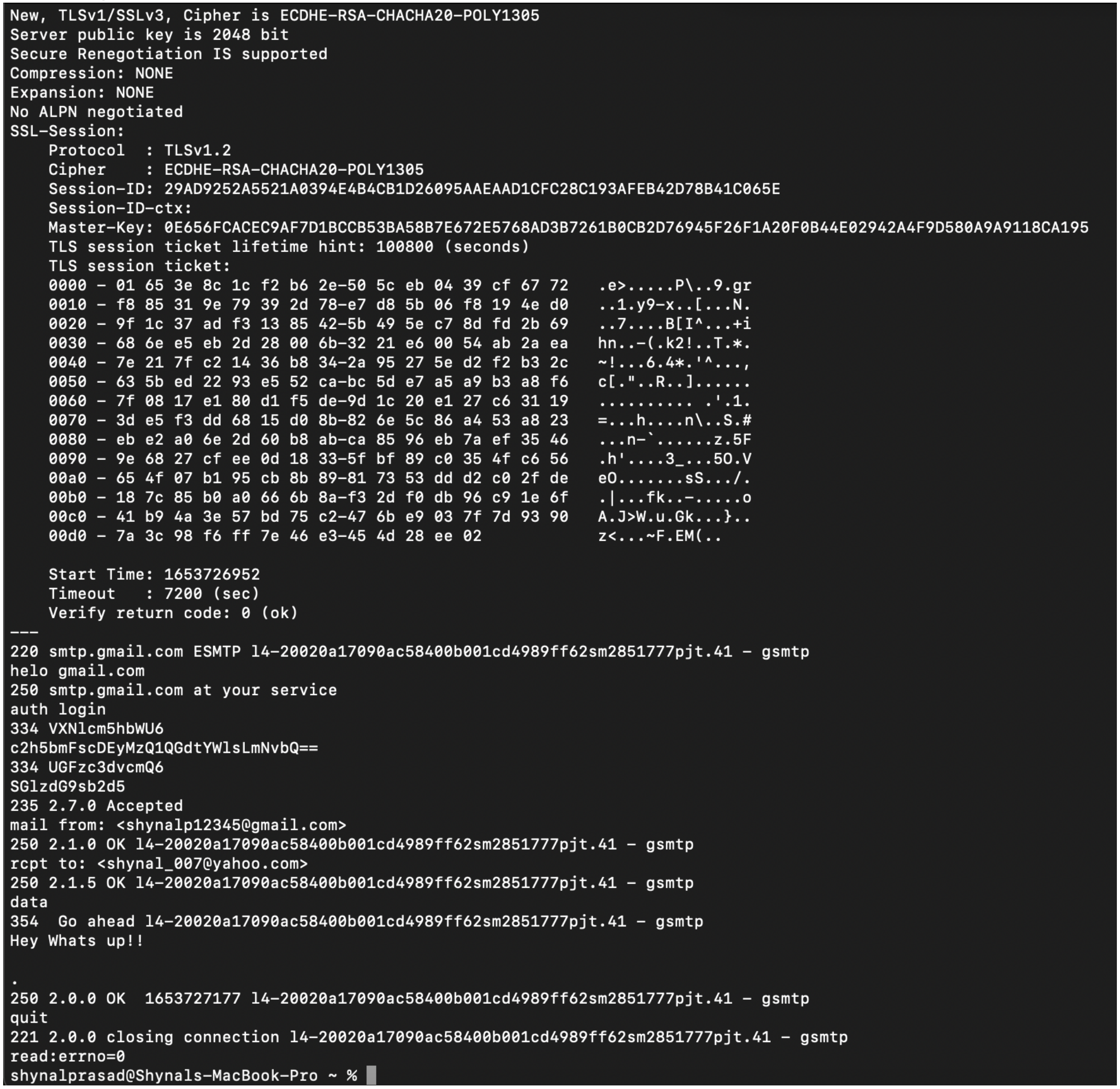
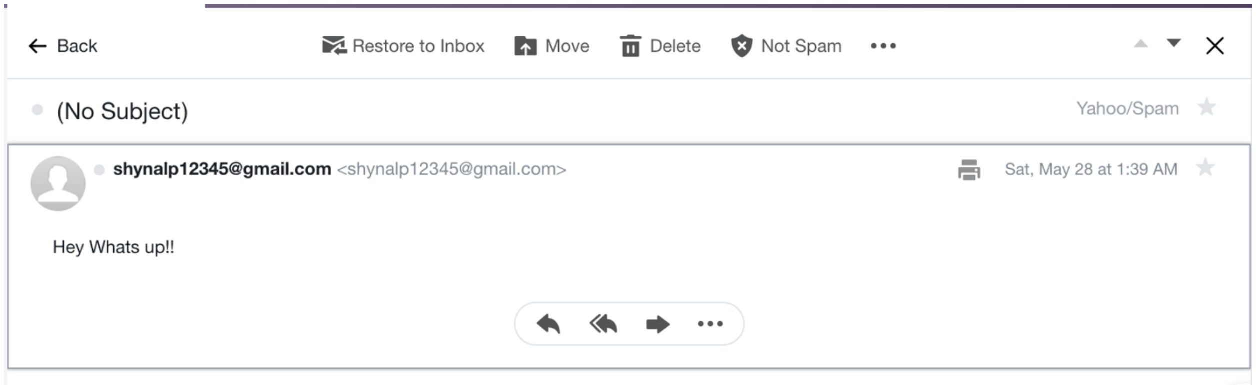
1 sending email and SMTP



```
New, TLSv1/SSLv3, Cipher is ECDHE-RSA-CHACHA20-POLY1305
Server public key is 2048 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
No ALPN negotiated
SSL-Session:
    Protocol  : TLSv1.2
    Cipher    : ECDHE-RSA-CHACHA20-POLY1305
    Session-ID: 29AD9252A5521A0394E4B4CB1D26095AAEAAD1CFC28C193AFEB42D78B41C065E
    Session-ID-ctx:
    Master-Key: 0E656FCACEC9AF7D1BCCB53BA58B7E672E5768AD3B7261B0CB2D76945F26F1A20F0B44E02942A4F9D580A9A9118CA195
    TLS session ticket lifetime hint: 100800 (seconds)
    TLS session ticket:
    0000 - 01 65 3e 8c 1c f2 b6 2e-50 5c eb 04 39 cf 67 72   .e>.....P\..9.gr
    0010 - f8 85 31 9e 79 39 2d 78-e7 d8 5b 06 f8 19 4e d0   ..1.y9-x..[...N.
    0020 - 9f 1c 37 ad f3 13 85 42-5b 49 5e c7 8d fd 2b 69   ..7....B[I^...+i
    0030 - 68 6e e5 eb 2d 28 00 6b-32 21 e6 00 54 ab 2a ea   hn..-(.k2!..T.*.
    0040 - 7e 21 7f c2 14 36 b8 34-2a 95 27 5e d2 f2 b3 2c   ~!...6.4*.'^...,
    0050 - 63 5b ed 22 93 e5 52 ca-bc 5d e7 a5 a9 b3 a8 f6   c[."..R..]......
    0060 - 7f 08 17 e1 80 d1 f5 de-9d 1c 20 e1 27 c6 31 19   .......... .'.1.
    0070 - 3d e5 f3 dd 68 15 d0 8b-82 6e 5c 86 a4 53 a8 23   =...h....n\..S.#
    0080 - eb e2 a0 6e 2d 60 b8 ab-ca 85 96 eb 7a ef 35 46   ...n-`......z.5F
    0090 - 9e 68 27 cf ee 0d 18 33-5f bf 89 c0 35 4f c6 56   .h'....3_...5O.V
    00a0 - 65 4f 07 b1 95 cb 8b 89-81 73 53 dd d2 c0 2f de   eO.......sS.../.
    00b0 - 18 7c 85 b0 a0 66 6b 8a-f3 2d f0 db 96 c9 1e 6f   .|...fk..-.....o
    00c0 - 41 b9 4a 3e 57 bd 75 c2-47 6b e9 03 7f 7d 93 90   A.J>W.u.Gk...}..
    00d0 - 7a 3c 98 f6 ff 7e 46 e3-45 4d 28 ee 02            z<...~F.EM(..

    Start Time: 1653726952
    Timeout   : 7200 (sec)
    Verify return code: 0 (ok)
---
220 smtp.gmail.com ESMTP l4-20020a17090ac58400b001cd4989ff62sm2851777pjt.41 - gsmtp
helo gmail.com
250 smtp.gmail.com at your service
auth login
334 VXNlcm5hbWU6
c2h5bmFscDEyMzQ1QGdtYWlsLmNvbQ==
334 UGFzc3dvcmQ6
SGlzdG9sb2d5
235 2.7.0 Accepted
mail from: <shynalp12345@gmail.com>
250 2.1.0 OK l4-20020a17090ac58400b001cd4989ff62sm2851777pjt.41 - gsmtp
rcpt to: <shynal_007@yahoo.com>
250 2.1.5 OK l4-20020a17090ac58400b001cd4989ff62sm2851777pjt.41 - gsmtp
data
354  Go ahead l4-20020a17090ac58400b001cd4989ff62sm2851777pjt.41 - gsmtp
Hey Whats up!!

.
250 2.0.0 OK  1653727177 l4-20020a17090ac58400b001cd4989ff62sm2851777pjt.41 - gsmtp
quit
221 2.0.0 closing connection l4-20020a17090ac58400b001cd4989ff62sm2851777pjt.41 - gsmtp
read:errno=0
shynalprasad@Shynals-MacBook-Pro ~ %
```



