3-Odei

September 26, 2019

1 Odei

1.1 Things to Remember:

1) Read the getting started before exploring this write-up.
2) All file paths shown are based on the computer used in this write-up.
3) Use the Resource page/pdf to see a list all websites and programs used in this write-up.

1.2 Odei 1

On the morning of August 15th 2018, there was a spike in network traffic. Check your network resources. What is the IP address of the requester?

1.2.1 Solution:

First, take a look at the pcap files for August 15, 2018:

```
[1]: Get-ChildItem -path "..\pcaps\" -Recurse -Filter "*2018-08-15*"
```

```


    Directory: C:\Users\Administrator\Artifacts\pcaps


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----        8/23/2018   3:32 PM      100003270 2018-08-15_06-22-41.pcap
-a----        8/23/2018   3:33 PM      100002467 2018-08-15_09-04-14.pcap
-a----        8/23/2018   3:33 PM      100005890 2018-08-15_09-27-04.pcap
-a----        8/23/2018   3:34 PM      100017730 2018-08-15_09-49-37.pcap
-a----        8/23/2018   3:34 PM      100000883 2018-08-15_10-11-46.pcap
-a----        8/23/2018   3:35 PM      100000004 2018-08-15_10-51-33.pcap
-a----        8/23/2018   3:36 PM      100000736 2018-08-15_12-02-09.pcap
-a----        8/23/2018   3:36 PM      100000312 2018-08-15_12-07-10.pcap
-a----        8/23/2018   3:37 PM      100000314 2018-08-15_12-19-00.pcap
-a----        8/23/2018   3:37 PM      100000484 2018-08-15_12-31-32.pcap
-a----        8/23/2018   3:38 PM      100000240 2018-08-15_12-44-14.pcap
-a----        8/23/2018   3:39 PM      100001528 2018-08-15_13-09-10.pcap
```

```
-a----        8/23/2018   3:39 PM      100000758 2018-08-15_14-16-10.pcap
```

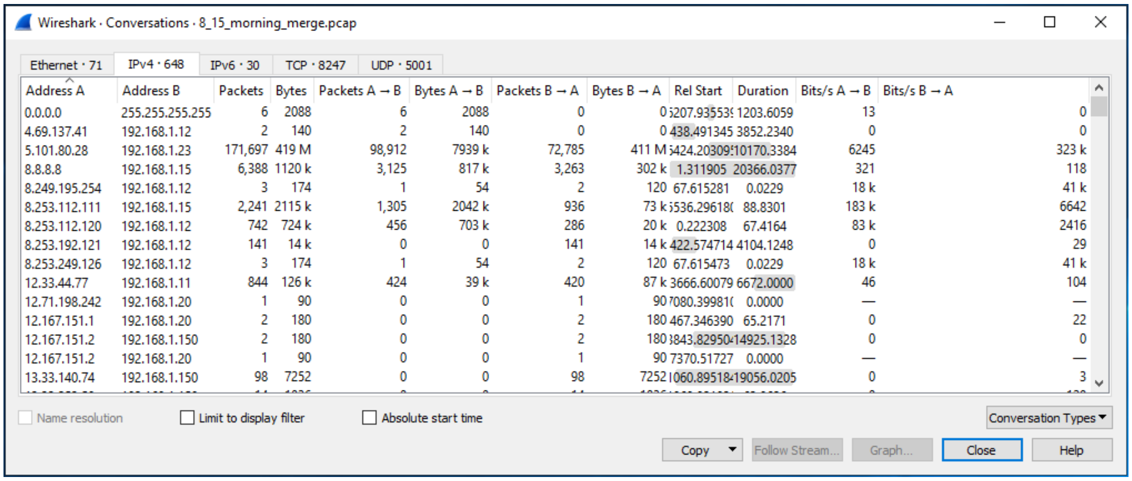
Let's merge the pcap files to make filtering easier:

[1]:

```
Set-Alias -Name merge -Value "C:\Program Files\Wireshark\mergecap.exe" |␣
↪Out-Null
Set-Alias -Name tshark -Value "C:\Program Files\Wireshark\tshark.exe" | Out-Null

merge -F pcap -w ..\pcaps\8_15_morning_merge.pcap ..\pcaps\2018-08-15_06-22-41.
↪pcap ..\pcaps\2018-08-15_09-04-14.pcap ..\pcaps\2018-08-15_09-27-04.pcap ..
↪\pcaps\2018-08-15_09-49-37.pcap ..\pcaps\2018-08-15_10-11-46.pcap ..
↪\pcaps\2018-08-15_10-51-33.pcap
write-host "Complete"
```

```
Complete
```

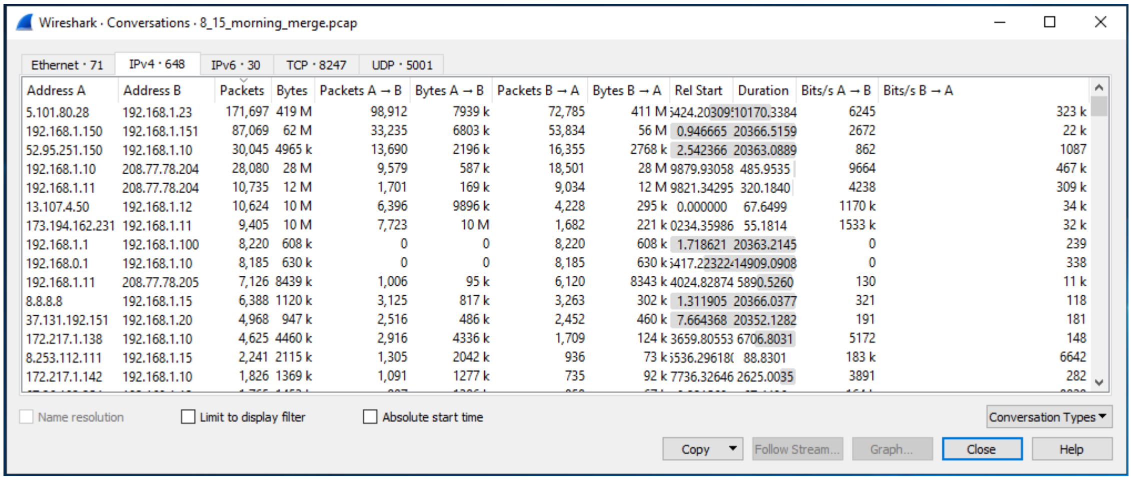
Open the merged morning pcap file in Wireshark. The question is asking for the IP of the requester that caused a spike in network traffic. Wireshark has a feature called conversations under the statistics tab. In the context of the question, conversations are important because the requester IP seems to have caused a spike in traffic. Conversations allows you to analyze the total packet transfers between two IP addresses.



Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
0.0.0.0	255.255.255.255	6	2088	6	2088	0	0	5207.93553	1203.6059	13	0
4.69.137.41	192.168.1.12	2	140	2	140	0	0	438.491345	3852.2340	0	0
5.101.80.28	192.168.1.23	171,697	419 M	98,912	7939 k	72,785	411 M	5424.20309	10170.3384	6245	323 k
8.8.8.8	192.168.1.15	6,388	1120 k	3,125	817 k	3,263	302 k	1.311905	20366.0377	321	118
8.249.195.254	192.168.1.12	3	174	1	54	2	120	67.615281	0.0229	18 k	41 k
8.253.112.111	192.168.1.15	2,241	2115 k	1,305	2042 k	936	73 k	5536.29618	88.8301	183 k	6642
8.253.112.120	192.168.1.12	742	724 k	456	703 k	286	20 k	0.222308	67.4164	83 k	2416
8.253.192.121	192.168.1.12	141	14 k	0	0	141	14 k	422.574714	4104.1248	0	29
8.253.249.126	192.168.1.12	3	174	1	54	2	120	67.615473	0.0229	18 k	41 k
12.33.44.77	192.168.1.11	844	126 k	424	39 k	420	87 k	3666.60079	6672.0000	46	104
12.71.198.242	192.168.1.20	1	90	0	0	1	90	7080.39981	0.0000	—	—
12.167.151.1	192.168.1.20	2	180	0	0	2	180	467.346390	65.2171	0	22
12.167.151.2	192.168.1.150	2	180	0	0	2	180	3843.82950	14925.1328	0	0
12.167.151.2	192.168.1.20	1	90	0	0	1	90	7370.51727	0.0000	—	—
13.33.140.74	192.168.1.150	98	7252	0	0	98	7252	1060.89518	19056.0205	0	3

Conversations has multiple tabs, but by looking at the number of packets in the IPv4 and IPv6 tabs, it can be deduced that this attack probably used IPv4 because of the large number of items inside it.

Inside the IPv4 tab, filter the content by largest packets. This will organize the data to show which conversation had the most packets transfered and in turn help show what would have caused a spike in traffic.



