

### Worksheet 3

Nama : Samuel Ludwig Ian

NPM : 1806191471

Kelas : Jarkomdat B

Link yang diakses :

- [Apache.org/index.html](http://Apache.org/index.html)
- [Facebook.com/](https://Facebook.com/)

#### A. Explain the flow of HTTP persistent in your experiment

Dalam eksperimen, saya menemukan adanya penanda dimana Connection yang diberikan adalah keep alive. Dalam melakukan koneksi kepada server, dilakukan **synchronization** yang berasal dari kita (remote) kepada tujuan (server) yang ditandai dengan flag [SYN]. Setelah itu, server menjawab dengan adanya **acknowledgement** atas **synchronization** request yang ditandai dengan flag [SYN, ACK]. Selanjutnya, computer kita (remote) memberikan **acknowledgement** juga dengan mengirimkan flag [ACK]. Setelah dilakukan **synchronization**, didapati bahwa kita mengirimkan (**mempush**) data dan telah berhasil menerima data (**acknowledged**) dengan adanya flag [PSH, ACK]. Komunikasi ini dilakukan selama beberapa kali dengan arah bolak-balik.

Saat kita melakukan request GET melalui protocol HTTP/1.1 dengan menjalankan perintah “GET / HTTP/1.1”, server menjawab request kita dengan konfirmasi (**acknowledgement**) yang ditandai oleh flag [ACK]. Kemudian, server merespon dengan protocol yang sama yaitu HTTP/1.1 dan memberi tahu bahwa resource yang mau kita akses sudah dipindahkan sehingga sebaiknya dilakukan redirect dengan adanya informasi “301 Moved Permanently”. Komputer kita mendapatkan pesan ini dan memberikan flag [ACK] untuk konfirmasi.

Karena komputer kita maupun server belum memberikan/mengirim data yang ditandai dengan flag [PSH], maka dikeluarkan flag **Continuation** yang menandakan bahwa dilakukan pengiriman data yang besar sehingga harus dipecah menjadi beberapa paket. Setelah itu, komputer dan server masih terus melakukan pengiriman (**push**) dan konfirmasi penerimaan data (**acknowledgement**) agar koneksi tetap tersambung dan tidak di alokasikan untuk resource lain yang ditandai dengan adanya flag [PSH, ACK] dan [ACK].

Saat dilakukan pemutusan koneksi dengan menekan tombol ctrl+c, dikeluarkan informasi **finish** yang di kirimkan (**push**) dengan flag [FIN, PSH, ACK] **acknowledgement** disini digunakan sebagai penanda konfirmasi dari pengiriman data sebelumnya. Setelah komputer menerima informasi bahwa server ingin memutus koneksi TCP yang tadi dibuat, maka komputer mengkonfirmasi (**acknowledgement**) dan mengirimkan informasi bahwa komputer juga ingin memutus koneksi (**finish**) dengan mengirimkan flag [FIN, ACK]. Koneksi TCP ditutup dengan adanya konfirmasi (**acknowledgement**) dari server yang menerima flag [FIN] yang dikirimkan komputer.

## B. Explain the differences of HTTP persistent and HTTP non persistent



Note: Saya menaruh garis merah sebagai penanda, bagian atas adalah HTTP Persistent dan bagian bawah adalah HTTP non-persistent

HTTP persistent dan HTTP non-persistent dapat dibedakan berdasarkan jenis koneksinya. HTTP persistent mempertahankan dan menunggu koneksi TCP setelah dilakukan request, namun HTTP non-persistent memutuskan koneksi TCP setelah request selesai direspon. HTTP non-persistent dapat diidentifikasi saat melakukan koneksi TCP melalui telnet yang ditandai oleh header **Connection: close**, sedangkan HTTP persistent dapat diidentifikasi dengan cara yang sama dengan jenis koneksi berbeda yakni **Connection: keep-alive**. Selain itu, untuk melakukan koneksi persistent, saya melakukan

koneksi TCP ke facebook.com dengan HTTP/1.1 dan untuk koneksi non-persistent, saya melakukan koneksi TCP ke facebook.com juga, namun dengan HTTP/1.0. Karena koneksi persistent mengalokasikan resource secara khusus untuk suatu komunikasi, maka koneksi dapat menjadi penuh dan resource menjadi macet, oleh karena itu umumnya digunakan HTTP/1.1 untuk persistent connection karena kecepatan HTTP/1.1 yang lebih tinggi dibanding dengan HTTP/1.0.

