Department of Electrical Engineering and Computer Science

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

6.858 Spring 2018

Quiz II

You have 120 minutes to answer the questions in this quiz. In order to receive credit you must answer each question as precisely as possible.

Some questions are harder than others, and some questions earn more points than others. You may want to skim them all through first, and attack them in the order that allows you to make the most progress.

If you find a question ambiguous, be sure to write down any assumptions you make. Be neat and legible. If we can't understand your answer, we can't give you credit!

Write your name and submission website email address on this cover sheet.

This is an open book, open notes, open laptop exam.
NO INTERNET ACCESS OR OTHER COMMUNICATION.

This quiz is printed double-sided.

Please do not write in the boxes below.

I (xx/15)	II (xx/12)	III (xx/6)	IV (xx/10)	V (xx/17)	VI (xx/12)	VII (xx/12)	VIII (xx/6)	IX (xx/6)	X (xx/4)	Total (xx/100)

Name:

Submission website email address:

You can answer the feedback questions on the back of the quiz before the official start time.

This page intentionally left blank.

I Multiple-choice questions

For all of the multiple choice questions, please mark **all** choices that apply.

1. [4 points]: Based on the paper *"SoK: SSL and HTTPS: Revisiting past challenges and evaluating certificates trust model enhancements"*, which of the following statements are true?

A. Valid DV certificates provide more confidence to a user that she is connecting to the intended party than valid EV certificates.

B. OCSP stapling allows a server to prove to a browser that its certificate hasn't been revoked.

C. DANE makes it difficult for an adversary to launch a SSL stripping attack.

D. Server key-pinning makes it harder for an adversary to convince a CA to mint a certificate for a site and launch an MITM attack on that site.

2. [4 points]: Based on the paper *"Click Trajectories: End-to-End Analysis of the Spam Value Chain"*, which of the following statements are true? "Spammers" here refer to operators of various parts of the "spam value chain."

A. Spammers run their spam-advertised web sites on compromised user machines that are part of a botnet.

B. Spammers need to register domain names in order for their spam-based advertisements to be effective.

C. Credit card network operators, such as Visa, perform random purchases to check whether transactions are correctly coded.

D. There is a high cost for spammers to switch acquiring banks.

This page intentionally left blank.

3. [4 points]: Based on Mark Silis and Jessica Murray's guest lecture, which of the following statements are true?

A. Portions of MIT's `18.*.*.*` address space are announced via BGP by Akamai to protect against denial-of-service attacks.

B. The biggest security impact of selling parts of `18.*.*.*` address space to Amazon was that AFS had hard-coded permissions allowing access from any net-18 address.

C. MIT uses a single ISP, allowing IS&T to outsource firewall and intrusion detection work to them.

D. MIT IS&T relies on backups to defend against malware that encrypts user data and holds it hostage until the user pays a ransom fee.

4. [3 points]: Based on the paper *"LAVA: Large-scale Automated Vulnerability Addition"*, which of the following statements are true?

A. The code added by LAVA is easy to detect.

B. Even if a bug can be detected, some bugs introduced by LAVA are unreachable (i.e., no input causes the buggy code to execute).

C. Even if a bug can be reached, some bugs introduced by LAVA are not exploitable (i.e., no input allows an adversary to execute arbitrary code).

This page intentionally left blank.

II Symbolic/concolic execution

Ben writes the following test case for lab 3:

```
def f(n):
   i = 0
   while i < n:
      print "1: Iteration"
      i += 1
   print "2: Done"
   return i

def test_f():
   v = f(fuzzy.mk_int("n"))

f_results = fuzzy.concolic_execs(test_f)
```

5. [6 points]: How many times would the `print "2: Done"` statement be executed? Explain where this number comes from, or why it's impossible to tell.

6. [6 points]: How many times would the `print "1: Iteration"` statement be executed? Explain where this number comes from, or why it's impossible to tell.

This page intentionally left blank.

III Web security

Ben Bitdiddle proposes a new approach to protect against cookie-stealing attacks, like the ones you did in lab 4. Ben's idea is to use the `path` attribute of the cookie to limit which pages can access the cookie through `document.cookie`. In particular, a page can access a cookie only if its URL matches the cookie's path. Ben also modifies Zoobar to set the login cookie to `/zoobar/index.cgi/login`.

7. [6 points]: Can an attacker still obtain a victim's cookie, as in lab 4, with Ben's new defense mechanism and Ben's modified Zoobar? Either explain why this is not possible, or describe a specific attack, including snippets of HTML and Javascript code the attacker could use.

This page intentionally left blank.

IV TCP hijacking

As described in the paper *A look back at "security problems in the TCP/IP protocol suite"*, the TCP/IP protocol for establishing a connection uses the following messages:

A. C → S: SYN(ISN_C), SRC=C

B. C ← S: SYN(ISN_S), ACK(ISN_C), SRC=S

C. C → S: ACK(ISN_s), SRC=C

D. C → S: byte0, ISN_C, SRC=C, ACK(ISN_s)

E. C → S: byte1, $ISN_C + 1$, SRC=C, ACK(ISN_s)

F. ...

TCP uses sequence numbers (as shown in message D and E) to detect duplicate bytes and deliver bytes to S in the order they were sent by C. As the paper describes, S increases the initial ISN by 128 every second and by 64 per new connection.

As described in the paper, this protocol is vulnerable to hijacking: an attacker can pretend to be C by suppressing message B and guessing the initial sequence number that S is proposing to C.

8. [5 points]: Assuming S isn't busy, how can an adversary guess the sequence number that it must send in message C efficiently? (Briefly describe.)

9. [5 points]: To defend against connection hijacking, Ben proposes to modify the TCP hand-shake protocol, replacing the initial ISN_S and ISN_C (in messages A and B) with random numbers, but still incrementing them sequentially as data is transmitted. Would this make hijacking harder? (Briefly explain.)

This page intentionally left blank.

V Secure handshake

Alisa layers the following protocol over TCP/IP to set up a secure channel that provides confidentiality and integrity:

A. C → S: connect with TCP

B. C ← S: PK_t, Sign(SK_A, {S: PK_S}), Sign(SK_S, {PK_t})

C. C → S: Encrypt(PK_t, $\{K, SN\}$)

D. C → S: c_1 = Encrypt(K, $\{msg_1, SN\}$), t = MAC(K, c_1)

E. C → S: c_2 = Encrypt(K, $\{msg_2, SN+1\}$), t = MAC(K, c_2)

F. ...

Public keys are denoted as PK, and their corresponding secret key as SK. A private key for symmetric ciphers is denoted as K. "SN" is a unique sequence number. { and } mark a message.

You can assume that everyone knows the PK_A of the certificate authority. Furthermore, as soon S doesn't need SK_t in the protocol anymore (i.e., right after processing message C), S deletes SK_t from memory.

10. [6 points]: Ben doesn't like certificate authorities so he suggests replacing message B with:

C ← S: PK_t, Sign(SK_S, {PK_t}).

That is, the server sends C a signed public key, but without the certificate. Ben also requires client C to store the PK_S in C's file system on the first successful connection to S. He further modifies the protocol to check the PK_S in message B against the stored key on subsequent connections.

What attack could an adversary launch? (Briefly describe the attack.)

This page intentionally left blank.

11. [6 points]: If the protocol doesn't include a unique *SN* in message C and D, what attack could an adversary launch? What concrete steps could an adversary take, which messages would he need to receive or send, and what bad thing would happen as a result?

12. [5 points]: Ben proposes to simplify the protocol by replacing PK_t with PK_s in message B and C. That is, Ben's protocol omits using PK_t. What attacks is Ben's protocol vulnerable to that Alissa's protocol isn't? What concrete steps could an adversary take, which messages would he need to receive or send, and what bad thing would happen as a result?

This page intentionally left blank.

VI Side channels

Consider the Spectre example implementation shown in Append A of the paper “Spectre attacks: exploiting speculative execution” by Kocher et al. In this example, the code attempts to learn the secret “secret” through a side-channel by running the function `victim_function`.

Suppose that you changed the value “256” to “128” on lines 49 and 81.

13. [6 points]: Would this code still print out the same secret?

14. [6 points]: Suppose that the secret value is an arbitrary AES key (recall that AES keys are 128-bit values). Would this attack work to recover the AES key?

This page intentionally left blank.

VII Bitcoin

At the beginning of the semester, Alyssa P. Hacker got 1 Bitcoin and challenged Ben Bitdiddle to steal it. Alyssa tells Ben the transaction in which she acquired 1 Bitcoin (and her public key), but Ben does not know Alyssa's private key (and Alyssa stores it in a way that Ben cannot obtain it).

Ben has been working hard on this problem, and found a friend that can lend him some chips that compute SHA-256 hashes much faster than existing Bitcoin miners—so much so that they give him about 60% of the "mining power" in Bitcoin! Ben's friend is willing to lend these chips to Ben for a few months.

15. [6 points]: Suppose Alyssa transfers her Bitcoin to her friend Charlie. Can Ben transfer Alyssa's Bitcoin to himself? Give a precise attack or explain why not.

16. [6 points]: Can Ben ensure that Alyssa cannot give her Bitcoin to someone else? Describe how, or explain why not.

This page intentionally left blank.

VIII Tor

Alyssa's nosy neighbor, Norbert, runs two popular web sites. Alyssa is visiting one of Norbert's web sites through Tor, but Norbert doesn't know which one. The only thing Norbert can observe is Alyssa's encrypted WiFi transmissions (he does not know Alyssa's WiFi password, or which Tor nodes she happens to use).

17. [6 points]: Describe an attack by which Norbert can determine which of Norbert's two sites Alyssa is visiting.

This page intentionally left blank.

IX Secure messaging

The 6.858 course staff design the following messaging protocol, which is a variation of the protocol discussed in lecture. Each user maintains a long-term signing key, *SK*, and knows the other users' corresponding public keys, *PK*. To start a conversation, each user chooses a fresh Diffie-Hellman key (say, *a* for Alice) and sends the corresponding public key (say, g^a for Alice) to the other user. Each user signs their public key (with their long-term signing key), to prevent MITM attacks, and then signs the other user's public key, to acknowledge its receipt:

A. A → B: g^a, "Alice", Sign(SK_{Alice}, $\{g^a\}$)

B. A ← B: g^b, "Bob", Sign(SK_{Bob}, $\{g^b\}$)

C. A → B: "ACK", Sign(SK_{Alice}, $\{g^b\}$)

D. A ← B: "ACK", Sign(SK_{Bob}, $\{g^a\}$)

The users (Alice and Bob in this example) then derive a shared secret key, g^{ab}, using Diffie-Hellman, and exchange messages by using `Seal()`, as described in the lecture notes. `Seal()` provides authenticated encryption: an adversary cannot determine the contents of the message, or construct another valid message, without knowing the shared key. Messages must unseal correctly (specifically, `Seal`'s MAC tag must match) before they are displayed to the recipient.

18. [6 points]: Describe how David, one of the 6.858 TAs, can send the message "Lecture is canceled" to Nickolai as if the message was sent by Frans. Assume there are no software bugs, the cryptographic primitives (signatures, Diffie-Hellman, and `Seal`) are secure, and David does not have physical access to Nickolai's or Frans's computer.

X 6.858

We'd like to hear your opinions about 6.858. Any answer, except no answer, will receive full credit.

19. [2 points]: Are there any papers in the second part of the semester that you think we should definitely remove next year? If not, feel free to say that.

20. [2 points]: Are there topics that we didn't cover this semester that you think 6.858 should cover in future years?

End of Quiz