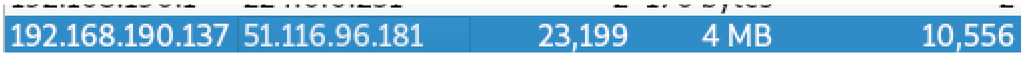
LetsDefend: Brute Force Attacks

The following writeup covers the Brute Force Attacks room hosted on LetsDefend. This room is entirely concerned with investigating a brute force attack.

Scenario: Our web server has been compromised, and it's up to you to investigate the breach. Dive into the system, analyse logs, dissect network traffic, and uncover clues to identify the attacker and determine the extent of the damage.

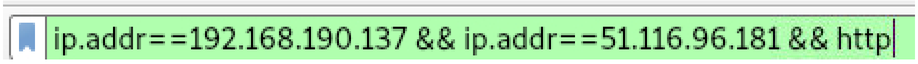
What is the IP address of the server targeted by the attacker's brute-force attack?

After opening up the pcap in Wireshark, I navigate to Statistics > Conversations, and opened up the IPv4 tab. Here we can see a very large number of traffic between 192.168.190.137 and 51.116.96.181:



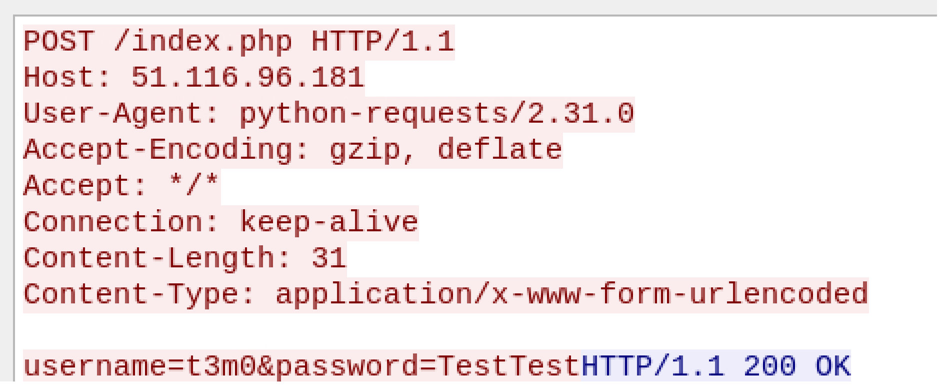
192.168.190.137	51.116.96.181	23,199	4 MB	10,556

Based on this, we can likely determine that 51.116.96.181 is the IP address being targeted, but lets confirm this. We can use the following display filter to see all http traffic between these two hosts:



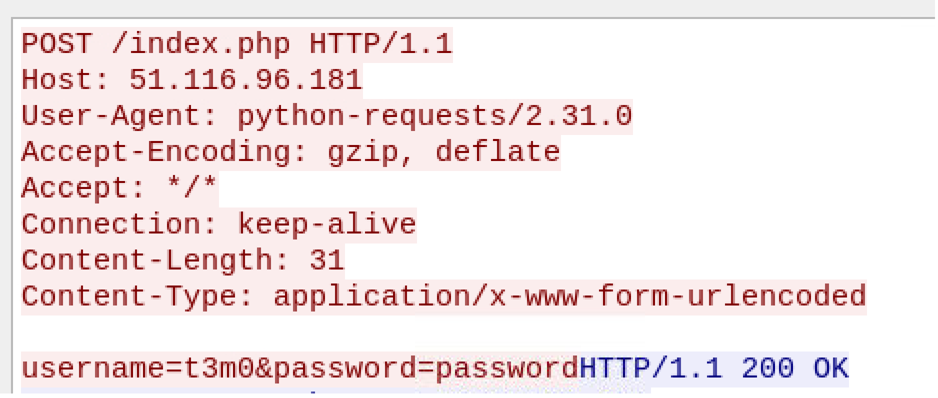
```
ip.addr==192.168.190.137 && ip.addr==51.116.96.181 && http
```

