jtsec

BEYOND IT SECURITY

SEGURIDAD OFENSIVA EN LA WEB
Una demostración práctica

UNIVERSIDAD DE GRANADA

¿Quién soy yo?

- **Javier Tallón:** Consultor experto del estándar de seguridad **Common Criteria** y de otros muchos estándares de seguridad dentro de las tecnologías de la información (FIPS 140-2, ITSEC, **ISO 27K1**, **SOC 2**, **ENS**,...). Javier ha trabajado como **evaluador** de seguridad en los mayores laboratorios nacionales y como **consultor** ha acompañado a diversas compañías en sus procesos de certificación de la ciberseguridad. Ha sido **ponente** en diversas conferencias en materia de seguridad informática y certificación y es profesor del máster de Ciberseguridad de la **UGR**.

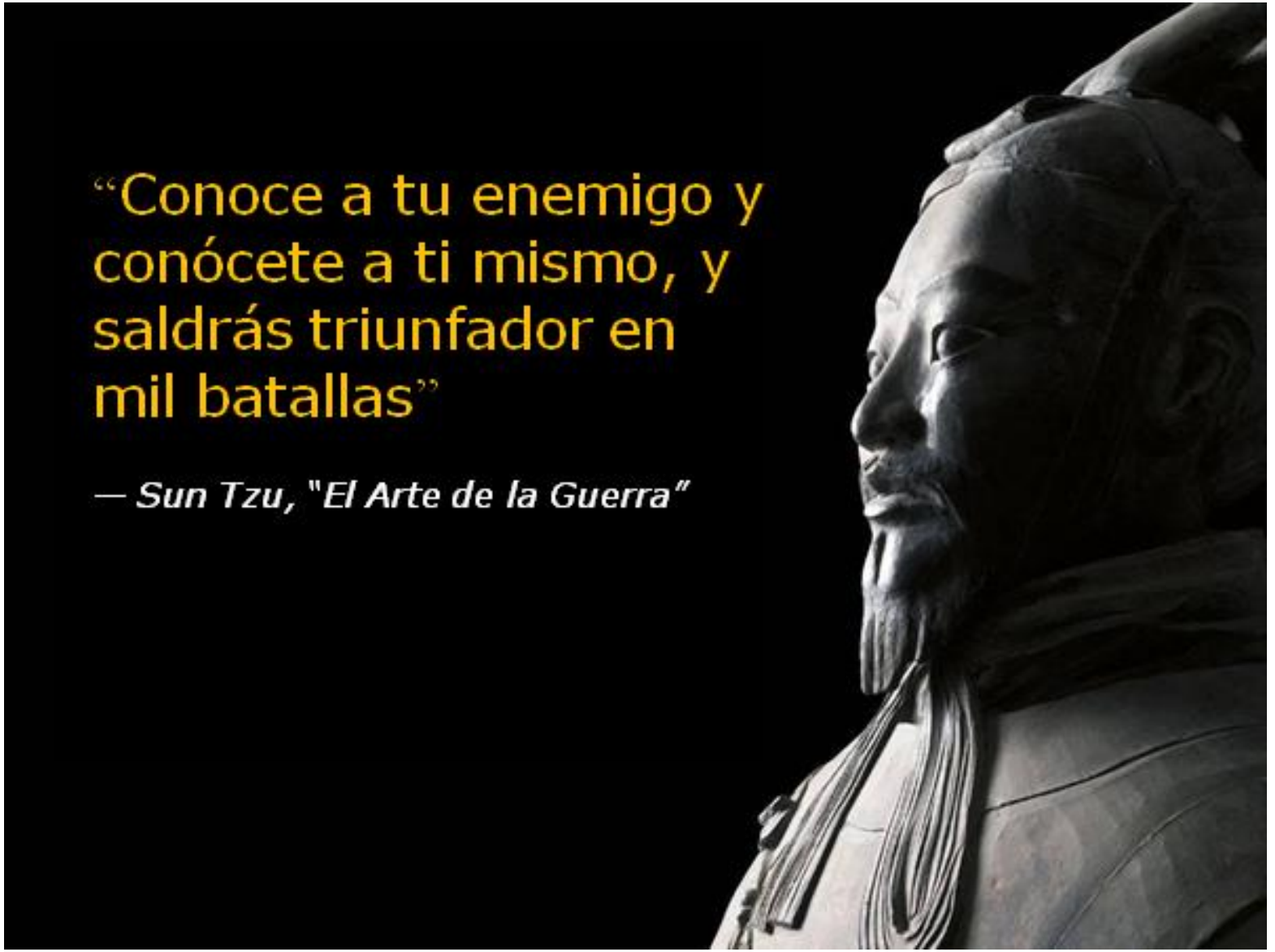
- En 2015 comienza a sentar las bases de lo que será jtsec. Actualmente ejerce las labores de **Jefe de Operaciones** (COO) de la oficina de **Granada** desde donde la empresa desarrolla la mayor parte del trabajo. Reconocido experto en diversas disciplinas de ciberseguridad (reversing, exploiting, web, …), asume la **dirección técnica** de la mayoría de los proyectos, dirigiendo y organizando el trabajo del equipo. Además lidera el área de **Investigación y Desarrollo**, fomentando la participación del equipo de jtsec en múltiples Congresos.

Ciberseguridad

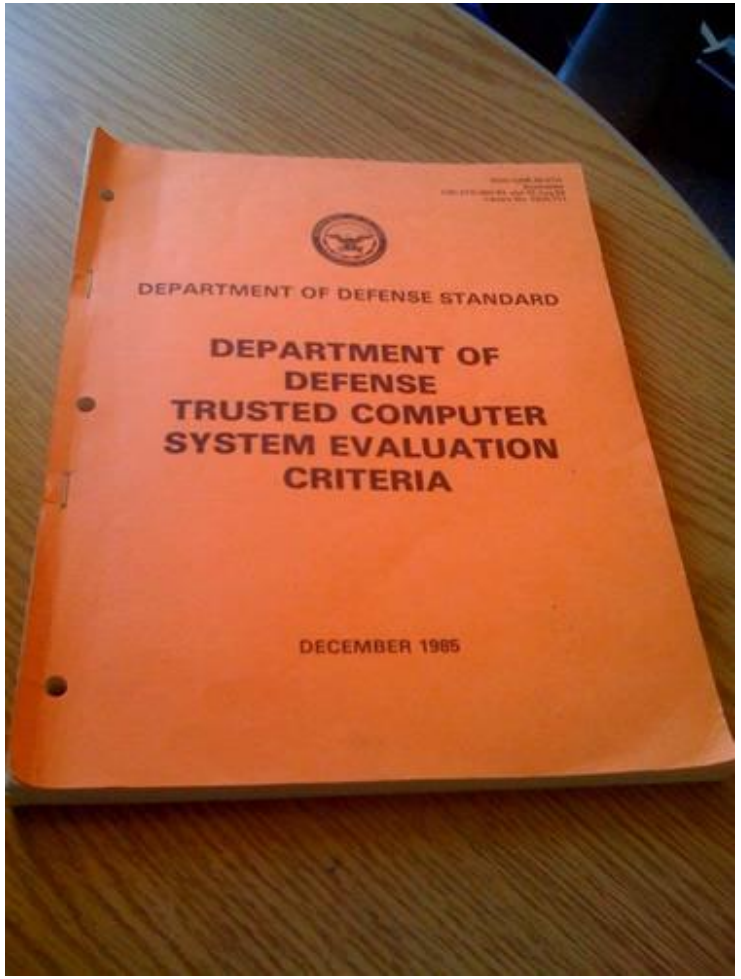
VS

- Proteger los activos de una organización
- Demostrar que no están suficientemente protegidos o las medidas implantadas no son eficaces

- **Seguri** **dversarial**
  approa ndividuals
  from at
- Hacer peores
  intenci
- Análisi
- Testing


"Conoce a tu enemigo y conócete a ti mismo, y saldrás triunfador en mil batallas"
— Sun Tzu, "El Arte de la Guerra"


