HW2.4. Memory Multiple Choice

We've seen in lecture and earlier in the homework that certain parts of memory are protected from direct user access, i.e. you'd most likely segfault core-dump if you try to illegally access memory addresses mapping to those locations.

Q1.1: Which parts of memory can fall under that category? Select all that apply.

☑ (a) Code/Text ✓

☑ (b) Stack ✓

☑ (c) Heap ✓

☑ (d) Data/Static ✓

Select all possible options that apply. ?

✓ 100%

Q1.2: What are general reasons such protections exist within a specific process? (We'll discuss this more in the OS/VM section of the course!) Select all that apply.

☑ (a) To protect the user from accidentally modifying code/data they didn't mean to ✓

□ (b) To ensure separate processes know about each other so they can run simultaneously

□ (c) To error instead of segfaulting

☑ (d) To properly segfault instead of erroring ✓

☑ (e) To prevent your process from accidentally modifying other unrelated aspects of the system ✓

Select all possible options that apply. ?

✓ 100%

Q1.3: Which of the following actions are possible if we were to remove all memory protections for a specific process? Select all that apply.

☑ (a) Change what the program does ✓

□ (b) Decrease the size of physical memory

☑ (c) Change what other programs stored can do ✓

☑ (d) Change stored data ✓

☑ (e) Change how the system functions ✓

Select all possible options that apply. ?

✓ 100%

Try a new variant

Correct answer

Q1.1: Which parts of memory can fall under that category? Select all that apply.
(a) Code/Text
(b) Stack
(c) Heap
(d) Data/Static

Q1.2: What are general reasons such protections exist within a specific process? (We'll discuss this more in the OS/VM section of the course!) Select all that apply.
(a) To protect the user from accidentally modifying code/data they didn't mean to
(d) To properly segfault instead of erroring
(e) To prevent your process from accidentally modifying other unrelated aspects of the system

Homework 2

Assessment overview

Total points:	20/100
Score:	20%

Question

Value:	5
History:	5
Awarded points:	5/5

Report an error in this question

Previous question

Next question

Attached files

No attached files

Attach a file

Attach text

Q1.3: Which of the following actions are possible if we were to remove all memory protections for a specific process? Select all that apply.
(a) Change what the program does
(c) Change what other programs stored can do
(d) Change stored data
(e) Change how the system functions

Q1.1: Pretty much any part of memory can cause a segmentation fault, depending on how much protection is added. The code section tends to be protected from reads and writes (since allowing a user to modify the code generally results in security flaws). Data, Heap, and Stack generally contain segments of memory allocated by the user, but also contain buffers/metadata which help the system keep track of everything, and can contain data/heap/stack data of other processes. Trying to access memory in those areas can cause segfaults as well, though this doesn't always happen.

Q1.2: From a debugging perspective, it is generally not correct behavior to access a memory location you didn't originally initialize. As such, segfaults allow you to catch bugs more easily. Generally, if a memory access that should segfault doesn't segfault, it ends up sending a completely random block of garbage data, and often different garbage data in each run of the code.

Q1.3: The other major reason why we have these protections is to protect the system and kernel from a malicious user. Under normal operation, a computer runs multiple programs, with a "manager" program known as the "kernel" which allocates time, handles memory, and polices programs. Without segfault protections, a malicious program could potentially access memory of another program, change another program's code, or even force the kernel to run arbitrary code. CS 162 goes into what the kernel actually does, and CS 161 discusses how malicious programs or malicious users can attack C code that doesn't segfault enough.

Submitted answer 4 correct: 100%
Submitted at 2022-09-03 08:30:33 (PDT)

ⓘ hide ^

Q1.1: Which parts of memory can fall under that category? Select all that apply.
(a) Code/Text ✓
(b) Stack ✓
(c) Heap ✓
(d) Data/Static ✓
✓ 100%

Q1.2: What are general reasons such protections exist within a specific process? (We'll discuss this more in the OS/VM section of the course!) Select all that apply.
(a) To protect the user from accidentally modifying code/data they didn't mean to ✓
(d) To properly segfault instead of erroring ✓
(e) To prevent your process from accidentally modifying other unrelated aspects of the system ✓
✓ 100%

Q1.3: Which of the following actions are possible if we were to remove all memory protections for a specific process? Select all that apply.
(a) Change what the program does ✓
(c) Change what other programs stored can do ✓
(d) Change stored data ✓
(e) Change how the system functions ✓
✓ 100%

Submitted answer 3 partially correct: 66%
Submitted at 2022-09-03 08:30:29 (PDT)

ⓘ show ⌄

Submitted answer 2 partially correct: 66%

ⓘ show ⌄

Submitted at 2022-09-03 08:30:20 (PDT)

Show/hide older submissions