Wireshark · Conversations · 8_15_morning_merge.pcap

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
5.101.80.28	192.168.1.23	171,697	419 M	98,912	7939 k	72,785	411 M	424.20309	10170.3384	6245	323 k
192.168.1.150	192.168.1.151	87,069	62 M	33,235	6803 k	53,834	56 M	0.946665	20366.5159	2672	22 k
52.95.251.150	192.168.1.10	30,045	4965 k	13,690	2196 k	16,355	2768 k	2.542366	20363.0889	862	1087
192.168.1.10	208.77.78.204	28,080	28 M	9,579	587 k	18,501	28 M	9879.93058	485.9535	9664	467 k
192.168.1.11	208.77.78.204	10,735	12 M	1,701	169 k	9,034	12 M	9821.34295	320.1840	4238	309 k
13.107.4.50	192.168.1.12	10,624	10 M	6,396	9896 k	4,228	295 k	0.000000	67.6499	1170 k	34 k
173.194.162.231	192.168.1.11	9,405	10 M	7,723	10 M	1,682	221 k	0234.35986	55.1814	1533 k	32 k
192.168.1.1	192.168.1.100	8,220	608 k	0	0	8,220	608 k	1.718621	20363.2145	0	239
192.168.0.1	192.168.1.10	8,185	630 k	0	0	8,185	630 k	417.22322	14909.0908	0	338
192.168.1.11	208.77.78.205	7,126	8439 k	1,006	95 k	6,120	8343 k	4024.82874	5890.5260	130	11 k
8.8.8.8	192.168.1.15	6,388	1120 k	3,125	817 k	3,263	302 k	1.311905	20366.0377	321	118
37.131.192.151	192.168.1.20	4,968	947 k	2,516	486 k	2,452	460 k	7.664368	20352.1282	191	181
172.217.1.138	192.168.1.10	4,625	4460 k	2,916	4336 k	1,709	124 k	3659.80553	6706.8031	5172	148
8.253.112.111	192.168.1.15	2,241	2115 k	1,305	2042 k	936	73 k	536.29618	88.8301	183 k	6642
172.217.1.142	192.168.1.10	1,826	1369 k	1,091	1277 k	735	92 k	7736.32646	2625.0035	3891	282

☐ Name resolution ☐ Limit to display filter ☐ Absolute start time

The largest amount of packets was transfered between Address A (5.101.80.28) and Address B (192.168.1.23). Referencing the network diagram for the competition, Address B (192.168.1.23) is the Orko Billpayer Site (Windows Server 2016). That leaves Address A (5.101.80.28) as the answer.

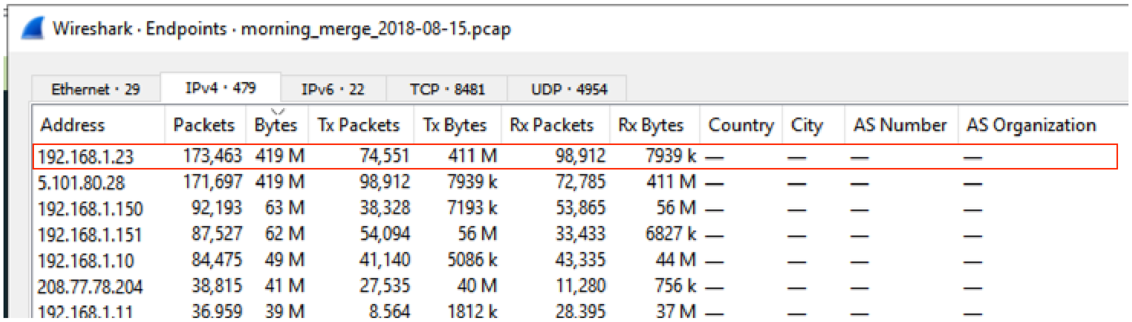
Answer: 5.101.80.28

1.3 Odei 2

What internal IP address was being consistently visited?

1.3.1 Solution:

Continue in the same merged morning pcap file. Go to **statistics>Endpoints**. In Endpoints click **IPv4** and organize it by **Bytes**.



Wireshark · Endpoints · morning_merge_2018-08-15.pcap

Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes	Country	City	AS Number	AS Organization
192.168.1.23	173,463	419 M	74,551	411 M	98,912	7939 k	—	—	—	—
5.101.80.28	171,697	419 M	98,912	7939 k	72,785	411 M	—	—	—	—
192.168.1.150	92,193	63 M	38,328	7193 k	53,865	56 M	—	—	—	—
192.168.1.151	87,527	62 M	54,094	56 M	33,433	6827 k	—	—	—	—
192.168.1.10	84,475	49 M	41,140	5086 k	43,335	44 M	—	—	—	—
208.77.78.204	38,815	41 M	27,535	40 M	11,280	756 k	—	—	—	—
192.168.1.11	36,959	39 M	8,564	1812 k	28,395	37 M	—	—	—	—

Notice the IP address with the most visits is **192.168.1.23**.

Answer: 192.168.1.23

1.4 Odei 3

What is the external IP of this website?

1.4.1 Solution:

From Odei 1, its known that the source IP Address is **5.101.80.28** and from the question its known that this source IP address is probably requesting data from this external IP. Continue in the merged morning pcap and filter by **ip.addr == 5.101.80.28 && http.request.method == GET** and by looking specifically at the **http.host** field.

[12]:

```
Set-Alias -Name tshark -Value "C:\Program Files\Wireshark\tshark.exe"
tshark -r "C:\Users\Administrator\Artifacts\pcaps\8_15_morning_merge.pcap" -Y "ip.addr == 5.101.80.28 && http.request.method == GET" -T "fields" -e "http.host" > "C:\Users\Administrator\Artifacts\pcaps\output1.txt"
Get-Content "C:\Users\Administrator\Artifacts\pcaps\output1.txt" -First 10
```

```
50.0.0.245
50.0.0.245
50.0.0.245
50.0.0.245
50.0.0.245
50.0.0.245
50.0.0.245
50.0.0.245
50.0.0.245
50.0.0.245
```

Answer: 50.0.0.245

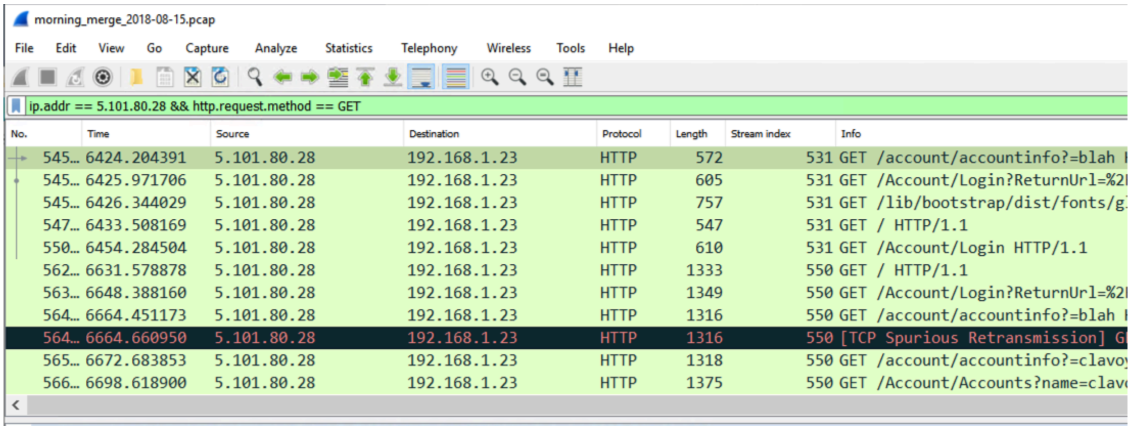
1.5 Odei 4

The requester tried getting account info for several user names. What is the first username they requested?

1.5.1 Solution:

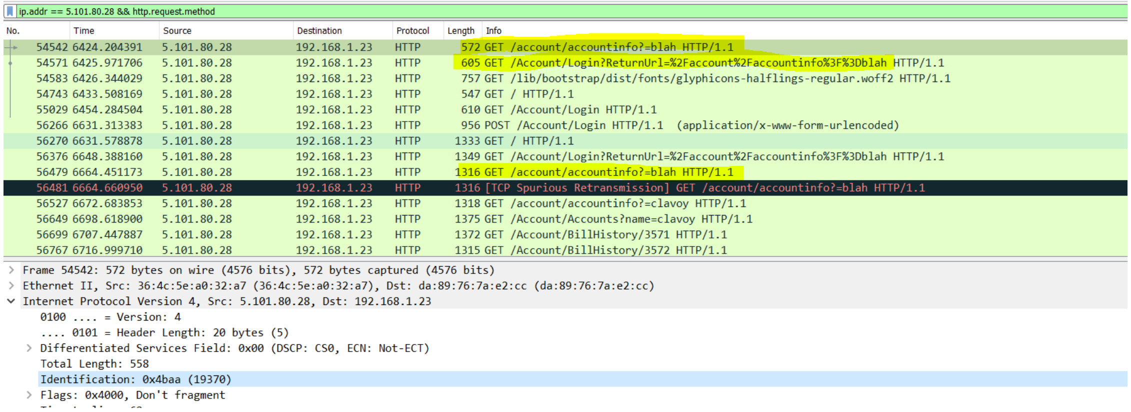
From Odei 1, it's known that the requesters IP address is **5.101.80.28**. From the context of the question, it's known that the username was requested by this IP, so it probably used a GET request.