If you follow the TCP stream of these HTTP requests, we can see the client (aka 192.168.190.137) sending several post requests containing credentials:



```
POST /index.php HTTP/1.1
Host: 51.116.96.181
User-Agent: python-requests/2.31.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Content-Length: 31
Content-Type: application/x-www-form-urlencoded

username=t3m0&password=TestTestHTTP/1.1 200 OK
```



```
POST /index.php HTTP/1.1
Host: 51.116.96.181
User-Agent: python-requests/2.31.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Content-Length: 31
Content-Type: application/x-www-form-urlencoded

username=t3m0&password=passwordHTTP/1.1 200 OK
```

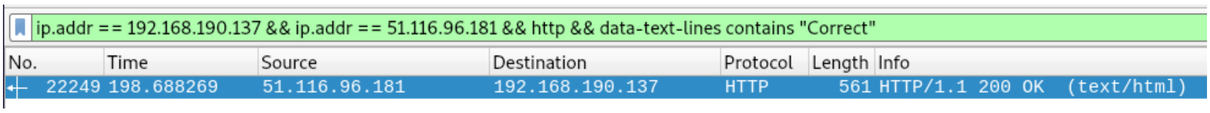
Etc, etc. Therefore, the answer is 51.116.96.181.

Which directory was targeted by the attacker's brute-force attempt?

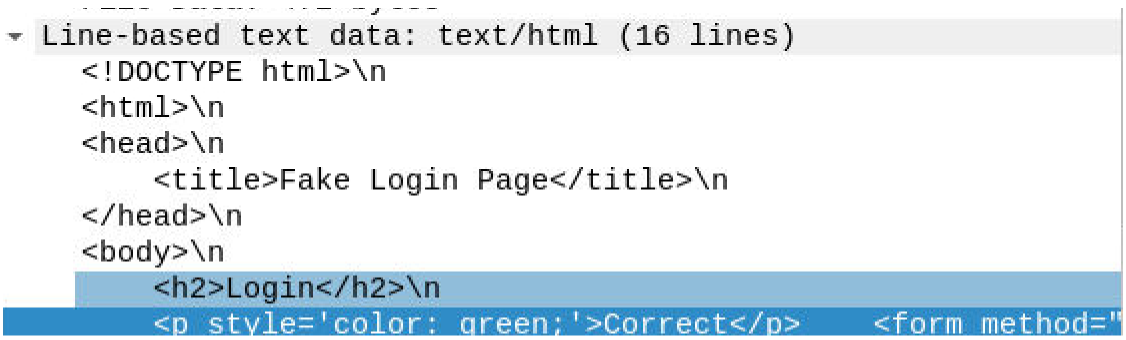
As found previously, the attacker is targeting /index.php.

Identify the correct username and password combination used for login.

If we look at the HTTP requests (specifically the responses from the web server), we can see that it will say "Incorrect" when the wrong credentials are answered. Therefore, we can use a filter like as follows to look for "Correct":

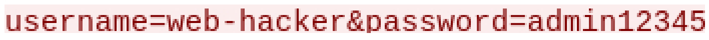


```
ip.addr == 192.168.190.137 && ip.addr == 51.116.96.181 && http && data-text-lines contains "Correct"
No.    Time        Source          Destination      Protocol  Length  Info
22249  198.688269  51.116.96.181   192.168.190.137  HTTP         561  HTTP/1.1 200 OK  (text/html)
```



```
Line-based text data: text/html (16 lines)
    <!DOCTYPE html>\n
    <html>\n
    <head>\n
        <title>Fake Login Page</title>\n
    </head>\n
    <body>\n
        <h2>Login</h2>\n
        <p style='color: green;'>Correct</p>    <form method="
```

If we follow the stream, we can see the correct credentials:

