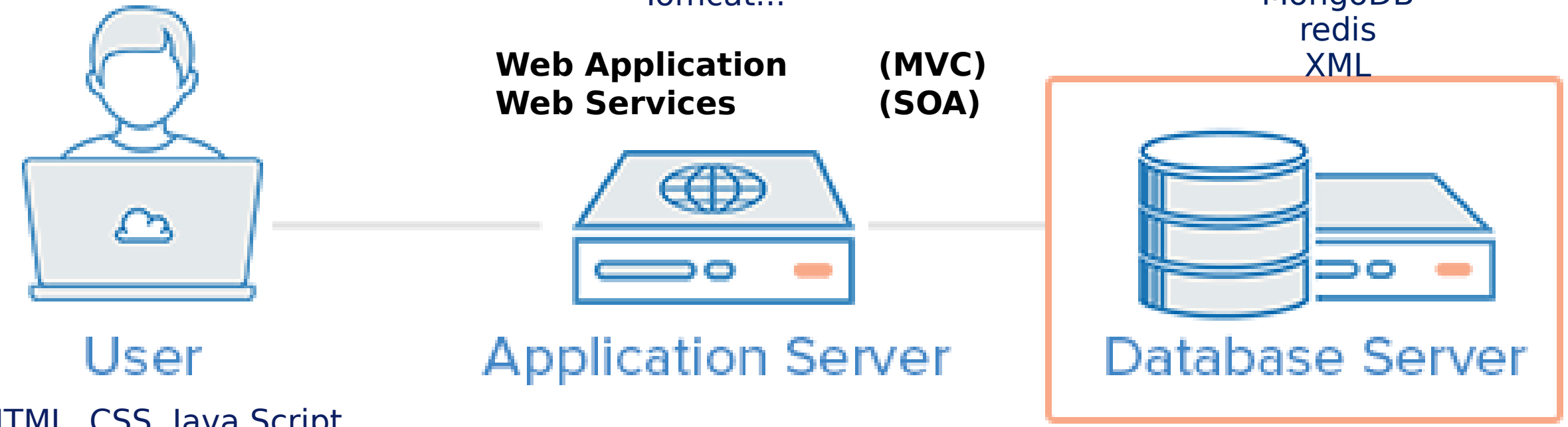
Software Engineer

ARQUITECTURA DE UNA APP WEB

Apache
ngnix
IIS
Tomcat...

MySQL
PostgreSQ
L
SQL Server
Oracle
MongoDB
redis
XML

Web Application (MVC)
Web Services (SOA)



HTML, CSS, Java Script

Vuejs
Angular
React

C#, PHP, Java, Go, Nodejs

ARQUITECTURA DE UNA APP WEB

La mayoría de los problemas de seguridad en los sistemas web se encuentran a nivel de aplicación y son el resultado de escritura

MANIFIESTO AGIL

Individuos e interacciones, sobre procesos y herramientas

Software funcionando, sobre documentación exhaustiva

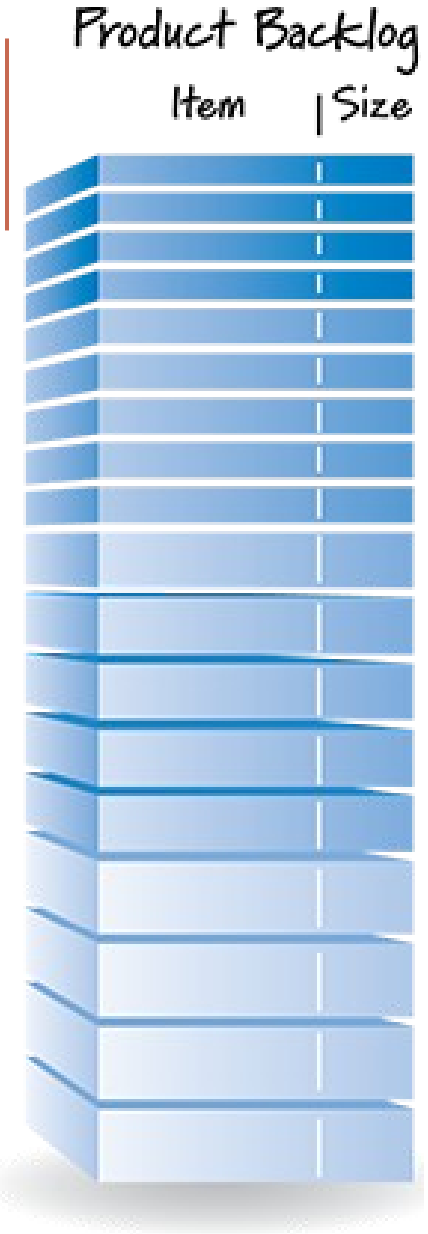
Colaboración con el cliente, sobre negociación contractual

Respuesta al cambio, sobre seguir un plan

DESARROLLAR **APLICACIONES WEB SEGURAS**

NO ES **UNA TAREA FÁCIL**

¿ POR QUÉ ?



LOS PROGRAMADORES...

NO SOLO SE DEBEN ENFOCAR EN CUMPLIR CON LOS REQUISITOS, SINO TAMBIÉN, DEBEN PRESTAR ATENCIÓN A LOS RIESGOS DE SEGURIDAD

"Aceptar el riesgo"

PERO...

¿ QUE HACEMOS LOS PROGRAMADORES CUANDO NOS PIDEN UNA APLICACIÓN ?

O, ¿Cuándo tenemos que resolver un problema?

Programar, y haber que sale!

La probabilidad que un mono escriba una poesía, NO ES CERO!

DESARROLLAR

SOFTWARE DE CALIDAD

SOFTWARE DE CALIDAD

CONCORDANCIA EXPLÍCITA

REQUISITOS FUNCIONALES

ESTÁNDARES DOCUMENTADOS

CARACTERÍSTICAS IMPLÍCITAS

ROBUSTEZ

FLEXIBILIDAD

SEGURIDAD

CONFIABILIDAD

NO SE TRATA DE PROGRAMAR ARTESANALMENTE...

