Peyrin & Ryan
Summer 2020

CS 161
Computer Security

HW7

To prepare you for the final format, Homework 7 is formatted exactly like the final. Please fill in your answers on the Homework 7 assignment on Gradescope.

The biggest difference from the midterm format is that the answer sheet is on Gradescope instead of on paper.

For questions with **circular bubbles**, you may select exactly *one* choice on the answer sheet.

○ Unselected option

● Only one selected option

For questions with **square checkboxes**, you may select *one* or more choices on the answer sheet.

■ You can select

■ multiple squares

For questions with a **large box**, you need to write a short answer in the corresponding text box on the answer sheet.

You have 1 week (170 minutes for the actual exam). There are 4 questions of varying credit (60 points total).

[NOTE: The Gradescope HW7 assignment is untimed, but the real final will be a timed assignment. See the "Sample Timed Answer Sheet" on Gradescope for an example.]

The Gradescope answer sheet assignment has a time limit of 170 minutes. Do not click "Start Assignment" until you're ready to start the exam. The password to decrypt the PDF is at the top of the answer sheet.

[NOTE: Most questions on this homework are from past exams, so we will grade the homework on completeness. These questions are similar to what you should expect on the final exam, so we recommend trying the questions as a practice exam before checking the solutions.]

The exam is open note. You can use an unlimited number of handwritten cheat sheets, but you must work alone.

Clarifications will be posted at https://cs161.org/clarifications.

Q1 *MANDATORY – Honor Code* (2 points)

Read the honor code on the Gradescope answer sheet and type your name. ***Failure to do so will result in a grade of 0 for this exam.***

Q2 *True/False* (16 points)

[NOTE: The first question on the exam will always be True/False.]

Each true/false is worth 2 points unless otherwise specified.

[NOTE: On Gradescope, every question will be labeled as being worth 1 point–you should ignore this. The real point values are on the exam PDF.]

Q2.1 TRUE or FALSE: If a victim is logged into a session on `https://bank.com/` in one tab and visits an attacker's website in another, the attacker can run JavaScript to load a form at `https://bank.com/transfer` and extract the CSRF token from it.

○ TRUE ● FALSE

Solution: False. SOP prevents this.

Q2.2 TRUE or FALSE: An on-path attacker can learn the request parameters of a GET request loaded over HTTPS.

○ TRUE ● FALSE

Solution: False. The request parameters will be encrypted.

Q2.3 TRUE or FALSE: TLS has end-to-end security, so it is secure against an attacker who steals the private key of the server.

○ TRUE ● FALSE

Solution: False. An attacker who's stolen the private key of the server could impersonate the server to the victim.

Q2.4 TRUE or FALSE: If the entire Internet stopped using HTTP POST requests and only allowed HTTP GET requests, CSRF attacks would still be possible.

● TRUE ○ FALSE

Solution: True. An attacker can force a victim to click on a link that generates an HTTP GET request with server-side effects.

Q2.5 TRUE or FALSE: It is secure for a server to generate session tokens based only on timestamp to the nearest second, as long as every user receives a unique token.

○ TRUE ● FALSE

Solution: False. Now an attacker can brute-force tokens and possibly log in as another user.

Q2.6 TRUE or FALSE: If every website uses TLS and every cookie has the secure flag set, clickjacking attacks are still possible.

● True ○ False

Solution: True. TLS defends against network attacks, not web/application layer attacks, and clickjacking attacks do not need cookies to succeed.

Q2.7 Assume you've set up a 3-relay Tor circuit to access some websites over HTTPS. A malicious adversary takes control of the entry relay, but the other two are honest and uncompromised. The adversary can now learn which website you are visiting.

○ True ● False

Solution: False, the entry relay can learn your identity but not which site you are visiting, and there is no way to correlate the two.

Q2.8 True or False: In Bitcoin, once your transaction is successfully added to a block that lives on the longest chain, you can be guaranteed that it will never be lost.

○ True ● False

Solution: False. The blockchain could fork and not include your transaction.

This is the end of Q2. Proceed to Q3 on your answer sheet.