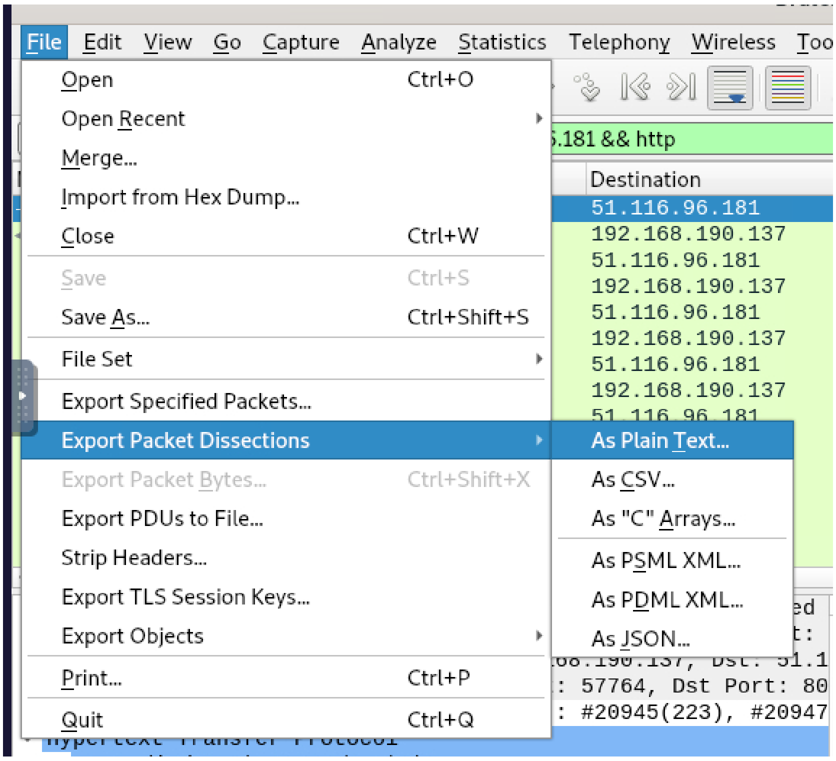


```
username=web-hacker&password=admin12345
```

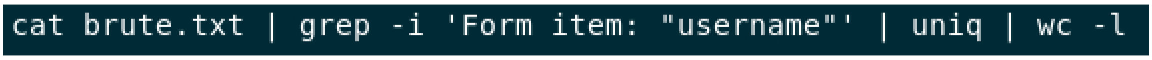
web-hacker:admin12345

How many user accounts did the attacker attempt to compromise via RDP brute-force?

For this question I used the help of the featured writeup. All you need to do is use the http filters shown previously, and navigate to File > Export Packet Dissections > As Plain Text:



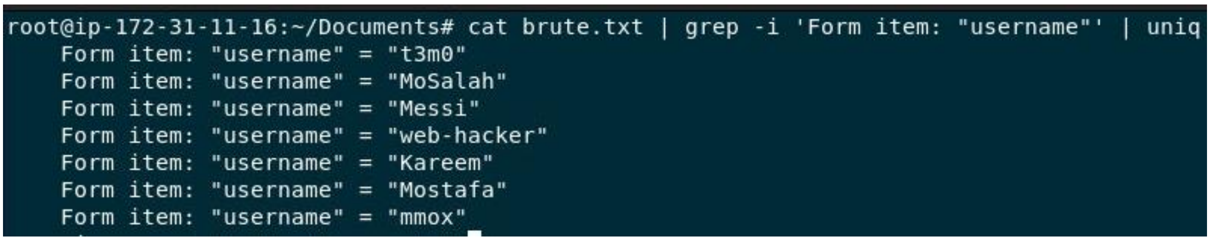
Save the file and call it whatever you want. Navigate to the director where you saved the file and enter the following command to search for new username entries:



```
cat brute.txt | grep -i 'Form item: "username"' | uniq | wc -l
```

```
7
```

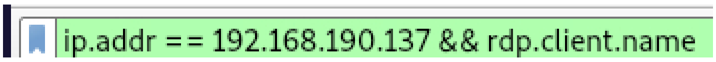
Or count manually:



```
root@ip-172-31-11-16:~/Documents# cat brute.txt | grep -i 'Form item: "username"' | uniq
    Form item: "username" = "t3m0"
    Form item: "username" = "MoSalah"
    Form item: "username" = "Messi"
    Form item: "username" = "web-hacker"
    Form item: "username" = "Kareem"
    Form item: "username" = "Mostafa"
    Form item: "username" = "mmox"
```

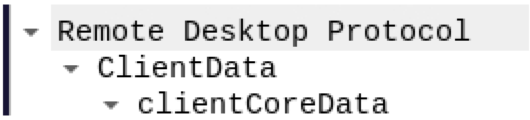
What is the "clientName" of the attacker's machine?

