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

LH Security of Real-World Systems

Main Summer Examinations 2023

Time allowed: 2 hours

[Answer all questions]

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

Consider the follow code:

```
1  #include<stdio.h>
2
3  void printSubString (int n, char string [ ] ) {
4      int i;
5      for (i=0;i<n;i++) {
6        printf("%hhX",string[i]); // Prints as hex
7      }
8      printf("\n");
9  }
10
11 int main ( ) {
12       char userinput[32];
13       int length;
14       while(1) {
15            printf("Enter your string:\n");
16            gets(userinput);
17            printf("Enter length:");
18            scanf("%d", &length); // Read in int from user
19            printSubString(length,userinput);
20       }
21 }
```

(a) Assume the code is run on x64 (64-bit). Draw a sketch of the stack when the printf on line 6 is executed. You should go down as far as the return address to the function that called main. Assume that the program has stack canaries. State any other assumptions you make. **[5 marks]**

(b) Now assume that the program is run with a non-executable stack, 32-bit, but without ASLR and stack canaries. How could you get this program to open a shell? If so explain how and what your inputs would be. **[5 marks]**

(c) Assume that program is run with a non-executable stack, 64-bit, ASLR and stack canaries. Could you get this program to open a shell? If so explain how and what your inputs would be. **[5 marks]**

(d) Explain how the program could be made safe from attacks without the protections mentioned above. **[5 marks]**

Question 2

A simplified version of the WPA protocol runs as follows:

AP $\rightarrow$ Client: N_{ap}
Client $\rightarrow$ AP: $N_c, MAC_{key}(N_{ap})$
AP $\rightarrow$ Client: $\{Groupkeys\}_{key} MAC_{key}(Groupkeys)$

where key equals a hash of the wi-fi password, N_{ap} and N_c.

(a) What would be the effect on the security of the protocol of the access point replacing its nonce (N_{ap}) with a static value? If an attack is possible say what it is, if no attack is possible say why. **[5 marks]**

(b) What would be the effect on the security of the protocol of the access point replacing its nonce (N_{ap}) with a 64-bit counter that increments by one each time the protocol runs? If an attack is possible say what it is, if no attack is possible say why. **[5 marks]**

(c) Explain why this protocol can be attacked when the password is low entropy (guessable). **[2 marks]**

(d) The Diffie Hellman protocol is normally run with a known generator g. However it can also be run with a generator produced from a password H(password). Design an improvement to the WPA protocol that uses this to provide protection for offline guessing attacks on weak passwords. Write down your protocol in Alice and Bob notation and explain why it is secure. **[8 marks]**

Question 3

An implementation of secure boot on a microcontroller receives a firmware image authenticated with a symmetric Message Authentication Code (MAC). For that purpose, a symmetric secret key `key` is shared between the microcontroller and the manufacturer that issues firmware updates. The MAC tag is stored in the first 16 byte of the firmware image. Your goal is to bypass the authentication check, which is implemented in the function `check_update()`. The C-pseudocode of that function is as follows:

```
1  // upd points to the received firmware image
2  // len is the length of upd
3  // MAX_SIZE is a constant
4  bool check_update(uint8_t* upd, size_t len) {
5    if(len < 16)
6       return false; // Update too small
7
8    if(len > MAX_SIZE)
9       return false; // Update too large
10
11   uint8_t mac_rcv[16];  // received MAC (first 16 byte)
12   uint8_t image[MAX_SIZE - 16]; // holds rest of image
13   uint8_t mac_comp[16]; // internally computed MAC
14
15   memcpy(mac_rcv, upd, 16); // copy MAC
16   memcpy(image, &upd[16], len - 16); // copy remainder
17
18   // compute MAC over image with stored key
19   compute_mac(mac_comp, image, key);
20
21   if(memcmp(mac_comp, mac_rcv, 16) != 0)
22      return false; // invalid MAC
23   else
24      return true;  // all good
25 }
```

(a) An adversary can inject a voltage glitch to make a comparison (e.g., $a < b$ or $a != b$) return the opposite result (e.g., $1 < 2$ would evaluate to false). Assume that the contents of `upd` and `len` are controlled by the adversary, and that the device does not use exploit mitigations like stack canaries etc. Explain how this can be used to achieve *arbitrary code execution* by exploiting a flaw within the `check_update()` function; and state on which line does the glitch needs to be injected. **[7 marks]**

(b) For the same scenario as in (a), explain how the adversary can alternatively make the function return `true` without achieving arbitrary code execution. **[5 marks]**

(c) Assume that if the function returns `false`, the device instantly sends an error message. Assume that all library functions used in the code are standard implementations without mitigations against timing attacks. Explain how the adversary can perform a timing attack to craft a malicious firmware image with a valid MAC, and state how many attempts are required in the worst case. **[8 marks]**

This page intentionally left blank.

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.