CS155: Computer Security **Spring 2022**

Take-home Final Exam

Instructions:

- **Please answer all six questions. You have three hours.**
- You may take the exam at any time during the exam window. You have three hours from the moment you begin until the moment you submit your answers on Gradescope.
- The exam is open book, open notes, and open laptops. However, you are expected to do the exam on your own. You may not interact, collaborate, or discuss the exam with another person during the exam window.
- To submit your answers please either (i) use the provided LaTeX template, or (ii) print out the exam and write your answers in the provided spaces, or (iii) write your answers on blank sheets of paper, but please **make sure to start each question on a new page.** When done, please upload your solutions to Gradescope (course code `N86BR6`).
- The LaTeX template for the final is available here. Please do not share the link with others.
- If you have questions, please post them privately on Ed and we will answer them as quickly as we can.
- Students are bound by the Stanford honor code. In particular, you are expected to do the exam on your own.

1. (*10 points*) ..... True or False

For each question, please write `T` or `F` in the space provided. No explanation needed.

_____ (a) Heap spraying is a technique used to exploit memory corruption in the heap.

_____ (b) A format string vulnerability lets the attacker read/write locations in memory that it should not have access to.

_____ (c) Marking the stack and the heap as non-executable memory prevents control hijacking attacks.

_____ (d) Consider the Spectre attack discussed in lecture 12. If memory caching were turned off, so that every memory read and write would go to main memory, then the attack would not be possible.

_____ (e) A cookie with no explicitly defined `Domain` will only be sent to the exact domain that set the cookie.

_____ (f) CORS performs a pre-flight check for all `POST` requests made through Javascript.

_____ (g) The purpose of Server Name Indication (SNI) field in the HTTPS `client-hello` message is to hide the domain name that the browser is requesting from the server.

_____ (h) DNSSEC protects users against DNS rebinding attacks.

_____ (i) Amplification-based DOS attacks can be prevented by all ISPs deploying ingress filtering for all their direct customers.

_____ (j) SMS-based two-factor authentication protects against phishing attacks.

2. (*24 points*) ..... Questions from all over with a short answer

(a) (*4 points*) Suppose a web site sets a cookie called `temp` in the browser. How can the web site instruct the browser on a subsequent visit to delete the cookie `temp`?

Your answer:

(b) (*4 points*) Suppose a web server has an exploitable memory corruption vulnerability. The web server was compiled using ProPolice to use random stack canaries. The web server is configured to restart automatically after a crash, without changing the canary value. Can an adversary extract the stack canary value from the server? If so, explain how. If not, explain why not.

Your answer:

(c) (*4 points*) A *confused deputy* vulnerability refers to a service that runs in a "high" security context. An attacker running in a "low" security context can call this service with an unexpected argument and cause an invalid operation to take place in the high security context. Slide 18 in lecture 18 gave an example of confused deputy using Android intents. Explain how the code from the lecture running in a victim app on the phone can be exploited by a malware app running on the same phone.

Your answer:

(d) (*4 points*) Suppose an attacker successfully steals the signing key of a certificate authority (CA). This lets the attacker issue certificates for any domain of its choice. Does Certificate Transparency (CT) prevent the attacker from using such a rogue certificate to carry out a man-in-the-middle (MiTM) attack against a web site that the attacker does not control? In particular, will a browser that only accepts CT certificates, accept the rogue certificate and establish an HTTPS connection with the attacker's site? Explain why or why not. Recall that the attacker can validly register its rogue certificate with a CT log, as required for CT compliance.

Your answer:

(e) (*4 points*) What is a TOCTOU bug and why is it a security vulnerability?

Your answer:

(f) (*4 points*) iPhones use a secondary secure processor that isolates secure components of the phone (e.g., Apple Wallet) and exposes only a hardened API to the primary processor. When the phone boots, the secure processor loads its firmware (i.e., the code it executes) from non-volatile memory. How do you think Apple prevents an attacker from replacing the firmware stored in non-volatile memory with malicious firmware that leaks sensitive data?
Hint: Think of a cryptographic mechanism that might help. Your solution should not prevent Apple from periodically installing firmware updates on the phone.