Untuk flow dari penggunaan non-persistent communication, proses berjalan seperti berikut: Client melakukan koneksi TCP → Request dari client → Response diberikan server → Koneksi TCP diputuskan → Client melakukan koneksi TCP → Request dari client → Response diberikan server → Koneksi TCP diputuskan → dst. Hal ini disebabkan oleh adanya pemutusan koneksi setelah request telah direspon oleh server pada HTTP non-persistent. Berbeda dengan HTTP non-persistent, HTTP persistent tetap membiarkan komunikasi TCP tersambung setelah request dari client telah di respon, sehingga flow atau prosesnya berjalan seperti berikut: Client melakukan koneksi TCP → Request dari client → Response diberikan server → Request dari client → Response diberikan server → ... → Koneksi TCP diputuskan.

### C. Bukti screenshot pengerjaan tugas

1. Link : facebook.com/  
Port : 80  
Request line : GET / HTTP/1.1  
Status line : HTTP/1.1 301 Moved Permanently  
Header lines : (Lihat gambar)  
Body/data : Server tidak memberikan

```
[*]-[ss27@parrot]-[~]
$telnet facebook.com 80
Trying 157.240.217.35...
Connected to facebook.com.
Escape character is '^]'.
GET / HTTP/1.1
Host:facebook.com

HTTP/1.1 301 Moved Permanently
Location: https://facebook.com/
Content-Type: text/html; charset="utf-8"
X-FB-Debug: M/Iy+0KioQsAYnLPjxp9NcXt/Y41LiIDa3dsZ3pfIHZSeBBP4WD5iks7fZLrXtB6AX60WQTMHgdPLbeQtmjtjzA==
Date: Mon, 05 Oct 2020 13:36:23 GMT
Alt-Svc: h3-29=":443"; ma=3600,h3-27=":443"; ma=3600
Connection: keep-alive
Content-Length: 0

^CConnection closed by foreign host.
```