(No Subject)

shynalp12345@gmail.com <shynalp12345@gmail.com> Sat, May 28 at 1:39 AM

Hey Whats up!!

1. What filter did you use to catch the traffic? Explain why you chose that filter.
   tcp filter, because we can use the port 465 to filter

2. What is the standard port for SMTP and why do we use port 465 in the example above?
   Port 25 is the standard port for SMTP. Port 465 is used for implicit TLS and can be used to facilitate secure communications for mail services.

3. Explain each line used in the command line, what it does, and why it is needed.

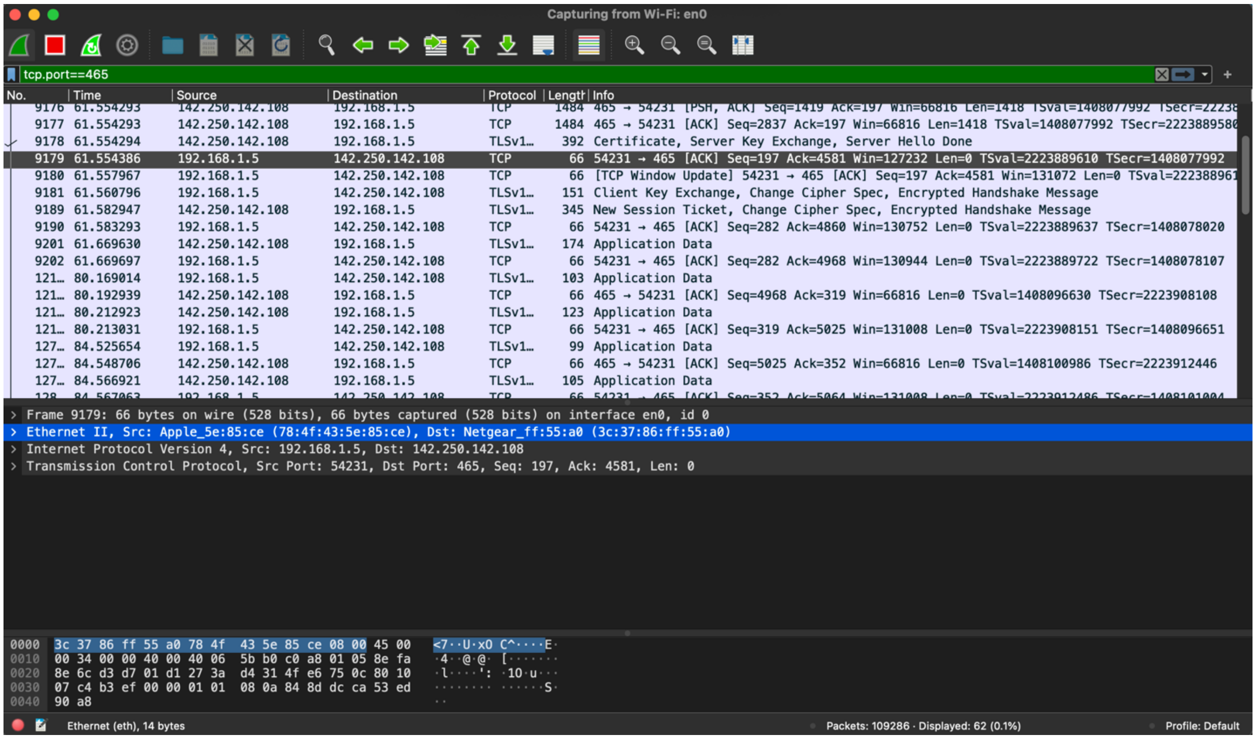
```
echo -ne shynalp12345@gmail.com | base64
--Convert the username to base64 encryption
echo -ne Histology | base64
--Convert the Password to base64 encryption
openssl s_client -connect smtp.gmail.com:465 -crlf
--to do transaction with gmail server
helo gmail.com
--to identify the gmail server
auth login
--authenticate
c2h5bmFscDEyMzQ1QGdtYWlsLmNvbQ==
--base 64 encrypted username
SGlzdG9sb2d5
--base 64 encrypted password
mail from: <mail.com>
--mail sent from this email address
rcpt to: <mail.com>
--mail received from this email address
data
--to enter the subject and body of the email
Quit
close the connection
```

4. How much back and forth communication do you see for establishing the connection?
   I can see 17 times back and forth communication for establishing the connection.