Continue in the merged morning pcap file and filter again by **ip.addr == 5.101.80.28 && http.request.method == GET.**.



```
ip.addr == 5.101.80.28 && http.request.method == GET

No.     Time            Source          Destination     Protocol  Length  Stream index  Info
545…  6424.204391   5.101.80.28   192.168.1.23   HTTP      572     531 GET /account/accountinfo?=blah H
545…  6425.971706   5.101.80.28   192.168.1.23   HTTP      605     531 GET /Account/Login?ReturnUrl=%2
545…  6426.344029   5.101.80.28   192.168.1.23   HTTP      757     531 GET /lib/bootstrap/dist/fonts/g
547…  6433.508169   5.101.80.28   192.168.1.23   HTTP      547     531 GET / HTTP/1.1
550…  6454.284504   5.101.80.28   192.168.1.23   HTTP      610     531 GET /Account/Login HTTP/1.1
562…  6631.578878   5.101.80.28   192.168.1.23   HTTP      1333    550 GET / HTTP/1.1
563…  6648.388160   5.101.80.28   192.168.1.23   HTTP      1349    550 GET /Account/Login?ReturnUrl=%2
564…  6664.451173   5.101.80.28   192.168.1.23   HTTP      1316    550 GET /account/accountinfo?=blah H
564…  6664.660950   5.101.80.28   192.168.1.23   HTTP      1316    550 [TCP Spurious Retransmission] G
565…  6672.683853   5.101.80.28   192.168.1.23   HTTP      1318    550 GET /account/accountinfo?=clavoy
566…  6698.618900   5.101.80.28   192.168.1.23   HTTP      1375    550 GET /Account/Accounts?name=clav
```

Look in the info column and notice that multiple packets are requesting account info for **blah**.



```
ip.addr == 5.101.80.28 && http.request.method

No.     Time          Source         Destination    Protocol  Length  Info
  54542 6424.204391   5.101.80.28    192.168.1.23   HTTP       572 GET /account/accountinfo?=blah HTTP/1.1
  54571 6425.971706   5.101.80.28    192.168.1.23   HTTP       605 GET /Account/Login?ReturnUrl=%2Faccount%2Faccountinfo%3F%3Dblah HTTP/1.1
  54583 6426.344029   5.101.80.28    192.168.1.23   HTTP       757 GET /lib/bootstrap/dist/fonts/glyphicons-halflings-regular.woff2 HTTP/1.1
  54743 6433.508169   5.101.80.28    192.168.1.23   HTTP       547 GET / HTTP/1.1
  55029 6454.284504   5.101.80.28    192.168.1.23   HTTP       610 GET /Account/Login HTTP/1.1
  56266 6631.313383   5.101.80.28    192.168.1.23   HTTP       956 POST /Account/Login HTTP/1.1  (application/x-www-form-urlencoded)
  56270 6631.578878   5.101.80.28    192.168.1.23   HTTP      1333 GET / HTTP/1.1
  56376 6648.388160   5.101.80.28    192.168.1.23   HTTP      1349 GET /Account/Login?ReturnUrl=%2Faccount%2Faccountinfo%3F%3Dblah HTTP/1.1
  56479 6664.451173   5.101.80.28    192.168.1.23   HTTP      1316 GET /account/accountinfo?=blah HTTP/1.1
  56481 6664.660950   5.101.80.28    192.168.1.23   HTTP      1316 [TCP Spurious Retransmission] GET /account/accountinfo?=blah HTTP/1.1
  56527 6672.683853   5.101.80.28    192.168.1.23   HTTP      1318 GET /account/accountinfo?=clavoy HTTP/1.1
  56649 6698.618900   5.101.80.28    192.168.1.23   HTTP      1375 GET /Account/Accounts?name=clavoy HTTP/1.1
  56699 6707.447887   5.101.80.28    192.168.1.23   HTTP      1372 GET /Account/BillHistory/3571 HTTP/1.1
  56767 6716.999710   5.101.80.28    192.168.1.23   HTTP      1315 GET /Account/BillHistory/3572 HTTP/1.1

> Frame 54542: 572 bytes on wire (4576 bits), 572 bytes captured (4576 bits)
> Ethernet II, Src: 36:4c:5e:a0:32:a7 (36:4c:5e:a0:32:a7), Dst: da:89:76:7a:e2:cc (da:89:76:7a:e2:cc)
v Internet Protocol Version 4, Src: 5.101.80.28, Dst: 192.168.1.23
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 558
    Identification: 0x4baa (19370)
  > Flags: 0x4000, Don't fragment
```

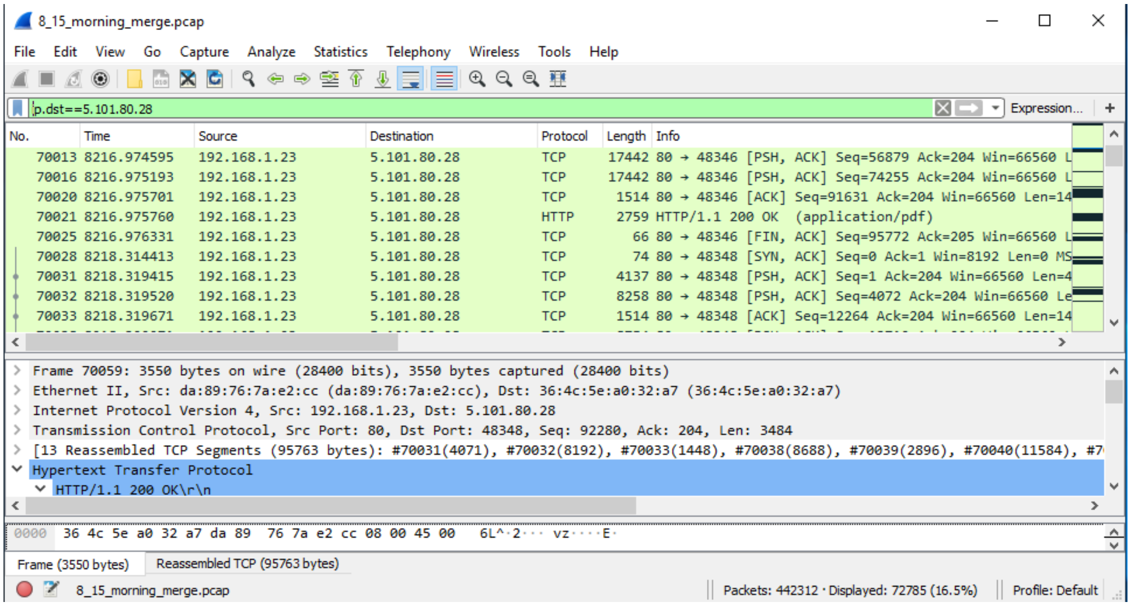
Answer: blah

1.6 Odei 5

What is the content type of the returned data from the URL that is consistently hit by the requester?

1.6.1 Solution:

It's known from Odei 1 that the IP address **5.101.80.28** is being used by the hacker and it can be deduced that this IP is most likely the destination.


8_15_morning_merge.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...

```
ip.dst==5.101.80.28
```

```
No.   Time         Source        Destination   Protocol Length Info
70013 8216.974595  192.168.1.23  5.101.80.28   TCP      17442 80 → 48346 [PSH, ACK] Seq=56879 Ack=204 Win=66560 L
70016 8216.975193  192.168.1.23  5.101.80.28   TCP      17442 80 → 48346 [PSH, ACK] Seq=74255 Ack=204 Win=66560 L
70020 8216.975701  192.168.1.23  5.101.80.28   TCP       1514 80 → 48346 [ACK] Seq=91631 Ack=204 Win=66560 Len=14
70021 8216.975760  192.168.1.23  5.101.80.28   HTTP      2759 HTTP/1.1 200 OK  (application/pdf)
70025 8216.976331  192.168.1.23  5.101.80.28   TCP         66 80 → 48346 [FIN, ACK] Seq=95772 Ack=205 Win=66560 L
70028 8218.314413  192.168.1.23  5.101.80.28   TCP         74 80 → 48348 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MS
70031 8218.319415  192.168.1.23  5.101.80.28   TCP       4137 80 → 48348 [PSH, ACK] Seq=1 Ack=204 Win=66560 Len=4
70032 8218.319520  192.168.1.23  5.101.80.28   TCP       8258 80 → 48348 [PSH, ACK] Seq=4072 Ack=204 Win=66560 Le
70033 8218.319671  192.168.1.23  5.101.80.28   TCP       1514 80 → 48348 [ACK] Seq=12264 Ack=204 Win=66560 Len=14
```

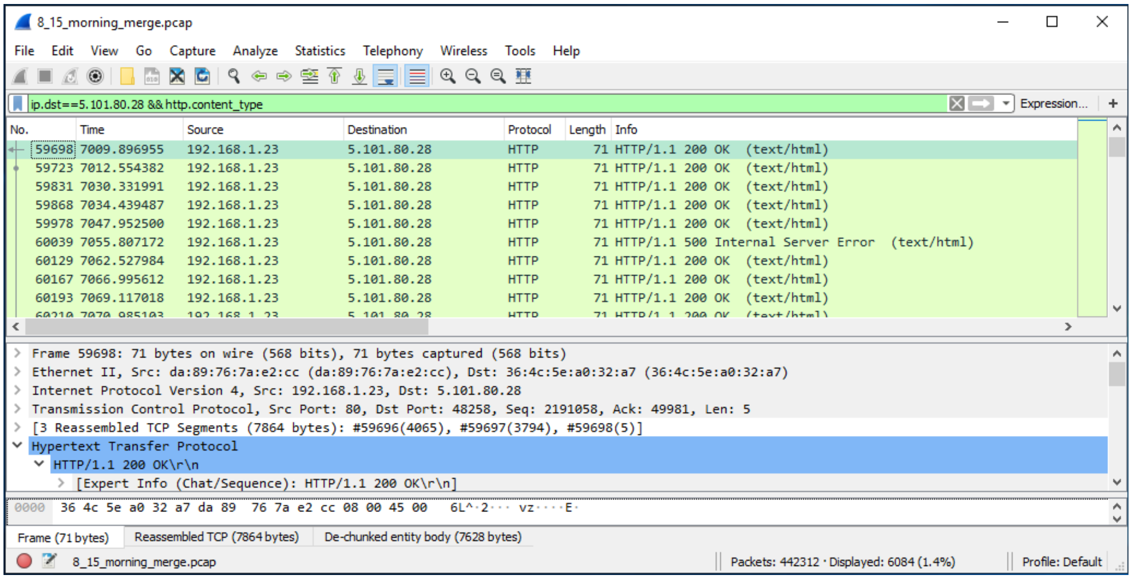
```
> Frame 70059: 3550 bytes on wire (28400 bits), 3550 bytes captured (28400 bits)
> Ethernet II, Src: da:89:76:7a:e2:cc (da:89:76:7a:e2:cc), Dst: 36:4c:5e:a0:32:a7 (36:4c:5e:a0:32:a7)
> Internet Protocol Version 4, Src: 192.168.1.23, Dst: 5.101.80.28
> Transmission Control Protocol, Src Port: 80, Dst Port: 48348, Seq: 92280, Ack: 204, Len: 3484
> [13 Reassembled TCP Segments (95763 bytes): #70031(4071), #70032(8192), #70033(1448), #70038(8688), #70039(2896), #70040(11584), #7
v Hypertext Transfer Protocol
  v HTTP/1.1 200 OK\r\n
```

```
0000  36 4c 5e a0 32 a7 da 89  76 7a e2 cc 08 00 45 00   6L^·2··· vz····E·
```

