

Elson Jian (918147110)
Jacob Dominick (917503904)

Submitted files:

proj3.py

proj3_gpt.py

[ChatGPT Link](#)

Code:

```
import requests

sid = '917503904'
headers = {
    'Student-Id': sid,
}

r =
requests.get('https://kartik-labeling-cvpr-0ed3099180c2.herokuapp.com/ecs1
52a_ass1', headers=headers)
print("Status Code:", r.status_code)
print(r.headers)
```

Output:

Status Code: 200

{'Connection': 'close', 'Server': 'BaseHTTP/0.6 Python/3.11.6', 'Date': 'Fri, 03 Nov 2023 03:51:47 GMT', 'Content-Type': 'text/plain', 'Ecs152a-Resp': '240836237', 'Via': '1.1 vegur'}

Wireshark (no proxy):

ip.addr == 52.5.82.174						
No.	Time	Source	Destination	Protocol	Length	Info
25	1.886514	10.0.0.223	52.5.82.174	TCP	66	55463 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
27	1.970122	52.5.82.174	10.0.0.223	TCP	66	443 → 55463 [SYN, ACK] Seq=0 Ack=1 Win=26883 Len=0 MSS=1460 SACK_PERM WS=4096
28	1.970193	10.0.0.223	52.5.82.174	TCP	54	55463 → 443 [ACK] Seq=1 Ack=1 Win=131328 Len=0
32	2.277192	10.0.0.223	52.5.82.174	TLSv1.2	571	Client Hello
34	2.365178	52.5.82.174	10.0.0.223	TCP	56	443 → 55463 [ACK] Seq=1 Ack=518 Win=28672 Len=0
35	2.365178	52.5.82.174	10.0.0.223	TLSv1.2	154	Server Hello
36	2.365178	52.5.82.174	10.0.0.223	TCP	1514	443 → 55463 [ACK] Seq=101 Ack=518 Win=28672 Len=1460 [TCP segment of a reassembled PDU]
37	2.365178	52.5.82.174	10.0.0.223	TCP	1514	443 → 55463 [ACK] Seq=1561 Ack=518 Win=28672 Len=1460 [TCP segment of a reassembled PDU]
38	2.365236	10.0.0.223	52.5.82.174	TCP	54	55463 → 443 [ACK] Seq=518 Ack=3021 Win=131328 Len=0
39	2.365312	52.5.82.174	10.0.0.223	TCP	1514	443 → 55463 [ACK] Seq=3021 Ack=518 Win=28672 Len=1460 [TCP segment of a reassembled PDU]
40	2.365328	10.0.0.223	52.5.82.174	TCP	54	55463 → 443 [ACK] Seq=518 Ack=4481 Win=131328 Len=0
41	2.367411	52.5.82.174	10.0.0.223	TLSv1.2	624	Certificate
42	2.367411	52.5.82.174	10.0.0.223	TLSv1.2	392	Server Key Exchange
43	2.367451	10.0.0.223	52.5.82.174	TCP	54	55463 → 443 [ACK] Seq=518 Ack=5389 Win=130304 Len=0
44	2.367575	52.5.82.174	10.0.0.223	TLSv1.2	63	Server Hello Done
45	2.369918	10.0.0.223	52.5.82.174	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
59	2.469246	52.5.82.174	10.0.0.223	TCP	56	443 → 55463 [ACK] Seq=5398 Ack=644 Win=28672 Len=0
62	2.469766	52.5.82.174	10.0.0.223	TLSv1.2	60	Change Cipher Spec
63	2.469766	52.5.82.174	10.0.0.223	TLSv1.2	99	Encrypted Handshake Message
64	2.469798	10.0.0.223	52.5.82.174	TCP	54	55463 → 443 [ACK] Seq=644 Ack=5449 Win=130304 Len=0
65	2.470143	10.0.0.223	52.5.82.174	TLSv1.2	296	Application Data
80	2.608246	52.5.82.174	10.0.0.223	TLSv1.2	770	Application Data
81	2.608246	52.5.82.174	10.0.0.223	TLSv1.2	122	Application Data
82	2.608246	52.5.82.174	10.0.0.223	TLSv1.2	85	Encrypted Alert
83	2.608278	10.0.0.223	52.5.82.174	TCP	54	55463 → 443 [ACK] Seq=886 Ack=6264 Win=131328 Len=0
84	2.609040	10.0.0.223	52.5.82.174	TCP	54	55463 → 443 [FIN, ACK] Seq=886 Ack=6264 Win=131328 Len=0
92	2.700485	52.5.82.174	10.0.0.223	TCP	56	443 → 55463 [FIN, ACK] Seq=6264 Ack=887 Win=32768 Len=0
93	2.700526	10.0.0.223	52.5.82.174	TCP	54	55463 → 443 [ACK] Seq=887 Ack=6265 Win=131328 Len=0

Can I tell what the secret key is?