5. What is the port your local machine is using between sending the two emails when communicating with the SMTP server?
   Port: 58444

6. Explain who sends the first FIN flag and how the quitting process works.
   Port 465 sends FIN flag to port 58444 and vice versa when quitting the process at the end of closing connection. The “quit” command let the connection knows that I don’t have more and close the connection

7. Add a screenshot of your Wireshark output and add it to your document.



Capturing from Wi-Fi: en0

tcp.port==465

```
No.    Time       Source           Destination      Protocol Length Info
9176   61.554293  142.250.142.108  192.168.1.5      TCP      1484   465 → 54231 [PSH, ACK] Seq=1419 Ack=197 Win=66816 Len=1418 TSval=1408077992 TSecr=22238
9177   61.554293  142.250.142.108  192.168.1.5      TCP      1484   465 → 54231 [ACK] Seq=2837 Ack=197 Win=66816 Len=1418 TSval=1408077992 TSecr=222388958
9178   61.554294  142.250.142.108  192.168.1.5      TLSv1…    392   Certificate, Server Key Exchange, Server Hello Done
9179   61.554386  192.168.1.5      142.250.142.108  TCP        66   54231 → 465 [ACK] Seq=197 Ack=4581 Win=127232 Len=0 TSval=2223889610 TSecr=1408077992
9180   61.557967  192.168.1.5      142.250.142.108  TCP        66   [TCP Window Update] 54231 → 465 [ACK] Seq=197 Ack=4581 Win=131072 Len=0 TSval=22238896
9181   61.560796  192.168.1.5      142.250.142.108  TLSv1…    151   Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
9189   61.582947  142.250.142.108  192.168.1.5      TLSv1…    345   New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
9190   61.583293  192.168.1.5      142.250.142.108  TCP        66   54231 → 465 [ACK] Seq=282 Ack=4860 Win=130752 Len=0 TSval=2223889637 TSecr=1408078020
9201   61.669630  142.250.142.108  192.168.1.5      TLSv1…    174   Application Data
9202   61.669697  192.168.1.5      142.250.142.108  TCP        66   54231 → 465 [ACK] Seq=282 Ack=4968 Win=130944 Len=0 TSval=2223889722 TSecr=1408078107
121…   80.169014  192.168.1.5      142.250.142.108  TLSv1…    103   Application Data
121…   80.192939  142.250.142.108  192.168.1.5      TCP        66   465 → 54231 [ACK] Seq=4968 Ack=319 Win=66816 Len=0 TSval=1408096630 TSecr=2223908108
121…   80.212923  142.250.142.108  192.168.1.5      TLSv1…    123   Application Data
121…   80.213031  192.168.1.5      142.250.142.108  TCP        66   54231 → 465 [ACK] Seq=319 Ack=5025 Win=131008 Len=0 TSval=2223908151 TSecr=1408096651
127…   84.525654  192.168.1.5      142.250.142.108  TLSv1…     99   Application Data
127…   84.548706  142.250.142.108  192.168.1.5      TCP        66   465 → 54231 [ACK] Seq=5025 Ack=352 Win=66816 Len=0 TSval=1408100986 TSecr=2223912446
127…   84.566921  142.250.142.108  192.168.1.5      TLSv1…    105   Application Data
128…   84.567063  192.168.1.5      142.250.142.108  TCP        66   54231 → 465 [ACK] Seq=352 Ack=5064 Win=131008 Len=0 TSval=2223912486 TSecr=1408101004

> Frame 9179: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface en0, id 0
> Ethernet II, Src: Apple_5e:85:ce (78:4f:43:5e:85:ce), Dst: Netgear_ff:55:a0 (3c:37:86:ff:55:a0)
> Internet Protocol Version 4, Src: 192.168.1.5, Dst: 142.250.142.108
> Transmission Control Protocol, Src Port: 54231, Dst Port: 465, Seq: 197, Ack: 4581, Len: 0

0000  3c 37 86 ff 55 a0 78 4f  43 5e 85 ce 08 00 45 00   <7··U·xO C^····E·
0010  00 34 00 00 40 00 40 06  5b b0 c0 a8 01 05 8e fa   ·4··@·@· [·······
0020  8e 6c d3 d7 01 d1 27 3a  d4 31 4f e6 75 0c 80 10   ·l····': ·1O·u···
0030  07 c4 b3 ef 00 00 01 01  08 0a 84 8d dc ca 53 ed   ········ ······S·
0040  90 a8                                              ··
```