Frame (3550 bytes)   Reassembled TCP (95763 bytes)

8_15_morning_merge.pcap   Packets: 442312 · Displayed: 72785 (16.5%)   Profile: Default

Filtering by **ip.dst==5.101.80.28** results in over 72,000 packets, additional filtering is needed. The question is asking for the "content type" of the returned data. Filtering by the IP destination (**ip.dst==5.101.80.28**) and content type (**http.content_type**) results in a little over 6000 packets.


8_15_morning_merge.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...

```
ip.dst==5.101.80.28 && http.content_type
```

```
No.   Time         Source        Destination   Protocol Length Info
59698 7009.896955  192.168.1.23  5.101.80.28   HTTP      71 HTTP/1.1 200 OK  (text/html)
59723 7012.554382  192.168.1.23  5.101.80.28   HTTP      71 HTTP/1.1 200 OK  (text/html)
59831 7030.331991  192.168.1.23  5.101.80.28   HTTP      71 HTTP/1.1 200 OK  (text/html)
59868 7034.439487  192.168.1.23  5.101.80.28   HTTP      71 HTTP/1.1 200 OK  (text/html)
59978 7047.952500  192.168.1.23  5.101.80.28   HTTP      71 HTTP/1.1 200 OK  (text/html)
60039 7055.807172  192.168.1.23  5.101.80.28   HTTP      71 HTTP/1.1 500 Internal Server Error  (text/html)
60129 7062.527984  192.168.1.23  5.101.80.28   HTTP      71 HTTP/1.1 200 OK  (text/html)
60167 7066.995612  192.168.1.23  5.101.80.28   HTTP      71 HTTP/1.1 200 OK  (text/html)
60193 7069.117018  192.168.1.23  5.101.80.28   HTTP      71 HTTP/1.1 200 OK  (text/html)
60210 7070.985103  192.168.1.23  5.101.80.28   HTTP      71 HTTP/1.1 200 OK  (text/html)
```

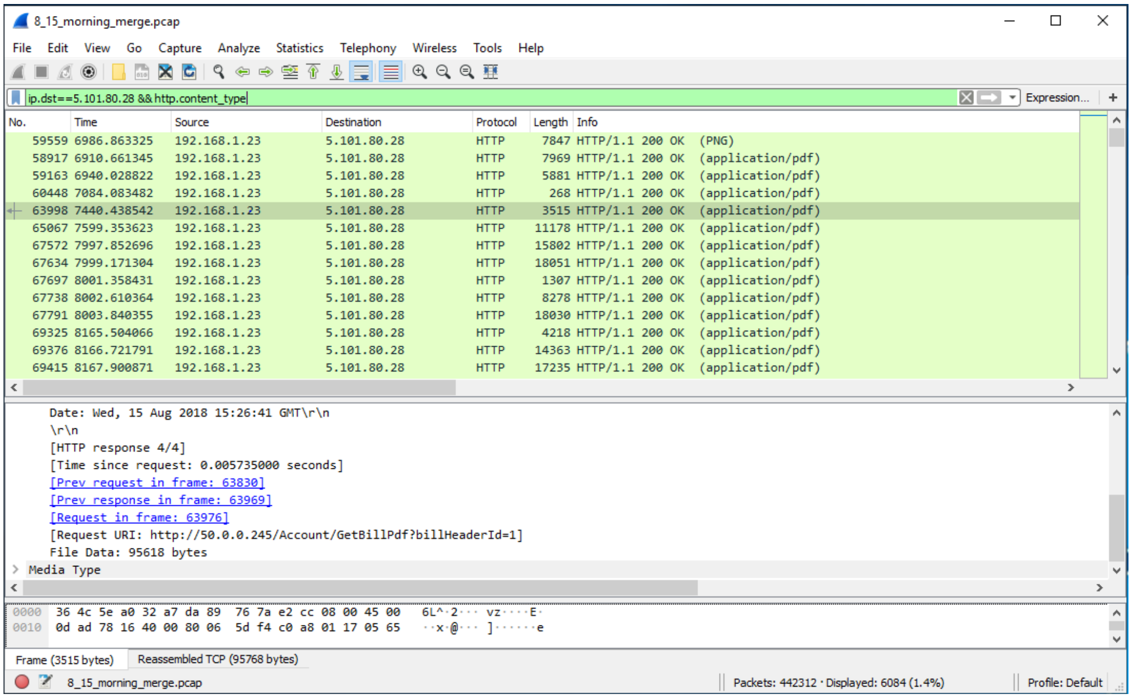
```
> Frame 59698: 71 bytes on wire (568 bits), 71 bytes captured (568 bits)
> Ethernet II, Src: da:89:76:7a:e2:cc (da:89:76:7a:e2:cc), Dst: 36:4c:5e:a0:32:a7 (36:4c:5e:a0:32:a7)
> Internet Protocol Version 4, Src: 192.168.1.23, Dst: 5.101.80.28
> Transmission Control Protocol, Src Port: 80, Dst Port: 48258, Seq: 2191058, Ack: 49981, Len: 5
> [3 Reassembled TCP Segments (7864 bytes): #59696(4065), #59697(3794), #59698(5)]
v Hypertext Transfer Protocol
  v HTTP/1.1 200 OK\r\n
    > [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
```

```
0000  36 4c 5e a0 32 a7 da 89  76 7a e2 cc 08 00 45 00   6L^·2··· vz····E·
```

Frame (71 bytes)   Reassembled TCP (7864 bytes)   De-chunked entity body (7628 bytes)

8_15_morning_merge.pcap   Packets: 442312 · Displayed: 6084 (1.4%)   Profile: Default

It's known from the question that this content type is being returned by a URL that is consistently hit by the requester. With that knowledge, organize the packets by the **info** column. This displays data a few different contents types, but primary **application/pdf**.


8_15_morning_merge.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...

```
ip.dst==5.101.80.28 && http.content_type
```

```
No.    Time           Source          Destination    Protocol Length Info
 59559 6986.863325    192.168.1.23    5.101.80.28    HTTP      7847 HTTP/1.1 200 OK  (PNG)
 58917 6910.661345    192.168.1.23    5.101.80.28    HTTP      7969 HTTP/1.1 200 OK  (application/pdf)
 59163 6940.028822    192.168.1.23    5.101.80.28    HTTP      5881 HTTP/1.1 200 OK  (application/pdf)
 60448 7084.083482    192.168.1.23    5.101.80.28    HTTP       268 HTTP/1.1 200 OK  (application/pdf)
 63998 7440.438542    192.168.1.23    5.101.80.28    HTTP      3515 HTTP/1.1 200 OK  (application/pdf)
 65067 7599.353623    192.168.1.23    5.101.80.28    HTTP     11178 HTTP/1.1 200 OK  (application/pdf)
 67572 7997.852696    192.168.1.23    5.101.80.28    HTTP     15802 HTTP/1.1 200 OK  (application/pdf)
 67634 7999.171304    192.168.1.23    5.101.80.28    HTTP     18051 HTTP/1.1 200 OK  (application/pdf)
 67697 8001.358431    192.168.1.23    5.101.80.28    HTTP      1307 HTTP/1.1 200 OK  (application/pdf)
 67738 8002.610364    192.168.1.23    5.101.80.28    HTTP      8278 HTTP/1.1 200 OK  (application/pdf)
 67791 8003.840355    192.168.1.23    5.101.80.28    HTTP     18030 HTTP/1.1 200 OK  (application/pdf)
 69325 8165.504066    192.168.1.23    5.101.80.28    HTTP      4218 HTTP/1.1 200 OK  (application/pdf)
 69376 8166.721791    192.168.1.23    5.101.80.28    HTTP     14363 HTTP/1.1 200 OK  (application/pdf)
 69415 8167.900871    192.168.1.23    5.101.80.28    HTTP     17235 HTTP/1.1 200 OK  (application/pdf)
```

```
    Date: Wed, 15 Aug 2018 15:26:41 GMT\r\n
    \r\n
    [HTTP response 4/4]
    [Time since request: 0.005735000 seconds]
    [Prev request in frame: 63830]
    [Prev response in frame: 63969]
    [Request in frame: 63976]
    [Request URI: http://50.0.0.245/Account/GetBillPdf?billHeaderId=1]
    File Data: 95618 bytes
> Media Type
```

```
0000  36 4c 5e a0 32 a7 da 89  76 7a e2 cc 08 00 45 00   6L^·2···  vz····E·
0010  0d ad 78 16 40 00 80 06  5d f4 c0 a8 01 17 05 65   ··x·@···  ]······e
```

```
Frame (3515 bytes)   Reassembled TCP (95768 bytes)
8_15_morning_merge.pcap   Packets: 442312 · Displayed: 6084 (1.4%)   Profile: Default
```