TENEMOS QUE APLICAR **INGENIERÍA** A NUESTRO PROYECTOS

Metodología

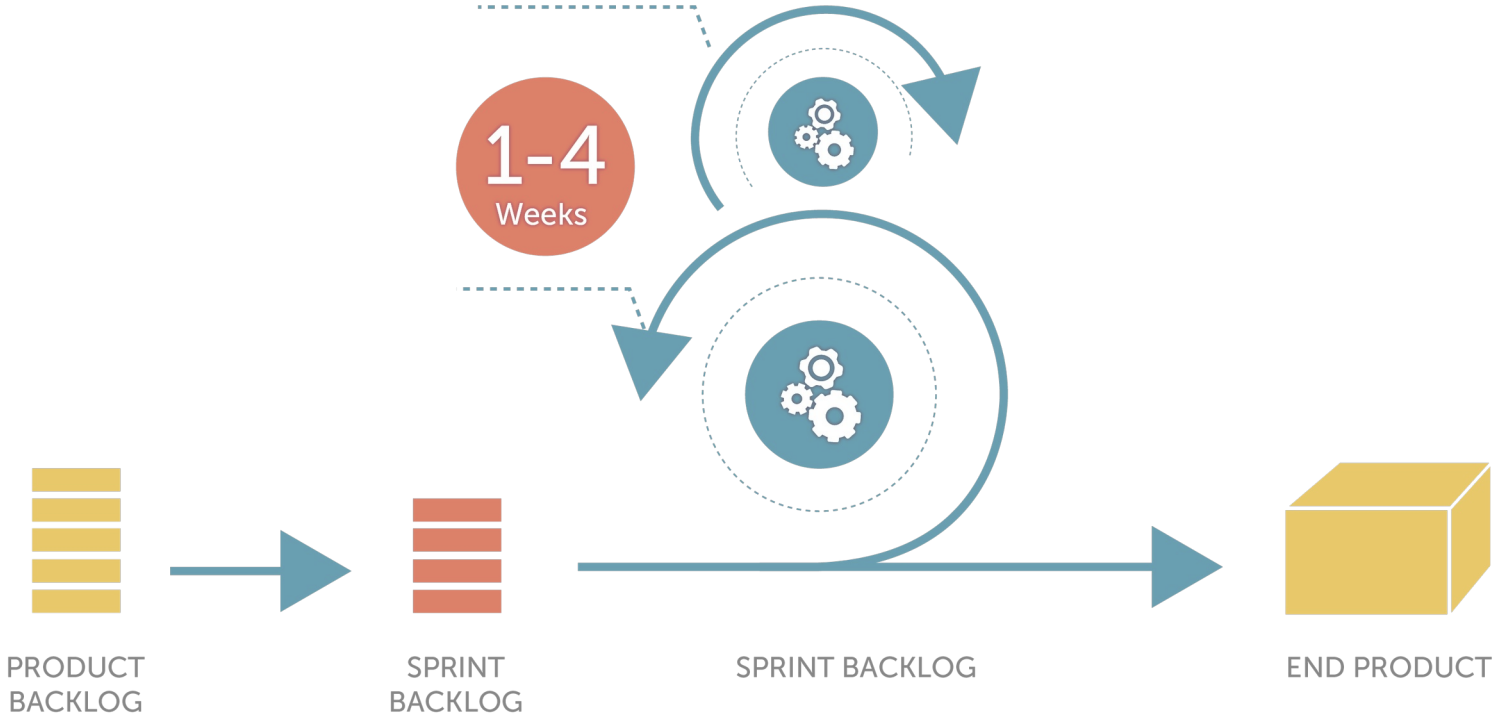
Ingeniería de Software

Modelado

Herramientas

LA SEGURIDAD DEBE ESTAR PRESENTE EN CADA FASE DEL DESARROLLO DEL SOFTWARE Y...

NO DEJARLO AL FINAL

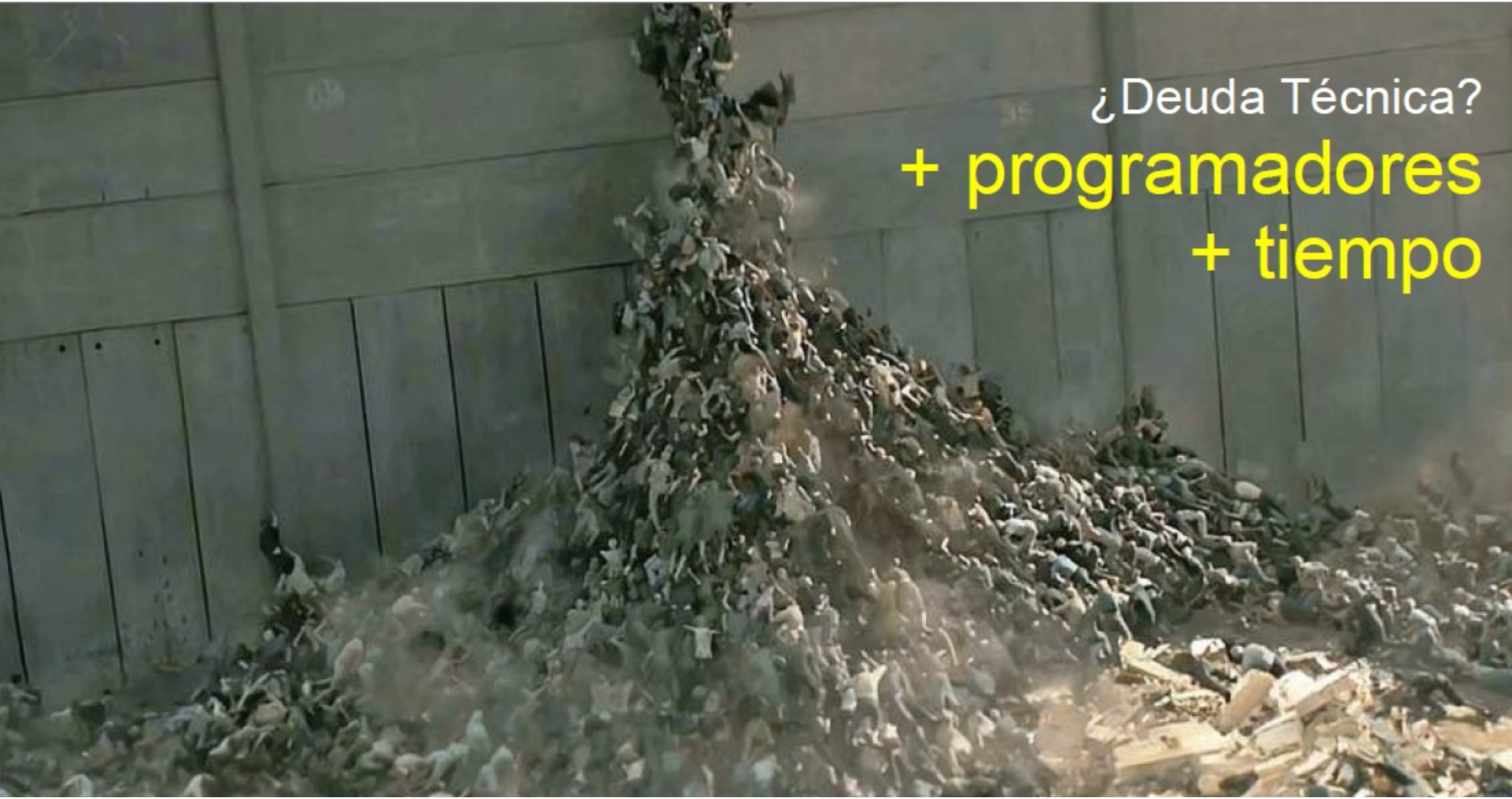


OBVIAR TODO ESTO,

REPERCUTE NEGATIVAMENTE EN LA

CALIDAD DEL PROYECTO

“Deuda Técnica”