Q3 *Infrastructure Week* (18 points)

For each public-key infrastructure (PKI) scheme, mark whether it provides the same trust guarantees as the standard PKI from lecture for all certificates, some certificates, or no certificates at all. Assume that everyone has the root certificate hardcoded into their machines.

Q3.1 (3 points) Each server can only sign the public keys of its grandchildren (two descendants below the current level). For example, the root server can sign the public key of `berkeley.edu` but not `.edu`, and the `.edu` server can sign the public key of `eecs.berkeley.edu` but not `berkeley.edu`.

[NOTE: Your answer sheet has six answer choices for every subpart, but not every question will have six answer choices. For example, for Q2.1 here, you should not use options (D), (E), and (F) on your answer sheet.]

○ (A) All certificates ○ (C) No certificates ○ (E) —

● (B) Some certificates ○ (D) — ○ (F) —

Solution: This works for any certificates located an even number of levels below the root. However, there is no way to create a path of trust from the root to a certificate located an odd number of levels below the root, such as `eecs.berkeley.edu`.

Q3.2 (3 points) As in the previous part, each server can only sign the public keys of its grandchildren. However, the root is additionally allowed to sign the public key of its direct children. For example, the root server can sign the public key of `.edu` and `berkeley.edu`. The `.edu` server can sign the public key of `eecs.berkeley.edu` but not `berkeley.edu`.

[NOTE: The answer choices on this subpart are circular bubbles, so you should only bubble in one option out of (G), (H), (I), on your answer sheet for Q2.2.]

● (G) All certificates ○ (I) No certificates ○ (K) —

○ (H) Some certificates ○ (J) — ○ (L) —

Solution: Skipping two levels at a time, all certificates must have a path of trust that ends at either root or a server one level below root. Since the root is allowed to sign public keys of servers one level below it, this scheme now works for all certificates.

Q3.3 (3 points) Same setup as the previous part, but an attacker has compromised a server one level below the root (e.g. `.edu`).

○ (A) All certificates ○ (C) No certificates ○ (E) —

● (B) Some certificates ○ (D) — ○ (F) —

Solution: Any certificate whose path to the root doesn't go through the compromised server (e.g. `google.com`) is unaffected, but a certificate whose path goes through the compromised server (e.g. `berkeley.edu`) cannot be trusted.

Q3.4 (3 points) The root handles all requests and sends the requested public key and a certificate directly through a TLS connection.

● (G) All certificates ○ (I) No certificates ○ (K) ——

○ (H) Some certificates ○ (J) —— ○ (L) ——

Solution: TLS provides end-to-end integrity.

Q3.5 (3 points) Instead of signing, use a cryptographic hash to create a certificate. For example, the root server signs the public key of `.edu` by hashing it.

○ (A) All certificates ● (C) No certificates ○ (E) ——

○ (B) Some certificates ○ (D) —— ○ (F) ——

Solution: Hashes don't provide integrity. An attacker can create a valid signature on their malicious public key just by hashing it.

Q3.6 (3 points) Instead of signing, use HMAC to create a certificate. For example, the root server signs the public key of `berkeley.edu` by applying `HMAC`(K, `berkeley.edu`), where K is the root's private signing key.

○ (G) All certificates ● (I) No certificates ○ (K) ——

○ (H) Some certificates ○ (J) —— ○ (L) ——

Solution: HMACs use symmetric keys, so there is no way for the signatures to be verified without knowing the server's secret key.

[NOTE: You may not need all blanks on the answer sheet for a question. You should leave Q2.5 and Q2.6 blank on your answer sheet. Also, since none of the subparts of this question asked for a short answer, you should leave the space below Q2 on your answer sheet blank.]

This is the end of Q3. Proceed to Q4 on your answer sheet.

Q4 *Network Security* (24 points)

Answer the following questions about network security.

Q4.1 (3 points) Bob connects his laptop to the DeCafe coffee shop's Wifi, which anyone nearby can join without a password. He browses to the website `http://www.foocorp.com`. At the table next to him is an evil attacker, Mallory, who has also joined the DeCafe Wifi network. What kind of threat model best describes Mallory when she first joins the network, with respect to Bob's connection with DeCafe router's?