**HEADER Lines**

*Note: Untuk nomor 1*

**Request Lines**

**Status Lines**

**HEADER Lines**

```

146 41.009110842 157.240.217.35 192.168.79.131 TCP 60 80 → 46858 [ACK] Seq=1 Ack=36 Win=64240 Len=0
147 41.865263285 192.168.79.131 157.240.217.35 HTTP 56 GET / HTTP/1.1
148 41.865779088 157.240.217.35 192.168.79.131 TCP 60 80 → 46858 [ACK] Seq=1 Ack=38 Win=64240 Len=0
149 42.057891312 157.240.217.35 192.168.79.131 HTTP 399 HTTP/1.1 301 Moved Permanently
150 42.057910675 192.168.79.131 157.240.217.35 TCP 54 46858 → 80 [ACK] Seq=38 Ack=346 Win=63895 Len=0
151 43.769506307 192.168.79.131 157.240.217.35 HTTP 59 Continuation
152 43.769739446 157.240.217.35 192.168.79.131 TCP 60 80 → 46858 [ACK] Seq=346 Ack=43 Win=64240 Len=0
153 43.807299414 157.240.217.35 192.168.79.131 TCP 1414 80 → 46858 [PSH, ACK] Seq=346 Ack=43 Win=64240 Len=0
154 43.807345614 192.168.79.131 157.240.217.35 TCP 54 46858 → 80 [ACK] Seq=43 Ack=1706 Win=63895 Len=0
155 43.807556113 157.240.217.35 192.168.79.131 TCP 1414 80 → 46858 [PSH, ACK] Seq=1706 Ack=43 Win=64240 Len=0
156 43.807562447 192.168.79.131 157.240.217.35 TCP 54 46858 → 80 [ACK] Seq=43 Ack=3066 Win=63895 Len=0
157 43.808085952 157.240.217.35 192.168.79.131 HTTP 439 HTTP/1.1 400 Bad Request (text/html)
158 43.808093747 192.168.79.131 157.240.217.35 TCP 54 46858 → 80 [ACK] Seq=43 Ack=3451 Win=63895 Len=0
159 43.808539801 157.240.217.35 192.168.79.131 TCP 60 80 → 46858 [FIN, PSH, ACK] Seq=3451 Ack=43 Win=64240 Len=0

> Frame 149: 399 bytes on wire (3192 bits), 399 bytes captured (3192 bits) on interface eth0, id 0
> Ethernet II, Src: VMware_f8:36:48 (00:50:56:f8:36:48), Dst: VMware_20:0c:29:20:d3:c1 (00:0c:29:20:d3:c1)
> Internet Protocol Version 4, Src: 157.240.217.35, Dst: 192.168.79.131
> Transmission Control Protocol, Src Port: 80, Dst Port: 46858, Seq: 1, Ack: 38, Len: 345
< Hypertext Transfer Protocol
  < HTTP/1.1 301 Moved Permanently\r\n
    Location: https://facebook.com/\r\n
    Content-Type: text/html; charset="utf-8"\r\n
    X-FB-Debug: M/Iy+0KioQsAYnLPjxp9NcXt/Y41LiIDa3dsZ3pfIHZSeBBP4Wd5iks7fZLrXtB6AX60WQTMHgdpLbeQtmtjzA==\r\n
    Date: Mon, 05 Oct 2020 13:36:23 GMT\r\n
    Alt-Svc: h3-29=":443"; ma=3600,h3-27=":443"; ma=3600\r\n
    Connection: keep-alive\r\n
  < Content-Length: 0\r\n
    [Content length: 0]
  \r\n
  [HTTP response 1/2]
  [Time since request: 0.192628027 seconds]
  [Request in frame: 147]
  [Next response in frame: 157]
  [Request URI: http://facebook.com/]
  
```

Note: Untuk nomor 1

2. Link : facebook.com/  
 Port : 80  
 Request line : GET / HTTP/1.0  
 Status line : HTTP/1.1 301 Moved Permanently  
 Header lines : (Lihat gambar)  
 Body/data : Server tidak memberikan

**HEADER Lines**

```

[*]-[ss27@parrot]-[~]
$telnet facebook.com 80
Trying 157.240.217.35...
Connected to facebook.com.
Escape character is '^]'.
GET / HTTP/1.0
Host:facebook.com

HTTP/1.1 301 Moved Permanently
Vary: Accept-Encoding
Location: https://facebook.com/
Content-Type: text/html; charset="utf-8"
X-FB-Debug: Wqjh403ClQRnU4I7KY30D3Hpru74bR1xwKwL62V0RCGk8W8U2rRHP+Lnm8DBqicdcMEscN6CgS0YM1vEEUNJMQ==
Date: Mon, 05 Oct 2020 13:36:35 GMT
Alt-Svc: h3-29=":443"; ma=3600,h3-27=":443"; ma=3600
Connection: close
Content-Length: 0

Connection closed by foreign host.
  
```

Note: Untuk nomor 2

No.	Time	Source IP	Destination IP	Protocol	Source Port	Destination Port	Details
172	51.434420707	157.240.217.35	192.168.79.131	TCP	60	80	→ 46860 [ACK] Seq=1 Ack=36 Win=64240 Len=0
173	52.178979569	192.168.79.131	157.240.217.35	HTTP	56	80	GET / HTTP/1.0
174	52.179279147	157.240.217.35	192.168.79.131	TCP	60	80	→ 46860 [ACK] Seq=1 Ack=38 Win=64240 Len=0
175	52.375328586	157.240.217.35	192.168.79.131	HTTP	417	80	HTTP/1.1 301 Moved Permanently
176	52.375369451	192.168.79.131	157.240.217.35	TCP	54	46860	→ 80 [ACK] Seq=38 Ack=364 Win=63877 Len=0
177	52.375834681	157.240.217.35	192.168.79.131	TCP	60	80	→ 46860 [FIN, PSH, ACK] Seq=364 Ack=38 Win=64240 Len=0
178	52.375871622	192.168.79.131	157.240.217.35	TCP	54	46860	→ 80 [FIN, ACK] Seq=38 Ack=365 Win=63877 Len=0
179	52.375982054	157.240.217.35	192.168.79.131	TCP	60	80	→ 46860 [ACK] Seq=365 Ack=39 Win=64239 Len=0

