Fabricio Rua Sanchez and Miguel Alvarez

The connection between the host and the user begins as usual with a TCP handshake consisting of 3 packets. Frames 1,2,3 show the TCP handshake. This consists of an initial **[SYN]** request sent by the user to the host. Followed by a **[SYN, ACK]** response from the host to the user acknowledging the user's request for a connection. The user then sends another **[ACK]** request to acknowledge the server's acknowledgement.

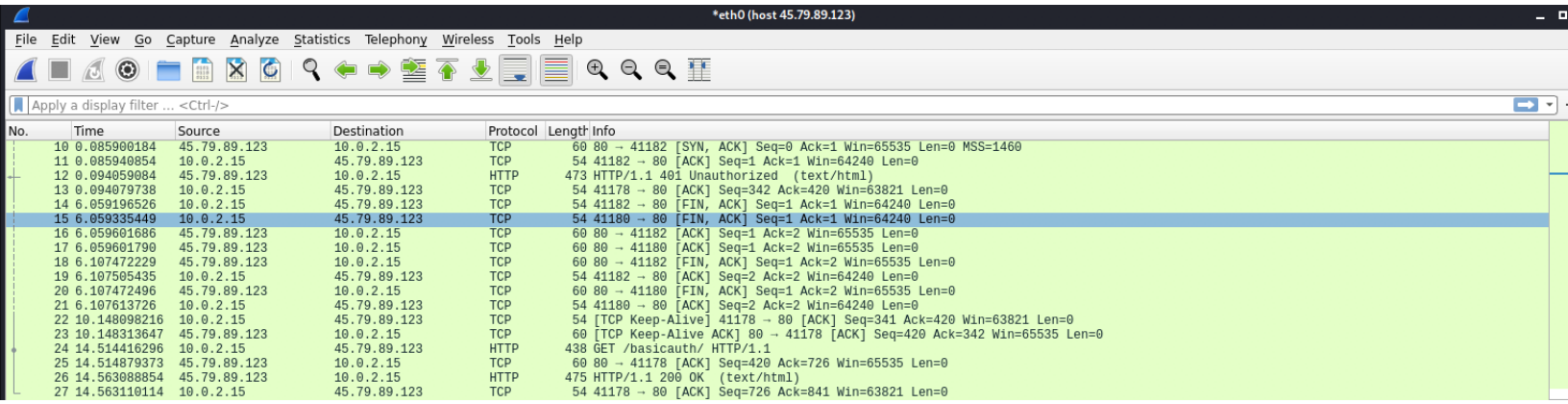
1	0.000000000	10.0.2.15	45.79.89.123	TCP	74	41270 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3670778802 TSecr=0 WS=128
2	0.047640500	45.79.89.123	10.0.2.15	TCP	60	80 → 41270 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
3	0.047680316	10.0.2.15	45.79.89.123	TCP	54	41270 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
4	0.098360216	10.0.2.15	45.79.89.123	HTTP	395	GET /basicauth/ HTTP/1.1
5	0.098701418	45.79.89.123	10.0.2.15	TCP	60	80 → 41270 [ACK] Seq=1 Ack=342 Win=65535 Len=0
6	0.146303737	45.79.89.123	10.0.2.15	HTTP	473	HTTP/1.1 401 Unauthorized (text/html)
7	0.146324238	10.0.2.15	45.79.89.123	TCP	54	41270 → 80 [ACK] Seq=342 Ack=420 Win=63821 Len=0
8	10.274530057	10.0.2.15	45.79.89.123	TCP	54	[TCP Keep-Alive] 41270 → 80 [ACK] Seq=341 Ack=420 Win=63821 Len=0
9	10.274754916	45.79.89.123	10.0.2.15	TCP	60	[TCP Keep-Alive ACK] 80 → 41270 [ACK] Seq=420 Ack=342 Win=65535 Len=0
10	18.947085101	10.0.2.15	45.79.89.123	HTTP	438	GET /basicauth/ HTTP/1.1
11	18.947582707	45.79.89.123	10.0.2.15	TCP	60	80 → 41270 [ACK] Seq=420 Ack=726 Win=65535 Len=0
12	18.996529501	45.79.89.123	10.0.2.15	HTTP	475	HTTP/1.1 200 OK (text/html)
13	18.996587790	10.0.2.15	45.79.89.123	TCP	54	41270 → 80 [ACK] Seq=726 Ack=841 Win=63821 Len=0
14	29.218244657	10.0.2.15	45.79.89.123	TCP	54	[TCP Keep-Alive] 41270 → 80 [ACK] Seq=725 Ack=841 Win=63821 Len=0
15	29.218398950	45.79.89.123	10.0.2.15	TCP	60	[TCP Keep-Alive ACK] 80 → 41270 [ACK] Seq=841 Ack=726 Win=65535 Len=0
16	39.491436454	10.0.2.15	45.79.89.123	TCP	54	[TCP Keep-Alive] 41270 → 80 [ACK] Seq=725 Ack=841 Win=63821 Len=0
17	39.491811901	45.79.89.123	10.0.2.15	TCP	60	[TCP Keep-Alive ACK] 80 → 41270 [ACK] Seq=841 Ack=726 Win=65535 Len=0
18	49.701818132	10.0.2.15	45.79.89.123	TCP	54	[TCP Keep-Alive] 41270 → 80 [ACK] Seq=725 Ack=841 Win=63821 Len=0