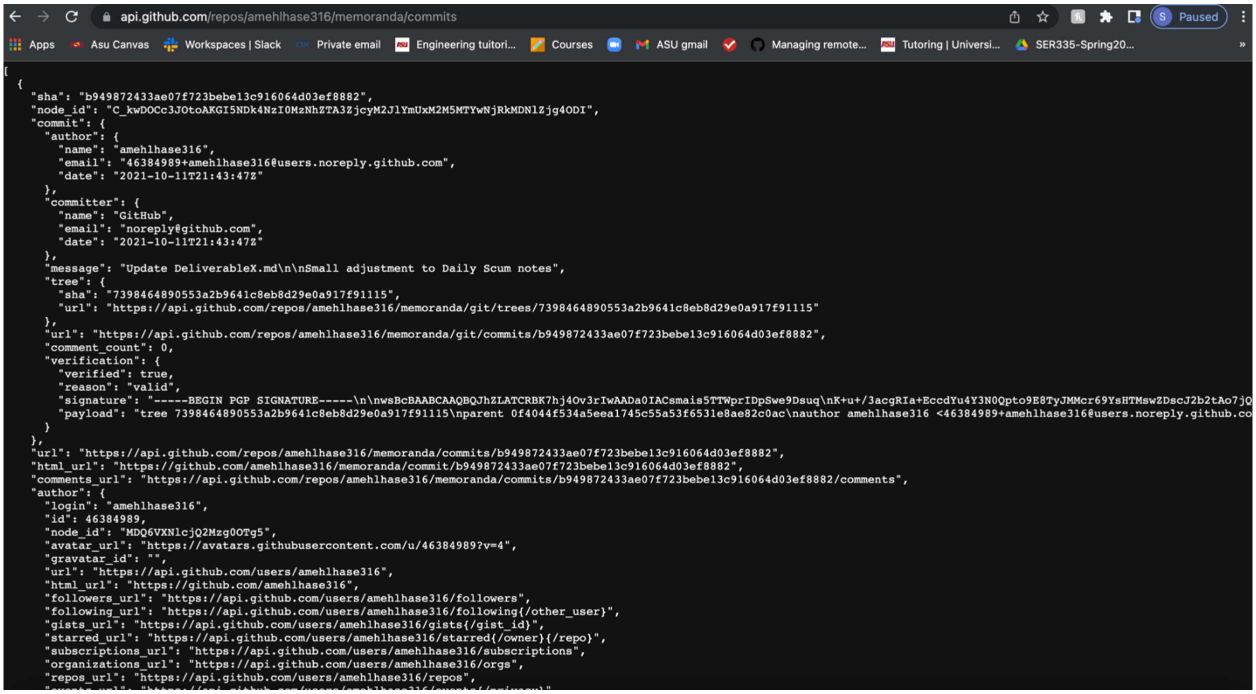
Ethernet (eth), 14 bytes

Packets: 109286 · Displayed: 62 (0.1%)

Profile: Default

2.. Understanding HTTP

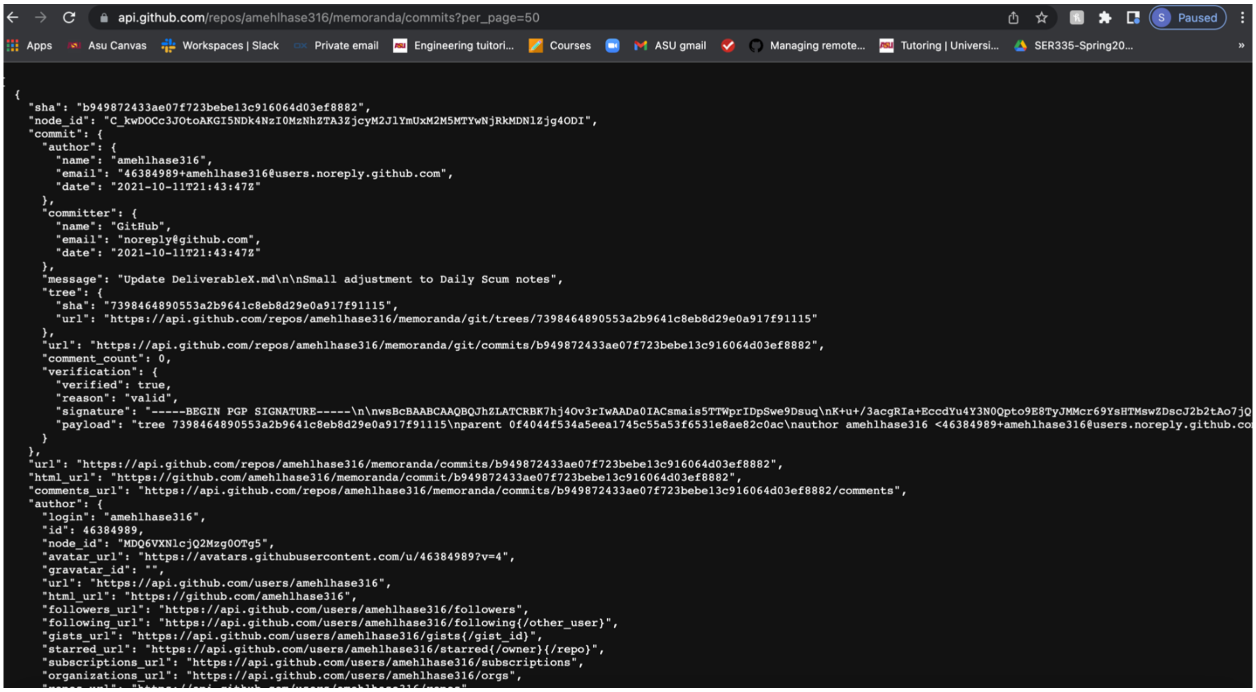
API call



api.github.com/repos/amehlhase316/memoranda/commits

```
[
  {
    "sha": "b949872433ae07f723bebe13c916064d03ef8882",
    "node_id": "C_kwDOCc3JOtoAKGI5NDk4NzI0MzNhZTA3ZjcyM2JlYmUxM2M5MTYwNjRkMDNlZjg4ODI",
    "commit": {
      "author": {
        "name": "amehlhase316",
        "email": "46384989+amehlhase316@users.noreply.github.com",
        "date": "2021-10-11T21:43:47Z"
      },
      "committer": {
        "name": "GitHub",
        "email": "noreply@github.com",
        "date": "2021-10-11T21:43:47Z"
      },
      "message": "Update DeliverableX.md\n\nSmall adjustment to Daily Scum notes",
      "tree": {
        "sha": "7398464890553a2b9641c8eb8d29e0a917f91115",
        "url": "https://api.github.com/repos/amehlhase316/memoranda/git/trees/7398464890553a2b9641c8eb8d29e0a917f91115"
      },
      "url": "https://api.github.com/repos/amehlhase316/memoranda/git/commits/b949872433ae07f723bebe13c916064d03ef8882",
      "comment_count": 0,
      "verification": {
        "verified": true,
        "reason": "valid",
        "signature": "-----BEGIN PGP SIGNATURE-----\n\nwsBcBAABCAAQBQJhZLATCRBK7hj4Ov3rIwAADa0IACsmais5TTWprIDpSwe9Dsuq\nK+u+/3acgRIa+EccdYu4Y3N0Qpto9E8TyJMMcr69YsHTMswZDscJ2b2tAo7jQ
        "payload": "tree 7398464890553a2b9641c8eb8d29e0a917f91115\nparent 0f4044f534a5eea1745c55a53f6531e8ae82c0ac\nauthor amehlhase316 <46384989+amehlhase316@users.noreply.github.co
      }
    },
    "url": "https://api.github.com/repos/amehlhase316/memoranda/commits/b949872433ae07f723bebe13c916064d03ef8882",
    "html_url": "https://github.com/amehlhase316/memoranda/commit/b949872433ae07f723bebe13c916064d03ef8882",
    "comments_url": "https://api.github.com/repos/amehlhase316/memoranda/commits/b949872433ae07f723bebe13c916064d03ef8882/comments",
    "author": {
      "login": "amehlhase316",
      "id": 46384989,
      "node_id": "MDQ6VXNlcjQ2Mzg0OTg5",
      "avatar_url": "https://avatars.githubusercontent.com/u/46384989?v=4",
      "gravatar_id": "",
      "url": "https://api.github.com/users/amehlhase316",
      "html_url": "https://github.com/amehlhase316",
      "followers_url": "https://api.github.com/users/amehlhase316/followers",
      "following_url": "https://api.github.com/users/amehlhase316/following{/other_user}",
      "gists_url": "https://api.github.com/users/amehlhase316/gists{/gist_id}",
      "starred_url": "https://api.github.com/users/amehlhase316/starred{/owner}{/repo}",
      "subscriptions_url": "https://api.github.com/users/amehlhase316/subscriptions",
      "organizations_url": "https://api.github.com/users/amehlhase316/orgs",
      "repos_url": "https://api.github.com/users/amehlhase316/repos",
```