  

> Transmission Control Protocol, Src Port: 80, Dst Port: 46860, Seq: 1, Ack: 80, Len: 0

▼ Hypertext Transfer Protocol

> HTTP/1.1 301 Moved Permanently\r\n

Vary: Accept-Encoding\r\n

Location: https://facebook.com/\r\n

Content-Type: text/html; charset="utf-8"\r\n

X-FB-Debug: Wqjh403ClQRnU4I7KY30D3Hpru74bR1xwkw1G2V0RCGkw8U2rRHP+Lnm8DBqicdcMEscN6CgS0YM1vEEUNJMqQ==\r\n

Date: Mon, 05 Oct 2020 13:36:35 GMT\r\n

Alt-Svc: h3-29=":443"; ma=3600,h3-27=":443"; ma=3600\r\n

Connection: close\r\n

> Content-Length: 0\r\n

\r\n

[HTTP response 1/1]

[Time since request: 0.196349017 seconds]

[Request in frame: 173]

[Request URI: http://facebook.com/]

Note: Untuk nomor 2



Wireshark interface showing a packet capture of a web session. The filter is set to `ip.addr == 157.240.217.35 || tcp.port == 46858 || tcp.port == 46860`. The packet list shows a series of TCP and HTTP packets between 192.168.79.131 and 157.240.217.35. A red line is drawn between packet 161 (HTTP 400 Bad Request) and packet 166 (TCP SYN). The packet details pane shows the structure of packet 140: Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol.