Community Chest
GET OUT
OF JAIL, FREE
THIS CARD MAY BE KEPT UNTIL NEEDED OR SOLD
JAIL



Common Criteria

Common Methodology for Information Technology Security Evaluation

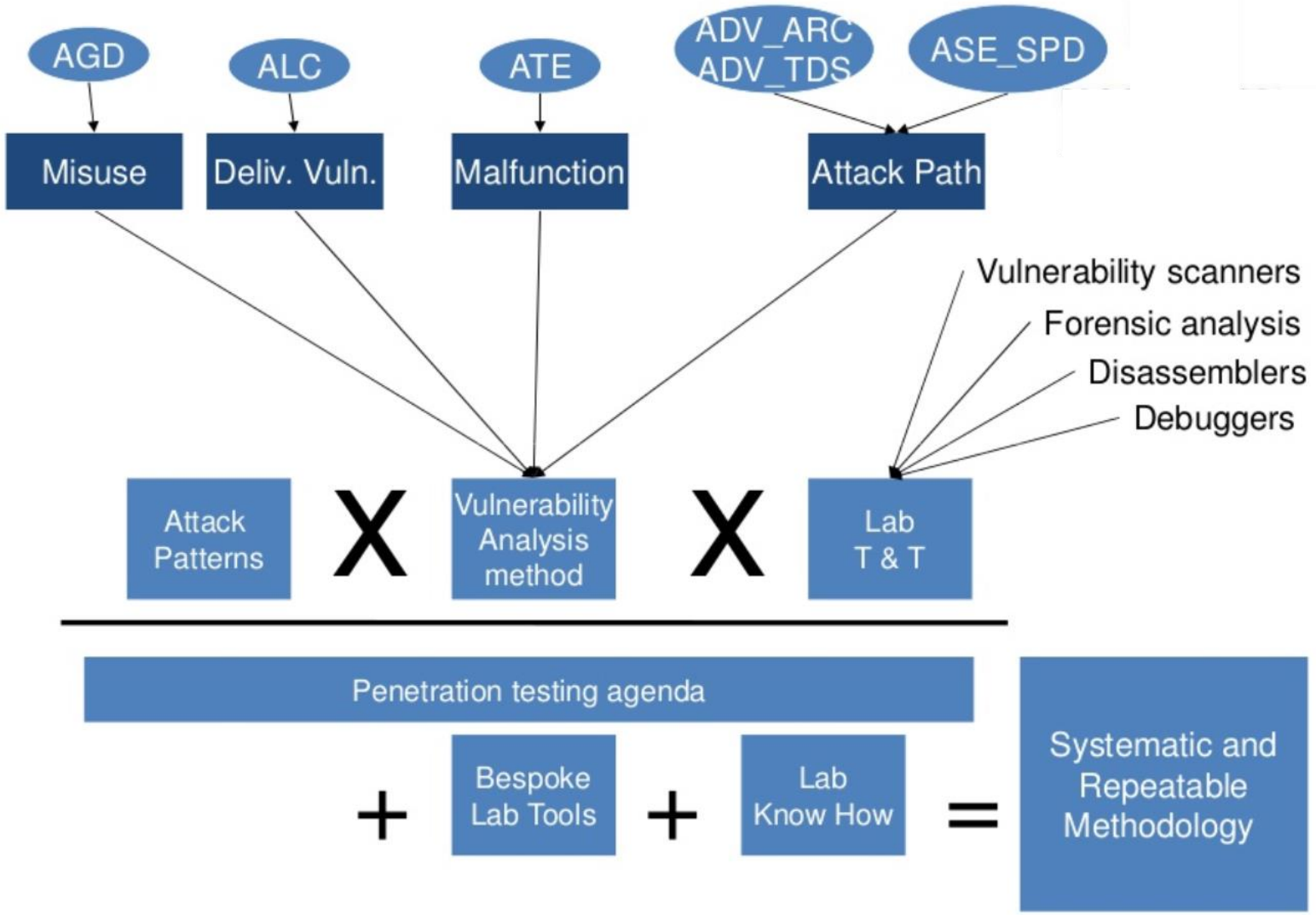
Evaluation methodology

September 2012

Version 3.1
Revision 4

CCMB-2012-09-004

centro criptológico nacional



Common Criteria

- Metodología de análisis de vulnerabilidades

OWASP

Open Web Application Security Project

MITRE

TOP 10

OWASP Top 10 - 2017
A1:2017-Injection
A2:2017-Broken Authentication
A3:2017-Sensitive Data Exposure
A4:2017-XML External Entities (XXE)
A5:2017-Broken Access Control
A6:2017-Security Misconfiguration
A7:2017-Cross-Site Scripting (XSS)
A8:2017-Insecure Deserialization
A9:2017-Using Components with Known Vulnerabilities
A10:2017-Insufficient Logging & Monitoring

Damn Vulnerable Web Application

AL TURRÓN!!!

A1. Inyección

- **¿Qué es?:** Los sitios web y las aplicaciones ocasionalmente necesitan **ejecutar** comandos en la base de datos o el sistema operativo subyacente para agregar o eliminar datos, ejecutar un script o iniciar otras aplicaciones. Si se añaden **entradas no verificadas** a una cadena de comandos o a un comando de base de datos, los atacantes pueden lanzar comandos a voluntad para tomar el control de un servidor, dispositivo o datos.

- **¿Cómo funciona?:** Si un sitio web, aplicación o dispositivo incorpora la entrada del usuario dentro de un comando, un atacante puede insertar un **comando de "carga útil"** directamente en dicha entrada. Si esa entrada no es verificada, un atacante entonces "inyecta" y ejecuta sus propios comandos

- **¿Y eso es malo?:** Una vez que los atacantes pueden realizar comandos, pueden controlar su sitio web, aplicaciones y datos.

A1. Inyección

- SQL Injection
- Command Injection
- LDAP Injection
- XML Injection
- Buffer Overflow

A1. Inyección

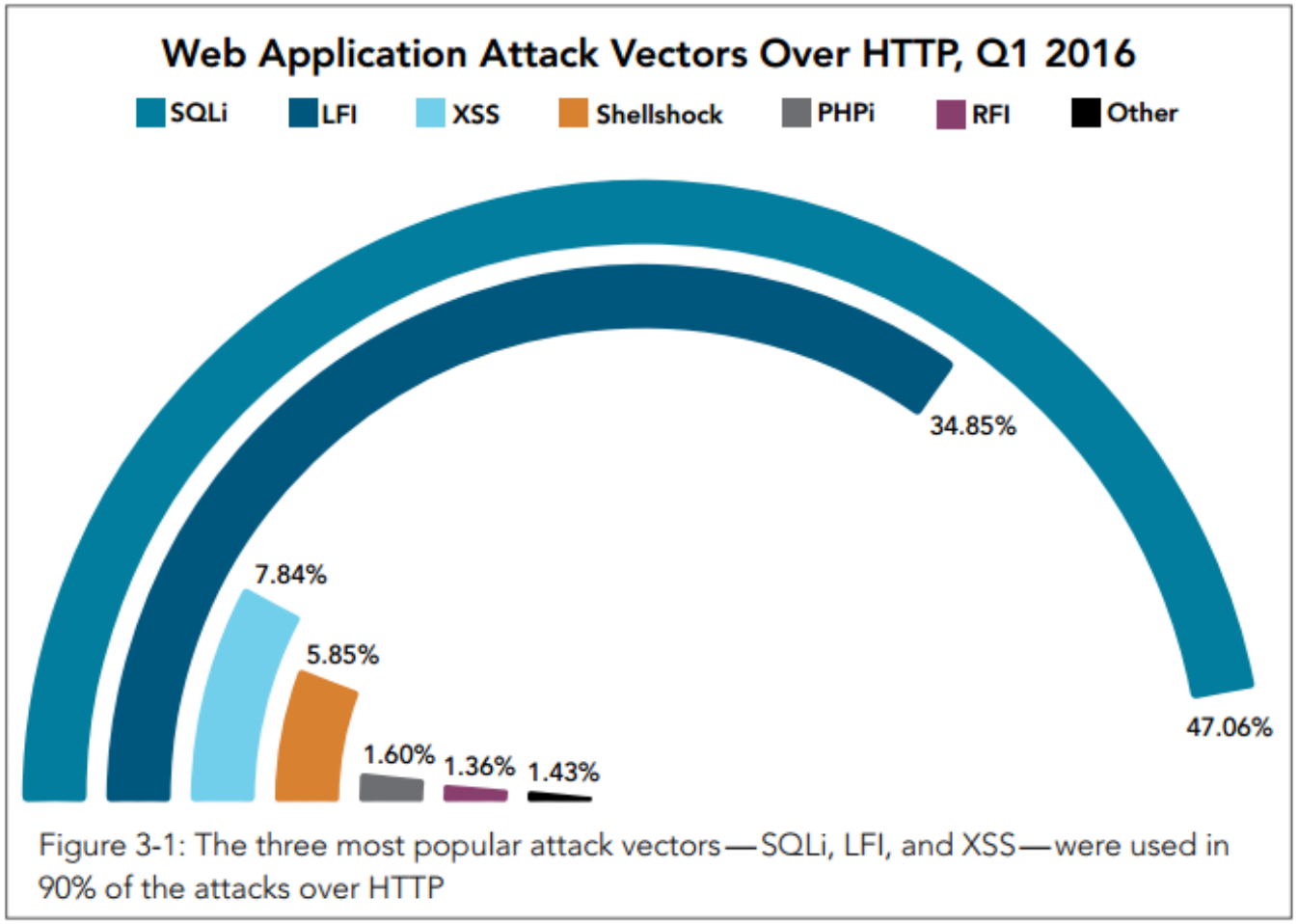
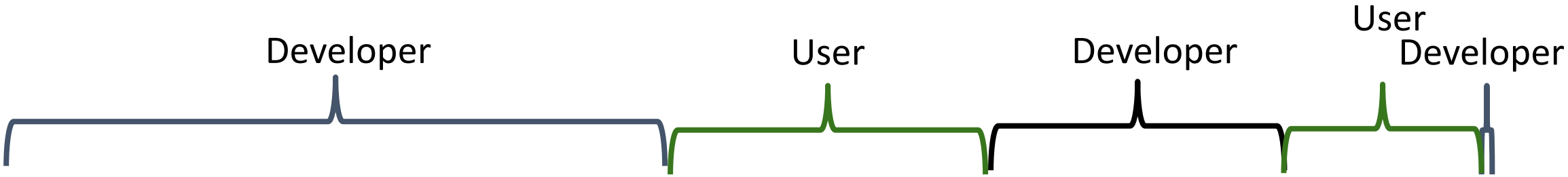


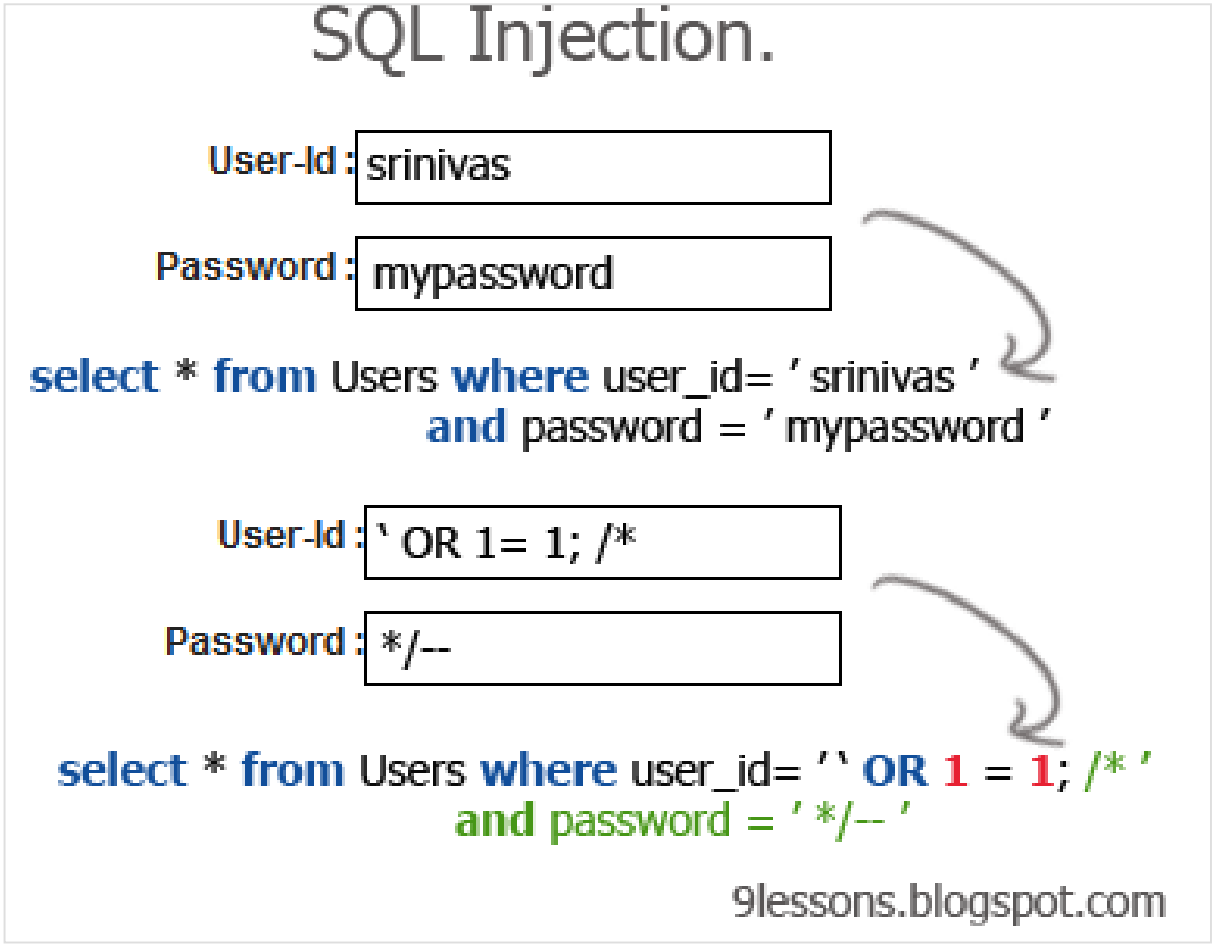
Figure 3-1: The three most popular attack vectors—SQLi, LFI, and XSS—were used in 90% of the attacks over HTTP

