UNIVERSITY OF LONDON

BSc EXAMINATION 2018

For Internal Students of
Royal Holloway

DO NOT TURN OVER UNTIL TOLD TO BEGIN

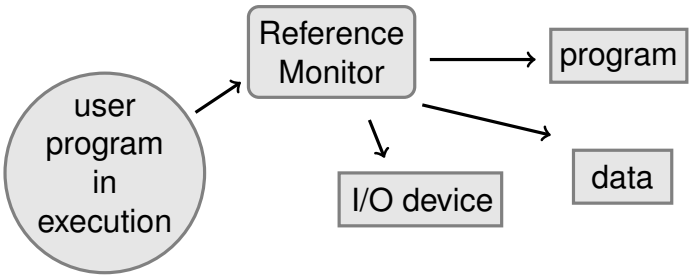
IY2840: Computer and Network Security
IY2840R: Computer and Network Security – PAPER FOR RESIT CANDIDATES

Time Allowed: **TWO hours**

Answer ALL questions
Calculators are NOT permitted

1. ACCESS CONTROL

The following figure is a reproduction from the Anderson Report from 1972.



(a) In the figure, which parts symbolize a **subject** and which symbolize an **object**?

[4 marks]

(b) Describe what is meant by the concept of a **Reference Monitor**.

[6 marks]

(c) The reference validation mechanism of an operating system is called the security kernel. Briefly, what is the relation between the notions of **Reference Monitor** and **Security Kernel**?

[4 marks]

(d) Which technical concepts of classic UNIX access control take the role of the **subjects** in the above sense? Which concepts take the role of the **objects**? What are the **security principals**?

[6 marks]

2. HARDWARE SECURITY

A hardware-implemented security concept that facilitates access control in operating systems is that of **protection rings** and was originally introduced with the MULTICS architecture.

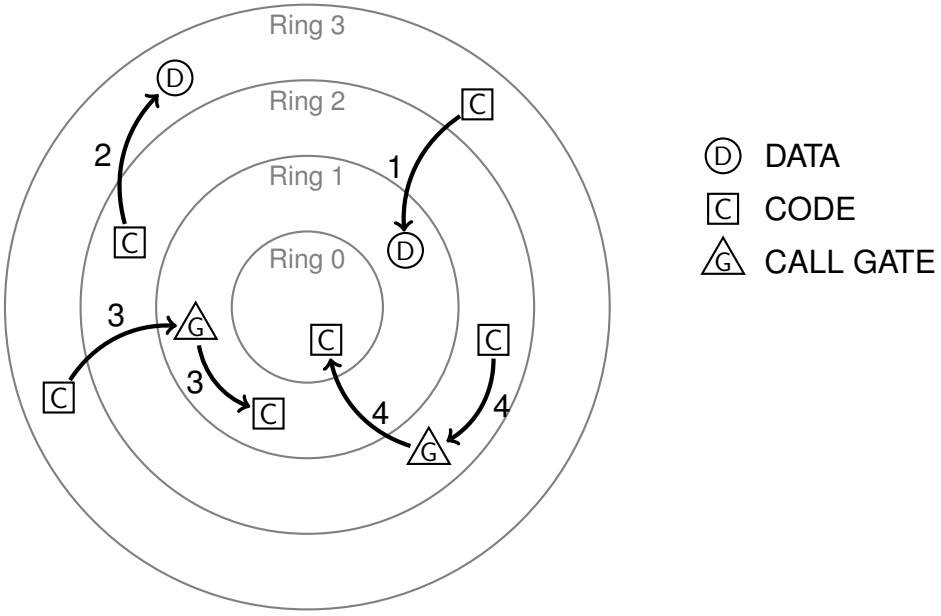
(a) Describe the protection ring architecture of MULTICS. Explain in particular, **to whom** and **how** it provides security. Mention **one example** where the ring architecture could be helpful.

[7 marks]

(b) Also more modern hardware designs follow the protection ring idea, but make their own adaptations and modifications to it. The x86 architecture supports four rings, and the following figure illustrates four access attempts that might be tried on that architecture. (The access attempts are numbered 1–4, and the numbers are indicated on the arrows.) Note specific symbols are used for representing the three concepts of **data**, **code**, and **call gates**. (Observe these symbols are slightly different than in the lecture slides.)

Which of the attempts will be **granted** by the CPU? Which of the attempts will be **denied**? Briefly **explain** your answer.

[8 marks]



(c) In the figure, which of the attempts is a case of **controlled invocation**? (No explanation needed.)

[2 marks]

(d) Early proposals for protection ring architectures were designed for a total of up to 64 rings, but current CPUs (e.g. the x86 family) support only a much smaller number. Give one brief argument **for** and one **against** a large number of rings.

[4 marks]

(e) Name one way to implement controlled invocation on x86 CPUs.