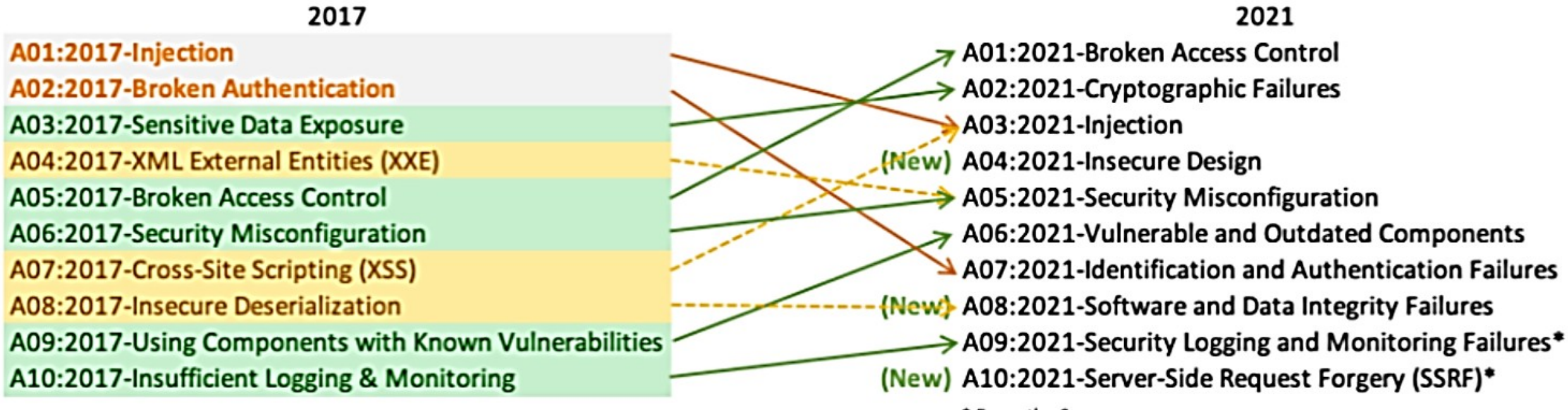

¿Deuda Técnica?
+ programadores
+ tiempo

La seguridad debería ser una tarea común

PERO...

LOS PROGRAMADORES SEGUIMOS COMETIENDO LOS MISMOS ERRORES

OWASP TOP 10


2017
A01:2017-Injection
A02:2017-Broken Authentication
A03:2017-Sensitive Data Exposure
A04:2017-XML External Entities (XXE)
A05:2017-Broken Access Control
A06:2017-Security Misconfiguration
A07:2017-Cross-Site Scripting (XSS)
A08:2017-Insecure Deserialization
A09:2017-Using Components with Known Vulnerabilities
A10:2017-Insufficient Logging & Monitoring
2021
A01:2021-Broken Access Control
A02:2021-Cryptographic Failures
A03:2021-Injection
(New) A04:2021-Insecure Design
A05:2021-Security Misconfiguration
A06:2021-Vulnerable and Outdated Components
A07:2021-Identification and Authentication Failures
(New) A08:2021-Software and Data Integrity Failures
A09:2021-Security Logging and Monitoring Failures*
(New) A10:2021-Server-Side Request Forgery (SSRF)*

VEAMOS ALGUNOS

ERRORES

QUE COMETEMOS LOS PROGRAMADORES

Muña v0.9

Es una aplicación web vulnerable para practicar seguridad y hacking ético de aplicaciones y servicios web.

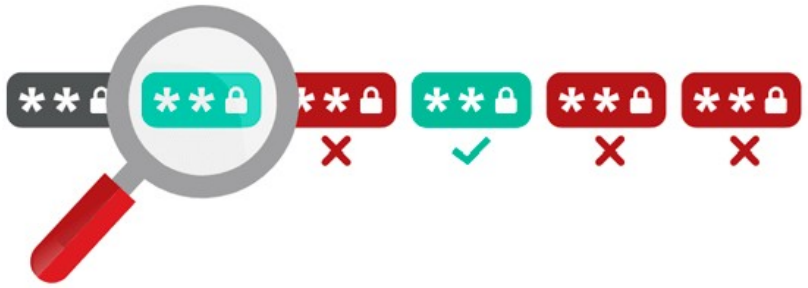
https://

Muña v0.9

Tulpar sql-xss-full
Web Vulnerability Scanner

MUÑA TIENE TANTOS **ERRORES** DE **SEGURIDAD** QUE SI LO **ESCANEAMOS** CON **TULPAR,** NOS DEVUELVE TODO TIPO DE RESULTADOS

Y SI LO ATACAMOS CON **SQLMAP,** NOS PODEMOS **ROBAR** TODA LA **BASE DE DATOS**



#1

ERRORES DE

VALIDACIÓN

Inyección
SQLi
XSS

ATAQUE: INYECCIÓN SQL

```
select    *
from  usuario
where
Cuenta = 'nose' and Clave = ' '  or
'
```

F y F ó V

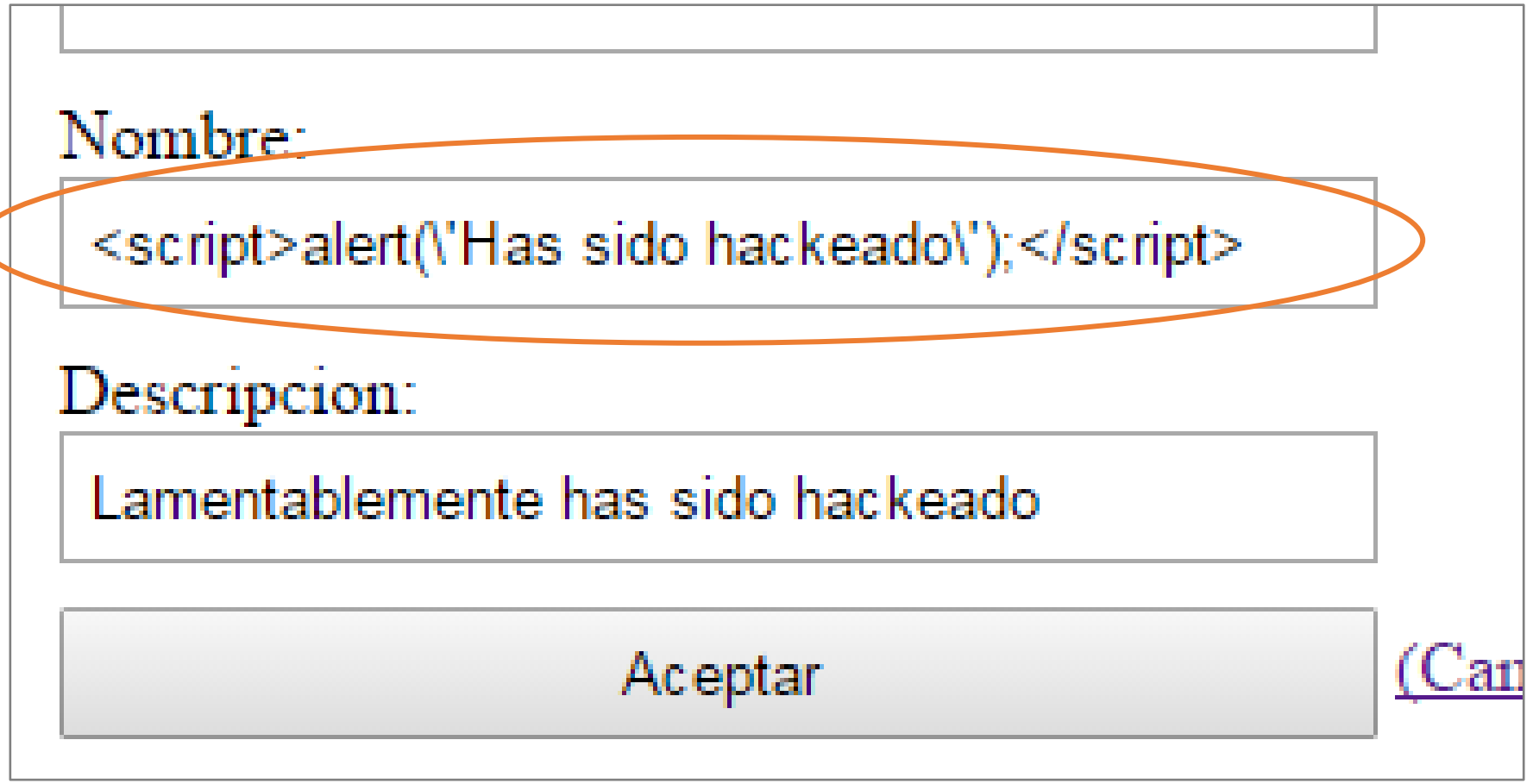
=V

```
select *
from  clientes
where id = -1'  or  true
```