To find the clientName of the attacker's machine, we can use the following display filter which looks for the attackers IP address along with RDP traffic that contains the client name:

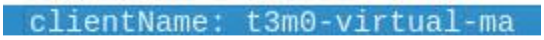


```
ip.addr == 192.168.190.137 && rdp.client.name
```

If you look at the packet details pane in any of these ClientData packets, we can see the clientName:



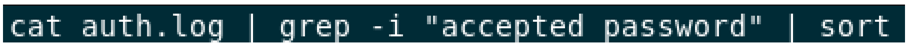
```
▾ Remote Desktop Protocol
  ▾ ClientData
    ▾ clientCoreData
```



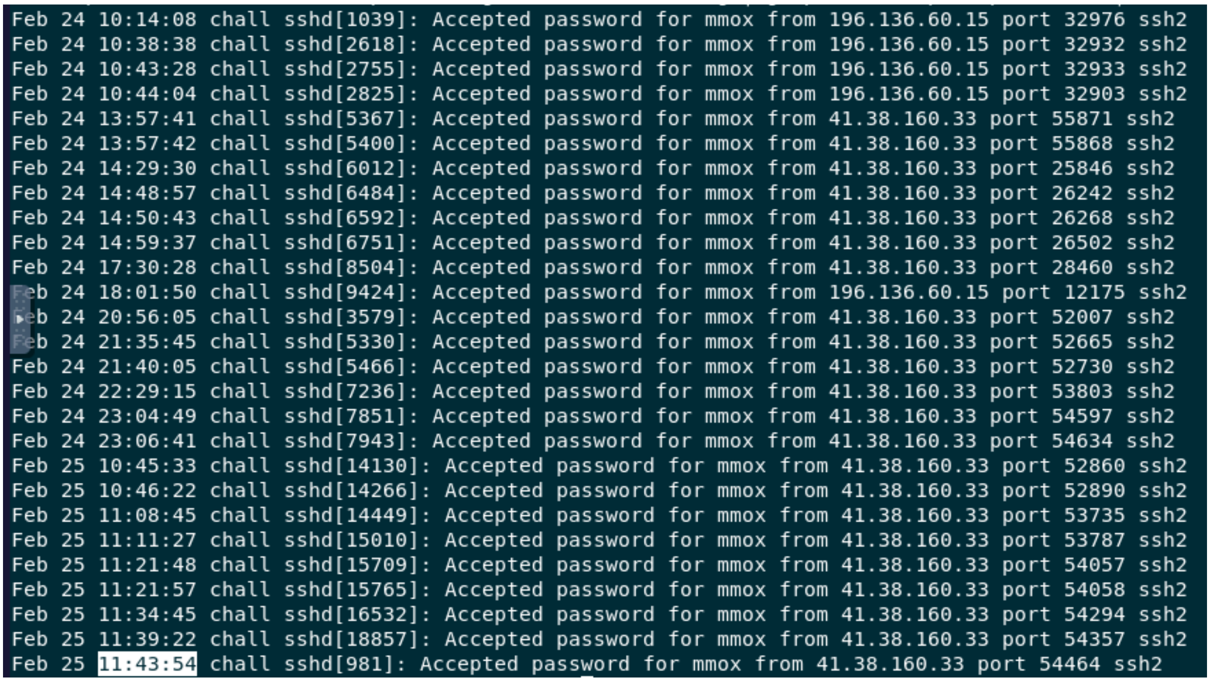
```
clientName: t3m0-virtual-ma
```

When did the user last successfully log in via SSH, and who was it?