Once connection is established, there is an attempted **GET/basicauth/ HTTP/1.1** method that is sent by the user on line 4. This is the user requesting access to the website with an empty username:password credential. The host acknowledges this request on line 5 and then on line 6 the host responds to the user's credentials with a **401 status code.** This code is telling the user that they do not have authorized access to the website with their credentials. The user acknowledges this and begins typing in their credentials. Each time the user types in a character it is sent to the host which acknowledges the request.

After typing in the username and password, the user calls **GET /basicauth/ HTTP/1.1** after which the server sends a TCP packet acknowledging the request. On line 26, the host returns a **200 OK** status code signifying that the username and password entered have been accepted and

the user has gained access to the website. This is followed by a TCP packet from the user to the host acknowledging the information.

The username and password credentials are not checked for authorization by the browser instead they are sent to the host. The host does not encrypt this password but it is reformatted to fit this format: **username:password**. This credential is used to authorize the user's access to the website. It can be found in the line where the user sends **GET /basicauth/ HTTP/1.1** method.



No.	Time	Source	Destination	Protocol	Length	Info
10	0.085900184	45.79.89.123	10.0.2.15	TCP	60	80 → 41182 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
11	0.085940854	10.0.2.15	45.79.89.123	TCP	54	41182 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
12	0.094059084	45.79.89.123	10.0.2.15	HTTP	473	HTTP/1.1 401 Unauthorized (text/html)
13	0.094079738	10.0.2.15	45.79.89.123	TCP	54	41178 → 80 [ACK] Seq=342 Ack=420 Win=63821 Len=0
14	6.059196526	10.0.2.15	45.79.89.123	TCP	54	41182 → 80 [FIN, ACK] Seq=1 Ack=1 Win=64240 Len=0
15	6.059335449	10.0.2.15	45.79.89.123	TCP	54	41180 → 80 [FIN, ACK] Seq=1 Ack=1 Win=64240 Len=0
16	6.059601686	45.79.89.123	10.0.2.15	TCP	60	80 → 41182 [ACK] Seq=1 Ack=2 Win=65535 Len=0
17	6.059601790	45.79.89.123	10.0.2.15	TCP	60	80 → 41180 [ACK] Seq=1 Ack=2 Win=65535 Len=0
18	6.107472229	45.79.89.123	10.0.2.15	TCP	60	80 → 41182 [FIN, ACK] Seq=1 Ack=2 Win=65535 Len=0
19	6.107505435	10.0.2.15	45.79.89.123	TCP	54	41182 → 80 [ACK] Seq=2 Ack=2 Win=64240 Len=0
20	6.107472496	45.79.89.123	10.0.2.15	TCP	60	80 → 41180 [FIN, ACK] Seq=1 Ack=2 Win=65535 Len=0
21	6.107613726	10.0.2.15	45.79.89.123	TCP	54	41180 → 80 [ACK] Seq=2 Ack=2 Win=64240 Len=0
22	10.148098216	10.0.2.15	45.79.89.123	TCP	54	[TCP Keep-Alive] 41178 → 80 [ACK] Seq=341 Ack=420 Win=63821 Len=0
23	10.148313647	45.79.89.123	10.0.2.15	TCP	60	[TCP Keep-Alive ACK] 80 → 41178 [ACK] Seq=420 Ack=342 Win=65535 Len=0
24	14.514416296	10.0.2.15	45.79.89.123	HTTP	438	GET /basicauth/ HTTP/1.1
25	14.514879373	45.79.89.123	10.0.2.15	TCP	60	80 → 41178 [ACK] Seq=420 Ack=726 Win=65535 Len=0
26	14.563088854	45.79.89.123	10.0.2.15	HTTP	475	HTTP/1.1 200 OK (text/html)
27	14.563110114	10.0.2.15	45.79.89.123	TCP	54	41178 → 80 [ACK] Seq=726 Ack=841 Win=63821 Len=0

