Solution 1.

Whenever client requests for any page from web server, web server send backs the OK 200 response back to the sender. So, to answer number of web servers we need to know which unique server served the client on port 80. So, I used **http.response && tcp.port==80** filter in wireshark.

List on the left shows the unique IP addresses of the servers, which acknowledge users with response on port 80. The count of these servers is 36.

44.156.34.47
144.173.165.161
144.173.165.190
157.181.81.12
146.37.61.88
225.17.57.128
151.243.109.81
44.156.93.198
157.181.83.44
83.53.44.213
41.14.20.98
50.7.4.102
50.7.4.192
197.249.126.226
51.160.153.127
147.126.238.118
55.120.31.89
151.121.203.166
80.186.115.48
157.67.141.79
96.40.233.220
55.117.247.113
51.60.232.95
43.40.207.224
44.160.254.249
50.123.132.194
236.47.252.130
154.202.173.149
45.237.242.133
45.237.242.136
55.117.247.73
55.117.247.119
157.181.83.149
197.114.61.167
96.42.150.141

Solution 2.

In this case, we need to look to put filter as **http.request.** After that, we can store the input in some csv file, say **malicious_src.txt** . Then, we need to use the following command:

$cat malicious_src.txt | grep ../../../ > malicious.txt

The above command can input the data with the "../../../" string into malicious.txt file and then we can diagnose, which user tries to try this attack.

152.153.157.125 is the one, which was attempting this type of attack.

Solution 3.

Since we know the ftp response code for negative response is 5**, so using the following command, we get all responses which of the form 5**.

ftp.response.code>=500 && ftp.response.code<600.

The user tries using wrong password four times with username as Administrator. The **four** passwords are as: **volley, ashley, bear and calvin,** which returns 530 as code.

Moreover, the user also tries to login without using password, which the server returns 503 response message and it is tried for **3** times. This leads to total unsuccessful attempts to 7.

Following is the excerpt, from the reading.

"6192","418.402013","55.117.247.119","83.95.78.96","FTP","66","Response: 530- *** ERROR ***"

"6194","418.581329","55.117.247.119","83.95.78.96","FTP","209","Response: 530- Only anonymous FTP is available on ftp.ICSI.Berkeley.EDU."

"6200","419.126243","55.117.247.119","83.95.78.96","FTP","231","Response: 530- *** ERROR ***"

"6203","419.304313","55.117.247.119","83.95.78.96","FTP","72","Response: 503 Login with USER first."

"6237","423.433761","55.117.247.119","83.95.78.96","FTP","66","Response: 530- *** ERROR ***"

"6240","423.614470","55.117.247.119","83.95.78.96","FTP","209","Response: 530- Only anonymous FTP is available on ftp.ICSI.Berkeley.EDU."

"6250","424.150906","55.117.247.119","83.95.78.96","FTP","231","Response: 530- *** ERROR ***"

Solution 4.

For FTP: Since the ftp response for successful login is 230. So, using the filter as:

ftp.response==230, we get all the successful attempts. Moreover, checking ftp, we get to know there were two successful attempts whose credentials are as:

Username: calrules

Password: thisissosecure

Username: anonymous

Password: <Blank>

For Telnet: We just need to put filter as “telnet” and analyze the telnet part corresponding to each request. In this, the user tried the username as “root” after repeated backspaces and password as “w00t”, but this was unsuccessful attempt.

Solution 5. To find the Apache server with oldest version, we used the following filter in wireshark:

http.server contains “Apache”

Then analyzed the HTTP header to check the Apache version and found following:

96.40.233.220 server runs on **Apache version 1.3.28** which is the oldest version.

Solution 6.

DNS works on port 53 at the server side. So, using the following filter gets all the clients that used the DNS request.

udp.dstport==53

Then we need to analyze all the requests from different clients and check the port number used by the clients to launch the request. Based on the analysis, the following result is deduced:

The IP address that use same port are:

- 96.40.233.194 with port 32927
- 96.40.189.51 with port 32927

Third one is 80.141.67.234 with use different ports.

Moreover, 96.40.189.117 with port 33814 use single query.

Solution 7. Ping request is ICMP “Echo request” packets. So, using “icmp” will show all the ping requests and responses.

So, we get the following analysis:

Source 1: 83.39.156.1 Destination 1: 55.117.246.18

Source 2: 172.21.191.36 Destination 2: 62.97.163.225