○ (A) Off-path attacker ○ (C) In-path attacker ○ (E) ——

● (B) On-path attacker ○ (D) None of the above ○ (F) ——

Solution: Other users on a Wifi network with shared passwords are On-path attackers since Wifi packets are broadcast (through the air).

Q4.2 (4 points) Bob returns home and types into his browser `www.foocorp.com`. Suppose that Mallory has managed to poison the DNS cache on Bob's laptop, such that it now thinks the IP address of `www.foocorp.com` is 6.6.6.6, which is the IP address of a server that Mallory controls.

Which of the following statements are true? Select all that apply.

[NOTE: The answer choices on this subpart are square bubbles, so you should bubble in any of options (G), (H), (I), (J) you think are correct, or only option (K) if you think all options are incorrect. You should leave option (L) on your answer sheet blank.]

□ (G) Mallory will be unable to steal Bob's cookies for `http://www.foocorp.com` if `http://www.foocorp.com` uses HTTP-Only cookies.

□ (H) Mallory will be unable to steal Bob's cookies for `http://www.foocorp.com` if `http://www.foocorp.com` uses a CSP policy that only allows scripts to be loaded from sources on `foocorp.com`

■ (I) Mallory will be unable to steal `foocorp.com` cookies marked with the `secure` flag.

□ (J) Mallory will be unable to inject JavaScript into `http://www.foocorp.com`

□ (K) None of the above

□ (L) ——

Solution: DNS results tell the machine which IP address a hostname / domain actually resides on. Thus, HTTP-only cookies do not protect Bob in this situation, since his browser will think that the server at 6.6.6.6 is actually foocorp.com (and thus send the cookies to Mallory's malicious server). Similarly, CSP provides no protection because Bob's machine will believe that 6.6.6.6 is foocorp.com and send the cookies by default.

Mallory won't be able to steal foocorp.com's cookies if they are marked as secure because they will ***ONLY*** ever be sent over HTTPS connections, which Mallory cannot establish since she does not have the private key that matches foocorp.com's certificate.

Mallory will be able to inject conect into http://www.foocorp.com, since Bob's browser will fetch the website's code (HTML, JS, etc.) from Mallory's malicious server. (Thus the answer that says she is **unable** to do so is false).

Mallory will be able to steal Bob's cookies even if foocorp uses HTTPS since Bob's browser does not necessarily know that foocorp uses HTTPS. When he browses to **www.foocorp.com** Mallory's malicious server can just start an unprotected HTTP session and serve him malicious content / receive Bob's cookies.

Q4.3 (5 points) Suppose that `foocorp.com` domain has the following four subdomains: (`www`, `alphabet`, `sushi`, `money`).
The attacker knows that `foocorp.com` has only four subdomains but does not know any of their names, and wishes to discover the subdomains using the zone enumeration attack discussed in class.

Assuming every DNS server uses plain NSEC, what is the minimum number of queries the attacker needs to make to `foocorp.com`'s nameservers in the worst-case for the attacker?

[NOTE: If a question asks for a short answer (empty box below the question on the exam), you should type your answer in the corresponding text box on Gradescope.]

○ (A) —— ○ (B) —— ○ (C) —— ○ (D) —— ○ (E) —— ○ (F) ——

Solution: Suppose that the attacker correctly guesses 'alphabet', 'money', and 'www' correctly on his first three tries. There are now 3 regions where the final domain might live: `alphabet -> money`, `money -> www`, and `www -> alphabet`. If the attacker searches the region from www -> alphabet (e.g., querying 'zzzzzz') and the region from alphabet -> money (e.g., querying 'bbbbbb'), she will learn that no subdomains exist in those regions. But she's now made 5 queries in total; when she searches the space between money -> www, the final subdomain sushi will be revealed, leading to a total of 6 queries. Since the attacker knows that there are exactly 4 subdomains, she is done.

Q4.4 (4 points) Suppose that a user Alice is browsing the Internet at home and Mallory is an on-path attacker.

In which of the following scenarios will Mallory be able to identify whether or not Alice is visiting a website on `foocorp.com`? Select all that apply.