API call page=50



api.github.com/repos/amehlhase316/memoranda/commits?per_page=50

```
[
  {
    "sha": "b949872433ae07f723bebe13c916064d03ef8882",
    "node_id": "C_kwDOCc3JOtoAKGI5NDk4NzI0MzNhZTA3ZjcyM2JlYmUxM2M5MTYwNjRkMDNlZjg4ODI",
    "commit": {
      "author": {
        "name": "amehlhase316",
        "email": "46384989+amehlhase316@users.noreply.github.com",
        "date": "2021-10-11T21:43:47Z"
      },
      "committer": {
        "name": "GitHub",
        "email": "noreply@github.com",
        "date": "2021-10-11T21:43:47Z"
      },
      "message": "Update DeliverableX.md\n\nSmall adjustment to Daily Scum notes",
      "tree": {
        "sha": "7398464890553a2b9641c8eb8d29e0a917f91115",
        "url": "https://api.github.com/repos/amehlhase316/memoranda/git/trees/7398464890553a2b9641c8eb8d29e0a917f91115"
      },
      "url": "https://api.github.com/repos/amehlhase316/memoranda/git/commits/b949872433ae07f723bebe13c916064d03ef8882",
      "comment_count": 0,
      "verification": {
        "verified": true,
        "reason": "valid",
        "signature": "-----BEGIN PGP SIGNATURE-----\n\nwsBcBAABCAAQBQJhZLATCRBK7hj4Ov3rIwAADa0IACsmais5TTWprIDpSwe9Dsuq\nK+u+/3acgRIa+EccdYu4Y3N0Qpto9E8TyJMMcr69YsHTMswZDscJ2b2tAo7jQ
        "payload": "tree 7398464890553a2b9641c8eb8d29e0a917f91115\nparent 0f4044f534a5eea1745c55a53f6531e8ae82c0ac\nauthor amehlhase316 <46384989+amehlhase316@users.noreply.github.co
      }
    },
    "url": "https://api.github.com/repos/amehlhase316/memoranda/commits/b949872433ae07f723bebe13c916064d03ef8882",
    "html_url": "https://github.com/amehlhase316/memoranda/commit/b949872433ae07f723bebe13c916064d03ef8882",
    "comments_url": "https://api.github.com/repos/amehlhase316/memoranda/commits/b949872433ae07f723bebe13c916064d03ef8882/comments",
    "author": {
      "login": "amehlhase316",
      "id": 46384989,
      "node_id": "MDQ6VXNlcjQ2Mzg0OTg5",
      "avatar_url": "https://avatars.githubusercontent.com/u/46384989?v=4",
      "gravatar_id": "",
      "url": "https://api.github.com/users/amehlhase316",
      "html_url": "https://github.com/amehlhase316",
      "followers_url": "https://api.github.com/users/amehlhase316/followers",
      "following_url": "https://api.github.com/users/amehlhase316/following{/other_user}",
      "gists_url": "https://api.github.com/users/amehlhase316/gists{/gist_id}",
      "starred_url": "https://api.github.com/users/amehlhase316/starred{/owner}{/repo}",
      "subscriptions_url": "https://api.github.com/users/amehlhase316/subscriptions",
      "organizations_url": "https://api.github.com/users/amehlhase316/orgs",
```

1. Explain the specific API calls you used.
   https://api.github.com/repos/amehlhase316/memoranda/commits

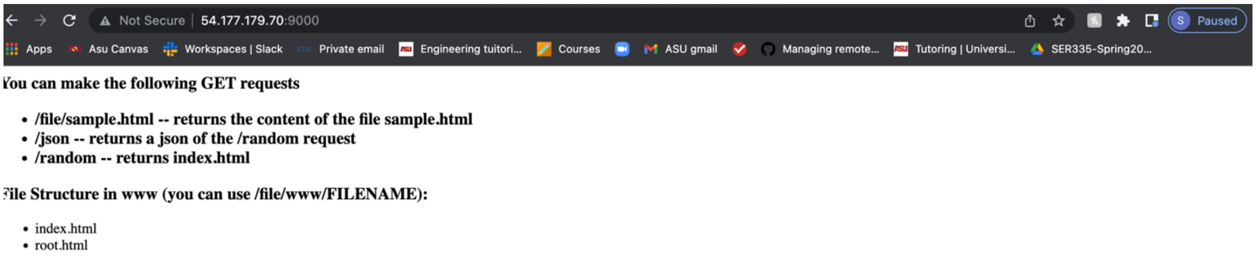
2. Explain the difference between stateless and a stateful communication.