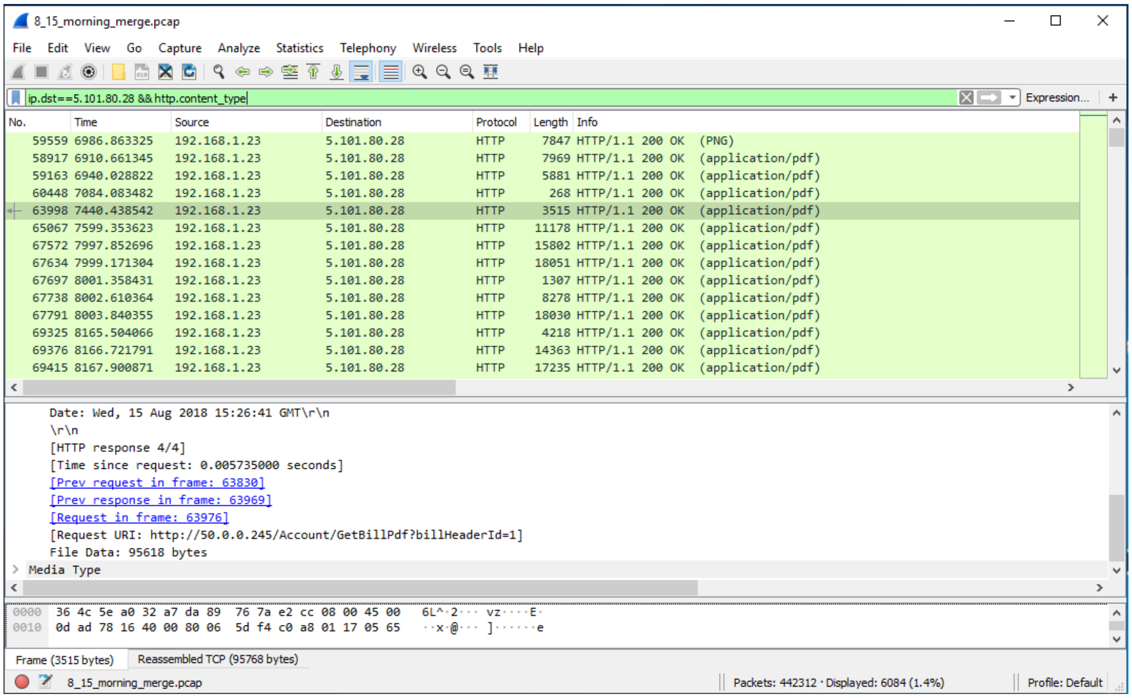
Answer: application/pdf

1.7 Odei 6

What is the the full request URL to get the PDF for user 1's bill?

1.7.1 Solution:

Starting from Odei 5, it's already known that PDFs have been uploaded and that these packets have a Request URI where packets are being delivered to.


8_15_morning_merge.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression…
+

ip.dst==5.101.80.28 && http.content_type

```
No.   Time        Source        Destination  Protocol Length Info
59559 6986.863325 192.168.1.23  5.101.80.28  HTTP      7847 HTTP/1.1 200 OK  (PNG)
58917 6910.661345 192.168.1.23  5.101.80.28  HTTP      7969 HTTP/1.1 200 OK  (application/pdf)
59163 6940.028822 192.168.1.23  5.101.80.28  HTTP      5881 HTTP/1.1 200 OK  (application/pdf)
60448 7084.083482 192.168.1.23  5.101.80.28  HTTP       268 HTTP/1.1 200 OK  (application/pdf)
63998 7440.438542 192.168.1.23  5.101.80.28  HTTP      3515 HTTP/1.1 200 OK  (application/pdf)
65067 7599.353623 192.168.1.23  5.101.80.28  HTTP     11178 HTTP/1.1 200 OK  (application/pdf)
67572 7997.852696 192.168.1.23  5.101.80.28  HTTP     15802 HTTP/1.1 200 OK  (application/pdf)
67634 7999.171304 192.168.1.23  5.101.80.28  HTTP     18051 HTTP/1.1 200 OK  (application/pdf)
67697 8001.358431 192.168.1.23  5.101.80.28  HTTP      1307 HTTP/1.1 200 OK  (application/pdf)
67738 8002.610364 192.168.1.23  5.101.80.28  HTTP      8278 HTTP/1.1 200 OK  (application/pdf)
67791 8003.840355 192.168.1.23  5.101.80.28  HTTP     18030 HTTP/1.1 200 OK  (application/pdf)
69325 8165.504066 192.168.1.23  5.101.80.28  HTTP      4218 HTTP/1.1 200 OK  (application/pdf)
69376 8166.721791 192.168.1.23  5.101.80.28  HTTP     14363 HTTP/1.1 200 OK  (application/pdf)
69415 8167.900871 192.168.1.23  5.101.80.28  HTTP     17235 HTTP/1.1 200 OK  (application/pdf)
```

```
    Date: Wed, 15 Aug 2018 15:26:41 GMT\r\n
    \r\n
    [HTTP response 4/4]
    [Time since request: 0.005735000 seconds]
    [Prev request in frame: 63830]
    [Prev response in frame: 63969]
    [Request in frame: 63976]
    [Request URI: http://50.0.0.245/Account/GetBillPdf?billHeaderId=1]
    File Data: 95618 bytes
> Media Type
```

```
0000  36 4c 5e a0 32 a7 da 89  76 7a e2 cc 08 00 45 00   6L^·2···  vz····E·
0010  0d ad 78 16 40 00 80 06  5d f4 c0 a8 01 17 05 65   ··x·@···  ]······e
```

Frame (3515 bytes)   Reassembled TCP (95768 bytes)

8_15_morning_merge.pcap   Packets: 442312 · Displayed: 6084 (1.4%)   Profile: Default

Examining the Request URIs from the application/pdfs, notice a full request URL of **http://50.0.0.245/Account/GetBillPdf?billHeaderId=1**.

Answer: http://50.0.0.245/Account/GetBillPdf?billHeaderId=1

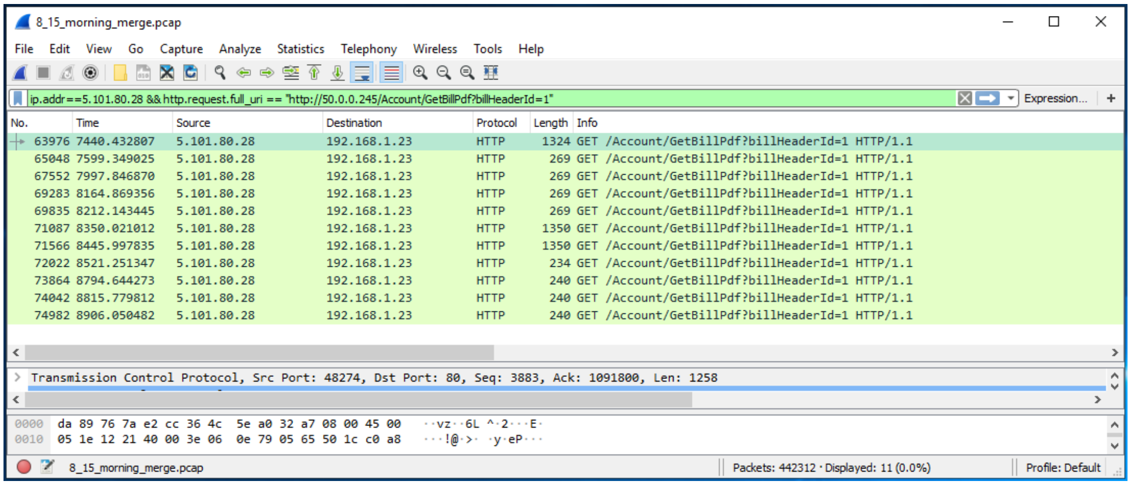
1.8 Odei 7

What did the attacker use to exfiltrate the PDFs?

1.8.1 Solution:

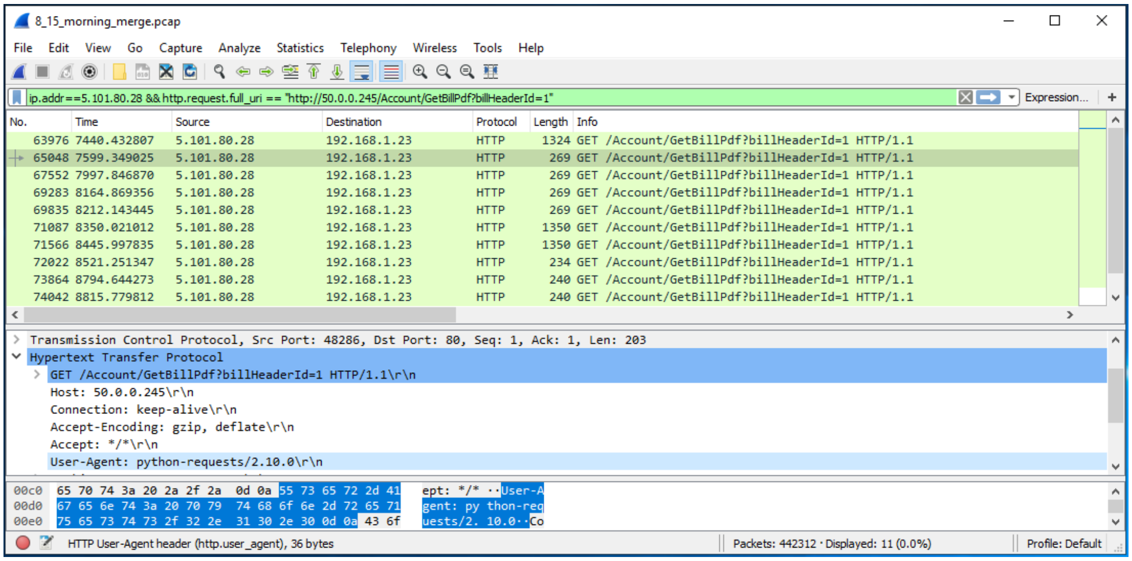
It's known from Odei 1 that the attacker uses the IP **5.101.80.28** and that the full request URL is **http://50.0.0.245/Account/GetBillPdf?billHeaderId=1** to get the PDFs.