In the HTTP specification document section 4.3.1 on page 23 it talks about the **GET** method. This is the main method for the retrieval of information in HTTP.In this case the user uses the **GET** method to acquire information about their credentials. The website has a file stored with sets of usernames and passwords, when the users calls **GET /basicauth/ HTTP/1.1** they are validating their username and password by retrieving and comparing their credentials with the stored set of authorized credentials. If the credentials do not match any of the authorized credentials the the user receives a **401** status code. In the HTTP 7235 document section 3.1 it states that status code 401 "indicates that the request has not been applied because it lacks valid authentication credentials for the target resource." In other words, the host is telling the user that

they will not have access to the web page with the given credentials. However, if the user's credentials are authorized then the host will send a **200** status code which "indicates that the request has succeeded" as stated in the HTTP 7231 document section 6.3.1.

The username and password credentials are not checked for authorization by the browser instead they are sent to the host. The host does not encrypt this password but it is reformatted to fit this format: **username:password**. This credential is used to authorize the user's access to the website. It can be found in the line where the user sends **GET /basicauth/ HTTP/1.1** method.

```
24 14.514416296  10.0.2.15      45.79.89.123   HTTP   438 GET /basicauth/ HTTP/1.1
25 14.514879373  45.79.89.123   10.0.2.15      TCP     60 80 → 41178 [ACK] Seq=420 Ack=726 Win=65535 Len=0
26 14.563088854  45.79.89.123   10.0.2.15      HTTP   475 HTTP/1.1 200 OK  (text/html)
27 14.563110114  10.0.2.15      45.79.89.123   TCP     54 41178 → 80 [ACK] Seq=726 Ack=841 Win=63821 Len=0

▾ [Timestamps]
    [Time since first frame in this TCP stream: 14.514416296 seconds]
    [Time since previous frame in this TCP stream: 4.366102649 seconds]
  TCP payload (384 bytes)
▾ Hypertext Transfer Protocol
  ▸ GET /basicauth/ HTTP/1.1\r\n
    Host: cs231.jeffondich.com\r\n
    User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8\r\n
    Accept-Language: en-US,en;q=0.5\r\n
    Accept-Encoding: gzip, deflate\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
  ▾ Authorization: Basic Y3MyMzE6cGFzc3dvcmQ=\r\n
      Credentials: cs231:password
    \r\n
    [Full request URI: http://cs231.jeffondich.com/basicauth/]
    [HTTP request 2/2]
    [Prev request in frame: 6]

0030  f9 4d 94 73 00 00 47 45  54 20 2f 62 61 73 69 63   ·M·s··GE T /
0040  61 75 74 68 2f 20 48 54  54 50 2f 31 2e 31 0d 0a   auth/ HT TP/
0050  48 6f 73 74 3a 20 63 73  32 33 31 2e 6a 65 66 66   Host: cs
0060  6f 6e 64 69 63 68 2e 63  6f 6d 0d 0a 55 73 65 72   ondich.c
0070  2d 41 67 65 6e 74 3a 20  4d 6f 7a 69 6c 6c 61 2f   -Agent:
0080  35 2e 30 20 28 58 31 31  3b 20 4c 69 6e 75 78 20   5.0 (X11 ; Linux
0090  78 38 36 5f 36 34 3b 20  72 76 3a 37 38 2e 30 29   x86_64;  rv:
00a0  20 47 65 63 6b 6f 2f 32  30 31 30 30 31 30 31 20    Gecko/2 0100101
00b0  46 69 72 65 66 6f 78 2f  37 38 2e 30 0d 0a 41 63   Firefox/
00c0  63 65 70 74 3a 20 74 65  78 74 2f 68 74 6d 6c 2c   cept: te xt/
00d0  61 70 70 6c 69 63 61 74  69 6f 6e 2f 78 68 74 6d   applicat ion/
00e0  6c 2b 78 6d 6c 2c 61 70  70 6c 69 63 61 74 69 6f   l+xml,ap
00f0  6e 2f 78 6d 6c 3b 71 3d  30 2e 39 2c 69 6d 61 67   n/xml;q=
0100  65 2f 77 65 62 70 2c 2a  2f 2a 3b 71 3d 30 2e 38   e/webp,* /
0110  0d 0a 41 63 63 65 70 74  2d 4c 61 6e 67 75 61 67   ··Accept -
0120  65 3a 20 65 6e 2d 55 53  2c 65 6e 3b 71 3d 30 2e   e: en-US
0130  35 0d 0a 41 63 63 65 70  74 2d 45 6e 63 6f 64 69   5··Accep t-
0140  6e 67 3a 20 67 7a 69 70  2c 20 64 65 66 6c 61 74   ng: gzip ,
0150  65 0d 0a 43 6f 6e 6e 65  63 74 69 6f 6e 3a 20 6b   e··Conne ction:
```

