Wagner Spring 2014 — CS 161 Computer Security — Midterm 1

PRINT your name: _____, _____
(last) (first)

I am aware of the Berkeley Campus Code of Student Conduct and acknowledge that academic misconduct will be reported to the Center for Student Conduct.

SIGN your name: _____

PRINT your class account login: `cs161-`_____ and SID: _____

Your TA's name: _____

Your section time: _____

Name of the person sitting to your left: _____ Name of the person sitting to your right: _____

You may consult one sheet of paper (double-sided) of notes. You may not consult other notes, textbooks, etc. Calculators, computers, and other electronic devices are not permitted. Please write your answers in the spaces provided in the test. We will not grade anything on the back of an exam page unless we are clearly told on the front of the page to look there.

You have 50 minutes. There are 6 questions, of varying credit (100 points total). The questions are of varying difficulty, so avoid spending too long on any one question.

Do not turn this page until your instructor tells you to do so.

Question:	1	2	3	4	5	6	Total
Points:	15	10	15	20	16	24	100
Score:							

Problem 1 ***Security principles*** **(15 points)**

For each scenario, identify the security principle illustrated by scenario and give a short one-sentence justification of your answer. Some scenarios may represent an example of following the principle; others may represent an example of violating the principle.

(a) Bike lock manufacturers tend to have a range of different kinds of locks for customers to choose from. The high-end ones are advertised for high-crime areas and the low-end options are advertised for low to moderate crime areas.

PRINCIPLE:

JUSTIFICATION:

(b) Some paranoid people will equip their houses with high fences, a gate that only opens with a password, a home security system, and a panic room.

PRINCIPLE:

JUSTIFICATION:

(c) A company called ES&S makes electronic voting machines for use in public elections. A special password is needed to upload software updates to their voting machines. This password is identical for every ES&S machine throughout the country, hard-coded in the code, and cannot be changed. The password is documented in manuals given to election officials who need to update the software on their voting machines.

PRINCIPLE:

JUSTIFICATION:

Problem 2 ***Multiple choice*** **(10 points)**

(a) Many security experts recommend using prepared statements in your code. Which of the following threats do prepared statements defend against? Circle all that apply.

XSS	Integer overflow
CSRF	SQL injection
Clickjacking	Polymorphic worms
Buffer overruns	Session fixation
None of the above	

(b) ROP (Return-Oriented Programming) attacks are one way to exploit memory-safety vulnerabilities. Which of the following defenses can defend against ROP attacks? Circle all that apply.

Non-executable stack	Random CSRF tokens
Same-origin policy	Memory-safe programming languages
Output escaping	
None of the above	

Problem 3 ***True/false*** **(15 points)**

In parts (a)–(e), circle true or false.

(a) TRUE or FALSE: The same-origin policy would prevent Javascript running on a page from `twitter.com` from reading the cookies for `twitter.com` and sending them to `evil.com`.

(b) TRUE or FALSE: The same-origin policy would prevent Javascript running on a page from `evil.com` from reading the cookies for `twitter.com` and sending them to `evil.com`.

To prevent SQL injection attacks, `www.sweetvids.com` uses input sanitization to remove the following characters from all user-provided text fields: `'=-`. However, they forgot to include `;` in the list, and as a result, some hacker figures out a way mount a successful SQL injection attack on their site.

Based on this, which of the following are accurate? Circle true or false.

(c) TRUE or FALSE: This vulnerability was a predictable consequence of using blacklisting: it's too easy to leave something out of a blacklist.

(d) TRUE or FALSE: This bug would not have been exploitable if all modern browsers used privilege separation and sandboxing, like Chrome does.

(e) TRUE or FALSE: If `www.sweetvids.com` had used address space layout randomization (ASLR), it would have been difficult or impossible for an attacker to exploit this vulnerability.

Problem 4 ***Web security*** **(20 points)**

`www.awesomevids.com` provides a way to search for cool videos. When presented with a URL such as:

`http://www.awesomevids.com/search.php?search=cats`

The server will return an HTML search results page containing:

`...searched for: <b> cats </b> ...`

In particular, the search phrase from the URL parameter is always included into the HTML exactly as found in the URL, without any changes.

(a) The site has a vulnerability. Describe it, in a sentence or two.

(b) Alice is a user of `www.awesomevids.com`. Describe how an attacker might be able to use this vulnerability to steal the cookies that Alice's browser has for `www.awesomevids.com`. You can assume that the attacker knows Alice's email address.

(c) The developers of `www.awesomevids.com` hear rumors of this vulnerability in their site, so they deploy framebusting on all of their pages. Does this prevent exploitation of the vulnerability? Why or why not? Circle yes or no, then provide a one- or two-sentence explanation of why or why not.

YES NO

Explanation (why or why not):

Problem 5 ***More web security*** **(16 points)**

You are the developer for a new fancy payments startup, CashBo, and you have been tasked with developing the web-based payment form. You have set up a simple form with two fields, the amount to be paid and the recipient of the payment. When a user clicks submit, the following request is made:

`http://www.cashbo.com/payment?amount=<dollar amount>&recipient=<username>`

You show this to your friend Eve, and she thinks there is a problem. She later sends you this message:

> Hey, check out this funny cat picture. http://tinyurl.com/as3fsjg

You click on this link, and later find out that you have paid Eve 1 dollar via CashBo.

(Background: Tinyurl is a URL redirection/shortener service that's open to the public. Thus, Eve was able to choose what URL the link above redirects to.)

(a) Name the type of vulnerability that Eve exploited to steal one dollar from you, in the story above.

(b) What did the tinyurl link redirect to?

(c) How could you, as the developer of CashBo, defend your web service from this sort of attack? Explain in one or two sentences.

Problem 6 ***Memory safety*** **(24 points)**

Assume all preconditions are met whenever the following function is called. You may also assume that the following code is executed on a 32-bit machine.

```
/* Copy every step'th character from src to dst */
/* Requires: src,dst are valid non-NULL pointers,
   n <= sizeof(src), n <= sizeof(dst) */
void vulncopy(char* dst, char* src, int n, int step) {
    for (int i = 0; i < n; i += step) {
        dst[i] = src[i];
    }
}
```

(a) This code has a memory-safety vulnerability. Describe it.

(b) What parameters could an attacker provide to `vulncopy()` to trigger a memory-safety violation? (Your input must comply with the preconditions for `vulncopy()`.)

(c) If the vulnerable code was compiled using a compiler that inserts stack canaries, would that prevent exploitation of this vulnerability? Answer yes or no. You do not need to justify your answer.

(d) If the vulnerable code was run with DEP (Data Execution Prevention), would that prevent exploitation of this vulnerability? Answer yes or no. You do not need to justify your answer.
Reminder: DEP uses the NX (non-executable) bit to mark the stack and heap regions as non-executable, so no page in memory is both writeable and executable.