A1. Inyección



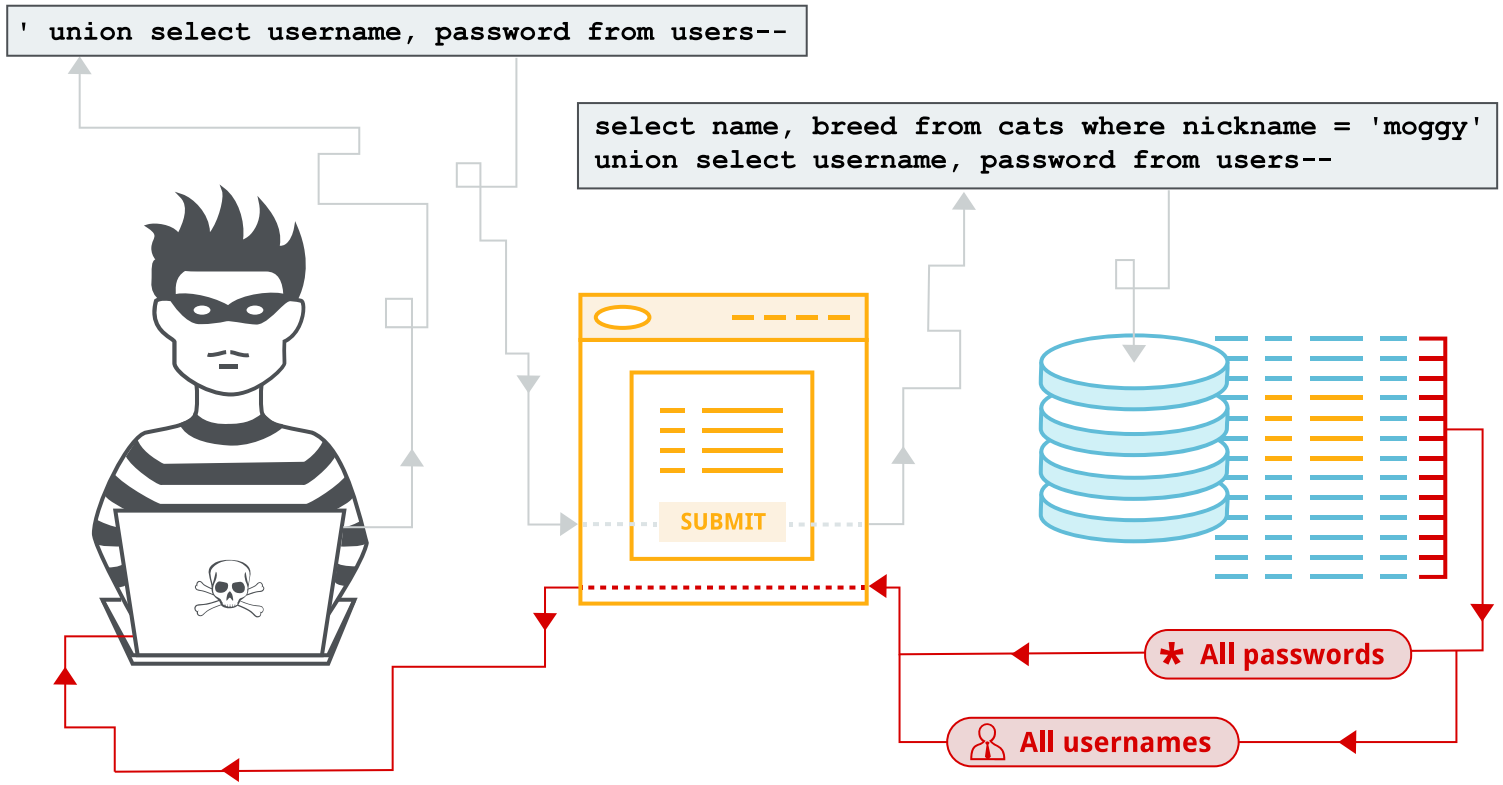
SELECT * FROM users WHERE user_id = 'mail@example.com' and password = 'mypassword'

A1. Inyección



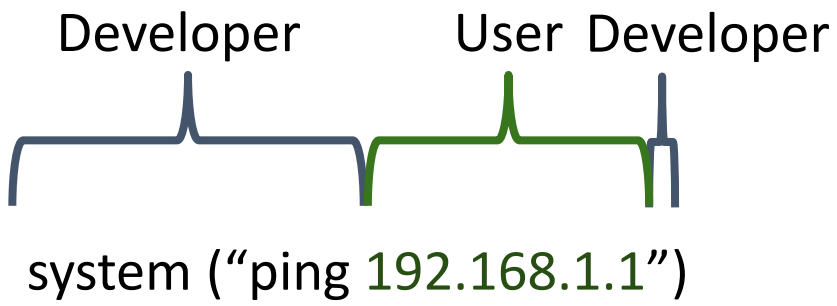
- admin'--
- admin'#
- ' or 1=1 --
- ' or 1=1 #
- ' or 1=1 /*
-) ' or '1' = '1
-) ' or ('1' = '1

