Assignment 4

Secure Programming, 2016

Elias Athanasopoulos - Sanjay Rawat
i.a.athanasopoulos@vu.nl - s.rawat@vu.nl

Question 1:

Initially in the class, we heard about a recent bug, called *HeartBleed*. Read the following documents and answer questions below:

Link A: http://www.dwheeler.com/essays/heartbleed.html

Link B: https://blog.hboeck.de/archives/868-How-Heartbleed-couldve-been-found.html

Link C: http://www.theregister.co.uk/2014/04/09/heartbleed_explained/

Link D:
https://git.openssl.org/gitweb/?p=openssl.git;a=commitdiff;h=731f431497f463f3a2a97236fe0187b11c44aead;ds=sidebyside

A. Which of the following CIA properties is precisely violated by this bug:

(i) Confidentiality (ii) Integrity (iii) availability (iv) none

B. Which of the following CWE ID corresponds to heartbleed bug?

(i) CWE-125 (ii) CWE-127 (iii) CWE-133 (iv) CWE-787

C. Why fuzzing could not detect Heartbleed bug?

(i) it is difficult to generate bug triggering network packets (ii) OpenSSL has encrypted messages/packets

(iii) there is no observable exceptional behavior (e.g., crash) by the server

(vi) Fuzzing did not know the input format

D. Why does address-sanitizer enabled binary help fuzzing in detecting Heartbleed like bug?

(i) Forming input packet is easier with this feature (ii) This makes OpenSSL sending hello packets in clear text, thus making fuzzing easier

(iii) it enables alert on out-of-bound array access (vi) address-sanitizer helps fuzzing knowing the input format of help message, thus generating better inputs

E: The patch to Heartbleed bug can be seen at Link D above (git repo for openssl). Read the patch for `/ssl/d1_both.c` and answer the following. Which of the following options is the *main* fix for the bug as far as detection of the wrong `payload_length` is concerned?

```
(i) if (1 + 2 + 16 > s->s3->rrec.length)     (ii) if (1 + 2 + payload + 16 > s->s3->rrec.length)

(iii)  if (write_length > SSL3_RT_MAX_PLAIN_LENGTH)  (vi) if (s->msg_callback)
```