HACKTHEBOX

Informe Técnico

Máquina Pressed

14 de abril del 2022

Índice

1. Antecedentes

El presente documento recoge los resultados obtenidos durante la fase de auditoría realizada a la máquina **Pressed** de la plataforma **HackTheBox**.

Figura 1: Dirección IP de la máquina

Dirección URL

Ir a la máquina.

2. Objetivos

Conocer el estado de seguridad actual del servidor **Pressed**, enumerando posibles vectores de explotación y determinado alcance e impacto que un atacante podria ocasionar sobre el sistema en producción.

2.1. Consideraciones

Una vez finalizadas las joranadas de auditoría, se llevará a cabo una fase de saneamientos y buenas prácticas con el objetivo de securizar el servidor y evitar ser victimas de un futuro ataque en base a los vectores explotados.

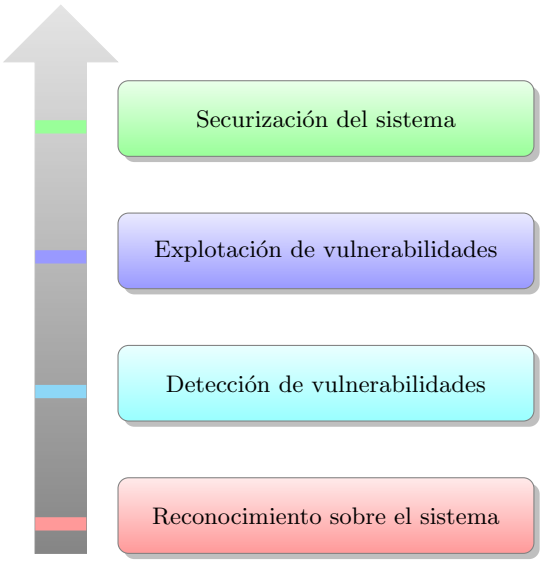

Figura 2: Flujo de trabajo

3. Analisis de vulnerabilidades

3.1. Vulnerabilidades encontradas

Se comenzó realizando un escaneo de puertos abiertos y escaneo de exhaustivo para poder ver como trabaja el sistema. Se observó el puerto 80 por lo cual decidí hacer un analisis de las tecnologias que tenia esta pagina y pude identificar que era Wordpress. Por lo cual procedí a utiizar la herramienta wpscan para ver el contenido y posibles vulnerabilidades. Me encontré con una ruta que contenia credenciales de administrador pero se debía modificar la contraseña por el año correspondiente para poder acceder, a pesar de acceder apareció un apartado con codigo de verificación de 3 intentos posibles, por lo que aplicar fuerza bruta no era una opción viable. Al indagar con respecto a Wordpress y sus posibles vias para ver información importante, investigue el archivo XML-RPC.php que me mostró wpscan, leyendo un poco más acerca de el vi varias funciones posibles para poder hacer un muy buen uso de este. Utilizando la función de listar metodos pude observar un metodo que me permitia manipular un apartado "posts.[el] cual es potencial a efectuar RCE. Haciendo uso de wordpress xmlrpc en python pude generar una comunicacion la cual me permitia obtener información importante para poder tratar de obtener RCE. Encontre una cadena en base64 que contenia codigo PHP en urlEncode por lo cual procedí a decodear todo y vi que tenia una instruccion en una ruta. Decidí crear otra variable con el contenido de la variable posts, pero en esta variable remplacé la cadena en base64 que tenía por uno que yo hice, que mi cadena la cual tenia de contenido el mismo codigo php pero con la excepción de que le indicaba hacer uso del CMD para poder efectuar el RCE. Creando esta variable de nombre malpost y efectuando lo anterior pude hacer RCE, esto me era posible de modificar porque ya habia encontrado las credenciales del administrador. Intenté conseguir una reverse shell pero no me fue posible debido a que la pagina tenia IPTABLES, por lo que se decidió hacer un script en bash el cual efectuaba el RCE pero en mi terminal. Al investigar más a fondo que contenia la maquina observé que contenia pkexec, así que hice uso de la herramienta pkwner. La cual metí en la maquina y efectuaba comandos a nivel de root y así pude obtener las dos banderas, pero lo importante aquí es obtener control de la maquina. Así que hice uso de la herramienta ttyoverhttp.py de S4vitar la cual modifique un poco para poder hacer uso en la maquina, este script funciona con una funcionalidad mkfifo que permite crer una forwarded shell la cual nos permite interactuar, navegar entre directorios). Pero para poder hacer uso de este script metí un archivo php con instucciones para poder hacer uso del CMD.