- Yes. Directly from the Python code, the response headers are printed which include 'Ecs152a-Resp' that contains the secret key **240836237**

Mitmproxy enabled

```
Flow Details
2023-11-02 21:19:01 GET https://kartik-labeling-cvpr-0ed3099180c2.herokuapp.com/ecs152a_ass1
+ 200 OK text/plain 39b 92ms

Request Response Detail
Report-To: {"group":"heroku-nel","max_age":3600,"endpoints":[{"url":"https://nel.heroku.com/reports?ts=1698985141&sid=1b10b0ff-8a76-4548-befa-353fc6c6c045&s=i2cwo4pkpC05dfh0boYrMTf5a%2ByC2QH1M4wmwKklBg%3D"}]}
Reporting-Endpoints: heroku-nel=https://nel.heroku.com/reports?ts=1698985141&sid=1b10b0ff-8a76-4548-befa-353fc6c6c045&s=i2cwo4pkpC05dfh0boYrMTf5a%2ByC2QH1M4wmwKklBg%3D
Nel: {"report_to":"heroku-nel","max_age":3600,"success_fraction":0.005,"failure_fraction":0.05,"response_headers":["Via"]}
Connection: close
Server: BaseHTTP/0.6 Python/3.11.6
Date: Fri, 03 Nov 2023 04:19:01 GMT
Content-Type: text/plain
Ecs152a-Resp: 240836237
Via: 1.1 vegur

Raw
You should look at the response headers
```

Wireshark (with mitmproxy decrypting traffic)

tcp.port == 80

No.	Time	Source	Destination	Protocol	Length	Info
44	1.424585	10.0.0.223	162.159.136.234	TLSv1.2	108	Application Data
45	1.443397	162.159.136.234	10.0.0.223	TCP	56	443 → 59489 [ACK] Seq=207 Ack=55 Win=7 Len=0
46	1.520590	162.159.136.234	10.0.0.223	TLSv1.2	88	Application Data
47	1.566911	10.0.0.223	162.159.136.234	TCP	54	59489 → 443 [ACK] Seq=55 Ack=241 Win=511 Len=0
57	2.455214	2601:200:c181:40a0::...	2600:1901:1:c36::...	TLSv1.2	113	Application Data
58	2.455730	10.0.0.223	54.159.116.102	TCP	66	59527 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
59	2.479713	2600:1901:1:c36::...	2601:200:c181:40a0::...	TCP	74	443 → 59481 [ACK] Seq=1 Ack=40 Win=280 Len=0
60	2.479713	2600:1901:1:c36::...	2601:200:c181:40a0::...	TLSv1.2	113	Application Data
63	2.534736	2601:200:c181:40a0::...	2600:1901:1:c36::...	TCP	74	59481 → 443 [ACK] Seq=40 Ack=40 Win=514 Len=0
64	2.545174	54.159.116.102	10.0.0.223	TCP	66	443 → 59527 [SYN, ACK] Seq=0 Ack=1 Win=26883 Len=0 MSS=1460 SACK_PERM WS=4096
65	2.545229	10.0.0.223	54.159.116.102	TCP	54	59527 → 443 [ACK] Seq=1 Ack=1 Win=131328 Len=0
66	2.573759	10.0.0.223	54.159.116.102	TLSv1.2	414	Client Hello
67	2.656310	54.159.116.102	10.0.0.223	TCP	56	443 → 59527 [ACK] Seq=1 Ack=361 Win=28672 Len=0
68	2.656803	54.159.116.102	10.0.0.223	TLSv1.2	158	Server Hello
69	2.657865	54.159.116.102	10.0.0.223	TCP	1514	443 → 59527 [ACK] Seq=105 Ack=361 Win=28672 Len=1460 [TCP segment of a reassembled PDU]
70	2.657865	10.0.0.223	54.159.116.102	TCP	54	59527 → 443 [ACK] Seq=361 Ack=1565 Win=131328 Len=0
71	2.659523	54.159.116.102	10.0.0.223	TCP	1514	443 → 59527 [ACK] Seq=1565 Ack=361 Win=28672 Len=1460 [TCP segment of a reassembled PDU]
72	2.659523	54.159.116.102	10.0.0.223	TCP	1514	443 → 59527 [ACK] Seq=3025 Ack=361 Win=28672 Len=1460 [TCP segment of a reassembled PDU]
73	2.659544	10.0.0.223	54.159.116.102	TCP	54	59527 → 443 [ACK] Seq=361 Ack=4485 Win=131328 Len=0
74	2.661486	54.159.116.102	10.0.0.223	TLSv1.2	624	Certificate
75	2.661486	54.159.116.102	10.0.0.223	TLSv1.2	392	Server Key Exchange
76	2.661486	54.159.116.102	10.0.0.223	TLSv1.2	63	Server Hello Done
77	2.661533	10.0.0.223	54.159.116.102	TCP	54	59527 → 443 [ACK] Seq=361 Ack=5402 Win=130304 Len=0
78	2.663008	10.0.0.223	54.159.116.102	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Finished
79	2.750583	54.159.116.102	10.0.0.223	TCP	56	443 → 59527 [ACK] Seq=5402 Ack=487 Win=28672 Len=0
80	2.750583	54.159.116.102	10.0.0.223	TLSv1.2	174	New Session Ticket
81	2.750679	54.159.116.102	10.0.0.223	TLSv1.2	105	Change Cipher Spec, Finished
82	2.750696	10.0.0.223	54.159.116.102	TCP	54	59527 → 443 [ACK] Seq=487 Ack=5573 Win=130304 Len=0
83	2.760159	10.0.0.223	54.159.116.102	HTTP	296	GET /ecs152a_ass1 HTTP/1.1
84	2.846709	54.159.116.102	10.0.0.223	TLSv1.2	801	
85	2.846709	54.159.116.102	10.0.0.223	TLSv1.2	85	Alert (Level: Warning, Description: Close Notify)
86	2.846765	10.0.0.223	54.159.116.102	TCP	54	59527 → 443 [ACK] Seq=729 Ack=6351 Win=131328 Len=0
87	2.849448	10.0.0.223	54.159.116.102	TCP	54	59527 → 443 [FIN, ACK] Seq=729 Ack=6351 Win=131328 Len=0
88	2.936371	54.159.116.102	10.0.0.223	TCP	56	443 → 59527 [FIN, ACK] Seq=6351 Ack=730 Win=32768 Len=0
89	2.936415	10.0.0.223	54.159.116.102	TCP	54	59527 → 443 [ACK] Seq=730 Ack=6352 Win=131328 Len=0
90	3.047517	2607:f8b0:4005:80c::...	2601:200:c181:40a0::...	TLSv1.2	158	Application Data