Your answer:

3. (*20 points*) ..... Memory tagging

ARM recently added a new security capability to ARM processors called *Memory Tagging Extension* or MTE. We discussed MTE in Lecture 3. Recall that MTE divides physical memory into granules, where each granule is 16 bytes long. Each 16-bytes granule is "tagged" by a 4-bit tag that is stored as metadata for the granule. New ARM instructions let an application set the 4-bit tag for a granule. In addition, the four top bits of every pointer contain a 4-bit tag. When a pointer is used to read or write an address in memory, the ARM memory management hardware checks that the 4-bit tag embedded in the pointer is equal to the 4-bit tag associated with the granule being accessed. If not, the memory management hardware raises an exception, which may cause the application to crash.

(a) (*4 points*) Consider the following C++ code:

```
1:   int func(char *inp) {
2:     char *s = new char(15);
3:     strcpy(s, inp);
4:     // do something
     }
```

Suppose that `malloc` on line (2) allocates the buffer `*s` on a single granule on the heap, and assigns a random 4-bit tag, say `0xA`, to that granule. The next 16-byte granule is assigned a different random 4-bit tag, say `0x5`. Moreover, `malloc` tags the pointer `s` with the 4-bit tag `0xA`. When MTE is enabled, explain what happens when the function `func` is called

- with `inp` pointing to a string of length 12 bytes;
- with `inp` pointing to a string of length 32 bytes.

Is this a desirable behavior?

Your answer:

(b) (*4 points*) Suppose the subsequent code in the function `func` does

```
5:      delete [] s;              // free the string *s
6:      Name *p = new Name;       // allocate an object on the heap
7:      s[2] = 'a';
```

Moreover, suppose that the object `*p` in line (6) is allocated in the same position as the recently freed buffer `*s`. What is the name of the potential vulnerability that line (7) introduces? Briefly explain how an attacker might be able to exploit this vulnerability. You can make whatever assumptions you need about the object `*p` and its location in memory.

Your answer:

(c) (*2 points*) Now, suppose that the `delete [] s` operation on line (5) frees the memory at `*s` on the heap and also changes the 4-bit tag assigned to that granule from `0xA` to a different random 4-bit tag, say `0x2`. For simplicity, assume that line (6) does not change the 4-bit tag assigned to the granule. Explain what will happen when the code from part (b) is executed. Is this the desired behavior?

Your answer:

(d) (*6 points*) Will re-tagging on free, as explained in part (c), prevent all the vulnerabilities of the type you identified in part (b)? If so, explain why. If not, give a short code snippet that contains a vulnerability of the type in part (b) which is still vulnerable despite the use of re-tagging on free. Briefly explain why your code snippet is vulnerable.

Your answer:

(e) (*4 points*) Suppose that all granules in a single stack frame are tagged with the same tag. However, adjacent stack frames are tagged with different tags. Will this prevent buffer overflow attacks on the stack? If so, explain why. If not, briefly explain how an attack might work.

Your answer:

4. (16 points) ..... Web text fragments and covert channels

Chrome 80 added a feature, called Web text fragments, that allows one to encode a search string in a URL. When the browser opens that URL, if the search string is found on the page, the page automatically scrolls to that location, and the string is highlighted. For example, to open the CS155 page at the "Projects" section, use the URL

`https://cs155.stanford.edu/#:~:text=Projects`

Feel free to try it out. If the string is not found on the page, no scrolling takes place.

As we saw throughout the course, benign-looking features can have significant security implications. In this question we will explore the risks associated with the Web text fragments feature.

Consider a page at `evil.com` that contains a link to a fictional site

```
<a href="https://bank.com" target="_blank"> Click me </a>
```

Suppose Bob visits `evil.com` and clicks on this link. The result is a new browser window at `bank.com`.

(a) (*2 points*) First, what browser security mechanism prevents `evil.com` from reading Bob's activity at the banking site?

Your answer:

(b) (*2 points*) Every DOM element has a `ScrollTop` property that indicates the number of pixels that the element has been scrolled to from its top position. Suppose `evil.com` could read the `ScrollTop` property of the newly opened window. Explain how `evil.com` could then use a Web text fragment to test if Bob is currently logged into `bank.com`. You may assume that if Bob is logged in then the word "Logout" appears on the page, and otherwise the word is not on the page.