Stateless Protocol is a network protocol in which Client send request to the server and server response back as per the given state. Stateful Protocol is a network protocol in which if client send a request to the server then it expects some kind of response, in case of no response then it resend the request

3 Setup your second system and run server on it

3.1 getting sample code onto your system

3.2 Running a sample Java WebServer

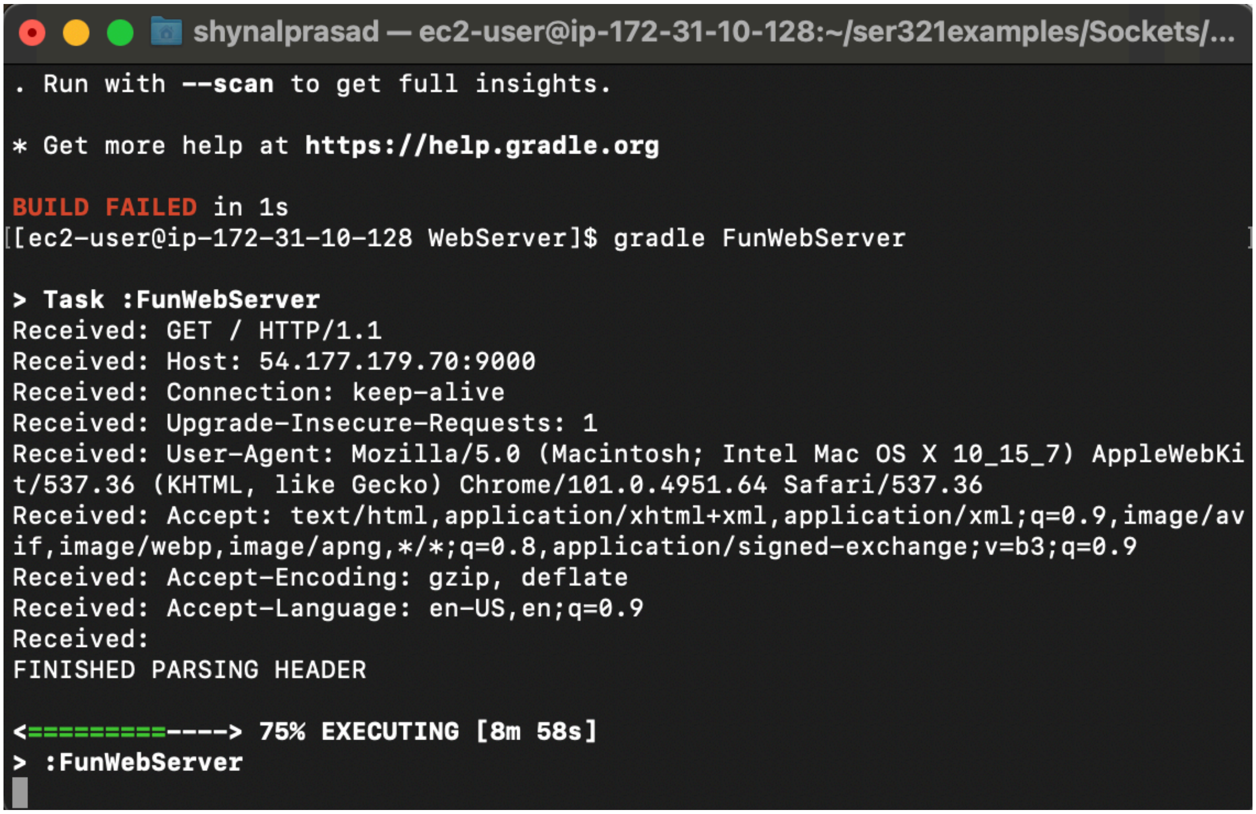


You can make the following GET requests

- **/file/sample.html -- returns the content of the file sample.html**
- **/json -- returns a json of the /random request**
- **/random -- returns index.html**

File Structure in www (you can use /file/www/FILENAME):

- index.html
- root.html


shynalprasad — ec2-user@ip-172-31-10-128:~/ser321examples/Sockets/...

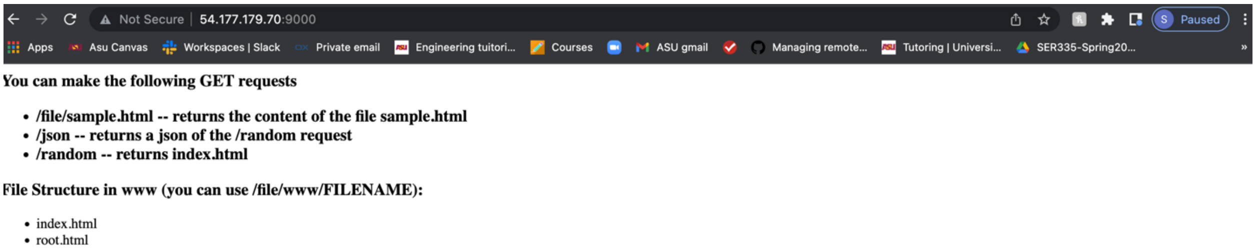
```
. Run with --scan to get full insights.

* Get more help at https://help.gradle.org

BUILD FAILED in 1s
[[ec2-user@ip-172-31-10-128 WebServer]$ gradle FunWebServer

> Task :FunWebServer
Received: GET / HTTP/1.1
Received: Host: 54.177.179.70:9000
Received: Connection: keep-alive
Received: Upgrade-Insecure-Requests: 1
Received: User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKi
t/537.36 (KHTML, like Gecko) Chrome/101.0.4951.64 Safari/537.36
Received: Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/av
if,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Received: Accept-Encoding: gzip, deflate
Received: Accept-Language: en-US,en;q=0.9
Received:
FINISHED PARSING HEADER

<=========----> 75% EXECUTING [8m 58s]
> :FunWebServer
```

1. What filter did you use? Explain why you chose that filter.
   tcp.port==9000 because that was the port defined to the web connection
2. What happens when you are on /random and click the "Random" button compared to the browser refresh (you can also use the command line output that the WebServer generates to answer this)?
   referesh doesn't create (rts) trace in wire shark
3. What kinds of response codes are you able to get through different requests to your server?
   200
4. Explain the response codes you get and why you get them.
   The code I got was 200 because we get the correct response from the server
5. When you do a *ipOfSecondMachine:9000* take a look what Wireshark generates as a server response. Are you able to find the data that the server sends back to you?
   Yes, was able to find data the server sent to me
6. Based on the above question explain why HTTPs is now more common than HTTP.
   because HTTPs is encrypted and HTTP is not encrytped
7. What port does the server listen to for HTTP requests in our case and is that the most common port for HTTP?
   In our case, the server listened to port 9000, but the common port is 80.
8. What local port is used when sending different requests to the WebServer? How does it differ to the traffic to your SMTP server from part 1?
   source port 57524 is used to send request to server. Same local ports are used when we are in part 2 but in part 1 different ports are used.

3.4 Setting up a "real" Web service

1. Check your traffic to your Webserver. What port is the traffic going to now? Is it the same as previously used or is it and should it be different?
   Traffic is going through port 9000 on server. It is the same as we used port 9000 in previous one too.