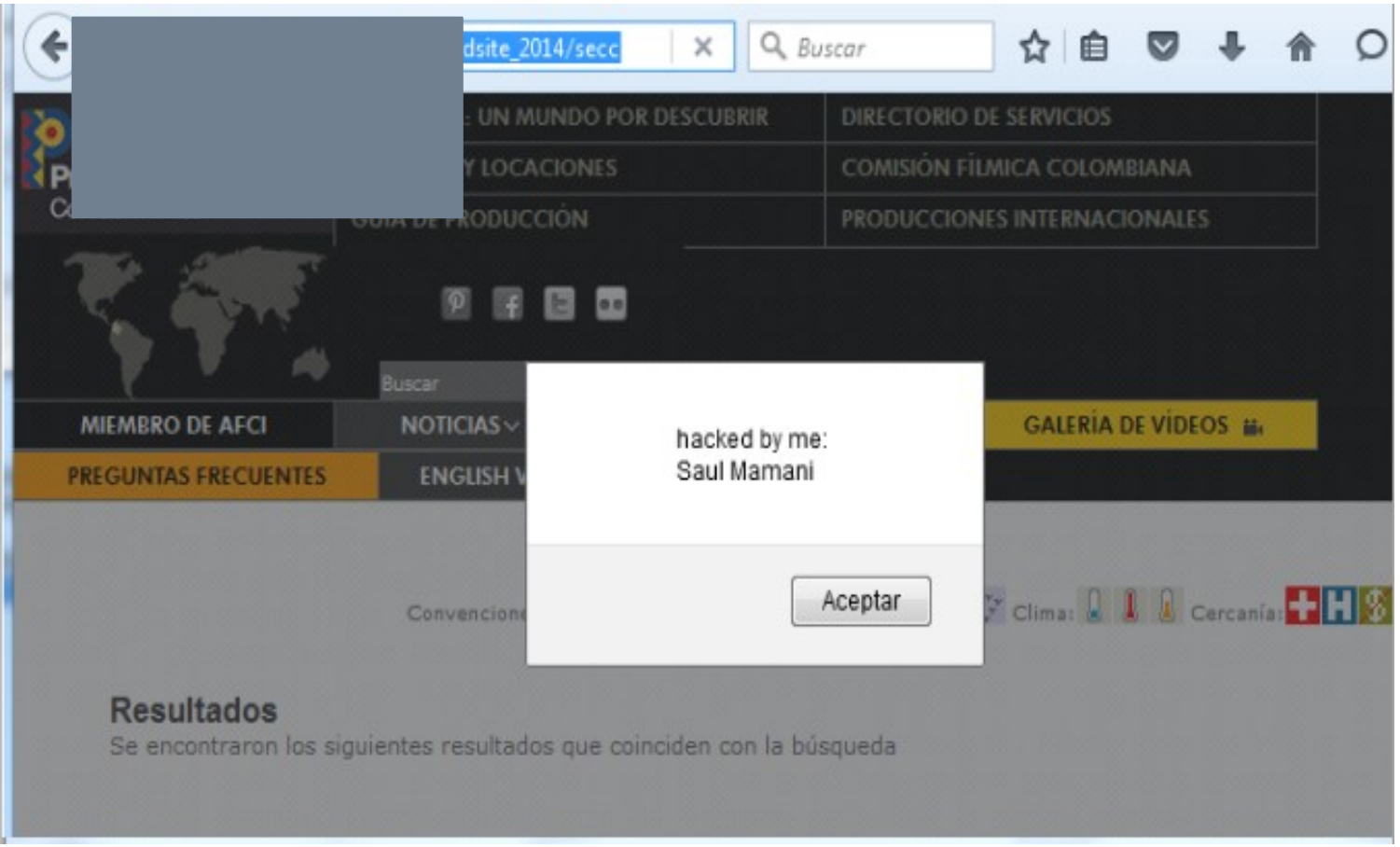
ATAQUE: INYECCIÓN SQL

ATAQUE: INYECCIÓN XSS

SON RELATIVAMENTE SENCILLOS DE REALIZAR



ATAQUE: INYECCIÓN



XSS Reflejados

XSS Almacenados

SOLUCIÓN: INYECCIÓN

VALIDAR Y FILTRAR DATOS

```php
function sanatizar($dato)
{
    // limpiando espacion en blanco
    $dato = trim($dato);
    // aplicando stripslashes si magic_quotes
    if(get_magic_quotes_gpc())
    {
        $dato = stripslashes($dato);
    }
    // un mySQL connection es requerido antes
    $dato = mysql_real_escape_string($dato);
    return $dato;
}
```

- EN EL CLIENTE Y EN EL SERVIDOR
- CONSULTAS SQL PARAMETRIZADAS
- LISTAS BLANCAS Y LISTAS NEGRAS

SOLUCIÓN: INYECCIÓN

ESPACAR LOS DATOS DE SALIDA

```
<td><?php echo utf8_encode($f['Cuenta']); ?></td>
<td><?php echo htmlentities(utf8_encode($f['Nombre'])); ?></td>
<td><?php echo htmlentities(utf8_encode($f['Descripcion'])); ?></td>
<td><?php echo utf8_encode($f['FechaRegistro']); ?></td>
<td><a href="perfil.php?id=<?php echo $f['Id']; ?>">ver</a></td>
```