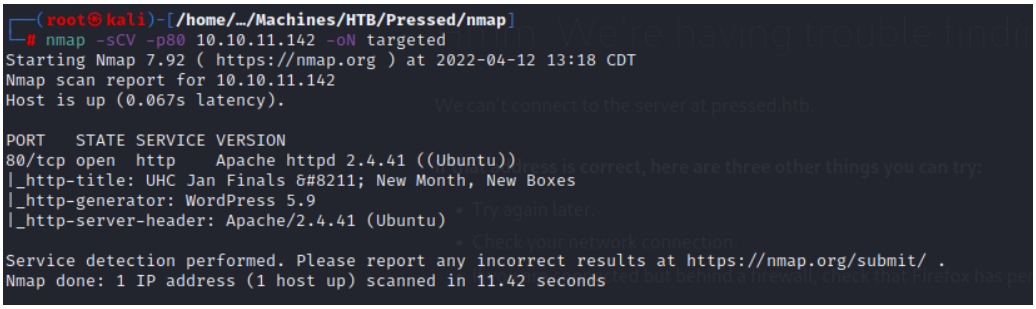

Figura 3: nmap

Figura 4: Haciendo uso de wpscan.

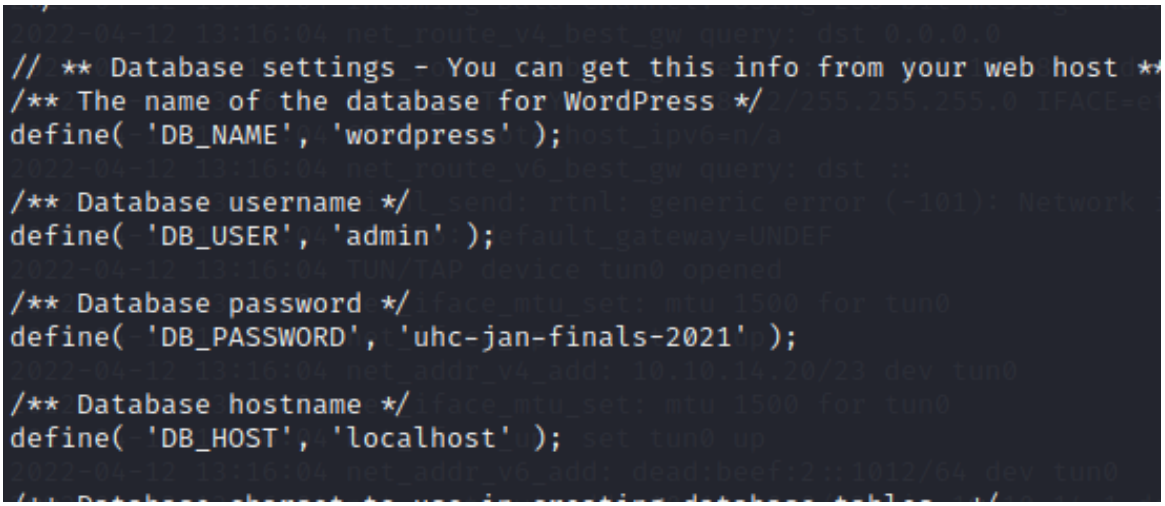

Figura 5: Obteniendo credenciales del archivo config.