2. Is it still HTTP or is it now HTTPs? Why?
   it is HTTP only because the web server configuration was setup to HTTP only

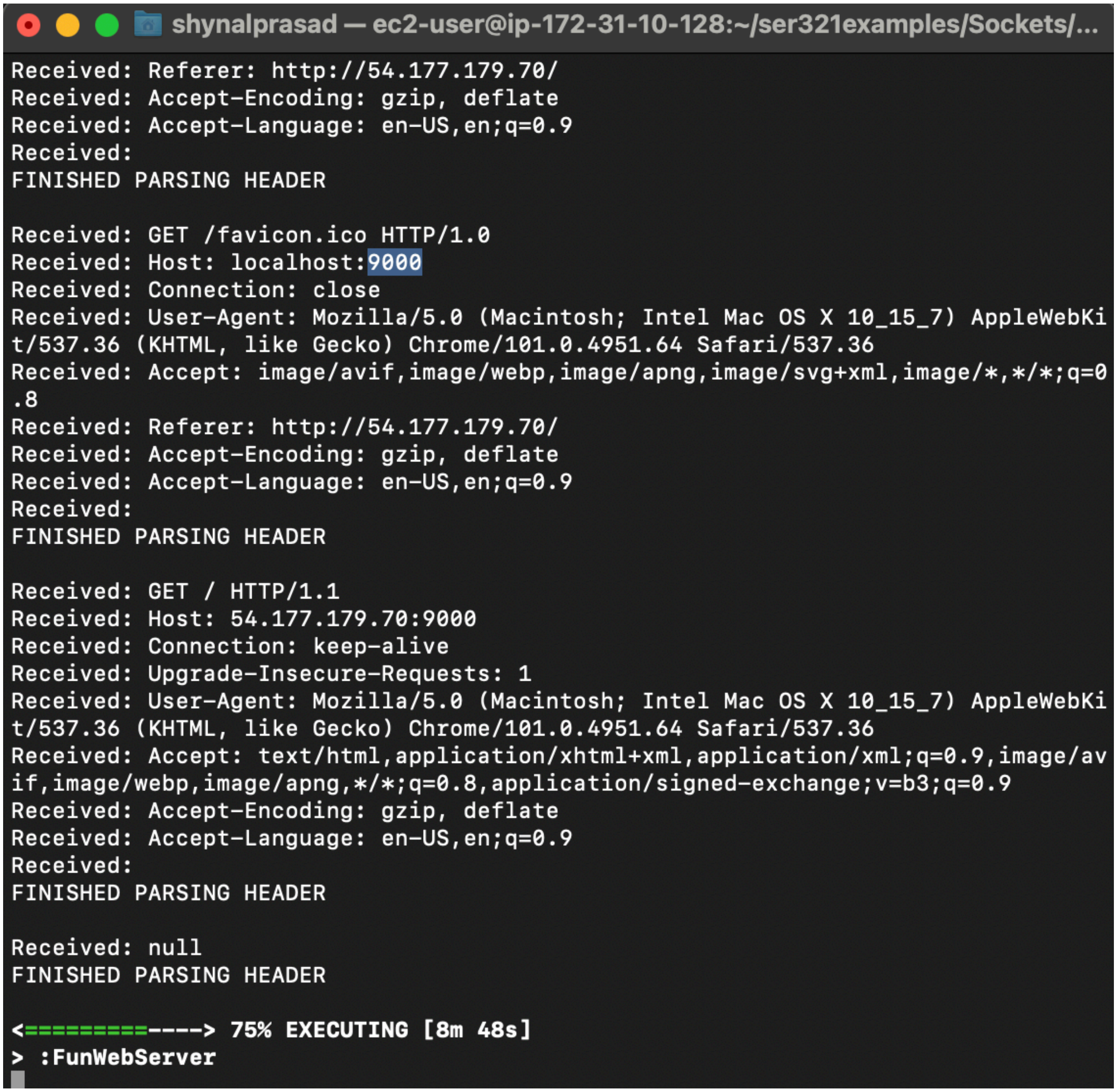


You can make the following GET requests

- **/file/sample.html -- returns the content of the file sample.html**
- **/json -- returns a json of the /random request**
- **/random -- returns index.html**

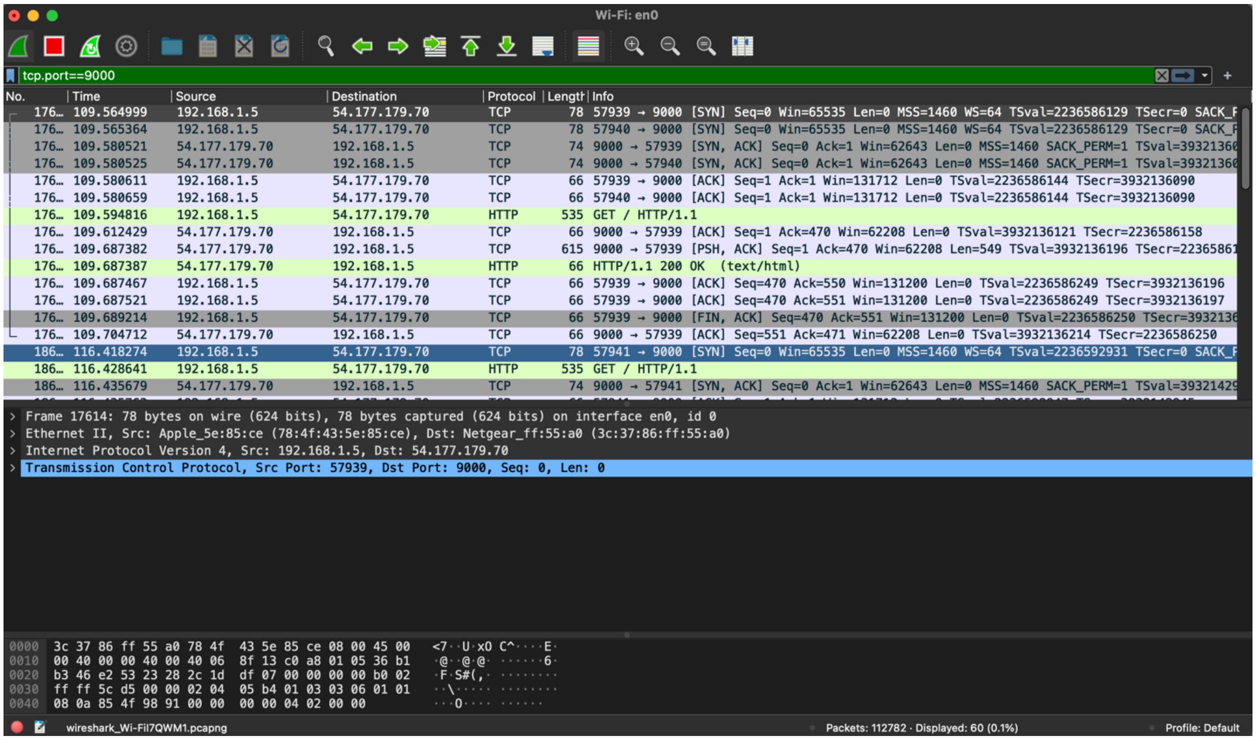
File Structure in www (you can use /file/www/FILENAME):