SOLUCIÓN: INYECCIÓN

ELIJA SIEMPRE DESARROLLAR CON UN FRAMEWORK

#2

AUSENCIA DE

EXCEPCIONES

Footpriting

Broken Access Control

ATAQUE: EXCEPCIONES

LOS MENSAJES DE ERROR, SIEMPRE MUESTRA INFORMACIÓN VULNERABLE

ASP.NET

```
System.ArgumentException: No se puede convertir saul en System.Double.
Nombre del par&#225;metro: type ---> System.FormatException: La cadena d
   en System.Number.StringToNumber(String str, NumberStyles options, Num
   en System.Number.ParseDouble(String value, NumberStyles options, Numb
   en System.Double.Parse(String s, NumberStyles style, NumberFormatInfo
```

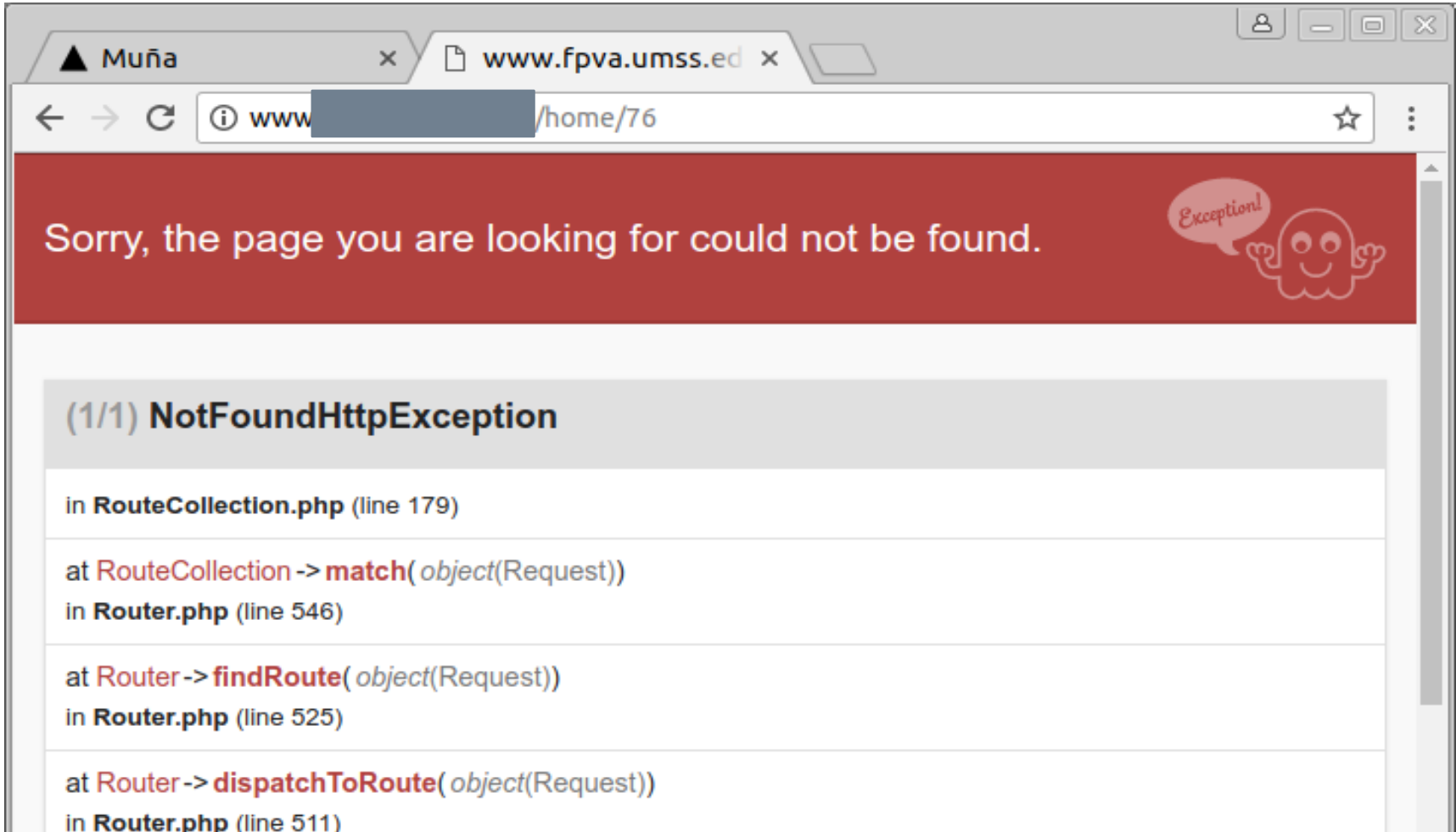
PHP

Warning: mysql_fetch_array() expects parameter 1 to be resource, boolean given in **C:\xampp\htdocs\malditaweb\perfil.php** on line **20**

ATAQUE: EXCEPCIONES

LARAVEL

LOS FRAMEWORK NO SE SALVAN...



SOLUCIÓN: EXCEPCIONES

MANEJO DE EXCEPCIONES

```
public String RaizCuadrada(string a)
{
    try
    {
        return Math.Sqrt(Convert.ToDouble(a)).ToString();
    }
    catch (Exception)
    {
        return "Ha ocurrido un Error";
    }
}
```

SOLUCIÓN: EXCEPCIONES

DESHABILITAR LOS MENSAJES DE ERROR

```
<customErrors mode="On" defaultRedirect="error.html" />
</system.web>
```