[2 marks]

(f) How many rings do the operating systems Windows and Linux use on x86 CPUs? Hint: it's the same number.

[2 marks]

3. UNIX SECURITY

Assume Alice and Bob have accounts on the same UNIX system. Their usernames are `alice` and `bob`, respectively. Alice and Bob are both member of groups `staff` and `cdrom`. None of them is member of group `admin`.

Assume a directory on that system that contains six files (`file1`–`file6`) with file mode and ownership displayed as follows:

```
$ stat --format="%a %A %U/%G %n" file[1-6]
0400 -r--------  alice/staff  file1
0440 -r--r-----  root/cdrom   file2
0777 -rwxrwxrwx  root/admin   file3
0044 ----r--r--  alice/staff  file4
0404 -r-----r--  root/cdrom   file5
0004 -------r--  root/admin   file6
```

Here, the left-most column (`%a`) lists the access permissions in octal, the second column (`%A`) lists the access permissions in ASCII, the third column (`%U/%G`) shows the user identifier and the group identifier of the file owner in combined form (in UID/GID format), and the right-most column shows the name of the file.

(a) List all files that ...

i. user `alice` can open for reading.

[2 marks]

ii. user `alice` can open for writing.

[2 marks]

iii. user `root` (with UID = 0) can open for reading.

[2 marks]

iv. user `root` (with UID = 0) can open for writing.

[2 marks]

(b) Consider now specifically `file3`. The file is owned by `root` and has all three `x`-bits set. Does this indicate the file is a set-user-ID (to `root`) file? Briefly explain.

[2 marks]

(c) Consider once more `file3`. The file represents a severe security risk and enables an attack. Explain why this is the case! Mention in particular: **Who** can run the attack? **What** precisely would the attacker do? **Who** would be the victim(s)?

[7 marks]

(d) This question is about advising the system administrator how to resolve the security issue caused by `file3` (see Question (c)). Assume the superuser (`root`, UID = 0) is ready to invoke the `chmod` command on `file3` to reconfigure the access permissions, but is not sure how to do this.

Propose one argument for the `chmod` command such that replacing `...` in

```
chmod ... file3
```

by that argument would resolve the security problem. You may specify the argument in octal or ASCII. There are many valid answers to this question.

[2 marks]

(e) Consider now `file1`. This question is on whether Alice can **delete** this file. From just the above information you cannot decide this. The access rights (`w` and `x`) of which entry of the file system would you need to know to answer the question?

[2 marks]

(f) Briefly, what does the `umask` influence?

[2 marks]

(g) Name one advanced security feature (related to access control) of modern operating systems like Linux.

[2 marks]

4. SOFTWARE SECURITY and SHELLCODE

The following assembly code (for the x86 architecture, 32 bit mode) is a "shellcode" as it will start a shell if executed in the right context, starting at line 1.

```
 1    jmp ahead
 2
 3    back:
 4       popl %ebx
 5       movl $0x0, %eax
 6       movl %ebx, 0x8(%ebx)
 7       leal 0x8(%ebx), %ecx
 8       movb %al, 0x7(%ebx)
 9       movl %eax, %edx
10       movl %eax, 0xc(%ebx)
11       movb $0xb, %al
12       int $0x80
13
14    ahead:
15       call back
16       .string "/bin/sh"
```

(a) Briefly, what is one typical application of a shellcode in computer security?

[2 marks]

(b) If the code is executed starting at line 1, what does the value of `%ebx` indicate when the code reaches line 5?

[2 marks]

(c) What is the effect of line 8?

[2 marks]

(d) If line 8 was replaced by `movb %ah, 0x7(%ebx)`, would the functionality of the shellcode change?

[2 marks]

(e) What is the meaning of value `0xb` in line 11?

[2 marks]

(f) The machine code (opcode) generated for line 5 has a property that is of disadvantage in certain attack scenarios. Which property is this? Briefly, how could this disadvantage be resolved?

[4 marks]

(g) When executed, the above shellcode will run the `/bin/sh` program file. In this question we assume the `sh` program was located at `/usr/bin/sh` instead of `/bin/sh`.

Some lines of the shellcode need to be adapted to make it run `/usr/bin/sh`, and only one of them is line 16. Which lines of the shellcode would need to be modified? (Just indicate the line numbers, an explanation is not required.)

[8 marks]

5. **SQL INJECTION**

SQL injection is an example of a Web Application exploit.

(a) Give a brief description (at most three sentences) of this attack and explain why it can succeed.

[4 marks]

(b) An online shopping site takes an email address as input to `$EMAIL` and constructs an SQL query as follows:

```
$query = "SELECT * FROM members WHERE email='$EMAIL'";
```

What would a malicious user enter as their `email` address in order to get the database to delete all entries from table `foo` (assuming the table exists)? (No need to explain the answer.)

[4 marks]

END