Protocol: TCP (6)
Header Checksum: 0x0000 [validation disabled]
[Header checksum status: Unverified]
Source Address: 10.0.0.223
Destination Address: 54.159.116.102

Transmission Control Protocol, Src Port: 59527, Dst Port: 443, Seq: 487, Ack: 5573, Len: 0
Source Port: 59527
Destination Port: 443
[Stream index: 4]
[Conversation completeness: Complete, WITH_DATA (31)]
[TCP Segment Len: 242]
Sequence Number: 487 (relative sequence number)
Sequence Number (raw): 2105915680
[Next Sequence Number: 729 (relative sequence number)]
Acknowledgment Number: 5573 (relative ack number)
Acknowledgment number (raw): 2387258995
0101 = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
Window: 509
[Calculated window size: 130304]
[Window size scaling factor: 256]
Checksum: 0xb6f0 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
[Timestamps]
[SEQ/ACK analysis]
TCP payload (242 bytes)
Transport Layer Security
Hypertext Transfer Protocol
GET /ecs152a_ass1 HTTP/1.1\r\n
Host: kartik-labeling-cvpr-0ed3099180c2.herokuapp.com\r\n
User-Agent: python-requests/2.31.0\r\n
Accept-Encoding: gzip, deflate\r\n
Accept: */*\r\n
Connection: keep-alive\r\n
Student-Id: 917503904\r\n

0000 47 45 54 20 2f 65 63 73 31 35 32 61 5f 61 73 73 GET /ecs152a_ass1
0010 31 20 48 54 54 50 2f 31 2e 31 0d 0a 48 6f 73 74 1 HTTP/1.1 ·Host
0020 3a 20 6b 61 72 74 69 6b 20 6c 61 62 65 6c 69 6e : kartik-labeling
0030 67 2d 63 76 70 72 2d 30 65 64 33 30 39 31 38 g-cvpr-0 ed309918
0040 30 63 32 2e 68 65 72 6f 6b 75 61 70 70 2e 63 6f 0c2.herokuapp.co
0050 6d 0d 0a 55 73 65 72 2d 41 67 65 6e 74 3a 20 70 m-User-Agent: p
0060 79 74 68 6f 6e 2d 72 65 71 75 65 73 74 73 2f 32 ython-requests/2
0070 2e 33 31 2e 30 0d 0a 41 63 63 65 70 74 2d 45 6e .31.0 ·A ccept-En
0080 63 6f 64 69 6e 67 3a 20 67 7a 69 70 2c 20 64 65 coding: gzip, de
0090 66 6c 61 74 65 0d 0a 41 63 63 65 70 74 3a 20 2a flate ·A ccept: *
00a0 2f 2a 0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e 3a 20 /* ·Conn ection:
00b0 6b 65 65 70 2d 61 6c 69 76 65 0d 0a 63 74 75 64 keep-alive ·Stud
00c0 65 6e 74 2d 49 64 3a 20 39 31 37 35 30 33 39 30 ent-Id: 917503904
00d0 3a 0d 0a 0d 0a 41 ··

Can I tell what the secret key is?

- From mitmproxy? Yes. You can see the response headers which include the secret key by clicking on the GET request from the terminal. Shown in the above screenshot of mitmproxy.
 - Again it is `'Ecs152a-Resp': '240836237'`
- From Wireshark? No. You can see the decrypted GET request that includes the student ID header, but the response is still encrypted.