Web.config
ASP.Net

```
error_reporting=~E_ALL & ~E_DEPRECATED & ~E_STRICT
```

php.ini

```
APP_KEY=
APP_DEBUG=false
APP_LOG_LEVEL=debug
```

LARAVEL
.env

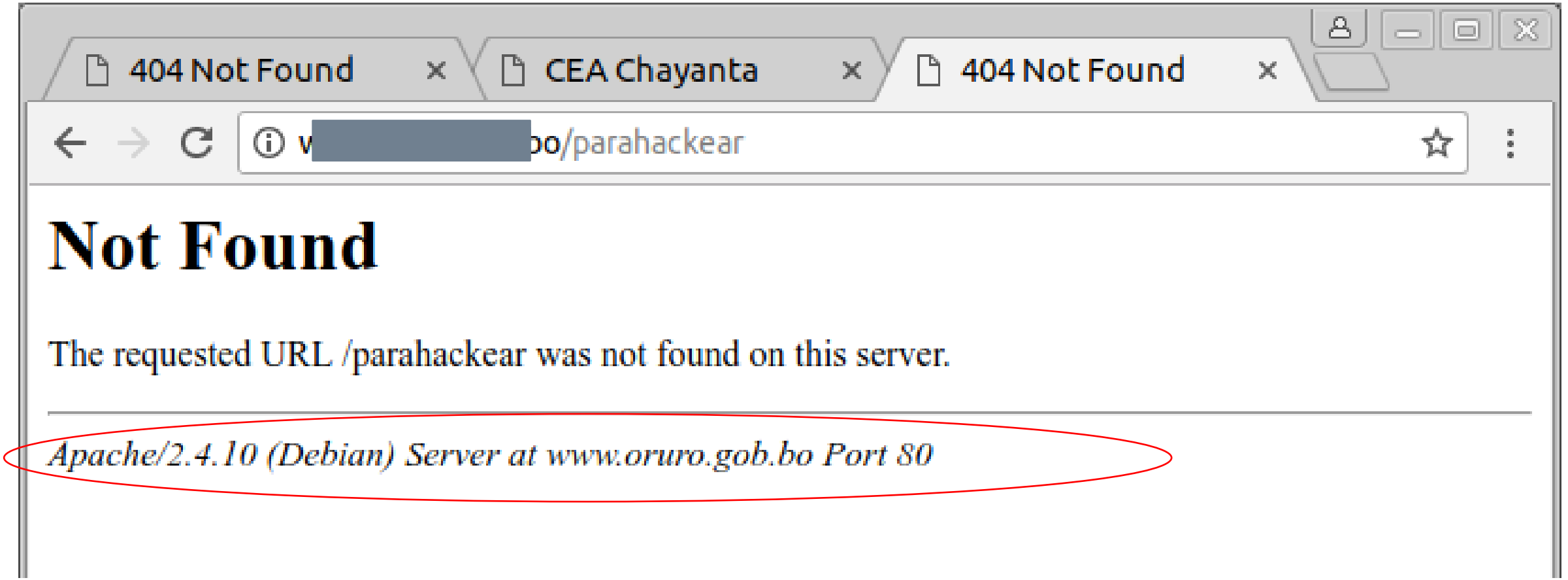
#3

MALA CONFIGURACIÓN

footpriting
exposición de recursos

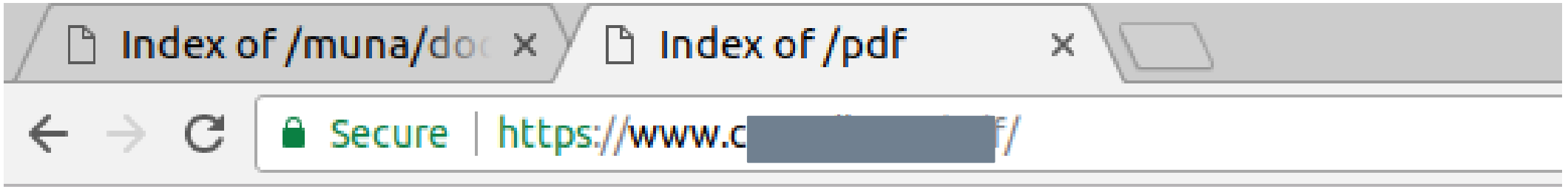
ATAQUE: CONFIGURACIÓN

LA MALA CONFIGURACIÓN, SIEMPRE MUESTRA INFORMACIÓN VULNERABLE



ATAQUE: CONFIGURACIÓN

CARPETAS QUE NO DEBERÍAN SER ACCESIBLES POR CUALQUIER TIPO DE USUARIO

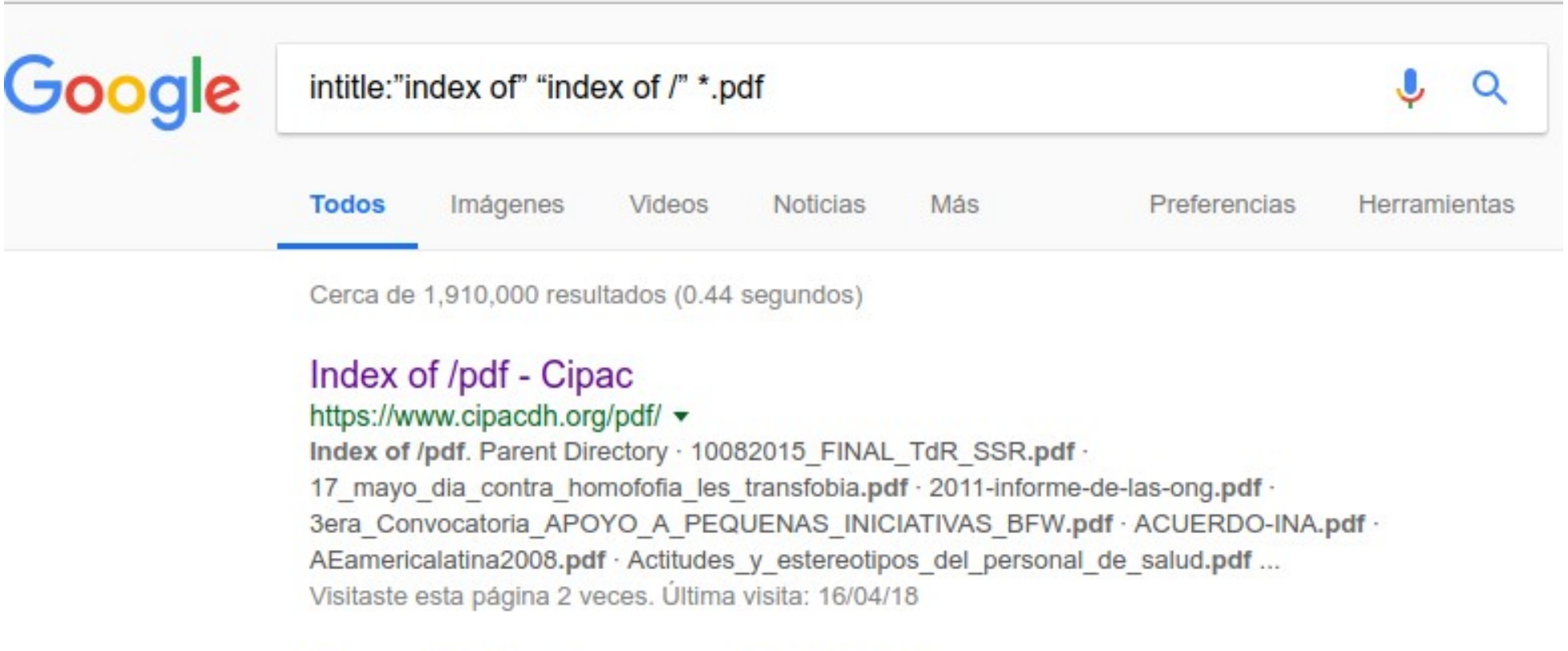


Index of /pdf

- Parent Directory
- 10082015_FINAL_TdR_SSR.pdf
- 17_mayo_dia_contra_homofofia_les_transfobia.pdf
- 2011-informe-de-las-ong.pdf

ATAQUE: CONFIGURACIÓN

ES RELATIVAMENTE FÁCIL BUSCAR ESTE TIPO DE ERRORES EN INTERNET



SOLUCIÓN: CONFIGURACIÓN

CONFIGURACIÓN DEL SERVIDOR APACHE

```
ServerSignature    Off
ServerTokens       Prod
```

```
#  nano /etc/apache2/apache2.conf
O en el: httpd.conf
```