A1. Inyección

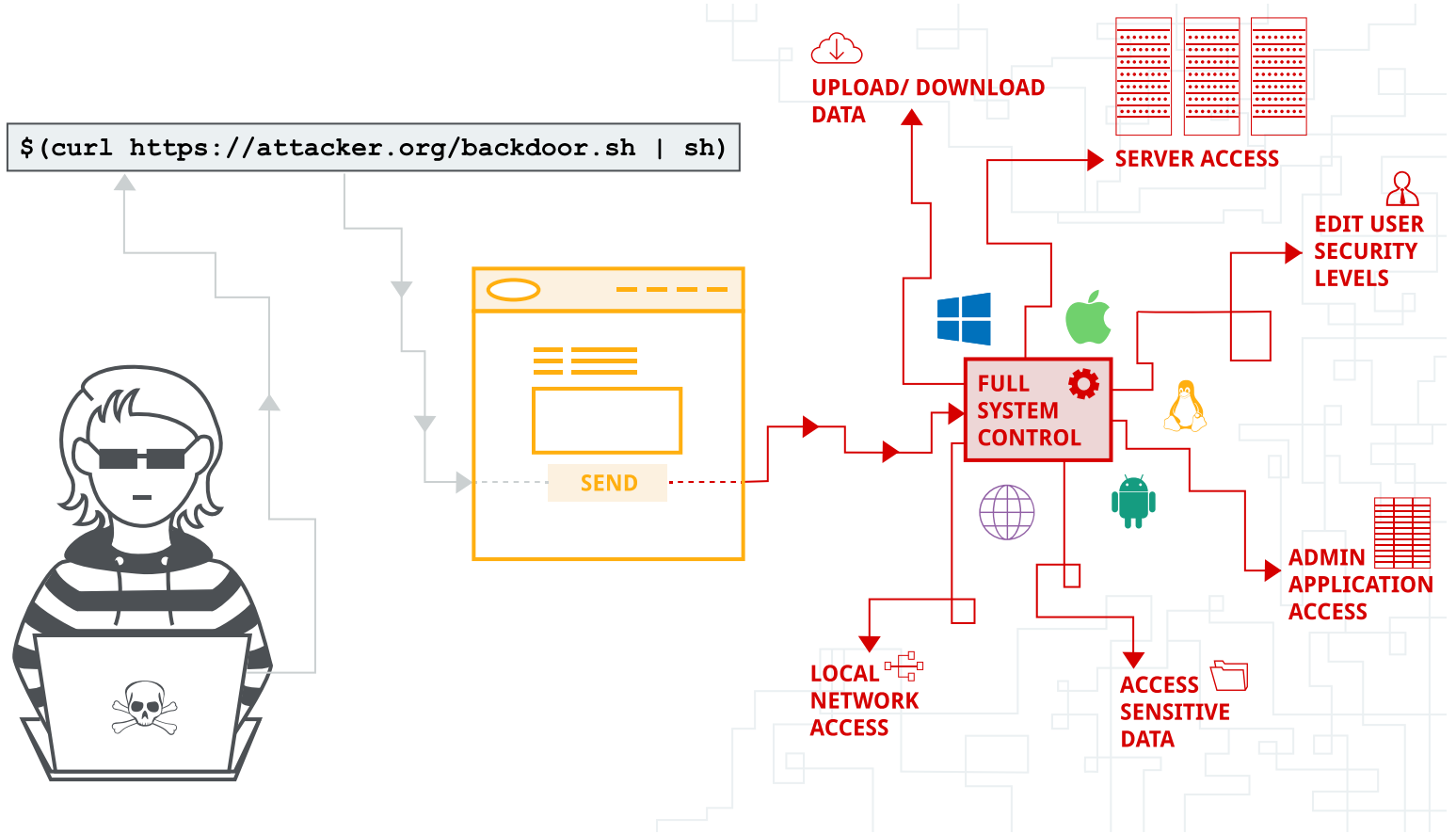



LIVE DEMO
DVWA

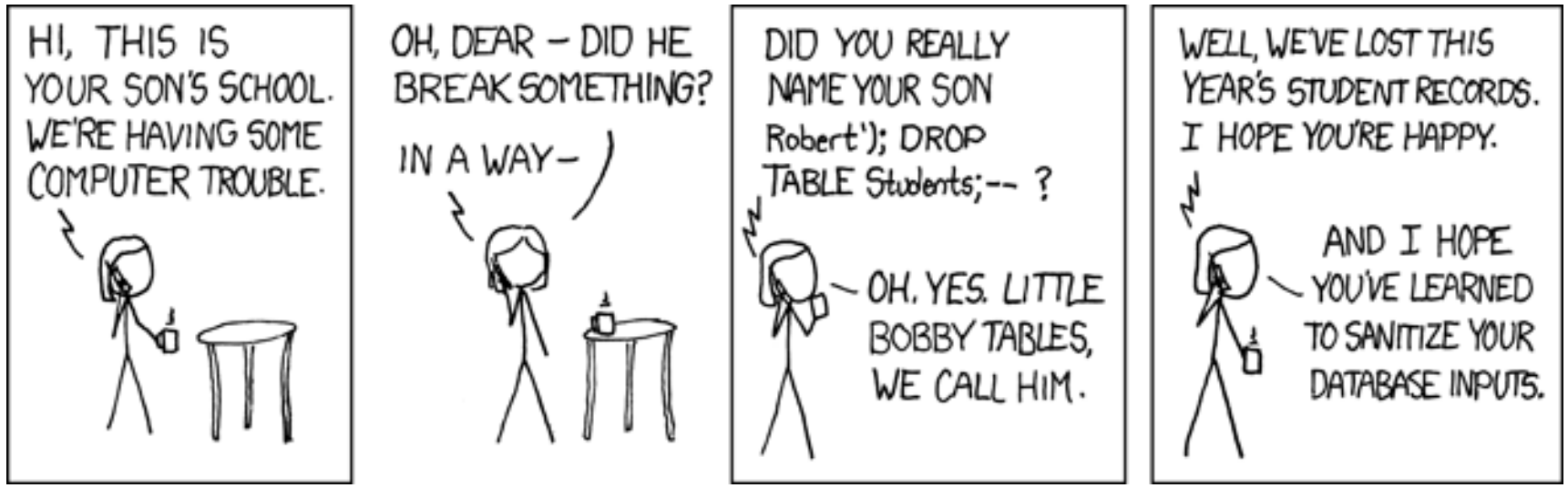
A1. Inyección



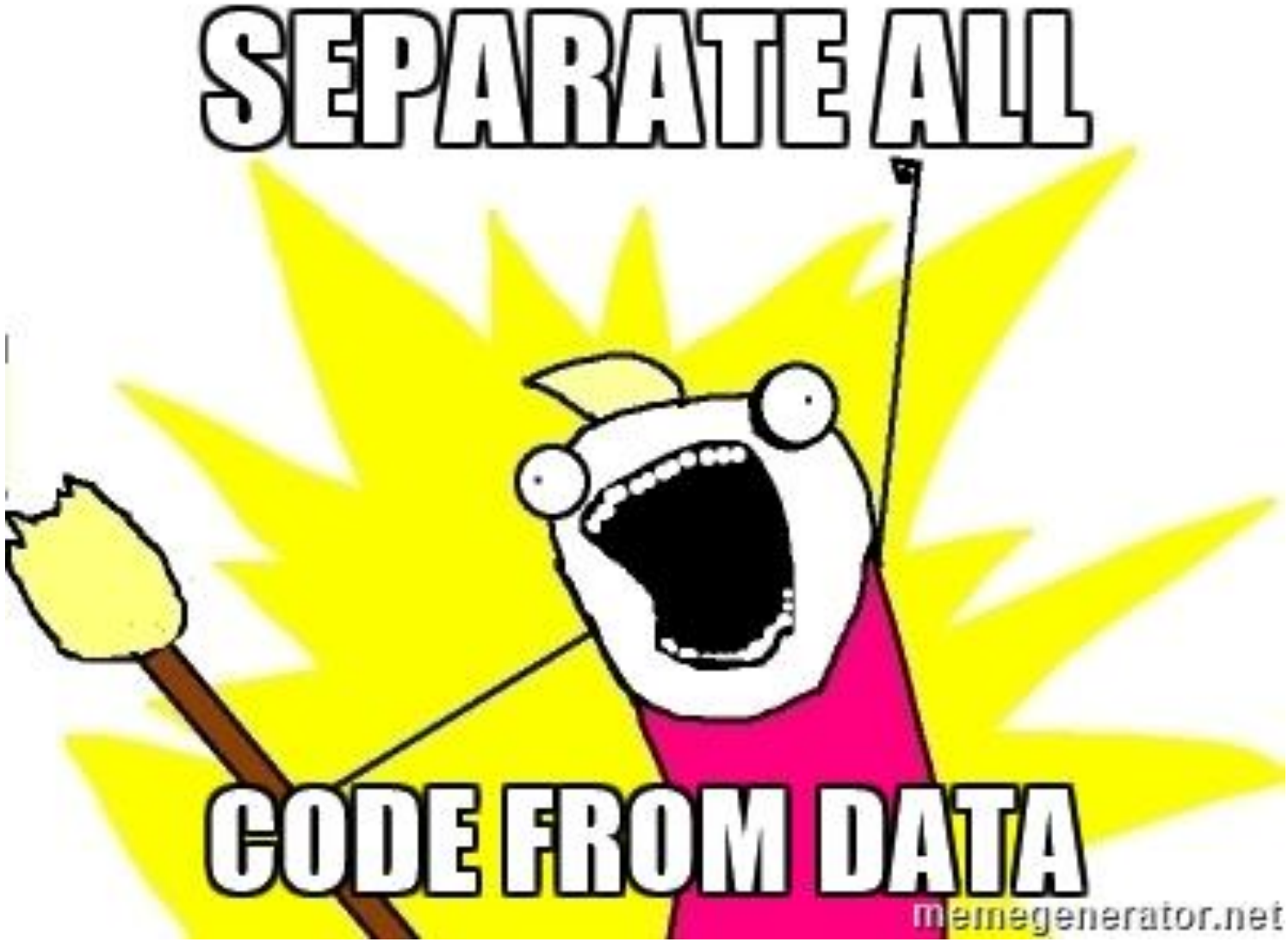
A1. Inyección




LIVE DEMO
DVWA


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.


IT'S NOT ABOUT
SANITIZATION
IT'S ABOUT PARAMETERIZATION!


SEPARATE ALL
CODE FROM DATA
memegenerator.net

A2. Perdida de autenticación / Autenticación rota

- **¿Qué es?:** La autenticación es el proceso para asegurarse de que **realmente eres tu** quien accede a tus cuentas y datos. Generalmente, es facilitado por una combinación de nombre de usuario y contraseña, pero la complejidad se añade cuando la gente olvida o cambia sus contraseñas o quiere actualizar sus direcciones de correo electrónico. Se vuelve aún más complejo a medida que un sitio, aplicación o dispositivo en sí mismo se hace más grande, más amplio y más conectado con otros sitios, aplicaciones o dispositivos.

- **¿Cómo funciona?:** En los ataques más simples, **las contraseñas pueden ser adivinadas o robadas** si se dejan desprotegidas. A medida que se añaden complejidades, los atacantes pueden encontrar otras áreas en las que las credenciales de usuario o las sesiones tienen protecciones inadecuadas y luego secuestrar el acceso de un usuario y, finalmente, sus datos.

- **¿Y eso es malo?:** Si los atacantes pueden **secuestrar la sesión** de un usuario o administrador, tienen acceso a todo lo disponible dentro de esa cuenta, desde los datos hasta el control de la cuenta.

A2. Perdida de autenticación / Autenticación rota

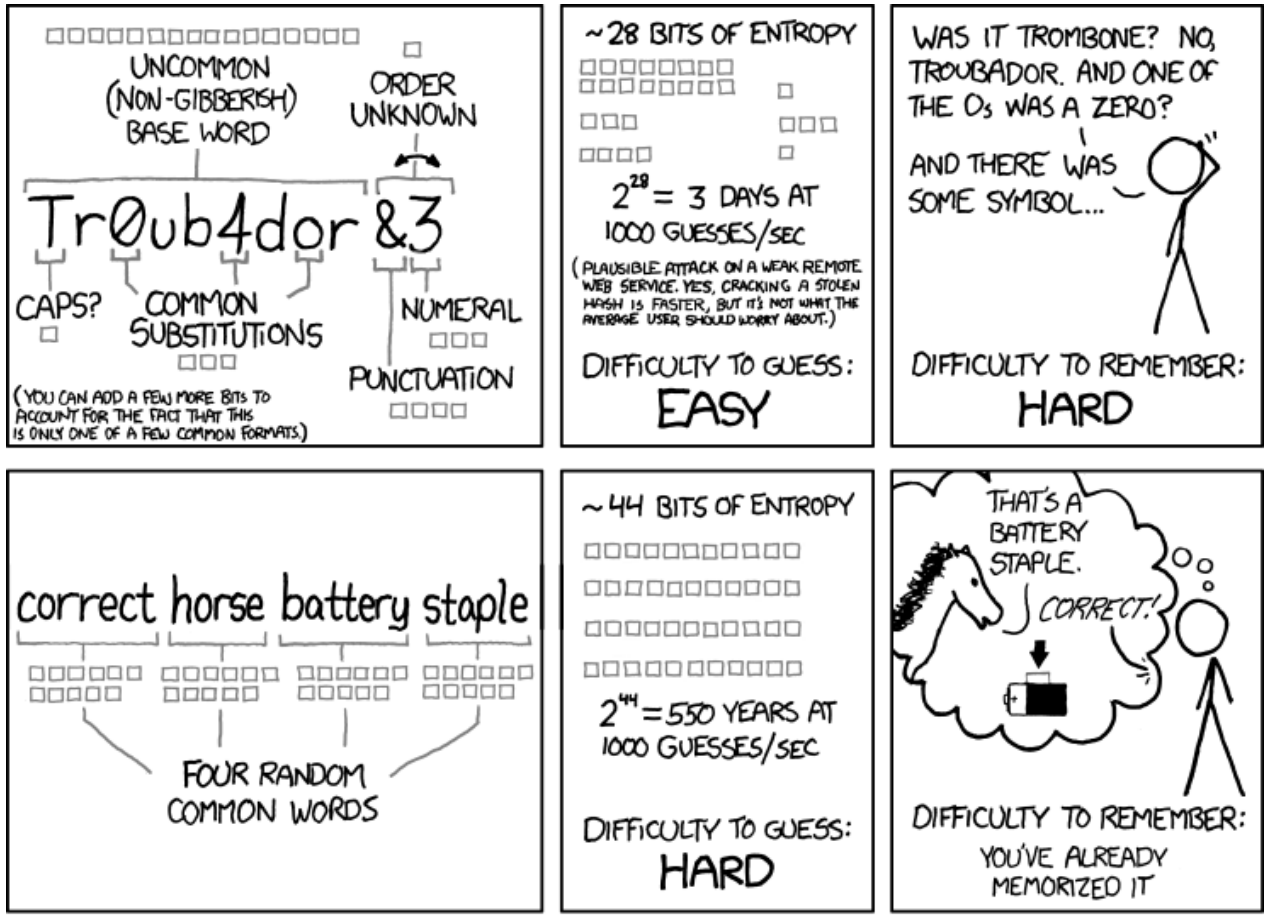
- **Autenticación:** proceso por el cual un sistema verifica la identidad de un usuario
- **Autorización:** función que especifica los derechos de acceso a los recursos a los que puede acceder un usuario.
- **Objetivo del atacante:** Bypass

A2. Perdida de autenticación / Autenticación rota

- **Ataques posibles**
  - Fuerza bruta (https://github.com/danielmiessler/SecLists/tree/master/Passwords)
  - Recuperación de contraseña mal implementada
  - Cookies de sesión que no caducan