Continue in the merged morning pcap file with the filter **ip.addr==5.101.80.28 && http.request.full_uri == "http://50.0.0.245/Account/GetBillPdf?billHeaderId=1"**. This filters the content down to 11 packets.


8_15_morning_merge.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression… +

```
ip.addr==5.101.80.28 && http.request.full_uri == "http://50.0.0.245/Account/GetBillPdf?billHeaderId=1"

No.   Time          Source        Destination   Protocol Length Info
63976 7440.432807   5.101.80.28   192.168.1.23  HTTP     1324 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
65048 7599.349025   5.101.80.28   192.168.1.23  HTTP      269 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
67552 7997.846870   5.101.80.28   192.168.1.23  HTTP      269 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
69283 8164.869356   5.101.80.28   192.168.1.23  HTTP      269 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
69835 8212.143445   5.101.80.28   192.168.1.23  HTTP      269 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
71087 8350.021012   5.101.80.28   192.168.1.23  HTTP     1350 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
71566 8445.997835   5.101.80.28   192.168.1.23  HTTP     1350 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
72022 8521.251347   5.101.80.28   192.168.1.23  HTTP      234 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
73864 8794.644273   5.101.80.28   192.168.1.23  HTTP      240 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
74042 8815.779812   5.101.80.28   192.168.1.23  HTTP      240 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
74982 8906.050482   5.101.80.28   192.168.1.23  HTTP      240 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1

> Transmission Control Protocol, Src Port: 48274, Dst Port: 80, Seq: 3883, Ack: 1091800, Len: 1258

0000  da 89 76 7a e2 cc 36 4c  5e a0 32 a7 08 00 45 00   ··vz··6L ^·2···E·
0010  05 1e 12 21 40 00 3e 06  0e 79 05 65 50 1c c0 a8   ···!@·>· ·y·eP···

8_15_morning_merge.pcap        Packets: 442312 · Displayed: 11 (0.0%)        Profile: Default
```

The question wants to know what the attacker used to exfiltrate the PDFs. When examining the filtered packets, locate the **user-agent** field under the Hypertext drop-down. This field will list the software (a software agent) that is acting on behalf of a user to send data. A few of the packet have a user agent of Mozilla, which is Firefox. However, the other packets have a user agent of Python-Requests. Knowing that Firefox is used for typical browsing purposes, it can be deduced that **python-requests** is what the attacker used to exfiltrate the PDFs.


8_15_morning_merge.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression… +

```
ip.addr==5.101.80.28 && http.request.full_uri == "http://50.0.0.245/Account/GetBillPdf?billHeaderId=1"

No.   Time          Source        Destination   Protocol Length Info
63976 7440.432807   5.101.80.28   192.168.1.23  HTTP     1324 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
65048 7599.349025   5.101.80.28   192.168.1.23  HTTP      269 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
67552 7997.846870   5.101.80.28   192.168.1.23  HTTP      269 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
69283 8164.869356   5.101.80.28   192.168.1.23  HTTP      269 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
69835 8212.143445   5.101.80.28   192.168.1.23  HTTP      269 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
71087 8350.021012   5.101.80.28   192.168.1.23  HTTP     1350 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
71566 8445.997835   5.101.80.28   192.168.1.23  HTTP     1350 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
72022 8521.251347   5.101.80.28   192.168.1.23  HTTP      234 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
73864 8794.644273   5.101.80.28   192.168.1.23  HTTP      240 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1
74042 8815.779812   5.101.80.28   192.168.1.23  HTTP      240 GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1

> Transmission Control Protocol, Src Port: 48286, Dst Port: 80, Seq: 1, Ack: 1, Len: 203
v Hypertext Transfer Protocol
  > GET /Account/GetBillPdf?billHeaderId=1 HTTP/1.1\r\n
    Host: 50.0.0.245\r\n
    Connection: keep-alive\r\n
    Accept-Encoding: gzip, deflate\r\n
    Accept: */*\r\n
    User-Agent: python-requests/2.10.0\r\n

00c0  65 70 74 3a 20 2a 2f 2a  0d 0a 55 73 65 72 2d 41   ept: */* ··User-A
00d0  67 65 6e 74 3a 20 70 79  74 68 6f 6e 2d 72 65 71   gent: py thon-req
00e0  75 65 73 74 73 2f 32 2e  31 30 2e 30 0d 0a 43 6f   uests/2. 10.0··Co

HTTP User-Agent header (http.user_agent), 36 bytes        Packets: 442312 · Displayed: 11 (0.0%)        Profile: Default
```

Answer: python-requests

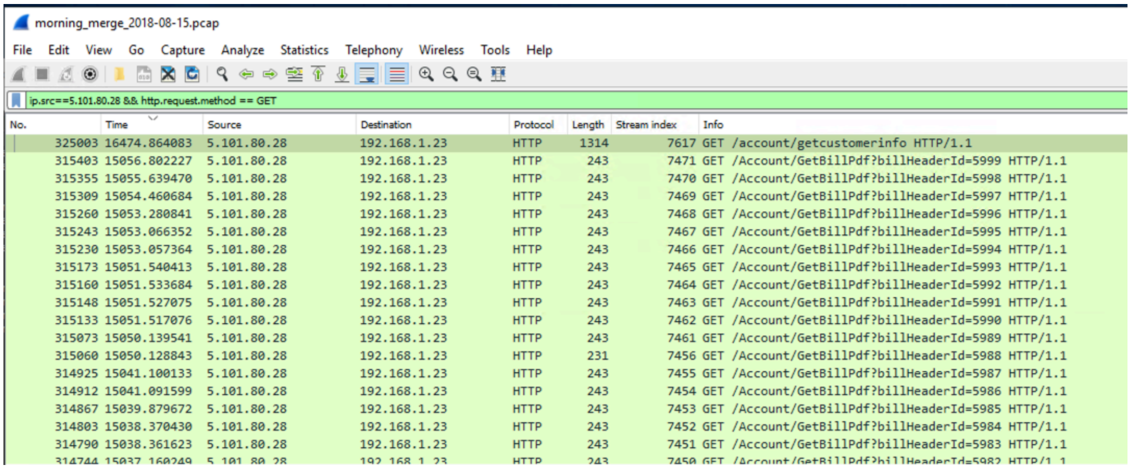
1.9 Odei 8

After the attacker exfiltrated the PDFs, the requester visited another URL that allowed them to access more personal information (such as SSN) from all of the users. What is the URL?

1.9.1 Solution:

It's known from the question that the attacker (5.101.80.28) visited the URL after exfiltrating the PDFs that were listed in Odei 5.