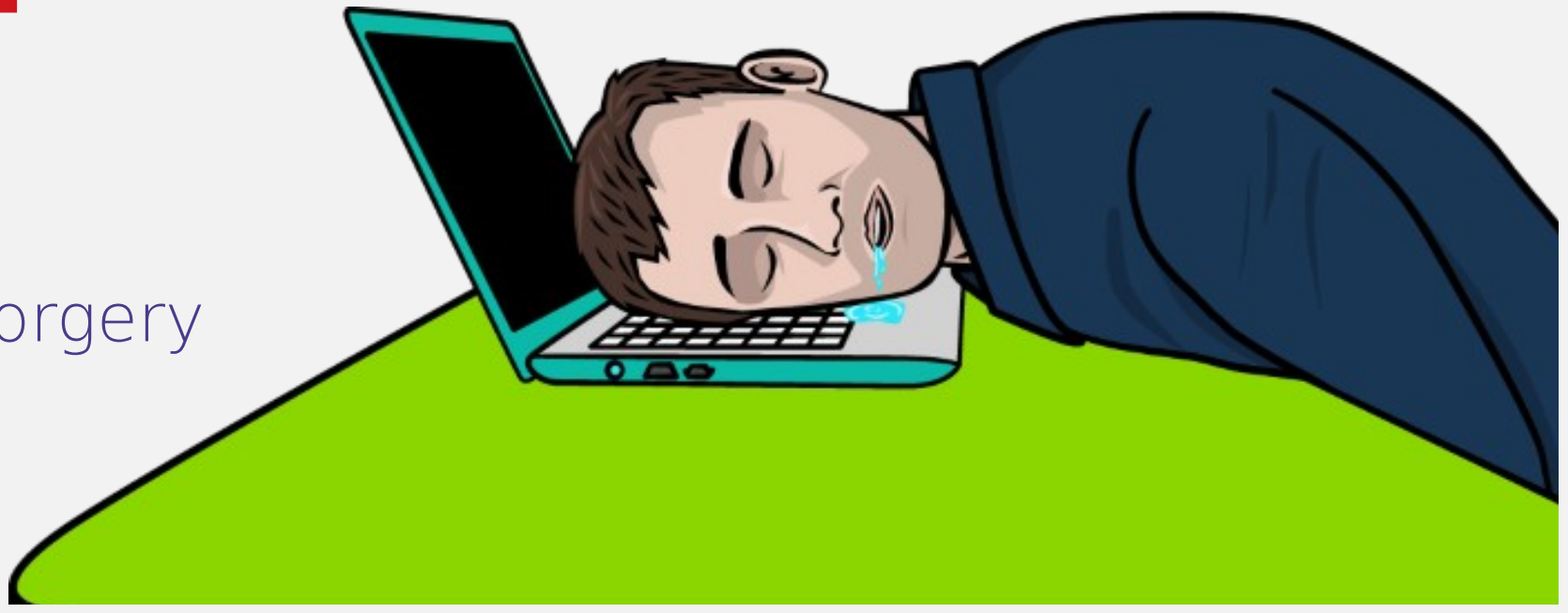
EN UN ARCHIVO **.htaccess**

#4

CONFIAR EN EL USUARIO

SPAM

DoS
Cross-Site Request Forgery



ATAQUE: SPAM

NO DEBEMOS CONFIAR EN LA BUENA FE DE LOS USUARIOS DEL SISTEMA

ACEPTAR EL RIESGO

ATAQUE: SPAM

ROBOT PARA INGRESO MASIVO DE DATOS

spampost_dos.py

```
1   import requests
2
3   for i in xrange(1, 11):
4       txtCuenta = 'hacker'+str(i)
5       txtClave = 'hack' + str(i)
6       txtNombre = 'protege'
7       txtDesc = 'tu aplicacion'
8
9       parametros = {"txtCuenta":txtCuenta,
10                    "txtClave":txtClave,
11                    "txtNombre":txtNombre,
12                    "txtDesc":txtDesc}
13
14      r = requests.post("http://localhost/muna/nuevoController.php",
15                        data=parametros)
16
17      print(r.text)
```

XSFR

SOLUCIÓN: SPAM

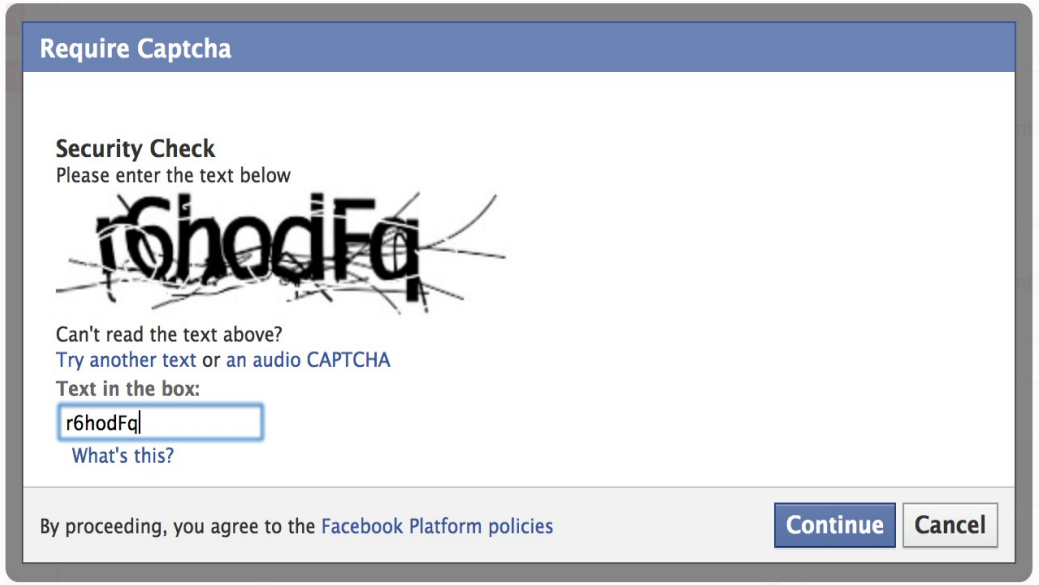
TOKEN PARA LOS REQUEST, LA AUTENTICACIÓN, Y EL INTERCAMBIO DE INFORMACIÓN

.env

```
APP_NAME=ctrlfood
APP_ENV=production
APP_KEY=base64:wWm2I60ftlp0a3ykDeWq+sChZib09hSDfXGBRUxsvMU=
APP_DEBUG=false
APP_URL=http://ctrlfood.com
```