To answer this question, we need to examine the auth.log file found in the ChallengeFile directory. We can grep for "accepted password" and sort the output to find the most recent successful login:



```
cat auth.log | grep -i "accepted password" | sort
```

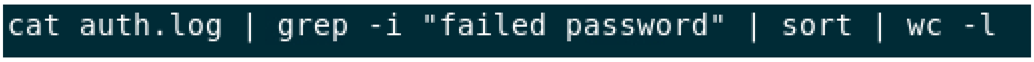


```
Feb 24 10:14:08 chall sshd[1039]: Accepted password for mmox from 196.136.60.15 port 32976 ssh2
Feb 24 10:38:38 chall sshd[2618]: Accepted password for mmox from 196.136.60.15 port 32932 ssh2
Feb 24 10:43:28 chall sshd[2755]: Accepted password for mmox from 196.136.60.15 port 32933 ssh2
Feb 24 10:44:04 chall sshd[2825]: Accepted password for mmox from 196.136.60.15 port 32903 ssh2
Feb 24 13:57:41 chall sshd[5367]: Accepted password for mmox from 41.38.160.33 port 55871 ssh2
Feb 24 13:57:42 chall sshd[5400]: Accepted password for mmox from 41.38.160.33 port 55868 ssh2
Feb 24 14:29:30 chall sshd[6012]: Accepted password for mmox from 41.38.160.33 port 25846 ssh2
Feb 24 14:48:57 chall sshd[6484]: Accepted password for mmox from 41.38.160.33 port 26242 ssh2
Feb 24 14:50:43 chall sshd[6592]: Accepted password for mmox from 41.38.160.33 port 26268 ssh2
Feb 24 14:59:37 chall sshd[6751]: Accepted password for mmox from 41.38.160.33 port 26502 ssh2
Feb 24 17:30:28 chall sshd[8504]: Accepted password for mmox from 41.38.160.33 port 28460 ssh2
Feb 24 18:01:50 chall sshd[9424]: Accepted password for mmox from 196.136.60.15 port 12175 ssh2
Feb 24 20:56:05 chall sshd[3579]: Accepted password for mmox from 41.38.160.33 port 52007 ssh2
Feb 24 21:35:45 chall sshd[5330]: Accepted password for mmox from 41.38.160.33 port 52665 ssh2
Feb 24 21:40:05 chall sshd[5466]: Accepted password for mmox from 41.38.160.33 port 52730 ssh2
Feb 24 22:29:15 chall sshd[7236]: Accepted password for mmox from 41.38.160.33 port 53803 ssh2
Feb 24 23:04:49 chall sshd[7851]: Accepted password for mmox from 41.38.160.33 port 54597 ssh2
Feb 24 23:06:41 chall sshd[7943]: Accepted password for mmox from 41.38.160.33 port 54634 ssh2
Feb 25 10:45:33 chall sshd[14130]: Accepted password for mmox from 41.38.160.33 port 52860 ssh2
Feb 25 10:46:22 chall sshd[14266]: Accepted password for mmox from 41.38.160.33 port 52890 ssh2
Feb 25 11:08:45 chall sshd[14449]: Accepted password for mmox from 41.38.160.33 port 53735 ssh2
Feb 25 11:11:27 chall sshd[15010]: Accepted password for mmox from 41.38.160.33 port 53787 ssh2
Feb 25 11:21:48 chall sshd[15709]: Accepted password for mmox from 41.38.160.33 port 54057 ssh2
Feb 25 11:21:57 chall sshd[15765]: Accepted password for mmox from 41.38.160.33 port 54058 ssh2
Feb 25 11:34:45 chall sshd[16532]: Accepted password for mmox from 41.38.160.33 port 54294 ssh2
Feb 25 11:39:22 chall sshd[18857]: Accepted password for mmox from 41.38.160.33 port 54357 ssh2
Feb 25 11:43:54 chall sshd[981]: Accepted password for mmox from 41.38.160.33 port 54464 ssh2
```

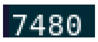
The answer is therefore mmox:11:43:54.

How many unsuccessful SSH connection attempts were made by the attacker?

All we need to do is count the number of log entries that contains "failed password". We can do this by entering:



```
cat auth.log | grep -i "failed password" | sort | wc -l
```



```
7480
```

What technique is used to gain access?

The technique being used is brute force, so if we search for this on MITRE ATT&CK we can see that the ATT&CK ID is T1110.

This was a really enjoyable room, and I believe it is suited towards both beginners and more experienced analysts. If you have a fundamental understanding of Wireshark, I believe that this room is a perfect starting point if you follow along with some sort of writeup.