No.	Time	Source	Destination	Protocol	Length	Info
140	37.204809916	192.168.79.131	157.240.217.35	TCP	74	46858 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK...
141	37.231199348	157.240.217.35	192.168.79.131	TCP	60	80 → 46858 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS...
142	37.231255339	192.168.79.131	157.240.217.35	TCP	54	46858 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
143	41.008793799	192.168.79.131	157.240.217.35	TCP	70	46858 → 80 [PSH, ACK] Seq=1 Ack=1 Win=64240 Len=16 [T...
144	41.009013424	157.240.217.35	192.168.79.131	TCP	60	80 → 46858 [ACK] Seq=1 Ack=17 Win=64240 Len=0
145	41.009026529	192.168.79.131	157.240.217.35	TCP	73	46858 → 80 [PSH, ACK] Seq=17 Ack=1 Win=64240 Len=19 [T...
146	41.009110842	157.240.217.35	192.168.79.131	TCP	60	80 → 46858 [ACK] Seq=1 Ack=36 Win=64240 Len=0
147	41.865263285	192.168.79.131	157.240.217.35	HTTP	56	GET / HTTP/1.1
148	41.865779088	157.240.217.35	192.168.79.131	TCP	60	80 → 46858 [ACK] Seq=1 Ack=38 Win=64240 Len=0
149	42.057891312	157.240.217.35	192.168.79.131	HTTP	399	HTTP/1.1 301 Moved Permanently
150	42.057910675	192.168.79.131	157.240.217.35	TCP	54	46858 → 80 [ACK] Seq=38 Ack=346 Win=63895 Len=0
151	43.769506307	192.168.79.131	157.240.217.35	HTTP	59	Continuation
152	43.769739446	157.240.217.35	192.168.79.131	TCP	60	80 → 46858 [ACK] Seq=346 Ack=43 Win=64240 Len=0
153	43.807299414	157.240.217.35	192.168.79.131	TCP	1414	80 → 46858 [PSH, ACK] Seq=346 Ack=43 Win=64240 Len=130...
154	43.807345614	192.168.79.131	157.240.217.35	TCP	54	46858 → 80 [ACK] Seq=43 Ack=1706 Win=63895 Len=0
155	43.807556113	157.240.217.35	192.168.79.131	TCP	1414	80 → 46858 [PSH, ACK] Seq=1706 Ack=43 Win=64240 Len=1...
156	43.807562447	192.168.79.131	157.240.217.35	TCP	54	46858 → 80 [ACK] Seq=43 Ack=3066 Win=63895 Len=0
157	43.808085952	157.240.217.35	192.168.79.131	HTTP	439	HTTP/1.1 400 Bad Request (text/html)
158	43.808093747	192.168.79.131	157.240.217.35	TCP	54	46858 → 80 [ACK] Seq=43 Ack=3451 Win=63895 Len=0
159	43.808539801	157.240.217.35	192.168.79.131	TCP	60	80 → 46858 [FIN, PSH, ACK] Seq=3451 Ack=43 Win=64240 L...
160	43.808572994	192.168.79.131	157.240.217.35	TCP	54	46858 → 80 [FIN, ACK] Seq=43 Ack=3452 Win=63895 Len=0
161	43.808665078	157.240.217.35	192.168.79.131	TCP	60	80 → 46858 [ACK] Seq=3452 Ack=44 Win=64239 Len=0
166	48.467745701	192.168.79.131	157.240.217.35	TCP	74	46860 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK...
167	48.492448369	157.240.217.35	192.168.79.131	TCP	60	80 → 46860 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS...
168	48.492507754	192.168.79.131	157.240.217.35	TCP	54	46860 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
169	51.433693470	192.168.79.131	157.240.217.35	TCP	70	46860 → 80 [PSH, ACK] Seq=1 Ack=1 Win=64240 Len=16 [T...
170	51.434081032	157.240.217.35	192.168.79.131	TCP	60	80 → 46860 [ACK] Seq=1 Ack=17 Win=64240 Len=0
171	51.434092087	192.168.79.131	157.240.217.35	TCP	73	46860 → 80 [PSH, ACK] Seq=17 Ack=1 Win=64240 Len=19 [T...
172	51.434420707	157.240.217.35	192.168.79.131	TCP	60	80 → 46860 [ACK] Seq=1 Ack=36 Win=64240 Len=0
173	52.178979569	192.168.79.131	157.240.217.35	HTTP	56	GET / HTTP/1.0
174	52.179279147	157.240.217.35	192.168.79.131	TCP	60	80 → 46860 [ACK] Seq=1 Ack=38 Win=64240 Len=0
175	52.375328586	157.240.217.35	192.168.79.131	HTTP	417	HTTP/1.1 301 Moved Permanently
176	52.375369451	192.168.79.131	157.240.217.35	TCP	54	46860 → 80 [ACK] Seq=38 Ack=364 Win=63877 Len=0
177	52.375834681	157.240.217.35	192.168.79.131	TCP	60	80 → 46860 [FIN, PSH, ACK] Seq=364 Ack=38 Win=64240 L...
178	52.375871622	192.168.79.131	157.240.217.35	TCP	54	46860 → 80 [FIN, ACK] Seq=38 Ack=365 Win=63877 Len=0
179	52.375982054	157.240.217.35	192.168.79.131	TCP	60	80 → 46860 [ACK] Seq=365 Ack=39 Win=64239 Len=0

Frame 140: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface eth0, id 0  
 Ethernet II, Src: VMware\_20:d3:c1 (00:0c:29:20:d3:c1), Dst: VMware\_f8:36:48 (00:50:56:f8:36:48)  
 Internet Protocol Version 4, Src: 192.168.79.131, Dst: 157.240.217.35  
 Transmission Control Protocol, Src Port: 46858, Dst Port: 80, Seq: 0, Len: 0

0000 00 50 56 f8 36 48 00 0c 29 20 d3 c1 08 00 45 10 .PV.6H.. ) ...E.  
 0010 00 3c fb 2f 40 00 00 06 b8 3c c0 a8 4f 83 9d f0 .<./@.@.<...0...  
 0020 d9 23 b7 0a 00 50 72 55 7c 3f 00 00 00 00 a0 02 .#...PrU |?...  
 0030 fa f0 87 6e 00 00 02 04 05 b4 04 02 08 0a 67 50 ...n....gP  
 0040 3c a4 00 00 00 00 01 03 03 0a <.....

wireshark-sniffed-packets.pcapng Packets: 179 · Displayed: 36 (20.1%) · Dropped: 0 (0.0%) Profile: Default

Note: Hasil filter untuk nomor 1 dan 2.