LIVE DEMO
DVWA

A2. Perdida de autenticación / Autenticación rota

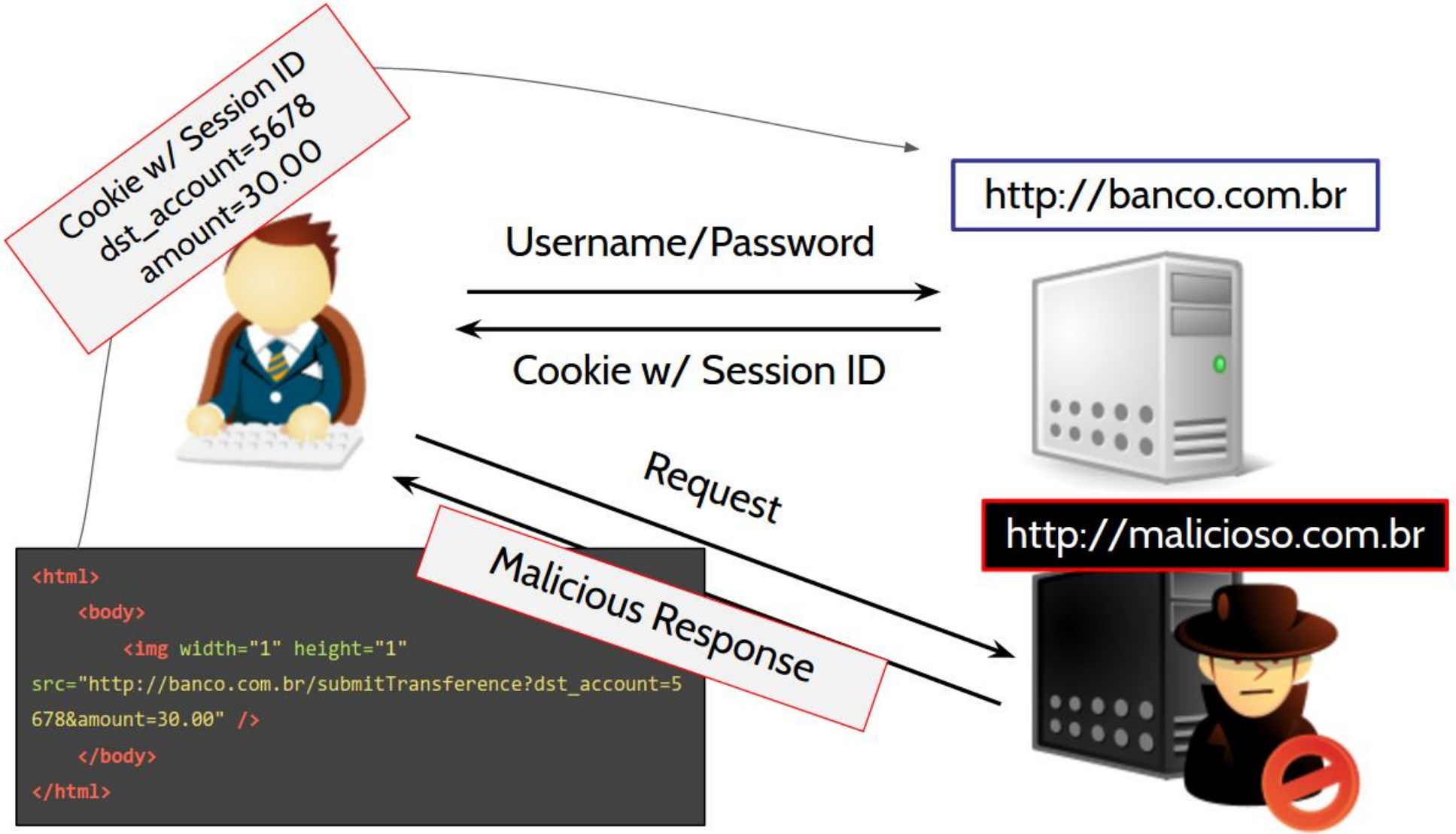


A2. Perdida de autenticación / Autenticación rota

- **Ataques posibles**
  - La web es stateless!
    - Robo de sesión (XSS, MITM, …)
    - Session IDs en URLs
    - Session IDs predecibles
    - CSRF


LIVE DEMO
DVWA

A2. Perdida de autenticación / Autenticación rota




LIVE DEMO
DVWA

A3. Exposición de datos sensibles

- **¿Qué es?:** Los datos sensibles, como los números de **tarjetas de crédito**, los datos de **salud** o las **contraseñas**, deben tener una protección adicional, dado el potencial de daño si caen en las manos equivocadas. Incluso hay regulaciones y normas diseñadas para proteger los datos confidenciales. Sin embargo, si los datos confidenciales se **almacenan**, se **transmiten** o se protegen con métodos inadecuados, pueden quedar expuestos a los atacantes.

- **¿Cómo funciona?:** Si los datos se almacenan o transfieren como **texto plano**, si se utiliza un cifrado más antiguo o más débil, o si los datos se descifran de forma descuidada, los atacantes pueden obtener acceso a los datos y explotarlos.

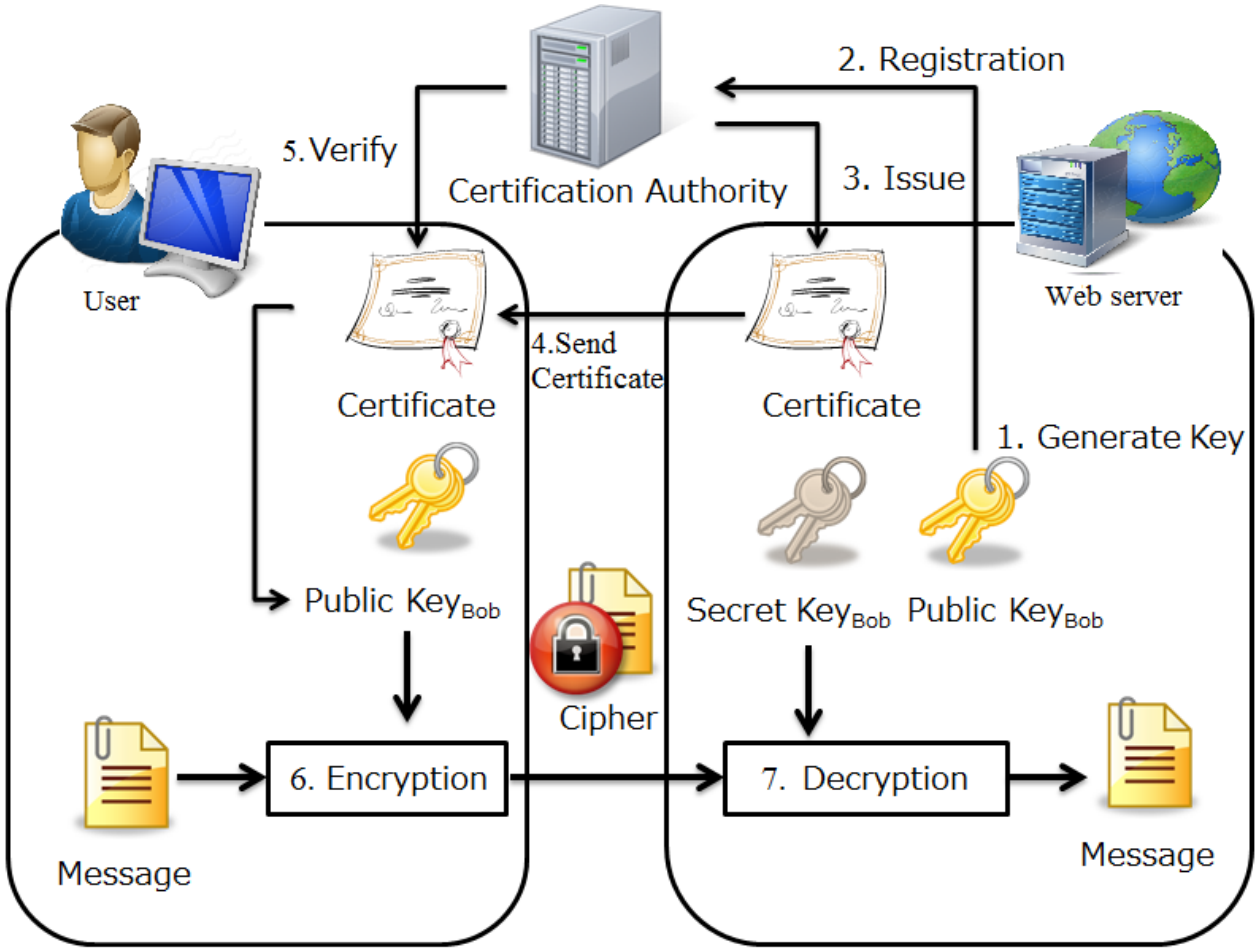
- **¿Y eso es malo?:** Una vez que un atacante tiene contraseñas y números de tarjeta de crédito, puede causar verdadero daño.

A3. Exposición de datos sensibles

- ¿Se transmite datos en **texto claro**?
- ¿Se utilizan **algoritmos** criptográficos obsoletos o **débiles**, ya sea por defecto o en código heredado?
- Por defecto, ¿se aplica cifrado? ¿se han establecido las directivas de seguridad o **encabezados** para el navegador web?
- ¿Estamos afectados por alguna **regulación** que nos obligue a proteger los datos de manera especial?


LIVE DEMO
DVWA

A3. Exposición de datos sensibles – En tránsito



HTTPS!

lenovo Superfish Search Visually

A3. Exposición de datos sensibles – Almacenados

- 2 procesadores Xeon E5, 64 GB de memoria, un SSD de 1 TB y 8 tarjetas EVGA GeForce GTX 1080 Founders Edition
- 500€ * 2 + 300€ + 126*4 + 740€ * 8 = 7724€ (feb'17)
- 341GH/s
  - 341 mil millones de hashes por segundo

A4. Entidades External XML (XXE)

- **¿Qué es?:** XML es un formato utilizado para describir diferentes elementos de datos. **XML** también utiliza "entidades" para ayudar a definir **datos relacionados**, pero las entidades pueden acceder a contenido local o remoto, tan inofensivo como extraer un precio actual de las acciones de un sitio web de terceros. Sin embargo, las entidades pueden ser utilizadas para solicitar datos o archivos locales, que luego podrían ser devueltos, incluso si esos datos nunca fueron destinados a un acceso externo.

- **¿Cómo funciona?:** Un atacante envía valores de búsqueda de datos maliciosos solicitando al sitio, dispositivo o aplicación que solicite y muestre datos de un **archivo local**. Si un desarrollador utiliza un nombre de archivo común o predeterminado en una ubicación común, el trabajo de un atacante es fácil.

- **¿Y eso es malo?:** Los atacantes pueden acceder a cualquier dato almacenado localmente, o pueden pivotar para atacar otros sistemas internos.


LIVE DEMO

A4. Entidades External XML (XXE)

- Billion Laughs

```
<?xml version="1.0"?>
<!DOCTYPE lolz [ <!ENTITY lol "lol">
<!ELEMENT lolz (#PCDATA)>
<!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
<!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
<!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
<!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
<!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
<!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
<!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;">
<!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;">
<!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;"> ]>
<lolz>&lol9;</lolz>
```

A5. Pérdida de control de acceso

- **¿Qué es?:** El control de acceso, o autorización, es la forma en que las aplicaciones web permiten que **diferentes usuarios accedan a diferentes contenidos**, datos o funciones. Es como si Netflix limitara a la gente en su plan "estándar" a contenido HD, mientras que los usuarios "premium" pueden ver 4K. Cuando se rompe, puedes acceder a más de lo que deberías.

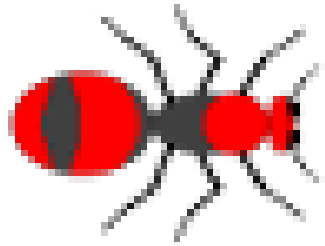
- **¿Cómo funciona?:** Al igual que con otras vulnerabilidades, los atacantes pueden obtener acceso a (y modificar) datos, cuentas y funciones que no deberían.

- **¿Y eso es malo?:** La plataforma del **sistema telemático de Justicia**, permitía el acceso a los casos judiciales a través de una simple URL que terminaba con un número creciente. Utilizando cualquier número se puede acceder al espacio privado de otros usuarios de la plataforma.

A5. Pérdida de control de acceso

- **Validación en el lado del cliente**
  - Cookies que establecen los permisos
  - Validaciones realizadas en javascript pero no en el servidor

- **File inclusion attacks**
  - LFI (Local File Inclusion)
  - RFI (Remote File Inclusion)



DVWA

A6. Configuración de Seguridad Incorrecta

- **¿Qué es?:** Exactamente como su nombre lo indica, una mala configuración de seguridad es cuando se han pasado por alto algunas vulnerabilidades. Esto incluye el uso de **credenciales predeterminadas**, dejar los archivos desprotegidos en servidores públicos, tener **fallos conocidos** pero no corregidos, y más, y en cualquier capa de la pila de software.

- **¿Cómo funciona?:** La gente está ocupada, se pierden cosas, se toman decisiones de priorización... y las vulnerabilidades se dejan sin control. **Prioridades**.

- **¿Y eso es malo?:** Facilita que incluso los atacantes novatos encuentren y accedan a tus valiosos sistemas y datos. Afortunadamente, la mayoría de estos tipos de vulnerabilidades también son fáciles de encontrar y arreglar. La botnet Mirai lograba propagarse gracias a credenciales por defecto en routers y otros dispositivos IoT, llegando a infectar más de 400.000 dispositivos.

A6. Configuración de Seguridad Incorrecta

- Passwords por defecto
- Directorios navegables
- Configuraciones inseguras de SSL/TLS
- Scripts de prueba
- Variables de debug
- Backups
- Subida de ficheros


LIVE DEMO
DVWA

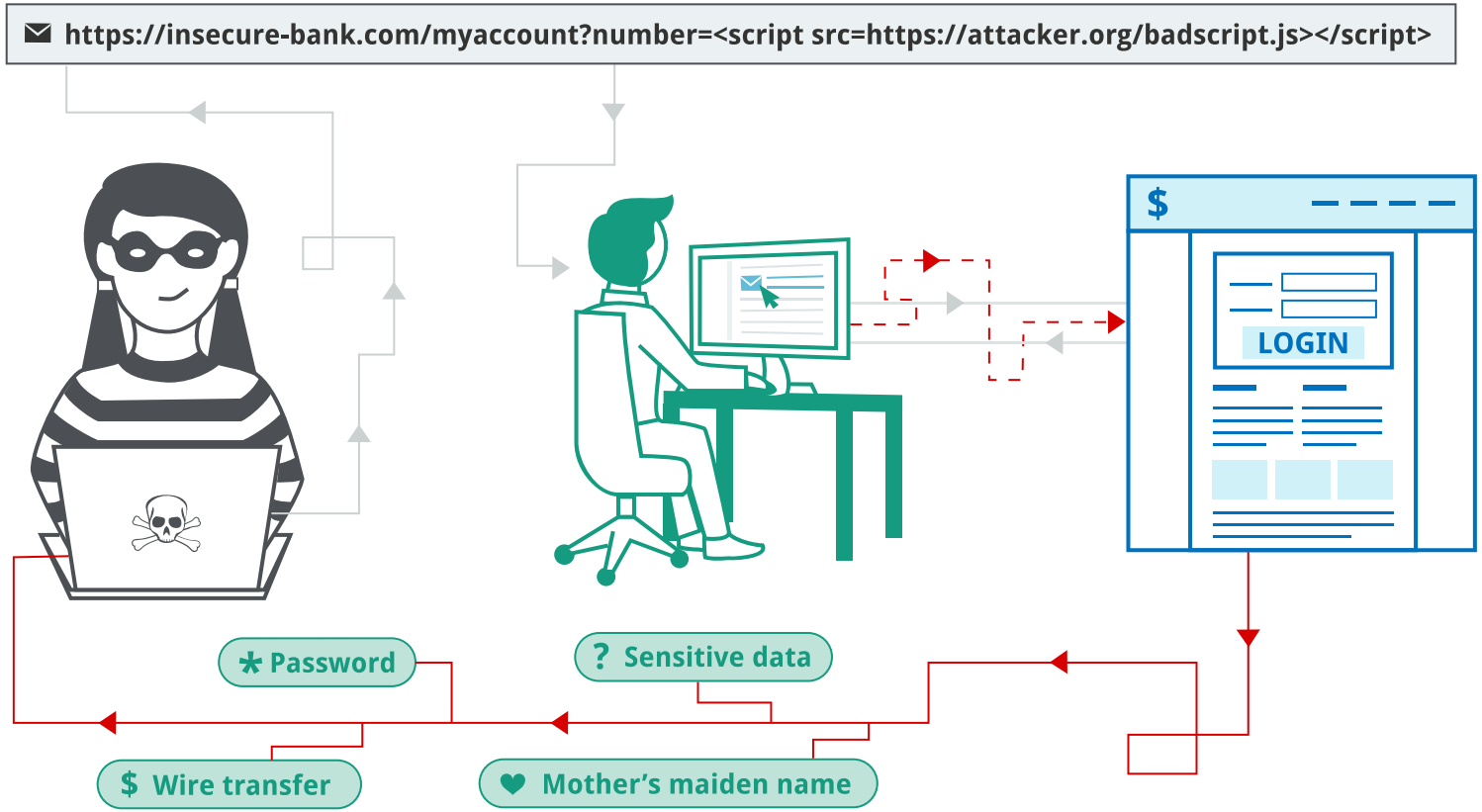
A7. Cross-Site Scripting (XSS)

- **¿Qué es?:** XSS permite añadir código malicioso a una página web o aplicación, por ejemplo, a través de comentarios del usuario o de **formularios** utilizados para definir la acción posterior. Dado que **HTML mezcla instrucciones de control, formato y el contenido** solicitado en el código fuente de la página web, permite una oportunidad para que el código no desinfectado se utilice en la página resultante.

- **¿Cómo funciona?:** Cuando una página web o una aplicación utiliza contenido introducido por el usuario como parte de una página resultante sin comprobar si hay algo malo, **un usuario malintencionado puede introducir contenido que incluya entidades HTML**.

- **¿Y eso es malo?:** Los atacantes pueden cambiar el comportamiento de una aplicación, dirigir datos a sus propios sistemas, o corromper o sobrescribir datos existentes.

A7. Cross-Site Scripting (XSS)



A7. Cross-Site Scripting (XSS)

- Reflejado
- Almacenado
- Basado en DOM

A7. Cross-Site Scripting (XSS)

- Account Hijacking
- Robar credenciales y otros datos sensibles
- Impersonar al usuario
- Obtener información de la víctima (navegador, plugins, …)
- Drive-by downloads
- Keylogger
- Escanear puertos de la red interna
- Modificar el contenido de la web o redirigir a otro sitio (phishing)
- Ejecutar un exploit del navegador
- Bypass de medidas anti CSRF