Once the user has access to the website, they are free to view all the files attached with it. The user can access one of the three txt files shown on the website **amateurs.txt**, **concrete.txt**, and **pigs.txt.** Each time one of these files is accessed the user and the host do another 3-step TCP handshake. An example of this can be seen in lines 44-46. When the user leaves to return to the initial page the connection is then ended with a TCP disconnection where the user sends a **[FIN]** request to the host and the host responds with **[FIN, ACK]** ending with the user sending a **[ACK]** request.

39	90.588160134	10.0.2.15	45.79.89.123	TCP	54	41274 → 80 [FIN, ACK] Seq=1 Ack=1 Win=64240 Len=0
40	90.588664991	45.79.89.123	10.0.2.15	TCP	60	80 → 41274 [ACK] Seq=1 Ack=2 Win=65535 Len=0
41	90.636471743	45.79.89.123	10.0.2.15	TCP	60	80 → 41274 [FIN, ACK] Seq=1 Ack=2 Win=65535 Len=0
42	90.636501111	10.0.2.15	45.79.89.123	TCP	54	41274 → 80 [ACK] Seq=2 Ack=2 Win=64240 Len=0
43	100.134038232	10.0.2.15	45.79.89.123	TCP	74	41276 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3670878936 TSecr=0 WS=128
44	100.135981621	10.0.2.15	45.79.89.123	TCP	74	41278 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3670878938 TSecr=0 WS=128
45	100.181710213	45.79.89.123	10.0.2.15	TCP	60	80 → 41276 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
46	100.181745093	10.0.2.15	45.79.89.123	TCP	54	41276 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
47	100.183678280	45.79.89.123	10.0.2.15	TCP	60	80 → 41278 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
48	100.183715503	10.0.2.15	45.79.89.123	TCP	54	41278 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
49	105.193148402	10.0.2.15	45.79.89.123	TCP	54	41278 → 80 [FIN, ACK] Seq=1 Ack=1 Win=64240 Len=0
50	105.193305155	10.0.2.15	45.79.89.123	TCP	54	41276 → 80 [FIN, ACK] Seq=1 Ack=1 Win=64240 Len=0
51	105.193541954	45.79.89.123	10.0.2.15	TCP	60	80 → 41278 [ACK] Seq=1 Ack=2 Win=65535 Len=0
52	105.193542068	45.79.89.123	10.0.2.15	TCP	60	80 → 41276 [ACK] Seq=1 Ack=2 Win=65535 Len=0
53	105.241846341	45.79.89.123	10.0.2.15	TCP	60	80 → 41278 [FIN, ACK] Seq=1 Ack=2 Win=65535 Len=0
54	105.241882465	10.0.2.15	45.79.89.123	TCP	54	41278 → 80 [ACK] Seq=2 Ack=2 Win=64240 Len=0
55	105.241846473	45.79.89.123	10.0.2.15	TCP	60	80 → 41276 [FIN, ACK] Seq=1 Ack=2 Win=65535 Len=0
56	105.242057992	10.0.2.15	45.79.89.123	TCP	54	41276 → 80 [ACK] Seq=2 Ack=2 Win=64240 Len=0