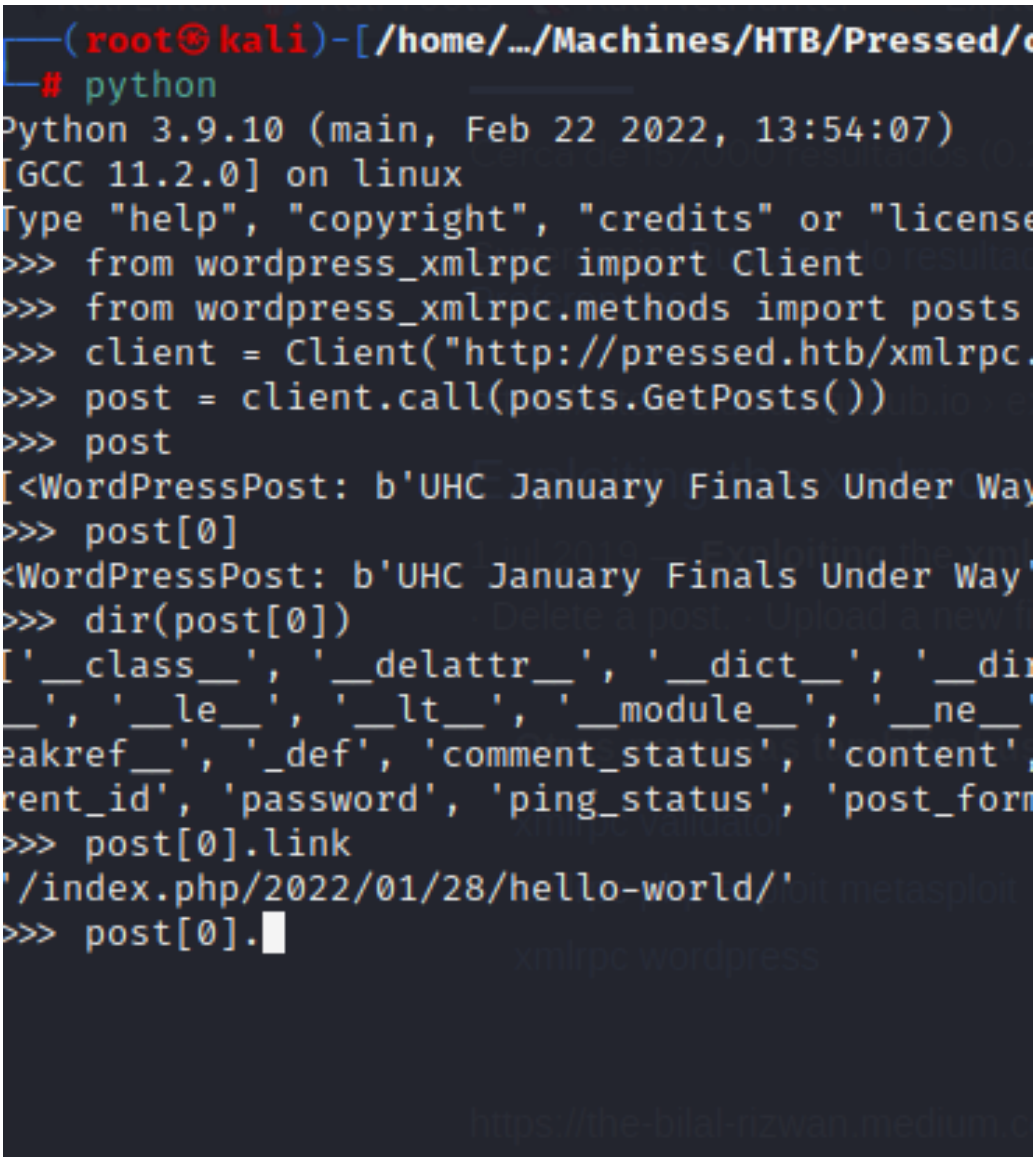

Figura 6: Haciendo uso de Wordpress xmlrpc en python.

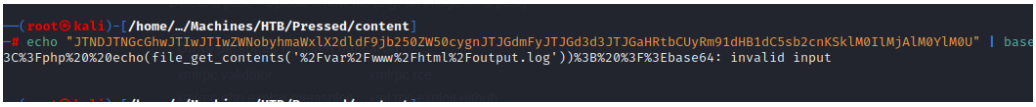

Figura 7: Visualizando el contenido codificado en base64 del metodo.

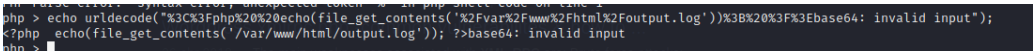

Figura 8: Contenido decodificado.

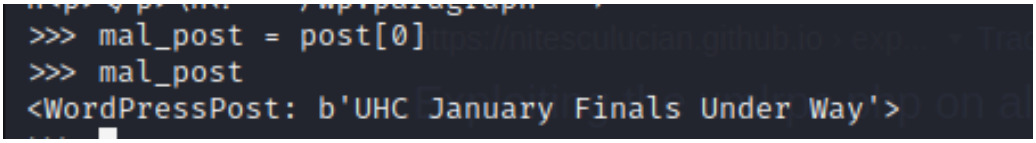

Figura 9: Creando variable malpost.

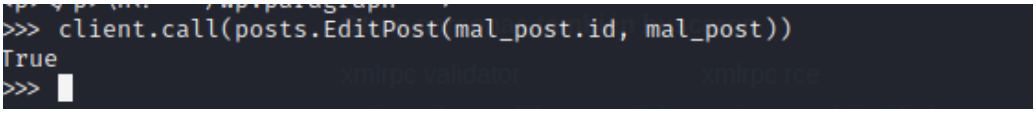

Figura 10: Editando post.

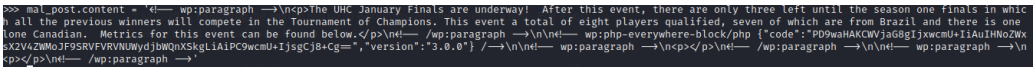

Figura 11: Introduciendo la instrucción de php en base64 para poder efectuar RCE.

Figura 12: RCE en la web.

Figura 13: RCE en la terminal con script en bash creado.

```
>>> from wordpress_xmlrpc.methods import media
>>> with open("pkwner.sh", "r") as f:
...     filename = f.read()
```

Figura 14: Creando variable malpost.

```
>>> from wordpress_xmlrpc.methods import media
>>> with open("pkwner.sh", "r") as f:
...     filename = f.read()
```

Figura 15: Creando variable malpost.

```
>>> from wordpress_xmlrpc.methods import media
>>> with open("pkwner.sh", "r") as f:
...     filename = f.read()
```

Figura 16: Pkwner.

```
>>> data_to_upload = { 'name': 'pkwner.png', 'bits': filename, 'type': '
'  }
>>> client.call(media.UploadFile(data_to_upload))
'attachment_id': '48', 'date_created_gmt': <DateTime '20220413T03:12:43
f0b9ef9d0>, 'parent': 0, 'link': '/wp-content/uploads/2022/04/pkwner.pn
': 'pkwner.png', 'caption': '', 'description': '', 'metadata': False, '
xt/plain', 'thumbnail': '/wp-content/uploads/2022/04/pkwner.png', 'id':
le': 'pkwner.png', 'url': '/wp-content/uploads/2022/04/pkwner.png'}
```

Figura 17: Subiendo el pkwner al servidor.

```
┌──(root㉿kali)-[/home/…/Machines/HTB/Pressed/content]
└─# base64 -w 0 cmd.php | xclip -sel clip
```

Figura 18: Codificando el codigo php en base 64 para poderlo meter en el servidor.

```
>>> data_to_upload = { 'name': 'pkwner.png', 'bits': filename, 'type': '
'  }
>>> client.call(media.UploadFile(data_to_upload))
'attachment_id': '48', 'date_created_gmt': <DateTime '20220413T03:12:43
f0b9ef9d0>, 'parent': 0, 'link': '/wp-content/uploads/2022/04/pkwner.pn
': 'pkwner.png', 'caption': '', 'description': '', 'metadata': False, '
xt/plain', 'thumbnail': '/wp-content/uploads/2022/04/pkwner.png', 'id':
le': 'pkwner.png', 'url': '/wp-content/uploads/2022/04/pkwner.png'}
```

Figura 19: Metiendo el archivo php codificado desde el script creado en bash.

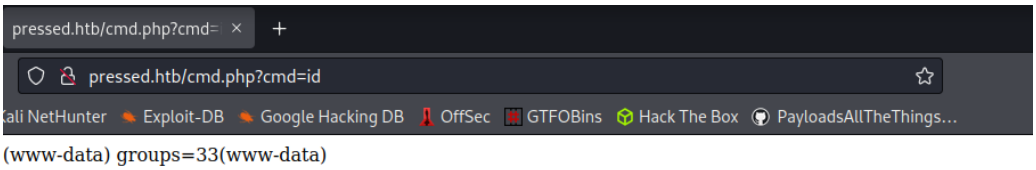

Figura 20: RCE por medio del archivo previamente introducido.

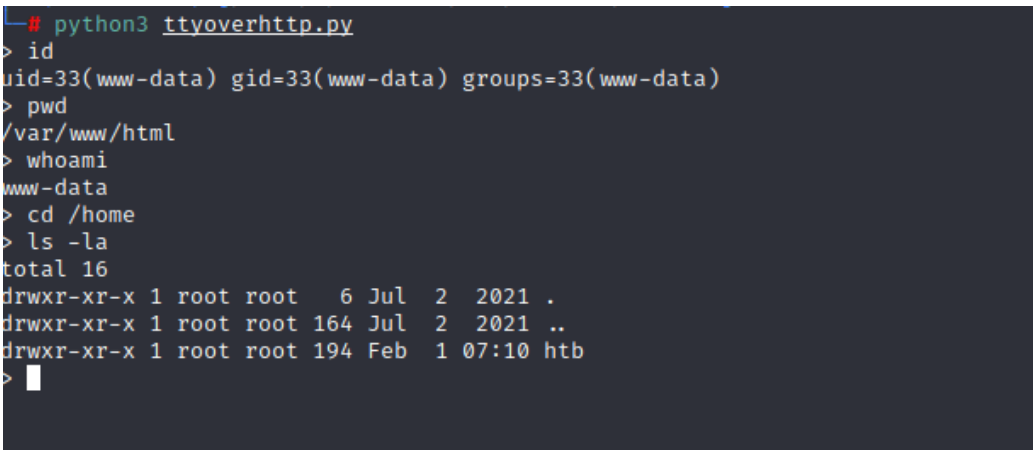

Figura 21: Uso del script de S4vitar, modificandolo para su uso y poder tener interaccion.

Para poder escalar de privilegios modifique la instruccion del pkwner para asignarle permisos SUID a la bash. Ejecute bash en la ttyoverhttp (el script). Y obtuve acceso como root.

Figura 22: Modificando la instruccion de la herramienta pkwner.

Figura 23: Maquina pwneada.