- index.html
- root.html



shynalprasad — ec2-user@ip-172-31-10-128:~/ser321examples/Sockets/...

```
Received: Referer: http://54.177.179.70/
Received: Accept-Encoding: gzip, deflate
Received: Accept-Language: en-US,en;q=0.9
Received:
FINISHED PARSING HEADER

Received: GET /favicon.ico HTTP/1.0
Received: Host: localhost:9000
Received: Connection: close
Received: User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKi
t/537.36 (KHTML, like Gecko) Chrome/101.0.4951.64 Safari/537.36
Received: Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0
.8
Received: Referer: http://54.177.179.70/
Received: Accept-Encoding: gzip, deflate
Received: Accept-Language: en-US,en;q=0.9
Received:
FINISHED PARSING HEADER

Received: GET / HTTP/1.1
Received: Host: 54.177.179.70:9000
Received: Connection: keep-alive
Received: Upgrade-Insecure-Requests: 1
Received: User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKi
t/537.36 (KHTML, like Gecko) Chrome/101.0.4951.64 Safari/537.36
Received: Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/av
if,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Received: Accept-Encoding: gzip, deflate
Received: Accept-Language: en-US,en;q=0.9
Received:
FINISHED PARSING HEADER

Received: null
FINISHED PARSING HEADER

<=========----> 75% EXECUTING [8m 48s]
> :FunWebServer
```



```
tcp.port==9000
```

```
No.    Time        Source          Destination     Protocol Length Info
176…   109.564999  192.168.1.5     54.177.179.70   TCP       78 57939 → 9000 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2236586129 TSecr=0 SACK_P
176…   109.565364  192.168.1.5     54.177.179.70   TCP       78 57940 → 9000 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2236586129 TSecr=0 SACK_P
176…   109.580521  54.177.179.70   192.168.1.5     TCP       74 9000 → 57939 [SYN, ACK] Seq=0 Ack=1 Win=62643 Len=0 MSS=1460 SACK_PERM=1 TSval=39321360
176…   109.580525  54.177.179.70   192.168.1.5     TCP       74 9000 → 57940 [SYN, ACK] Seq=0 Ack=1 Win=62643 Len=0 MSS=1460 SACK_PERM=1 TSval=39321360
176…   109.580611  192.168.1.5     54.177.179.70   TCP       66 57939 → 9000 [ACK] Seq=1 Ack=1 Win=131712 Len=0 TSval=2236586144 TSecr=3932136090
176…   109.580659  192.168.1.5     54.177.179.70   TCP       66 57940 → 9000 [ACK] Seq=1 Ack=1 Win=131712 Len=0 TSval=2236586144 TSecr=3932136090
176…   109.594816  192.168.1.5     54.177.179.70   HTTP     535 GET / HTTP/1.1
176…   109.612429  54.177.179.70   192.168.1.5     TCP       66 9000 → 57939 [ACK] Seq=1 Ack=470 Win=62208 Len=0 TSval=3932136121 TSecr=2236586158
176…   109.687382  54.177.179.70   192.168.1.5     TCP      615 9000 → 57939 [PSH, ACK] Seq=1 Ack=470 Win=62208 Len=549 TSval=3932136196 TSecr=22365861
176…   109.687387  54.177.179.70   192.168.1.5     HTTP      66 HTTP/1.1 200 OK  (text/html)
176…   109.687467  192.168.1.5     54.177.179.70   TCP       66 57939 → 9000 [ACK] Seq=470 Ack=550 Win=131200 Len=0 TSval=2236586249 TSecr=3932136196
176…   109.687521  192.168.1.5     54.177.179.70   TCP       66 57939 → 9000 [ACK] Seq=470 Ack=551 Win=131200 Len=0 TSval=2236586249 TSecr=3932136197
176…   109.689214  192.168.1.5     54.177.179.70   TCP       66 57939 → 9000 [FIN, ACK] Seq=470 Ack=551 Win=131200 Len=0 TSval=2236586250 TSecr=3932136
176…   109.704712  54.177.179.70   192.168.1.5     TCP       66 9000 → 57939 [ACK] Seq=551 Ack=471 Win=62208 Len=0 TSval=3932136214 TSecr=2236586250
186…   116.418274  192.168.1.5     54.177.179.70   TCP       78 57941 → 9000 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2236592931 TSecr=0 SACK_P
186…   116.428641  192.168.1.5     54.177.179.70   HTTP     535 GET / HTTP/1.1
186…   116.435679  54.177.179.70   192.168.1.5     TCP       74 9000 → 57941 [SYN, ACK] Seq=0 Ack=1 Win=62643 Len=0 MSS=1460 SACK_PERM=1 TSval=39321429
```

```
> Frame 17614: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface en0, id 0
> Ethernet II, Src: Apple_5e:85:ce (78:4f:43:5e:85:ce), Dst: Netgear_ff:55:a0 (3c:37:86:ff:55:a0)
> Internet Protocol Version 4, Src: 192.168.1.5, Dst: 54.177.179.70
> Transmission Control Protocol, Src Port: 57939, Dst Port: 9000, Seq: 0, Len: 0
```

```
0000  3c 37 86 ff 55 a0 78 4f  43 5e 85 ce 08 00 45 00   <7··U·xO C^····E·
0010  00 40 00 00 40 00 40 06  8f 13 c0 a8 01 05 36 b1   ·@··@·@· ······6·
0020  b3 46 e2 53 23 28 2c 1d  df 07 00 00 00 00 b0 02   ·F·S#(,· ········
0030  ff ff 5c d5 00 00 02 04  05 b4 01 03 03 06 01 01   ··\····· ········
0040  08 0a 85 4f 98 91 00 00  00 00 04 02 00 00         ···O···· ······
```

wireshark_Wi-Fil7QWM1.pcapng · Packets: 112782 · Displayed: 60 (0.1%) · Profile: Default

3.6 Some programming on your WebServer

3.6.1 Multiply

We used the error code 400 because the user is entering invalid input.

3.6.2