

Name:

ID:

## Malware – Master on Cybersecurity

Final Exam, January 12th 2021

This exam is individual, you cannot receive any external help to perform it. Use of any external source will be punished accordingly

Don't forget to specify your full name and Identity Card number on top of this page. You **don't** need to do that for all the pages on the exam

The exam punctuation goes from 0 to 10, where 0 is no correct answer and 10 is the perfect exam

Each question has its value indicated with all the subsections' values as well

The exam may be resolved using one of the following languages:

- Catalan
- Spanish
- English

**It is mandatory to explain and develop all your answers to get the full punctuation**

**Duration: 1 hour and 55 minutes (No extension will be granted)**

### Question 1 – Infection Propagation (2.5 points)

Answer the following question regarding infection propagation lesson.

1. Indicate the main differences between Local and Remote exploits. Clearly indicating which is the purpose of each one as well. **(0.5 Points)**

2. Describe what are buffer overflows and how they may be exploited in the real world. (0.75 Points)

3. Describe what are heap overflows and how they may be exploited in the real world. (0.75 Points)

4. List the usual limitations when creating shellcode, discussing the reasons why it needs to be this way. (0.5 Points)

## Question 2 – Obfuscation (2.5 points)

Answer the following question regarding obfuscation techniques lesson.

1. Given the following code:

```
0000000000401000 <_start>:
401000: 6a 01          push 0x1
401002: 58            pop rax
401003: 50            push rax
401004: 5f            pop rdi
401005: be 76 69 6c 0a mov esi,0x0a6c6976
40100a: 48 c1 c6 08    rol rsi,0x8
40100e: 48 83 f6 45    xor rsi,0x45
401012: 56            push rsi
401013: 54            push rsp
401014: 5e            pop rsi
401015: 6a 05          push 0x5
401017: 5a            pop rdx
401018: 0f 05          syscall
40101a: 6a 3c          push 0x3c
40101c: 58            pop rax
40101d: 48 31 ff      xor rdi,rdi
401020: 0f 05          syscall
```

Obfuscate it using the placeholder technique studied in class. Remember the following opcodes:

jmp → eb

mov rax, [LITERAL VALUE] → 48 b8

(1 Point)

2. Regarding metamorphic viruses. Describe the technique known as RegSwap

(0.5 Points)

**3.** What is a polymorphic virus?

**(0.5 Points)**

**4.** Describe what are the anti-emulator techniques and outline how they work.

**(0.5 Points)**

### Question 3 – AntiVirus (1 point)

Answer the following question regarding the antivirus lesson.

1. Indicate which part of an antivirus is the most vulnerable to attacks and discuss why is that. **(0.5 Points)**

2. Discuss the basis of behavioural virus detection seen in class. **(0.5 Points)**

#### Question 4 – Malware Categorization (2 points)

Answer the following questions related with the malware categorization lesson.