Saya menaruh garis merah sebagai penanda, bagian atas adalah HTTP Persistent dengan request menggunakan HTTP/1.1 dan bagian bawah adalah HTTP non-persistent dengan menggunakan HTTP/1.0



Note: Flow Graph untuk nomor 1 dan 2

Saya menaruh garis merah sebagai penanda, bagian atas adalah HTTP Persistent dengan request menggunakan HTTP/1.1 dan bagian bawah adalah HTTP non-persistent dengan menggunakan HTTP/1.0



3. Link : [apache.org/index.html](http://apache.org/index.html)  
Port : 80  
Request line : GET /index.html HTTP/1.0  
Status line : HTTP/1.1 200 OK  
Header lines : (Lihat gambar)  
Body/data : (Lihat gambar)

```
[*]-[ss27@parrot]-[~]
$telnet apache.org 80
Trying 95.216.24.32...
Connected to apache.org.
Escape character is '^]'.
GET /index.html HTTP/1.0
Host:apache.org

HTTP/1.1 200 OK
Date: Mon, 05 Oct 2020 13:36:11 GMT
Server: Apache/2.4.18 (Ubuntu)
Last-Modified: Mon, 05 Oct 2020 13:10:27 GMT
ETag: "14ac6-5b0ec34c08739"
Accept-Ranges: bytes
Content-Length: 84678
Vary: Accept-Encoding
Cache-Control: max-age=3600
Expires: Mon, 05 Oct 2020 14:36:11 GMT
Connection: close
Content-Type: text/html
```

**HEADER Lines**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="description" content="Home page of The Apache Software Foundation">

  <link rel="apple-touch-icon" sizes="57x57" href="/favicons/apple-touch-icon-57x57.png">
  <link rel="apple-touch-icon" sizes="60x60" href="/favicons/apple-touch-icon-60x60.png">
  <link rel="apple-touch-icon" sizes="72x72" href="/favicons/apple-touch-icon-72x72.png">
  <link rel="apple-touch-icon" sizes="76x76" href="/favicons/apple-touch-icon-76x76.png">
  <link rel="apple-touch-icon" sizes="114x114" href="/favicons/apple-touch-icon-114x114.png">
  <link rel="apple-touch-icon" sizes="120x120" href="/favicons/apple-touch-icon-120x120.png">
  <link rel="apple-touch-icon" sizes="144x144" href="/favicons/apple-touch-icon-144x144.png">
  <link rel="apple-touch-icon" sizes="152x152" href="/favicons/apple-touch-icon-152x152.png">
  <link rel="apple-touch-icon" sizes="180x180" href="/favicons/apple-touch-icon-180x180.png">
  <link rel="icon" type="image/png" href="/favicons/favicon-32x32.png" sizes="32x32">
  <link rel="icon" type="image/png" href="/favicons/favicon-194x194.png" sizes="194x194">
  <link rel="icon" type="image/png" href="/favicons/favicon-96x96.png" sizes="96x96">
  <link rel="icon" type="image/png" href="/favicons/android-chrome-192x192.png" sizes="192x192">
  <link rel="icon" type="image/png" href="/favicons/favicon-16x16.png" sizes="16x16">
  <link rel="manifest" href="/favicons/manifest.json">
  <link rel="shortcut icon" href="/favicons/favicon.ico">
  <meta name="msapplication-TileColor" content="#603cba">
  <meta name="msapplication-TileImage" content="/favicons/mstile-144x144.png">
  <meta name="msapplication-config" content="/favicons/browserconfig.xml">
  <meta name="theme-color" content="#282661">

  <title>Welcome to The Apache Software Foundation!</title>
  <link href="https://fonts.googleapis.com/css?family=Montserrat:300,600" rel="stylesheet">
  <link href="/css/min.bootstrap.css" rel="stylesheet">
  <link href="/css/styles.css" rel="stylesheet">
```

**Body/data**

*Note: Untuk nomor 3*



