

Cyber Security Coursework - Protocol Analysis

James Donohue - james.donohue@bbc.co.uk

Introduction

This is my report for the 'protocol analysis' part of the Cyber Security coursework assignment.

The third-party tools used in this report were:

- `file(1)` - command-line tool common on UNIX-based systems for identifying file types
- Wireshark v2.2.4 - graphical network protocol analyser

`capture1.pcap`

The `file` tool identifies this capture as follows:

```
tcpdump capture file (little-endian) - version 2.4 (Ethernet, capture length 65535)
```

From this we can learn that the capture was created on a little-endian processor (such as the Intel x86 family). The 'capture length' indicates that captured data was limited to 65535 bytes per packet (Lamping, Sharpe, and Warnicke 2014).

Opening the file in Wireshark shows it contains 44 packets. Using the *Statistics > Endpoints* report we can see the packets relate to 6 different Ethernet addresses and 5 IPv4 addresses.

Find the username & password from the HTTP Basic type of authentication

By using the `http` view filter in Wireshark to show only HTTP packets (see Figure 1) we can see two separate HTTP connections are made from host 192.168.0.3 to port 80 on host 192.168.0.1. The first of these connections contains a single request (`GET /`) which results in a `401 Access Denied` response.

The second connection (coloured in purple on Figure 1) contains two requests made using HTTP Basic authentication. The content of these requests can be seen most easily using the **Follow > TCP Stream** feature (see Figure 2).

Basic HTTP authentication, originally specified in HTTP/1.0, allows for simple challenge-response authentication of web clients using the `Authorization` header. The value after the scheme name `Basic` is a Base64 encoding of the username and password, separated by a colon. As no encryption is performed on the password, the Basic scheme is explicitly non-secure (Berners-Lee, Fielding, and Frystyk 1996). In this case the client sends the following header:

```
Authorization: Basic YWRtaW5pc3RyYXRvcjpwQHNzdzByZA==
```

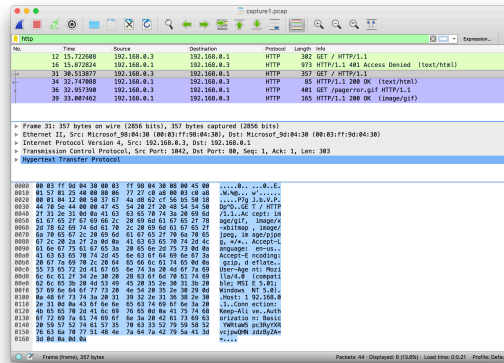


Figure 1: Wireshark filtering for only HTTP packets in capture1.pcap

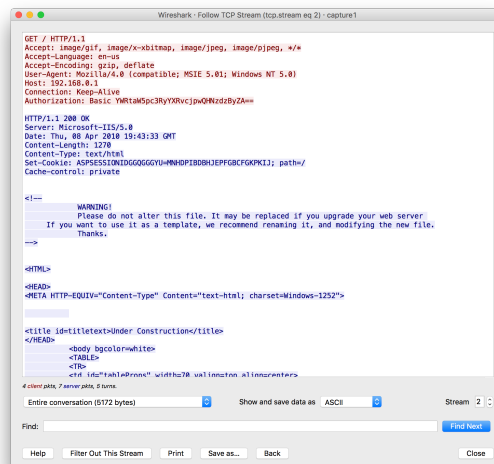


Figure 2: Wireshark Follow TCP Stream showing HTTP Basic authentication in capture1.pcap

The server responds with HTTP/1.1 200 OK, indicating that the authentication attempt was successful. The plaintext of the Base64 value can easily be determined using a trivial fragment of Python:

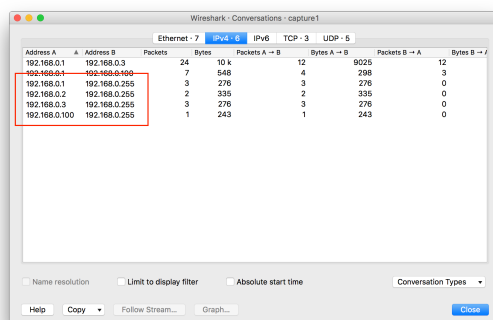
```
>>> import base64
>>> encoded = 'YWRtaW5pc3RyYXRvcjpwQHNzdzByZA=='
>>> base64.b64decode(encoded)
'administrator:p@ssw0rd'
```

Therefore the username sent was administrator and the password was p@ssw0rd.

Which hosts appear to be sending broadcast IP packets?

In IPv4, broadcast packets are those sent to a special address which causes them to be received by all hosts on a given subnet. The special address has all its unmasked bits set to 1 (Mogul 1984), which means ending in a sequence of one or more 255s when printed in typical ‘dotted quad’ form.

The Wireshark *Statistics > Conversations* view (shown in Figure 3) shows that four hosts are sending packets to 192.168.0.255, which is the broadcast address for the 192.168.0.0/24 network.



The image shows the Wireshark 'Conversations' view for capture1.pcap. The 'IPv4' tab is selected, and the 'UDP' protocol is chosen. The table displays conversations between hosts. The following table represents the data shown in the image, with broadcast traffic (to 192.168.0.255) highlighted by a red box.

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A
192.168.0.1	192.168.0.3	24	10 k	12	9025	12	0
192.168.0.1	192.168.0.100	7	548	4	288	3	0
192.168.0.1	192.168.0.255	3	276	3	276	0	0
192.168.0.2	192.168.0.255	2	335	2	335	0	0
192.168.0.3	192.168.0.255	3	276	3	276	0	0
192.168.0.100	192.168.0.255	1	243	1	243	0	0

Figure 3: Wireshark ‘Conversations’ view for capture1.pcap with broadcast traffic highlighted

The hosts that appear to be sending broadcast IP packets (via UDP) are therefore:

192.168.0.1
192.168.0.2
192.168.0.3
192.168.0.100

The packets are UDP datagrams on ports 137 and 138 (NetBIOS), meaning we can infer that these hosts are likely to be running Windows.

References

Berners-Lee, T., R. Fielding, and H. Frystyk. 1996. “RFC1945: Hypertext Transfer Protocol – HTTP/1.0.” <https://tools.ietf.org/search/rfc1945>.

Lamping, Ulf, Richard Sharpe, and Ed Warnicke. 2014. “Wireshark User’s Guide: Capturing with

Tcpdump for Viewing with Wireshark.” https://www.wireshark.org/docs/wsug_html_chunked/AppToolstcpdump.html.

Mogul, Jeffrey. 1984. “RFC919: Broadcasting Internet Datagrams.” <https://tools.ietf.org/html/rfc919>.