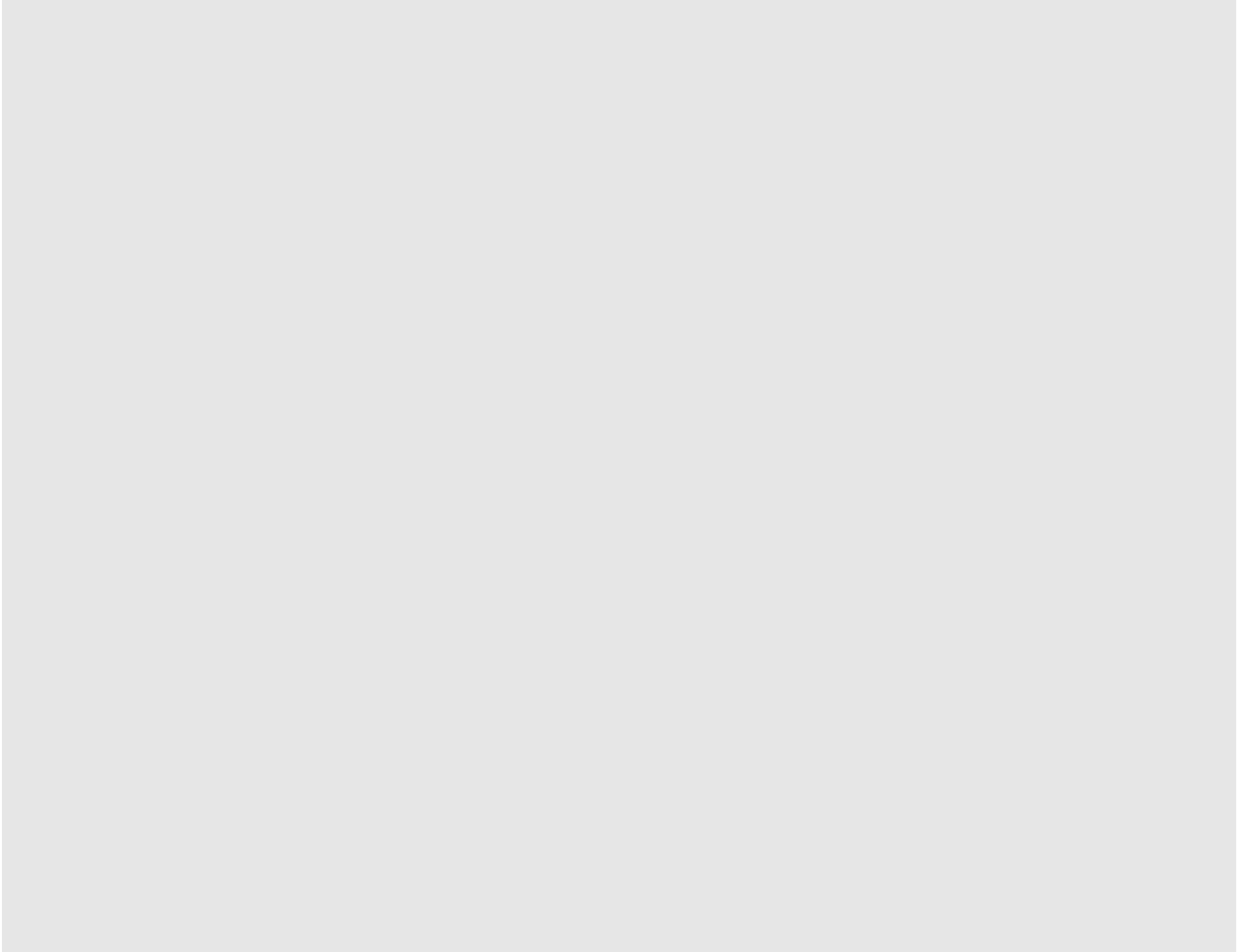
1. Detail the main differences between Worm and Virus.

(0.5 Points)

2. Explain the life-cycle of a Virus.

(0.75 Points)

3. Discuss about rootkits, particularly about the stealth techniques and the different types. (0.75 Points)



## Question 5 – General Theory (2 points)

Answer the following question marking the appropriate cell. Each question has only one valid response.

Each correct answer gives 0.5 points. **WRONG ANSWERS SUBTRACT 0.25 points, you can decide to leave blank answers. The minimum punctuation for the test is 0 (it doesn't affect the punctuation of other questions).**

1. Regarding canary values:
  - ☐ a) Protect from buffer overflow attacks by placing compiler generated values on entry to a function and validating them after returning to the caller
  - ☐ b) Protect from buffer overflow attacks by placing compiler generated values on entry to a function and validating them before returning to the caller
  - ☐ c) Protect from buffer overflow attacks by protecting the stack against execution
  
2. In code obfuscation, disassemblers based on linear sweep:
  - ☐ a) Are not able to disassemble the following code successfully:  

```
jmp B
db B8
B:
mov eax, 0x10
push eax
call C
C:
...
```
  - ☐ b) Are able to disassemble code in option a successfully
  - ☐ c) Are the most robust brand of disassemblers
  
3. Related to the Return Oriented Programming:
  - ☐ a) Uses the particularities of `leave`; `ret` in Intel processors to run code from the stack.
  - ☐ b) It is based on searching for gadgets in existing libraries, while jumping there to execute the particular exploit. The stack is used as indirection to the particular gadgets.
  - ☐ c) It is a libc exclusive technique that uses gadgets in the library to perform system calls through return values on the stack.
  
4. DLL process injection is a technique that:
  - ☐ a) It is a technique used to substitute the running code of a victim's process by another one allowing remote code invocation.
  - ☐ b) The goal is to embed a DLL into a remote running process which will infect it by running arbitrary code. This can be used by trojans to obfuscate its presence.
  - ☐ c) An evil process, hooks a DLL to a particular external process which allows a function to be invoked when a particular event is triggered. This can be used for example by keyloggers.