```
<div class="col-sm-2">
  <h5 class="white">Legal</h5>
  <ul class="list-unstyled white" role="menu">
    <li><a href="/legal/">Legal Affairs</a></li>
    <li><a href="/legal/dmca.html">DMCA</a></li>
    <li><a href="/licenses/">Licensing</a></li>
    <li><a href="/foundation/marks/">Trademark Policy</a></li>
    <li><a href="/foundation/records/">Public Records</a></li>
    <li><a href="/foundation/policies/privacy.html">Privacy Policy</a></li>
    <li><a href="/licenses/exports/">Export Information</a></li>
    <li><a href="/foundation/license-faq.html">Licensing FAQ</a></li>
    <li><a href="/foundation/contact.html">Contacts</a></li>
  </ul>
</div>

<div class="col-sm-1">
</div>

</div>
<hr class="col-lg-12 hr-white" />
<div class="row">
  <div class="col-lg-12">
    <p class="text-center">Copyright ©169; 2020 The Apache Software Foundation, Licensed under the <a class="
enses/LICENSE-2.0">Apache License, Version 2.0</a>.</p>
    <p class="text-center">Apache and the Apache feather logo are trademarks of The Apache Software Foundation
    </div>
  </div>
</div>

</footer>

<!-- / Footer -->

<script src="/js/jquery-2.1.1.min.js"></script>
<script src="/js/bootstrap.js"></script>
<script src="/js/slideshow.js"></script>
<script>
  (function($){
    $(document).ready(function(){
      $('ul.dropdown-menu [data-toggle=dropdown]').on('click', function(event) {
        event.preventDefault();
        event.stopPropagation();
        $(this).parent().siblings().removeClass('open');
        $(this).parent().toggleClass('open');
        console.log('WOrrked');
      });
    });
  })(jQuery);
</script>
</body>
</html>
Connection closed by foreign host.
```

Body/data

Note: Untuk nomor 3

The image shows a Wireshark packet capture of an HTTP transaction. The packet list on the left shows several TCP and HTTP packets. Red arrows point from labels to specific packets: 'Request Lines' points to packet 56 (GET /index.html), 'Status Lines' points to packet 133 (HTTP/1.1 200 OK), 'HEADER Lines' points to packets 134 and 135 (HTTP headers), and 'Body/data' points to packet 136 (HTML body content).

No.	Time	Source	Destination	Protocol	Length	Info
49	26.575584972	192.168.79.131	95.216.24.32	TCP	74	50860 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SA...
50	26.787810040	95.216.24.32	192.168.79.131	TCP	60	80 → 50860 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 ...
51	26.787836243	192.168.79.131	95.216.24.32	TCP	54	50860 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
52	30.891318132	192.168.79.131	95.216.24.32	TCP	80	50860 → 80 [PSH, ACK] Seq=1 Ack=1 Win=64240 Len=26...
53	30.891582892	95.216.24.32	192.168.79.131	TCP	60	80 → 50860 [ACK] Seq=1 Ack=27 Win=64240 Len=0
54	30.891614653	192.168.79.131	95.216.24.32	TCP	71	50860 → 80 [PSH, ACK] Seq=27 Ack=1 Win=64240 Len=1...
55	30.891708169	95.216.24.32	192.168.79.131	TCP	60	80 → 50860 [ACK] Seq=1 Ack=44 Win=64240 Len=0
56	31.582926037	192.168.79.131	95.216.24.32	HTTP	56	GET /index.html HTTP/1.0
57	31.583141510	95.216.24.32	192.168.79.131	TCP	60	80 → 50860 [ACK] Seq=1 Ack=46 Win=64240 Len=0
58	31.794386994	95.216.24.32	192.168.79.131	TCP	4134	80 → 50860 [PSH, ACK] Seq=1 Ack=46 Win=64240 Len=4...
59	31.794428326	192.168.79.131	95.216.24.32	TCP	54	50860 → 80 [ACK] Seq=46 Ack=4081 Win=61320 Len=0
60	31.794990770	95.216.24.32	192.168.79.131	TCP	2774	80 → 50860 [PSH, ACK] Seq=4081 Ack=46 Win=64240 Le...
61	31.795000180	192.168.79.131	95.216.24.32	TCP	54	50860 → 80 [ACK] Seq=46 Ack=6801 Win=62780 Len=0
130	32.292134686	192.168.79.131	95.216.24.32	TCP	54	50860 → 80 [ACK] Seq=46 Ack=82961 Win=65535 Len=0
131	32.294108143	95.216.24.32	192.168.79.131	TCP	1414	80 → 50860 [PSH, ACK] Seq=82961 Ack=46 Win=64240 L...
132	32.294126020	192.168.79.131	95.216.24.32	TCP	54	50860 → 80 [ACK] Seq=46 Ack=84321 Win=65535 Len=0
133	32.294794003	95.216.24.32	192.168.79.131	HTTP	136	HTTP/1.1 200 OK (text/html)
134	32.294866183	192.168.79.131	95.216.24.32	TCP	54	50860 → 80 [FIN, ACK] Seq=46 Ack=85024 Win=65535 L...
135	32.295020547	95.216.24.32	192.168.79.131	TCP	60	80 → 50860 [ACK] Seq=85024 Ack=47 Win=64239 Len=0

**Request Lines**

**Status Lines**

**HEADER Lines**

**Body/data**

Note: Untuk nomor 3, html tidak di screenshot semuanya karena terlalu panjang

Wireshark interface showing a packet capture of traffic filtered by `ip.addr == 95.216.24.32 || tcp.port == 50860`.

The packet list shows 98 packets, all of which are TCP connections to 95.216.24.32 on port 50860. The first packet (No. 49) is a SYN packet from 192.168.79.131 to 95.216.24.32. The subsequent packets are ACKs and data transfers.

Packet 49 details:

- Frame 49: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface eth0, id 0
- Ethernet II, Src: VMware\_20:d3:c1 (00:0c:29:20:d3:c1), Dst: VMware\_f8:36:48 (00:50:56:f8:36:48)
- Internet Protocol Version 4, Src: 192.168.79.131, Dst: 95.216.24.32
- Transmission Control Protocol, Src Port: 50860, Dst Port: 80, Seq: 0, Len: 0

The packet bytes pane shows the raw data of the SYN packet:

```
0000 00 50 56 f8 36 48 00 0c 29 20 d3 c1 08 00 45 10  .PV.6H.. ) ...E.
0010 00 3c dd 07 40 00 00 06 d5 80 c0 a8 4f 83 5f d8  <..@.@...0...
0020 18 20 c6 ac 00 50 2f 44 1a 77 00 00 00 00 a0 02  ..P/D..w....
0030 fa f0 88 52 00 00 02 04 05 b4 04 02 08 0a 37 fe  ...R.....7...
0040 f7 00 00 00 00 00 01 03 03 0a                      ....
```

Wireshark status: Packets: 179 · Displayed: 87 (48.6%) · Dropped: 0 (0.0%) · Profile: Default

Note: Hasil filter untuk apache.org.





Note: Hasil Flow Graph untuk apache.org.



## **D. Sources**

[https://linuxhint.com/http\\_wireshark/](https://linuxhint.com/http_wireshark/)

<https://osqa-ask.wireshark.org/questions/13384/display-http-header#:~:text=Wireshark%20captures%20full%20packets%20by,was%20processed%20by%20the%20proxy.>

<https://www.the-art-of-web.com/system/telnet-http11/>

<https://unix.stackexchange.com/questions/237635/using-telnet-to-get-website-header/237648>

[https://www.juniper.net/documentation/en\\_US/junos/topics/topic-map/tcp-configure-features.html#:~:text=The%20FIN%20flag%20indicates%20the,recipient%2C%20depending%20on%20the%20OS.](https://www.juniper.net/documentation/en_US/junos/topics/topic-map/tcp-configure-features.html#:~:text=The%20FIN%20flag%20indicates%20the,recipient%2C%20depending%20on%20the%20OS.)

<http://www.freekb.net/Article?id=938>