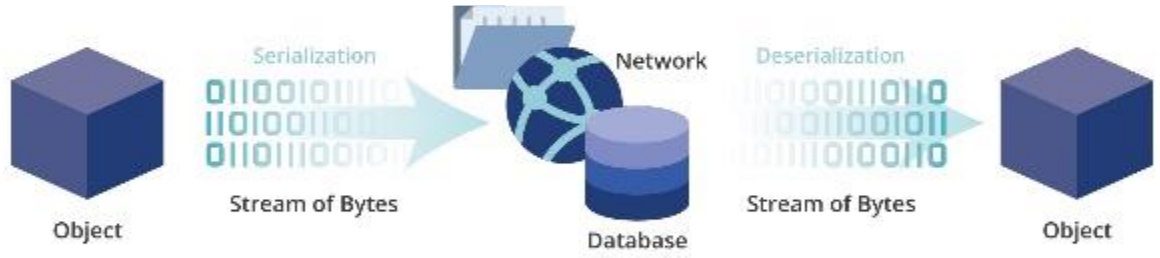

LIVE DEMO
DVWA

A8. Deserialización Insegura

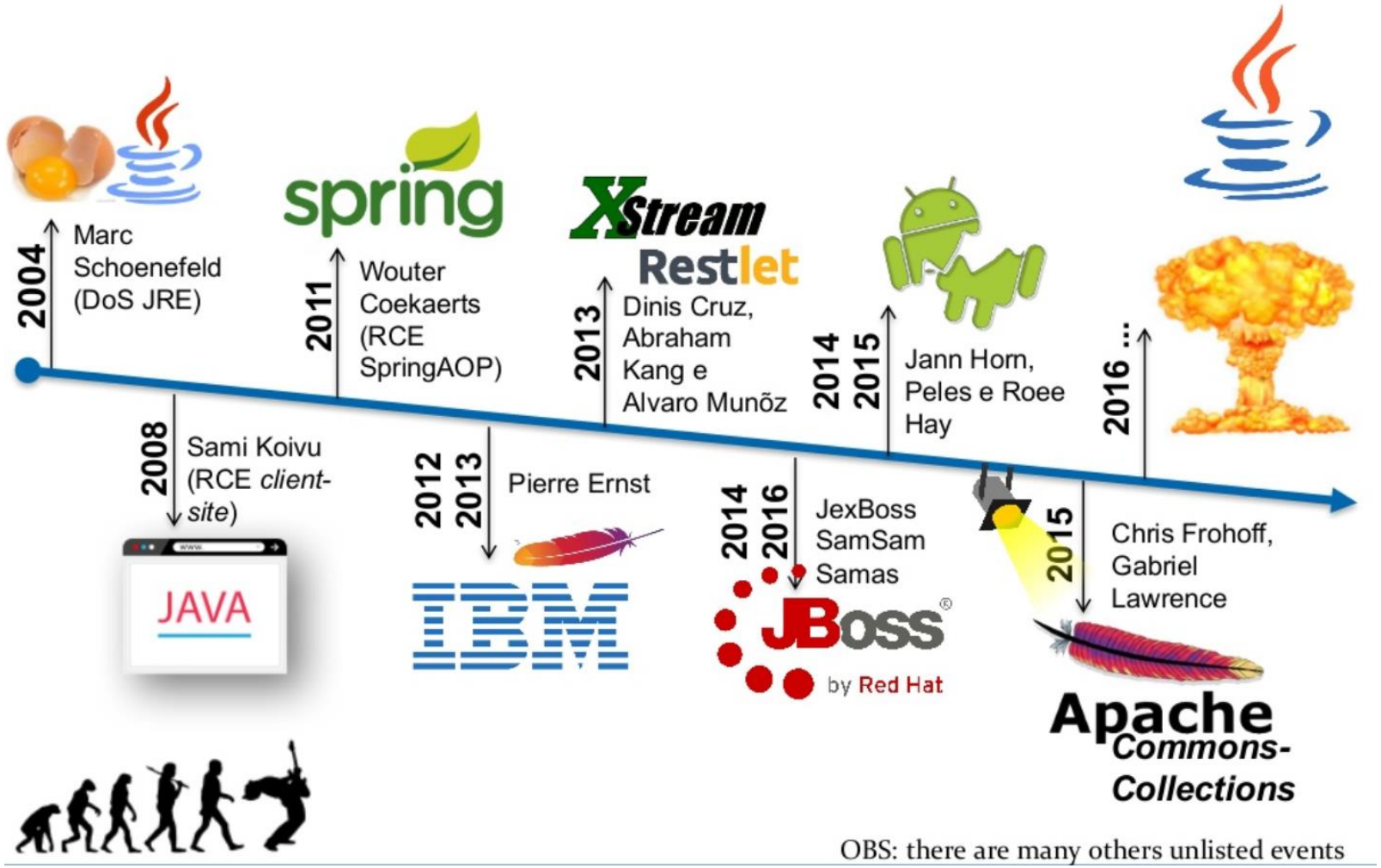
- **¿Qué es?:** Antes de que los datos se almacenen o transmitan, los bits a menudo se serializan para que puedan ser restaurados posteriormente a la estructura original de los datos. Volver a ensamblar una serie de bits en un archivo u objeto se denomina deserialización.

- **¿Cómo funciona?:** Los datos deserializados pueden modificarse para incluir código malicioso, lo que puede causar problemas si la aplicación no verifica la fuente o el contenido de los datos antes de la deserialización.

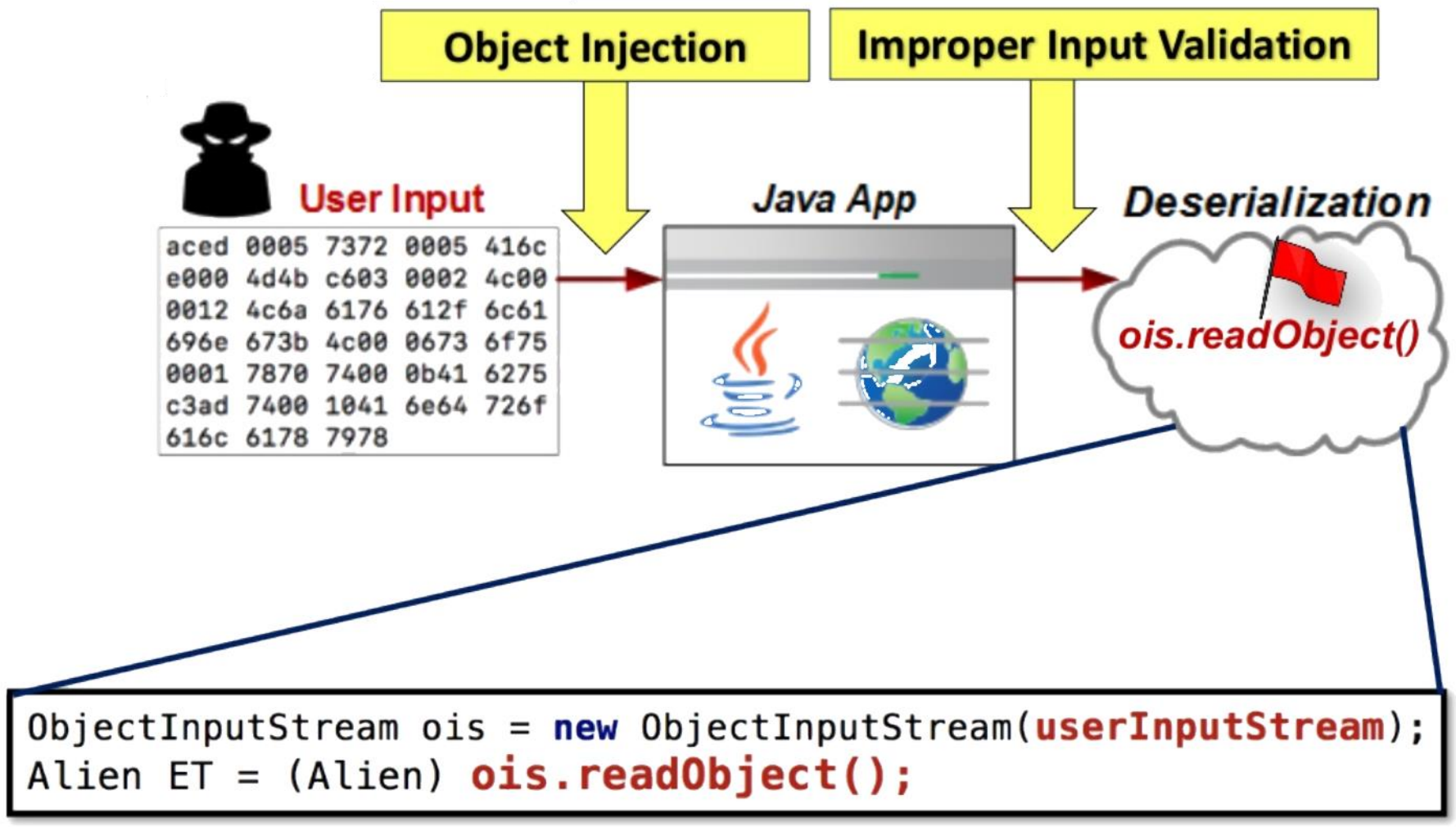
- **¿Y eso es malo?:** Los atacantes pueden construir objetos ilegítimos que ejecutan comandos dentro de una aplicación infectada.



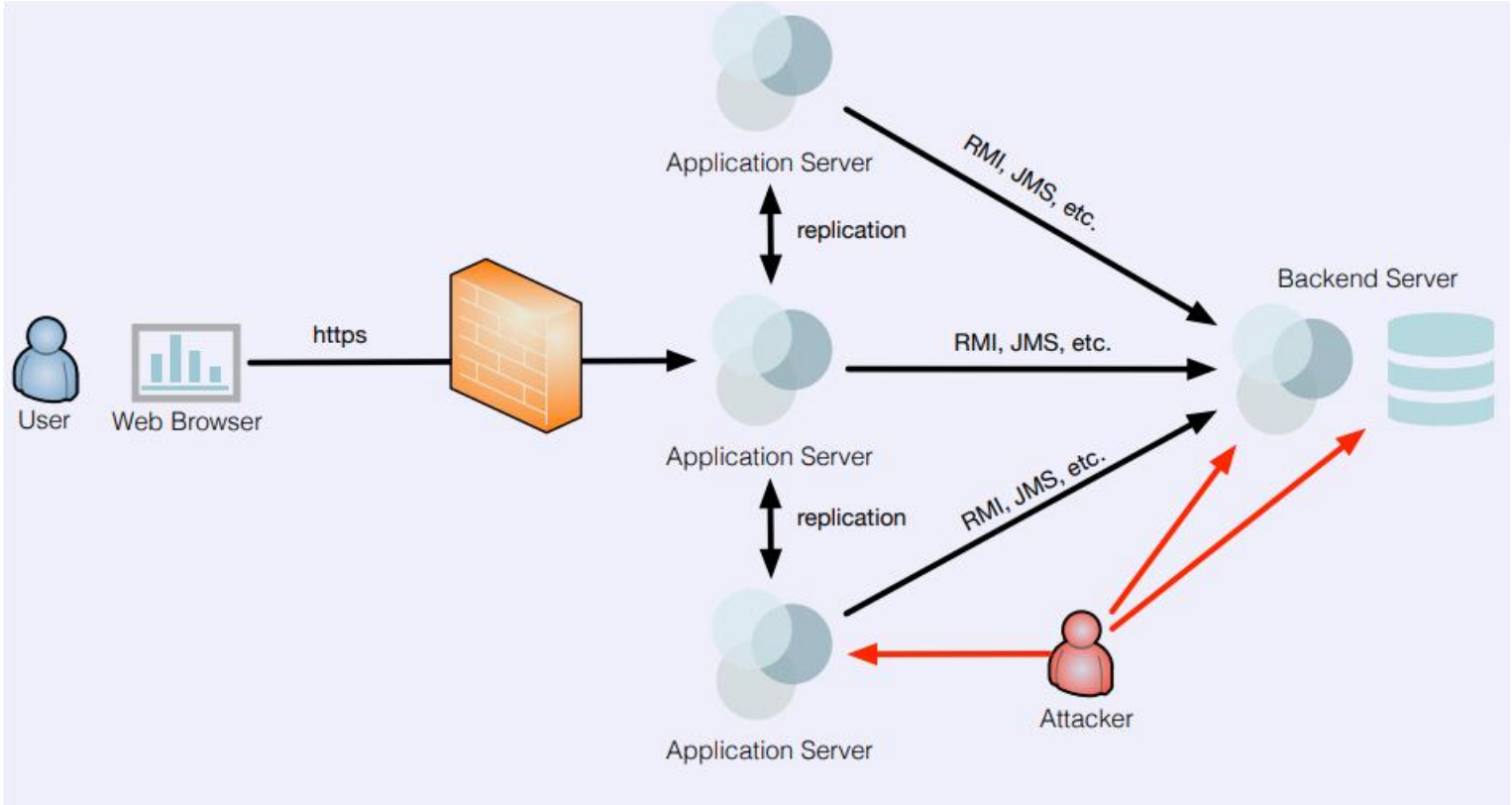
A8. Deserialización Insegura



A8. Deserialización Insegura



A8. Deserialización Insegura



JexBoss

Jboss (and Java Deserialization Vulnerabilities) verify and Exploitation Tool

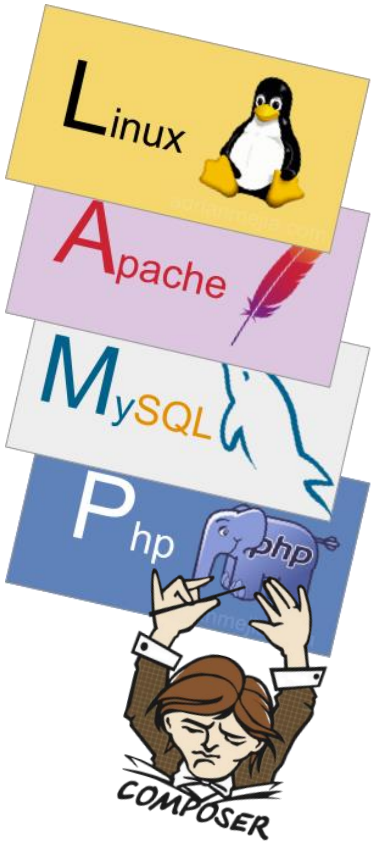
A9. Uso de Componentes con Vulnerabilidades Conocidas