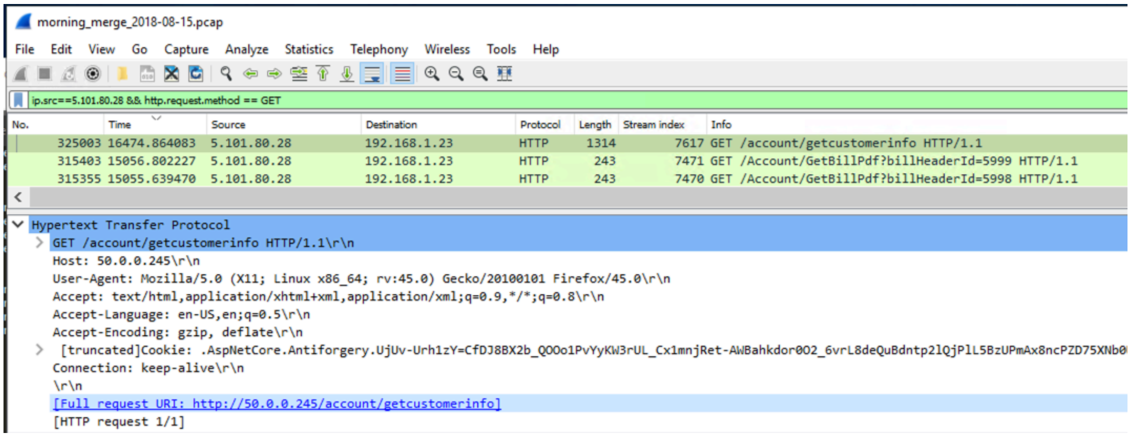
Continue in the merged morning pcap file and filter by the attackers IP and the GET request method: **ip.addr==5.101.80.28 && http.request.method == GET**. Since the attacker visited this URL after exfiltrating the PDF files, organize the filtered content by the time column.



```
ip.src==5.101.80.28 && http.request.method == GET
```

```
No.     Time            Source          Destination     Protocol  Length  Stream index  Info
 325003 16474.864083 5.101.80.28     192.168.1.23     HTTP      1314      7617 GET /account/getcustomerinfo HTTP/1.1
 315403 15056.802227 5.101.80.28     192.168.1.23     HTTP       243      7471 GET /Account/GetBillPdf?billHeaderId=5999 HTTP/1.1
 315355 15055.639470 5.101.80.28     192.168.1.23     HTTP       243      7470 GET /Account/GetBillPdf?billHeaderId=5998 HTTP/1.1
 315309 15054.460684 5.101.80.28     192.168.1.23     HTTP       243      7469 GET /Account/GetBillPdf?billHeaderId=5997 HTTP/1.1
 315260 15053.280841 5.101.80.28     192.168.1.23     HTTP       243      7468 GET /Account/GetBillPdf?billHeaderId=5996 HTTP/1.1
 315243 15053.066352 5.101.80.28     192.168.1.23     HTTP       243      7467 GET /Account/GetBillPdf?billHeaderId=5995 HTTP/1.1
 315230 15053.057364 5.101.80.28     192.168.1.23     HTTP       243      7466 GET /Account/GetBillPdf?billHeaderId=5994 HTTP/1.1
 315173 15051.540413 5.101.80.28     192.168.1.23     HTTP       243      7465 GET /Account/GetBillPdf?billHeaderId=5993 HTTP/1.1
 315160 15051.533684 5.101.80.28     192.168.1.23     HTTP       243      7464 GET /Account/GetBillPdf?billHeaderId=5992 HTTP/1.1
 315148 15051.527075 5.101.80.28     192.168.1.23     HTTP       243      7463 GET /Account/GetBillPdf?billHeaderId=5991 HTTP/1.1
 315133 15051.517076 5.101.80.28     192.168.1.23     HTTP       243      7462 GET /Account/GetBillPdf?billHeaderId=5990 HTTP/1.1
 315073 15050.139541 5.101.80.28     192.168.1.23     HTTP       243      7461 GET /Account/GetBillPdf?billHeaderId=5989 HTTP/1.1
 315060 15050.128843 5.101.80.28     192.168.1.23     HTTP       231      7456 GET /Account/GetBillPdf?billHeaderId=5988 HTTP/1.1
 314925 15041.100133 5.101.80.28     192.168.1.23     HTTP       243      7455 GET /Account/GetBillPdf?billHeaderId=5987 HTTP/1.1
 314912 15041.091599 5.101.80.28     192.168.1.23     HTTP       243      7454 GET /Account/GetBillPdf?billHeaderId=5986 HTTP/1.1
 314867 15039.879672 5.101.80.28     192.168.1.23     HTTP       243      7453 GET /Account/GetBillPdf?billHeaderId=5985 HTTP/1.1
 314803 15038.370430 5.101.80.28     192.168.1.23     HTTP       243      7452 GET /Account/GetBillPdf?billHeaderId=5984 HTTP/1.1
 314790 15038.361623 5.101.80.28     192.168.1.23     HTTP       243      7451 GET /Account/GetBillPdf?billHeaderId=5983 HTTP/1.1
 314744 15037.160249 5.101.80.28     192.168.1.23     HTTP       243      7450 GET /Account/GetBillPdf?billHeaderId=5982 HTTP/1.1
```

The only entry past the PDF exfiltrations is **/account/getcustomerinfo**. Locate the URL by clicking on this packet and navigating to the drop-down **Hyptertext Transfer Protocol**. From there the full request URI is **http://50.0.0.245/account/getcustomerinfo**.



```
ip.src==5.101.80.28 && http.request.method == GET
```

```
No.     Time            Source          Destination     Protocol  Length  Stream index  Info
 325003 16474.864083 5.101.80.28     192.168.1.23     HTTP      1314      7617 GET /account/getcustomerinfo HTTP/1.1
 315403 15056.802227 5.101.80.28     192.168.1.23     HTTP       243      7471 GET /Account/GetBillPdf?billHeaderId=5999 HTTP/1.1
 315355 15055.639470 5.101.80.28     192.168.1.23     HTTP       243      7470 GET /Account/GetBillPdf?billHeaderId=5998 HTTP/1.1

Hypertext Transfer Protocol
  > GET /account/getcustomerinfo HTTP/1.1\r\n
    Host: 50.0.0.245\r\n
    User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
    Accept-Language: en-US,en;q=0.5\r\n
    Accept-Encoding: gzip, deflate\r\n
  >  [truncated]Cookie: .AspNetCore.Antiforgery.UjUv-Urh1zY=CfDJ8BX2b_QOOo1PvYyKW3rUL_Cx1mnjRet-AWBahkdor0O2_6vrL8deQuBdntp2lQjPlL5BzUPmAx8ncPZD75XNb0
    Connection: keep-alive\r\n
    \r\n
    [Full request URI: http://50.0.0.245/account/getcustomerinfo]
    [HTTP request 1/1]
```

Since this get request is made by the attacker after the PDF file exfiltration, it can be deduced that this is the URL the attacker visited.

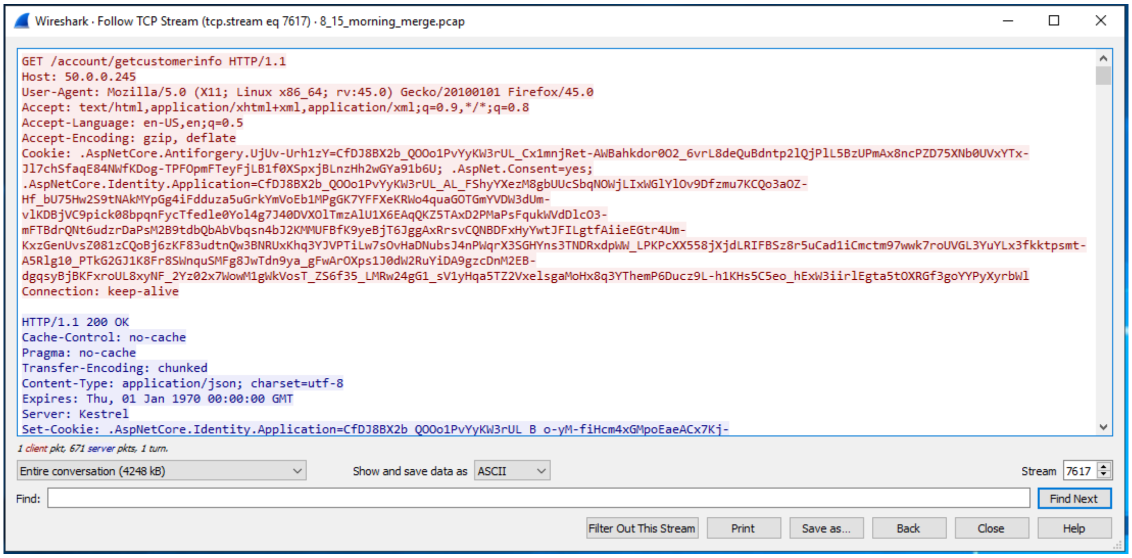
Answer: http://50.0.0.245/account/getcustomerinfo

1.10 Odei 9

What data type was returned from the last attack?

1.10.1 Solution:

Starting from the same packet from Odei 8 with the URL http://50.0.0.245/account/getcustomerinfo, right click and follow the TCP Stream. The TCP stream is needed because it logs streams of data between connections during a transfer. Since the question is asking for the data type that was returned, the stream should provide this information.



Wireshark · Follow TCP Stream (tcp.stream eq 7617) · 8_15_morning_merge.pcap

```
GET /account/getcustomerinfo HTTP/1.1
Host: 50.0.0.245
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: .AspNetCore.Antiforgery.UjUv-Urh1zY=CfDJ8BX2b_QOOo1PvYyKW3rUL_Cx1mnjRet-AWBahkdor0O2_6vrL8deQuBdntp2lQjPlL5BzUPmAx8ncPZD75XNb0UVxYTx-
Jl7chSfaqE84NWfKDog-TPFOpmFTeyFjLB1f0XSpxjBLnzHh2wGYa91b6U; .AspNet.Consent=yes;
.AspNetCore.Identity.Application=CfDJ8BX2b_QOOo1PvYyKW3rUL_AL_FShyYXezM8gbUUcSbqNOWjLIxWGlYlOv9Dfzmu7KCQo3aOZ-
Hf_bU75Hw2S9tNAkMYpGg4iFdduza5uGrkYmVoEb1MPgGK7YFFXeKRWo4quaGOTGmYVDW3dUm-
vlKDBjVC9pick08bpqnFycTfedle0Yol4g7J40DVXOlTmzAlU1X6EAqQKZ5TAxD2PMaPsFqukWVdDlcO3-
mFTBdrQNt6udzrDaPsM2B9tdbQbAbVbqsn4bJ2KMMUFBfK9yeBjT6JggAxRrsvCQNBDFxHyYwtJFILgtfAiieEGtr4Um-
KxzGenUvsZ081zCQoBj6zKF83udtnQw3BNRUxKhq3YJVPTiLw7sOvHaDNubsJ4nPWqrX3SGHYns3TNDRxdpWW_LPKPcXX558jXjdLRIFBSz8r5uCad1iCmctm97wwk7roUVGL3YuYLx3fkktpsmt-
A5Rlg10_PTkG2GJ1K8Fr8SWnquSMFg8JwTdn9ya_gFwArOXps1J0dW2RuYiDA9gzcDnM2EB-
dgqsyBjBKFxroUL8xyNF_2Yz02x7WowM1gWkVosT_ZS6f35_LMRw24gG1_sV1yHqa5TZ2VxelsgaMoHx8q3YThemP6Ducz9L-h1KHs5C5eo_hExW3iirlEgta5tOXRGf3goYYPyXyrbWl
Connection: keep-alive

HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Transfer-Encoding: chunked
Content-Type: application/json; charset=utf-8
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Server: Kestrel
Set-Cookie: .AspNetCore.Identity.Application=CfDJ8BX2b QOOo1PvYyKW3rUL B o-yM-fiHcm4xGMpoEaeACx7Kj-
```

1 client pkt, 671 server pkts, 1 turn.