■ (G) Alice's machine and local DNS resolver randomize the source port of DNS queries; `foocorp.com`'s NS server use DNS (without DNSSEC); `foocorp.com` does not use HTTPS

■ (H) Alice's machine and local DNS resolver use a fixed source port for every DNS query; `foocorp.com`'s NS server uses DNSSEC with plain NSEC; `foocorp.com` does not use HTTPS

■ (I) Alice's machine and local DNS resolver use a fixed source port for every DNS query; `foocorp.com`'s NS server uses DNSSEC with NSEC3; `foocorp.com` does not use HTTPS

■ (J) Alice's machine and local DNS resolver use a fixed source port for every DNS query; `foocorp.com`'s NS server uses DNSSEC with NSEC3; `foocorp.com` uses HTTPS

□ (K) None of the above

□ (L) ——

Solution: DNS does not provide confidentiality for a user's queries, regardless of whether or not DNSSEC is used. Thus an on-path attacker can observe that Alice is visiting a foocorp.com website if Alice's browser ever issues a DNS query for a subdomain on foocorp.com Moreover, HTTPS doesn't hide the src and destination IP addresses of the connection. So even if Alice's browser never issues a DNS query about foocorp.com's sites, an on-path attacker can simply check whether the destination IP address of any of Alice's sessions matches an IP address for foocorp.com. As a result, none of the solutions hide any destination site that Alice is visiting from Mallory.

Q4.5 (5 points) `FooCorp` has chosen to use very short TTLs in all of their DNS responses. Which of the following statements are true? Select all that apply.

□ (A) Short TTLs help protect against attacks where `FooCorp`'s DNS servers have been compromised

□ (B) Assuming all DNS servers used DNSSEC with plain NSEC, then `FooCorp`'s decision to use short TTLs will increase the amount of work that the DNS servers of `FooCorp`'s parent zone need to perform

■ (C) Short TTLs increase the number of requests `FooCorp`'s DNS servers need to support

□ (D) Short TTLs help protect against DNS cache poisoning attacks by an on-path attacker

□ (E) Short TTLs help protect against blind-spoofing attacks

□ (F) None of the above

Solution: FooCorp's decision to use short TTLs will actually hurt / cause attacks to be worse in the case of a compromise of their DNS servers, since the short TTLs will mean that users will be more likely to request new DNS records during the window of compromise. And during the window of compromise, the attacker has full control to set DNS responses to whatever they want (including the TTL values, regardless of what FooCorp's original policy was).

Short TTLs do not provide any mitigation against spoofing attacks: if a spoofed DNS response succeeds, the attacker gets to set their own TTL; so the attacks involving on-path and blind-spoofing attacks are incorrect.

Short TTLs do not involve the parent zone in NSEC.

Q4.6 (5 points) FooCorp hosts *all* of its servers on machines provided by CheapCloud: a large, but unreliable, cloud hosting provider. CheapCloud suffers from two major problems: (i) they have frequent data breaches; and (ii) they often need to assign new IP addresses to their customers' servers. Nevertheless, CheapCloud promptly notifies their customers whenever either of these events occurs.

Which of the following designs or techniques can FooCorp use to help mitigate some of the security issues caused specifically by CheapCloud's poor environment? Select all that apply.

■ (G) FooCorp uses plain DNS and sets short TTLs for all of its DNS responses

□ (H) FooCorp uses RSA-based TLS

□ (I) FooCorp uses DNSSEC with plain NSEC

■ (J) FooCorp uses DHE-based TLS

□ (K) FooCorp uses DNSSEC with NSEC3

□ (L) None of the above

Solution: Short TTLs will be useful here: because CheapCloud frequently issues new IP addresses to customer servers, that means that the DNS records for FooCorp will frequently be incorrect and point to the server of another customer. Short TTLs force resolvers to clear their cache entries more frequently, and thus issue new queries and receive the updated information more often.

Diffie-Hellman provides forward secrecy, which will help mitigate the frequent breaches that CheapCloud's servers face: even if an attacker obtains a private key from a FooCorp server in one of these data breaches, forward secrecy protects the confidentiality of past TLS sessions.

DNSSEC isn't relevant to CheapCloud's environment since we're not worried about spoofed DNS records; the bigger concern stems from the frequently changing IP addresses, which short TTLs addresses.

This is the end of Q4. You have reached the end of the exam.