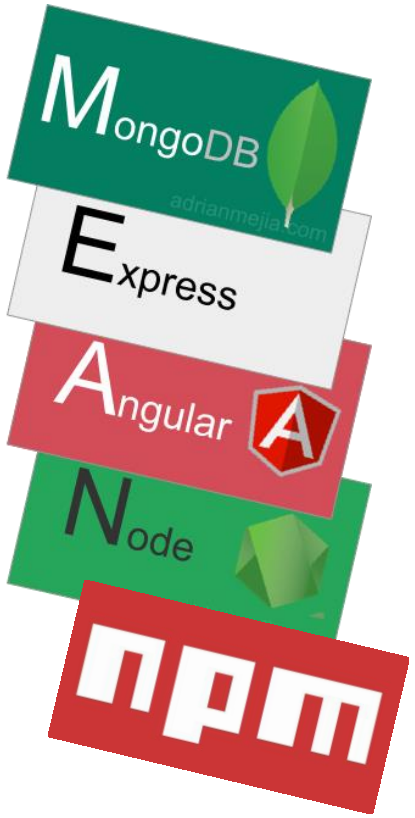
- **¿Qué es?:** Cuando se conocen las vulnerabilidades, los proveedores generalmente las solucionan con un **parche o una actualización**. El proceso de actualización del software elimina o mitiga dicha vulnerabilidad.

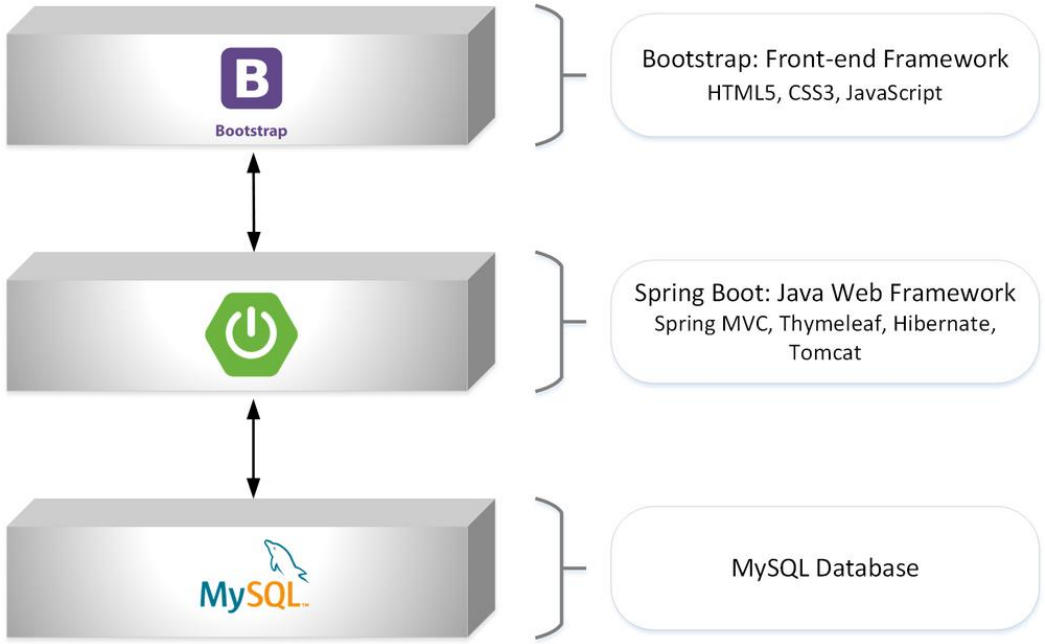
- **¿Cómo funciona?:** Las organizaciones a veces no mantienen el software actualizado, especialmente si sus **frameworks son grandes o complejos**, o si esto requeriría un esfuerzo significativo para validar sus sistemas o productos después de una actualización. Cuando un exploit se hace público o se libera un parche, los atacantes saben que algunas organizaciones no actuarán inmediatamente. Los atacantes ahora tienen **una ventana, de días a años**, para buscar sistemas o aplicaciones en los que la vulnerabilidad conocida sigue existiendo.

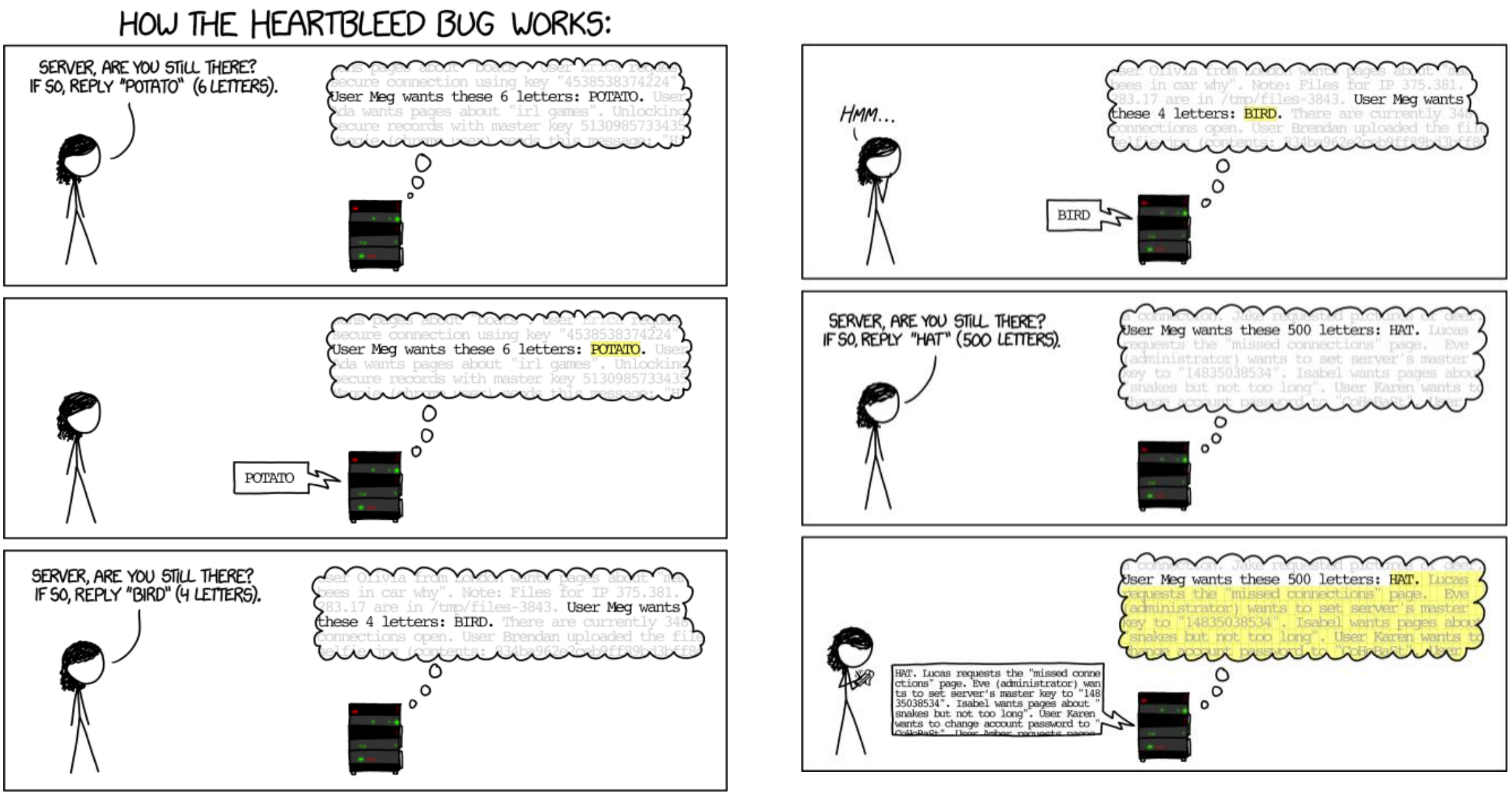
- **¿Y eso es malo?:** Debido a que es información pública, los atacantes tienen un camino de ataque claro para explotar y las organizaciones tienen pocas excusas para dejar el camino abierto.

A9. Uso de Componentes con Vulnerabilidades Conocidas








HOW THE HEARTBLEED BUG WORKS:
SERVER, ARE YOU STILL THERE? IF SO, REPLY "POTATO" (6 LETTERS).
User Meg wants these 6 letters: POTATO.
POTATO
SERVER, ARE YOU STILL THERE? IF SO, REPLY "BIRD" (4 LETTERS).
User Meg wants these 4 letters: BIRD.
HMM...
BIRD
SERVER, ARE YOU STILL THERE? IF SO, REPLY "HAT" (500 LETTERS).
User Meg wants these 500 letters: HAT.
HAT. Lucas requests the "missed connections" page. Eve (administrator) wants to set server's master key to "14835038534". Isabel wants pages about "snakes but not too long". User Karen wants to change account password to "


LIVE DEMO

A10. Registro y Monitoreo Insuficientes

- **¿Qué es?:** Si no estás buscando atacantes o actividades sospechosas, **no las vas a encontrar**.

- **¿Cómo funciona?:** El software y los sistemas tienen capacidades de monitorización para que las organizaciones puedan ver los inicios de sesión, las transacciones, el tráfico y mucho más. Al **monitorizar actividades sospechosas**, como inicios de sesión fallidos, las organizaciones pueden potencialmente ver y detener actividades sospechosas.

- **¿Y eso es malo?:** Los atacantes confían en la falta de monitorización para explotar las vulnerabilidades antes de que sean detectadas. Sin la supervisión y el registro para mirar hacia atrás y ver lo que sucedió, los atacantes pueden causar daños ahora y en el futuro.

jtsec: Beyond IT Security
c/ Abeto s/n Edificio CEG Oficina 2B
CP 18230 Granada – Atarfe – Spain
hola@jtsec.es

@jtsecES
www.jtsec.es

¡Gracias por aguantar!