SOLUCIÓN: SPAM

UTILICE **CAPTCHA** EN LOS FORMULARIOS IMPORTANTES



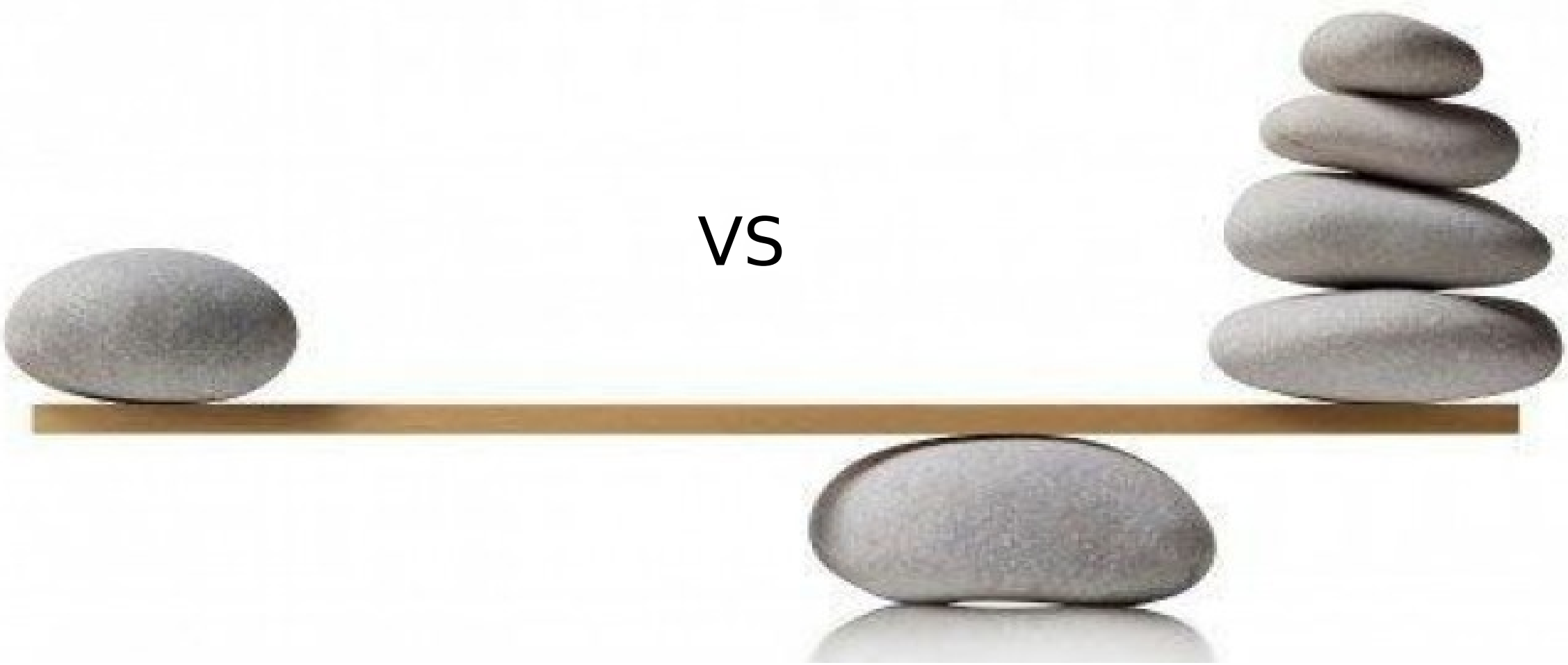
AUTENTICACIÓN EN DOS PASOS O DE **DOBLE FACTOR**

LOS SISTEMAS DEBEN SER SEGUROS

PERO, NO DEBEN AFECTAR LA USABILIDAD

DE LOS USUARIOS

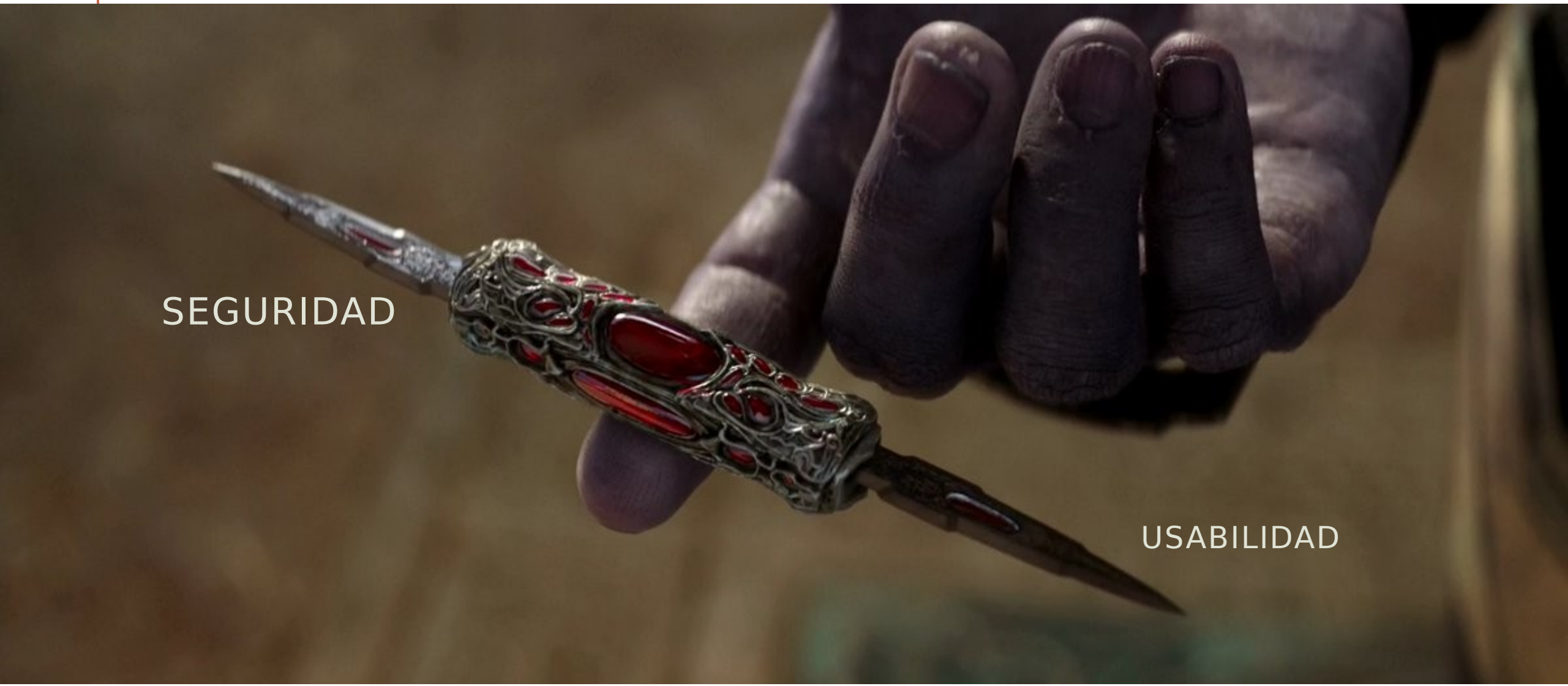
USABILIDAD
SEGURIDAD



Imaginen: ¿Cómo sería el sistema mas seguro del mundo?

Play

PERFECTAMENTE **EQUILIBRADO**, COMO TODO DEBE ESTAR



LOS **MÉTODOS,** PROCESOS Y LAS **HERRAMIENTAS** PUEDEN TENER **FALLAS**.

PERO DEFINITIVAMENE Y AL FINAL DE CUENTAS...

4TO. CONGRESO INTERNACIONAL

SEGURIDAD INFORMÁTICA

28 - 30 de Noviembre - La Paz, Bolivia

GRACIAS

#LA_CULPA_ES_DE_LAS_PERSONAS

#CIDSI 2022