Entire conversation (4248 kB)

Show and save data as ASCII

Stream 7617

Find:

The top red text shows the get request from the attacker. The blue text shows what is returned from the host machine. In the blue text there is a field called **content-type**. This field displays what type of data was returned to the host. In this case that field shows **application/json**. Since the question is asking what data type was sent back, it can be deduced that **json** is the data type.

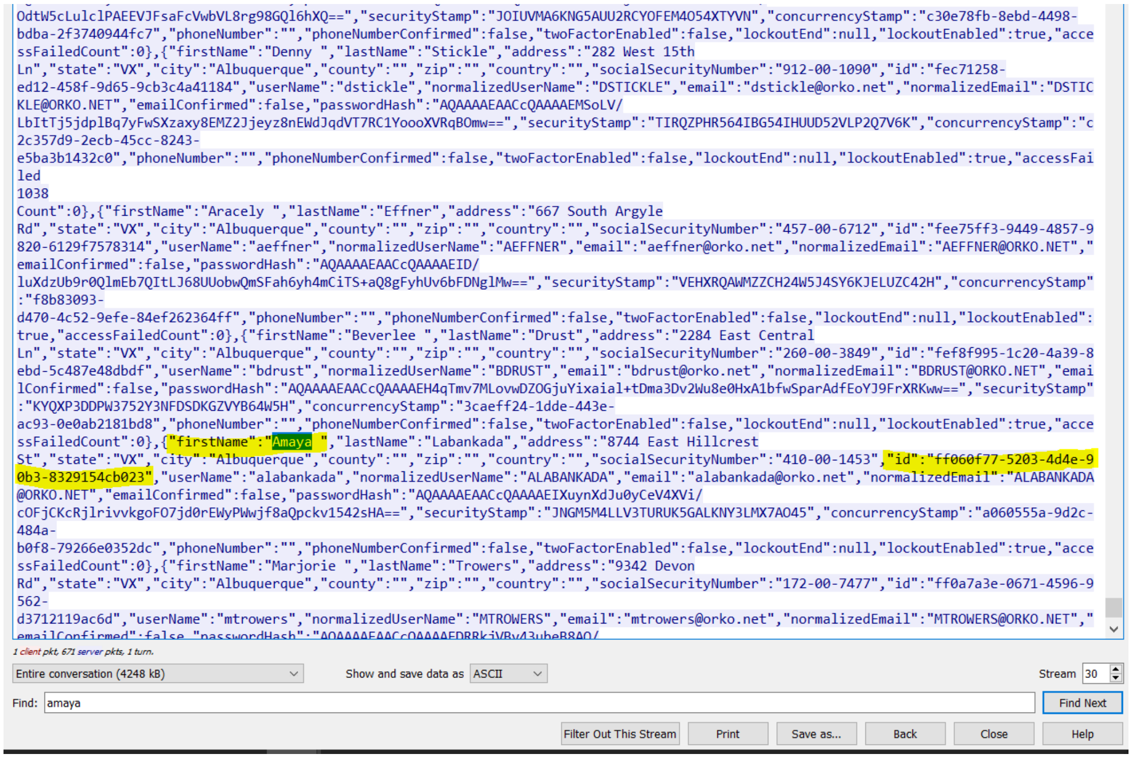
Answer: json

1.11 Odei 10

What is Amaya Labankada's unique id within the customer data?

1.11.1 Solution:

Continue in the TCP stream from Odei 9. It's known that the data is formatted in JSON from Odei 9. Scrolling through the TCP stream, you'll notice usernames, email addresses, and much more. Use the find option in the TCP stream window to search for Amaya. This search shows a json package with Amaya's data and a field called **id**. This field contains the ID **ff060f77-5203-4d4e-90b3-8329154cb023**. It can be deduced that this is the unique identifier for Amaya.



```
OdtW5cLulclPAEEVJFsaFcVwbVL8rg98GQl6hXQ==","securityStamp":"JOIUVMA6KNG5AUU2RCYOFEM4O54XTYVN","concurrencyStamp":"c30e78fb-8ebd-4498-
bdba-2f3740944fc7","phoneNumber":"","phoneNumberConfirmed":false,"twoFactorEnabled":false,"lockoutEnd":null,"lockoutEnabled":true,"acce
ssFailedCount":0},{"firstName":"Denny ","lastName":"Stickle","address":"282 West 15th
Ln","state":"VX","city":"Albuquerque","county":"","zip":"","country":"","socialSecurityNumber":"912-00-1090","id":"fec71258-
ed12-458f-9d65-9cb3c4a41184","userName":"dstickle","normalizedUserName":"DSTICKLE","email":"dstickle@orko.net","normalizedEmail":"DSTIC
KLE@ORKO.NET","emailConfirmed":false,"passwordHash":"AQAAAAEAACcQAAAAEMSoLV/
LbItTj5jdplBq7yFwSXzaxy8EMZ2Jjeyz8nEWdJqdVT7RC1YoooXVRqBOmw==","securityStamp":"TIRQZPHR564IBG54IHUUD52VLP2Q7V6K","concurrencyStamp":"c
2c357d9-2ecb-45cc-8243-
e5ba3b1432c0","phoneNumber":"","phoneNumberConfirmed":false,"twoFactorEnabled":false,"lockoutEnd":null,"lockoutEnabled":true,"accessFai
led
1038
Count":0},{"firstName":"Aracely ","lastName":"Effner","address":"667 South Argyle
Rd","state":"VX","city":"Albuquerque","county":"","zip":"","country":"","socialSecurityNumber":"457-00-6712","id":"fee75ff3-9449-4857-9
820-6129f7578314","userName":"aeffner","normalizedUserName":"AEFFNER","email":"aeffner@orko.net","normalizedEmail":"AEFFNER@ORKO.NET","
emailConfirmed":false,"passwordHash":"AQAAAAEAACcQAAAAEID/
luXdzUb9r0QlmEb7QItLJ68UUobwQmSFah6yh4mCiTS+aQ8gFyhUv6bFDNglMw==","securityStamp":"VEHXRQAWMZZCH24W5J4SY6KJELUZC42H","concurrencyStamp"
:"f8b83093-
d470-4c52-9efe-84ef262364ff","phoneNumber":"","phoneNumberConfirmed":false,"twoFactorEnabled":false,"lockoutEnd":null,"lockoutEnabled":
true,"accessFailedCount":0},{"firstName":"Beverlee ","lastName":"Drust","address":"2284 East Central
Ln","state":"VX","city":"Albuquerque","county":"","zip":"","country":"","socialSecurityNumber":"260-00-3849","id":"fef8f995-1c20-4a39-8
ebd-5c487e48dbdf","userName":"bdrust","normalizedUserName":"BDRUST","email":"bdrust@orko.net","normalizedEmail":"BDRUST@ORKO.NET","emai
lConfirmed":false,"passwordHash":"AQAAAAEAACcQAAAAEH4qTmv7MLovwDZOGjuYixaial+tDma3Dv2Wu8e0HxA1bfwSparAdfEoYJ9FrXRKww==","securityStamp"
:"KYQXP3DDPW3752Y3NFDSDKGZVYB64W5H","concurrencyStamp":"3caeff24-1dde-443e-
ac93-0e0ab2181bd8","phoneNumber":"","phoneNumberConfirmed":false,"twoFactorEnabled":false,"lockoutEnd":null,"lockoutEnabled":true,"acce
ssFailedCount":0},{"firstName":"Amaya ","lastName":"Labankada","address":"8744 East Hillcrest
St","state":"VX","city":"Albuquerque","county":"","zip":"","country":"","socialSecurityNumber":"410-00-1453","id":"ff060f77-5203-4d4e-9
0b3-8329154cb023","userName":"alabankada","normalizedUserName":"ALABANKADA","email":"alabankada@orko.net","normalizedEmail":"ALABANKADA
@ORKO.NET","emailConfirmed":false,"passwordHash":"AQAAAAEAACcQAAAAEIXuynXdJu0yCeV4XVi/
cOFjCKcRjlrivvkgoFO7jd0rEWyPWwjf8aQpckv1542sHA==","securityStamp":"JNGM5M4LLV3TURUK5GALKNY3LMX7AO45","concurrencyStamp":"a060555a-9d2c-
484a-
b0f8-79266e0352dc","phoneNumber":"","phoneNumberConfirmed":false,"twoFactorEnabled":false,"lockoutEnd":null,"lockoutEnabled":true,"acce
ssFailedCount":0},{"firstName":"Marjorie ","lastName":"Trowers","address":"9342 Devon
Rd","state":"VX","city":"Albuquerque","county":"","zip":"","country":"","socialSecurityNumber":"172-00-7477","id":"ff0a7a3e-0671-4596-9
562-
d3712119ac6d","userName":"mtrowers","normalizedUserName":"MTROWERS","email":"mtrowers@orko.net","normalizedEmail":"MTROWERS@ORKO.NET","
emailConfirmed":false,"passwordHash":"AQAAAAEAACcQAAAAEDRRkjVBy43ubeB8AO/
```

1 client pkt, 671 server pkts, 1 turn.

Entire conversation (4248 kB)

Show and save data as: ASCII

Stream: 30

Find: amaya

Answer: ff060f77-5203-4d4e-90b3-8329154cb023