Calculators may be used in this examination provided they are not capable of being used to store alphabetical information other than hexadecimal numbers

UNIVERSITY OF BIRMINGHAM

School of Computer Science

Secure Software and Hardware Systems

Main Summer Examinations 2024

Time allowed: 2 hours

[Answer all questions]

Turn Over

Note

Answer ALL questions. Each question will be marked out of 20. The paper will be marked out of 60, which will be rescaled to a mark out of 100.

Question 1

In your job at a software company you responsible for reviewing the source-code produced by your colleagues. Your latest task requires you to review the following C code:

```
 1 #include <stdlib.h>
 2 #include <string.h>
 3 #include <stdio.h>
 4 #include <stdint.h>
 5
 6 void main(int argc, char *argv[]) {
 7   uint32_t in_len=4;
 8   void *in = malloc(in_len*sizeof(in_len));
 9   uint64_t sec_len=1;
10   void *sec = malloc(sec_len*sizeof(sec_len));
11
12   strcpy(sec, "12345");
13   if (argc > 1)
14     strcpy(in, argv[1]);
15
16   printf("debug: size of uint64_t=%d bytes and of *sec data=%d bytes\n",
17          sizeof(uint64_t),sizeof(sec));
18   printf("sec < addr: %p, value: %s >\n", sec, sec);
19   printf("in < addr: %p, value: %s >\n", in, in);
20 }
```

(a) Briefly describe the two common types of *buffer overflows* which can affect software. **[6 marks]**

(b) When run with the input '`test`' the program produces the following output:

```
% ./a.out test
 debug: size of uint64_t=8 bytes and of *sec data=8 bytes
 sec < addr: 0x7fffd30a92c0, value: 12345 >
 in < addr: 0x7fffd30a92a0, value: test >
```

Explain how '`sec`' can store the value '12345' considering that '`sec_len`' has a length of 1. What is the longest string that can fit into `sec` "safely"? **[4 marks]**

(c) Now consider the execution for the input '`abcdefghijklmnopqrstuvwxyzABCDEF`':

```
% ./a.out abcdefghijklmnopqrstuvwxyzABCDEF
 debug: size of uint64_t=8 bytes and of *sec data=8 bytes
 sec < addr: 0x7fffdd58f2c0, value:  >
 in < addr: 0x7fffdd58f2a0, value: abcdefghijklmnopqrstuvwxyzABCDEF >
```

This input has removed the value stored in '`sec`'.

(i) Provide a brief technical explanation on what causes this behaviour. **[3 marks]**

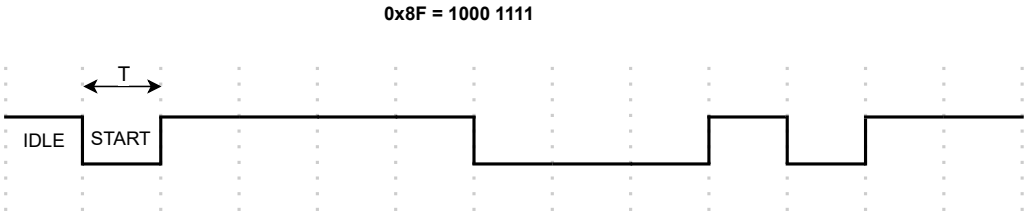
(ii) What would be the content value of '`sec`' for the input '`abcdefghijklmnopqrstuvwxyzABCDEFGHIJ`' **[3 marks]**

(d) Propose two fixes that would improve the source-code above and prevent the undefined behaviour observed. **[4 marks]**

Question 2

You got moved from the software team of the company to the firmware team, and are now responsible for evaluating the security of embedded devices manufactured by the company. One day, you have to do an assessment of the newest "IoT wheather station". The system only consists of a single sensor to get the temperature, a display to show it to the user, and a UART interface for debugging purposes.

(a) You attach to the UART port and observe the following communication of a single data byte (`0x8f`). Can you identify the key parameters used for this transmission (number of data bits, number and configuration of parity bits, number of stop bits)? Consider the full signal as shown as part of the transmission.



[3 marks]

(b) Given the provided information, what programming model would you use if you were to implement the system? Briefly justify your answer in 1–2 sentences.

[3 marks]

(c) You are tasked to run a black-box fuzzing campaign against the physical device.

(i) Name and briefly explain three challenges you are likely to encounter when fuzz testing the physical device.

[6 marks]

(ii) According to the device's datasheet, it is expected to respond to every message received via UART either with an `ACK` or a `NACK`. How would you detect whether your fuzzing input triggered a vulnerability? Name and briefly explain 2 different possibilities.

[4 marks]

(iii) Your fuzzing campaign found a stack-based buffer overflow leading to a crash of the device due to a corrupted program counter. Your co-worker claims that enabling the memory protection unit on the device would have prevented this vulnerability. Can you confirm or refute this claim and pitch one additional defense to prevent the device from crashing? Justify your answer in at most 4 sentences.

[4 marks]

Question 3

The following (pseudocode) implementation of a cipher called XTEA (with 64-bit block size and 128-bit key) runs on a processor with caches:

```
0: uint32_t key[4] = ...; // 128-bit key
1: uint32_t v0 = ...; // first half of plaintext
2: uint32_t v1 = ...; // second half of plaintext
3: uint32_t sum = 0;
4:
5: for (uint8_t i = 0; i < 32; i++) {
6:     v0 += (((v1 << 4) ^ (v1 >> 5)) + v1) ^ (sum + key[sum & 3]);
7:     sum += 0x9E3779B9;
8:     v1 += (((v0 << 4) ^ (v0 >> 5)) + v0) ^ (sum + key[(sum >> 11) & 3]);
9: }
10: ...
```

(a) Assume the above implementation consumes 20,000 cycles to encrypt a 64-bit block. An AES-128 implementation takes 30,000 cycles. If higher performance is the only goal, which implementation is better? Explain your answer, using a suitable metric to compare the two implementations. **[5 marks]**

(b) Briefly name and explain one physical method to inject faults into a microcontroller, and a corresponding hardware of software countermeasure. **[4 marks]**

(c) Assume the adversary can inject a fault into the above implementation such that the for loop is aborted after one iteration (i.e., it only runs once for i = 0, resulting in outputs `v0'`, `v1'`). Given the inputs `v0`, `v1` and the outputs `v0'`, `v1'`, explain how the adversary can then compute 32-bit of the key, i.e., `key[0]`. **[7 marks]**

(d) Is this implementation vulnerable to a cache attack where the adversary can observe which elements of an array are accessed? Briefly explain your answer. **[4 marks]**

End of Paper

Do not complete the attendance slip, fill in the front of the answer book or turn over the question paper until you are told to do so

Important Reminders

- Coats/outwear should be placed in the designated area.
- Unauthorised materials (e.g. notes or Tippex) must be placed in the designated area.
- Check that you do not have any unauthorised materials with you (e.g. in your pockets, pencil case).
- Mobile phones and smart watches **must** be switched off and placed in the designated area or under your desk. They must not be left on your person or in your pockets.
- You are not permitted to use a mobile phone as a clock. If you have difficulty seeing a clock, please alert an Invigilator.
- You are not permitted to have writing on your hand, arm or other body part.
- Check that you do not have writing on your hand, arm or other body part – if you do, you must inform an Invigilator immediately
- Alert an Invigilator immediately if you find any unauthorised item upon you during the examination.

Any students found with non-permitted items upon their person during the examination, or who fail to comply with Examination rules may be subject to Student Conduct procedures.