Your answer:

(c) (*2 points*) How do you think Chrome prevents the attack you described in part (b)?

Your answer:

(d) (*4 points*) For security reasons, the browser will only search for the text fragment search string in the top level frame. It will not search for the search string in an embedded iframe. Explain how `evil.com` could mount your attack from part (b) if the search were applied to an embedded iframe. You may assume that `evil.com` can open `bank.com` as an iframe in the `evil.com` page. Moreover, you may assume that scrolling the iframe scrolls the entire page.

Your answer:

(e) (*2 points*) Suppose that using a covert channel, `evil.com` could measure the time that it takes the browser to open a link in a new window. Again, explain how `evil.com` could test if Bob is currently logged into the banking site. You may assume that the search for the string terminates quickly when the string is found at the top of the page compared to when the string is not found on the page at all.

Your answer:

(f) (*4 points*) How would you suggest that Chrome defend against your attack from part (e)?

Your answer:

For completeness, we note that a web page can prevent text fragments from being applied to it by including the HTTP header

```
Document-Policy: force-load-at-top
```

This header forces the page to always open at the top scroll position.

5. (*14 points*) ..... Secure Communication

SMTP is a network protocol that's used to deliver email from one organization to another. In the protocol, the sending mail server accepts messages from a sending user, establishes a TCP connection with the destination domain's email server, and then delivers the message. Unfortunately, SMTP did not include any built-in security protections. Instead, later additional protocols were introduced to improve email security.

DomainKeys Identified Mail (DKIM) is an opt-in protocol in which a mail server can cryptographically sign the email messages it sends with a private key and then attach the signature to the messages it sends. If a recipient receives a message with a signature, it looks up the sender's corresponding public key using a specially named DNS record. Using the server's public key, the recipient validates that the message originated from the server and that it hasn't been altered in transit.

Sender Policy Framework (SPF) is a complimentary email security protocol that allows organizations to whitelist a specific set of IP addresses that are allowed to send mail on behalf of a domain. This prevents attackers from sending mail on behalf of the organization. When a recipient any mail server receives any message, it looks up the allowed IP addresses for the domain through a specifically named record and drops messages that are not from one of the IPs.

(a) (*3 points*) What is a security fundamental flaw in the DKIM protocol and how would you extend the protocol to prevent this attack? Hint: Think about how the attacker might modify the message.

Your answer:

(b) (*3 points*) What about the SMTP protocol prevents off-path attackers from being able to spoof SMTP connections from an SPF-whitelisted IP address in an undetectable manner?

Your answer:

(c) (*2 points*) Organizations oftentimes whitelist all IP addresses that belong to the organization in their SPF records. How might an attacker take advantage of that error to send spam even if the mail server is secure?

Your answer:

(d) (*3 points*) Even if all of a senders' infrastructure is perfectly secure, an attacker may be able to send spam as that organization by mounting an attack against *other* Internet infrastructure or protocols. Describe how an attacker could send spam as an organization without compromising any of that organization's infrastructure.

Your answer:

(e) (*3 points*) End-to-End Encryption protocols like PGP, OTR, and Signal encrypt the contents of individual messages over email and chat. What information do they fail to protect? Why might this be a concern to a whistleblower?

Your answer:

6. (*16 points*) ..... Tor Network

As described in Lecture 16, Tor is community-run network that enables anonymous communication.

(a) (*3 points*) Is it safe to visit websites over (unencrypted) HTTP using Tor? Explain why or why not, and if not, which nodes in the Tor network is the user vulnerable from attack from.

Your answer:

(b) (*3 points*) If the NSA compromised a single popular Tor entrance node, what information could they collect about users? What will they not be able to see?

Your answer:

(c) (*3 points*) If an attacker was able to view all traffic that entered and exited the Tor network but none of the traffic within the Tor network, what information might an attacker be able to guess about a website? How could the Tor network protect against this attack?

Your answer:

(d) (*3 points*) Does using four Tor relays provide any additional tradeoffs over using three relays?

Your answer:

(e) (*4 points*) Malware has started to use Tor hidden services to hide command and control (C2) servers that compromised bots connect to to request instructions. What benefits does provide attackers? Consider both the benefits on the compromised bot-side (2 pts) and the C2-side (2 pts).

Your answer: