Department of Electrical Engineering and Computer Science

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

6.033 Computer Systems Engineering: Spring 2014

Quiz II

There are 18 questions and 17 pages in this quiz booklet. Answer each question according to the instructions given. You have **100 minutes** to answer the questions.

For true/false and yes/no questions, there will be no negative points for incorrect answers.

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 in the space below. Write your initials at the bottom of each page.

THIS IS AN OPEN BOOK, OPEN NOTES, OPEN LAPTOP QUIZ, BUT DON'T USE YOUR LAPTOP FOR COMMUNICATION WITH OTHERS. TURN YOUR NETWORK DEVICES OFF.

CIRCLE your recitation section number:

10:00	1. Butler/Eirik	2. Katrina/Pratiksha	3. Arvind/Qian
11:00	4. Butler/Eirik	5. Arvind/Qian	6. Katrina/Pratiksha
12:00	11. Mark/Lixin		
1:00	7. Karen/Bryan	9. Peter/Tiffany	12. Mark/Lixin
2:00	8. Karen/Bryan	10. Peter/Tiffany	

Do not write in the boxes below

1-2 (xx/14)	3-5 (xx/14)	6-10 (xx/18)	11-15 (xx/20)	16-18 (xx/16)	Total (xx/82)

Name:

Initials:

I Lecture Questions

1. [12 points]: Indicate whether each of the following statements about computer security are true or false.

(Circle True or False for each choice.)

A. True / False Password salting improves security as long as the underlying hash function is invertible.
Answer: False.

B. True / False When an certification authority like Verisign is compromised, online organizations that have acquired certificates from the compromised authority can be impersonated. Organizations that obtained their certificates from other trusted authorities, however, do not become susceptible to impersonation.
Answer: False.

C. True / False Suppose Alice and Bob wish to exchange their public keys on a wireless channel. Assume only one attacker, Eve, exists in the system. Eve can snoop on the wireless medium but cannot transmit any packets or signals. In this case, Alice and Bob can transmit their public keys in the clear on the wireless channel and later use public key cryptography to communicate securely.
Answer: True.

D. True / False Consider a modification to BGP in which each AS has a public/private key pair. Every AS knows the correct public key every other AS. Each AS also knows the origin AS for each prefix, i.e., the AS that owns each IP prefix. This modified BGP changes BGP routing announcements such that each AS along a path concatenates its AS number with the time of the day and the IP prefix and signs the concatenated text with its private key. Thus for prefix P, instead of the BGP announcement being $P : AS1, AS2; AS3$, the modified announcement would be

$$P : (AS1, time, P)_{SK_{AS1}}; (AS2, time, P)_{SK_{AS2}}; (AS3, time, P)_{SK_{AS3}}$$

where P is the routed prefix, $time$ refers to the current time, and $(\ldots)_{SK_{ASj}}$ denotes the text being signed with the private key of ASj. Assume all ASes have synchronized clocks and that all private keys are secure. The modified BGP ensures that if a rogue AS tries to hijack the route to prefix P, other ASes that receive the announcement can detect that the route is fake.
Answer: False.

E. True / False Cross-site scripting attacks require the victim's browser to have cookies enabled.
Answer: False.

F. True / False A TCP SYN Flood attack on a server may cause the server to drop legitimate customers but does not reveal private server data to the attacker.
Answer: True.

2. [2 points]: According to the lecture by Hal Abelson, why are "the lights burning late in Mountain View" right now? (In other words, according to Hal, why are employees at Google staying late and working hard?)

(Circle the BEST answer)

A. Because of the recent revelations that Google has cooperated with the NSA to provide information about US citizens.

B. Because of ongoing patent disputes between Google and Apple over mobile phone patents.

C. Because of a recent ruling in which it was found that Google could be compelled to remove certain search results from their website.

D. Because they are scrambling to prepare for the widespread public release of Google Glass.

Answer: B

Initials:

3. [3 points]: Consider a transaction processing system running three transactions, T1, T2, and T3, which read and write three data items, A, B, and C, employing two-phase locking (at data-item granularity, with reader-writer locks), write-ahead logging, and log-based recovery.

Suppose the log on disk has the following contents immediately prior to a system crash:

Begin T1	Begin T2	Begin T3	T1 WA Before: 0 After: 10	T2 WB Before: 0 After: 5	COMMIT T2	T1 WB Before: 5 After: 10	T3 WC Before: 5 After: 10	ABORT T3

Here the notation `Tx Wy` means that transaction `x` wrote to data item `y`, with the before and after images of `y` shown below the write operation.

After the system recovers, what values will A, B, and C contain?

(Write the value of each record in the spaces below.)

A. _______

B. _______

C. _______

Answer: 0, 5, 5. Only transaction 2 commits, so only its effects are visible. T3's WC shows that the before value of C was 5.

Initials:

4. [8 points]: Consider a transaction processing system running three transactions, T1, T2, and T3, which read and write three data items, A, B, and C, employing two-phase locking (at data-item granularity, with reader-writer locks), write-ahead logging, and log-based recovery (this is the same setup as in the previous problem.)

Suppose the system runs for some time, and then deadlocks (with none of the three transactions able to make progress.) You know that the previous two statements that successfully completed were a write to A by T1 and a read of B by T2, and there have been no commits or aborts since these statements completed.

Which of the following could explain the observed behavior:

(Circle True or False for each choice.)

A. True / False T1 is waiting for T3 to release a lock on B; T2 is waiting for T1 to release a lock on A; T3 is waiting for T2 to release a lock on C.
Answer: True. T3 (and T2) could have a read lock on B, which still would have allowed T2 to read B but would have prevented T1 from writing it.

B. True / False T1 is waiting for T2 to release a lock on B; T2 is waiting for T1 to release a lock on A; T3 is waiting for T1 to release a lock on A.
Answer: True. Both T2 and T3 can be waiting for A.

C. True / False T1 is waiting for T2 to release a lock on A; T2 is waiting for T1 to release a lock on A; T3 is waiting for T2 to release a lock on C.
Answer: False. T1 can't be waiting for T2 since it was able to write to A.

D. True / False T1 is waiting for T2 to release a lock on C; T2 is waiting for T3 to release a lock on B; T3 is waiting for T1 to release a lock on A.
Answer: True. T2 might want to upgrade its read lock on B.

5. [3 points]: Which of the following statements best describes the role of two-phase commit in transaction processing?

(Circle the BEST answer)

A. It provides all-or-nothing atomicity when data is partitioned across multiple machines.

B. It improves the availability of a multi-node transaction processing system.

C. It prevents two transactions from concurrently modifying the same data item.

D. It ensures that multiple nodes will agree on the outcome of some transaction at exactly the same time.

Answer: A. D is not a good choice because nodes do not agree at exactly the same time.

Initials:

II PNUTS

Alices stores some personal information on a Yahoo! website that uses PNUTS to store user data, in particular the name of her current employer and the current email address she is using for work. She creates a single record, `r`, that contains both of these values, so that she can change one or both values with a single write operation. She wants any application using that information to see a consistent version of the two. PNUTS read operations on `r` will return both its current version number and current set of values.

6. [5 points]: Which of the following ways of reading the data are guaranteed to leave a consistent version in the variables `emp` and `email` after they complete? Here `r.employer` denotes reading the employer field from `r`.

(Circle "True" for each approach that produces consistent results, and "False" for ones that do not.)

A. True / False

```
(v1, emp) = read-any(r).employer;
(v2, email) = read-any(r).email;
```

B. True / False

```
(v1, emp) = read-any(r).employer;
(v2, email) = read-critical(v1,r).email;
```

C. True / False

```
(v1, emp) = read-latest(r).employer;
(v2, email) = read-critical(v1,r).email;
```

D. True / False

```
while (true) do {
    (v1, emp) = read-any(r).employer;
    (v2, email) = read-any(r).email;
    if (v1 == v2):
       break
}
```

E. True / False There is no way to get a version that is guaranteed to be consistent.

Answer: F, F, F, T, F. (a)-(c) can give inconsistent results if the two reads go to different replicas. The test in (d) ensures that both reads are getting the same version.

Initials:

Alice changes her design so that the employer name and e-mail address are kept in two separate records, `r1` and `r2`, respectively.

7. [5 points]: Now which of the following ways of reading the data (which are the same as in the previous question, rewritten to use `r1` and `r2` instead of `r.employer` or `r.email`) are guaranteed to leave a consistent version in the variables `emp` and `email` after they complete?

(Circle "True" for each approach that produces consistent results, and "False" for ones that do not.)

A. True / False

```
(v1, emp) = read-any(r1);
(v2, email) = read-any(r2);
```

B. True / False

```
(v1, emp) = read-any(r1);
(v2, email) = read-critical(v1,r2);
```

C. True / False

```
(v1, emp) = read-latest(r1);
(v2, email) = read-critical(v1,r2);
```

D. True / False

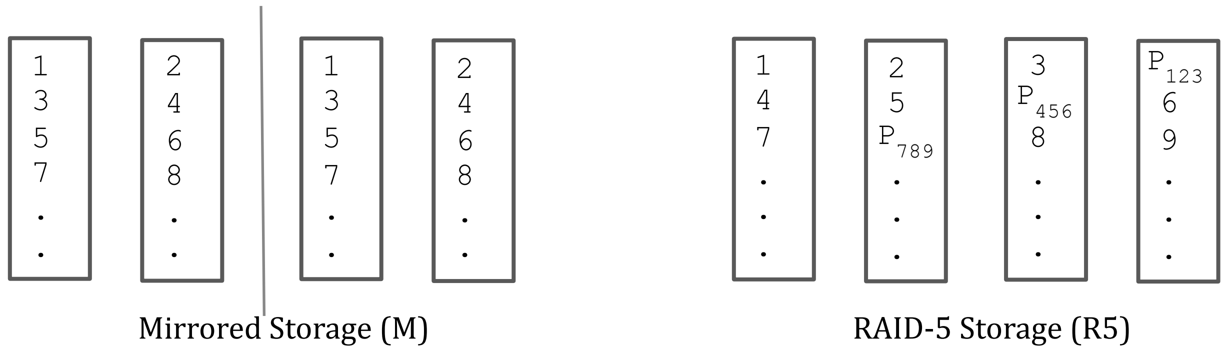
```
while (true) do {
    (v1, emp) = read-any(r1);
    (v2, email) = read-any(r2);
    if (v1 == v2):
       break
}
```

E. True / False Theres no way to get a version that is guaranteed to be consistent.

Answer: F, F, F, F, T. PNUTS doesnt maintain any consistency between separate records.

Initials:

III RAID



Consider the two storage organizations shown above. One of which uses mirroring (M) and one of which uses RAID 5 (R5). Here each rectangle represents a disk. The numbers represent blocks of data and show how the blocks are striped across different disks. P_{ijk} represents the parity block for data blocks i, j and k. It is assumed that disk controllers can detect an erroneous data block while reading it using checksums. If a disk fails then all the blocks on that disk become inaccessible.

8. [4 points]: Answer the following questions about read performance of these two configurations in the absence of failures.

A. What is the maximum number of distinct data blocks that can be read concurrently (where each read is processed at the same time on a different disk) by these two schemes?

(Write your answer in the spaces below.)

M:_____________ R5:_____________

Answer: M: 4, R5: 4.

B. How long is the longest sequence of consecutively numbered data blocks that can be read concurrently by these two schemes?

(Write your answer in the spaces below.)

M:_____________ R5:_____________

Answer: M: 4, R5: 4.
For both A and B, we can read 3,4,5,6; However, we can read 1,2,3,4 from M but not from R5.

Initials:

9. [2 points]: How many disk blocks are needed to store 6000 blocks of user data in these two systems? **(Write your answer in the spaces below.)**

M:______________ R5:______________

Answer: M: 12000. R5: 8000. For M one extra block per block is needed but for R5 one extra block is needed only for every three blocks.

10. [2 points]: In case a bad data block is detected while reading, how many disks have to be read to reconstruct the bad block in these two systems (excluding the read to find the bad block in the first place) ? **(Write your answer in the spaces below.)**

M:______________ R5:______________

Answer: M: 1. R5: 3. For R5, the three remaining blocks have to be read to reconstruct the data/parity.

IV Spanner

11. [4 points]: In Spanner, a client reads the value associated with a particular key by calling `read(key k, timestamp t)`. Which properties ensure that this read is consistent?

(Circle ALL that apply)

A. Each record has a master that controls the writes to that record (thus allowing the reads to be consistent).

B. Reads in Spanner are full Paxos reads.

C. The value will not be returned from the call to read until an operation has occurred after time t.

D. None. reads in Spanner are not guaranteed to be consistent

Answer: Only C.
a) is false; record masters are a feature of PNUTS not Spanner
b) is false; reads in Spanner are not full Paxos reads (which is part of how Spanner provides good performance on reads)
d) is false, since (c) is true

Recall that Spanner's TrueTime protocol supports an API where `time.now()` returns an interval `(earliest,latest)`, rather than a single value, where the actual time is between `earliest` and `latest`. For the following problems, assume that the length of the interval (`latest - earliest`) is *always* between 250 ms and 500 ms, inclusive.

12. [2 points]: In the best case, about how long will a transaction that does a single write take, from the time a lock is acquired to the time the write is committed?

(Circle the BEST answer)

A. $<$ 250 ms

B. 250 ms

C. 500 ms

D. $>$ 500 ms

A is correct. It should be clear that C and D are incorrect, so I'll just explain why B is incorrect.
The lock that is held when doing the write can be released as soon as tt.now().earliest is $>$ s.now().latest.
For concreteness, say that s.now().earliest = 0, and s.now().latest = 250. The "true" value of s, which is when the lock was acquired, is then between 0 and 250.
Let t = tt.now(). The write is finished when t.earliest $>$ 250. In this scenario, the "true" value of t will be between 250 and 500.
The crux of this problem comes from noticing that the true values of s and t need not be exactly in the middle of those ranges.

Initials:

The best case occurs when s and t are both near 250. Say s = 240, t = 260. The write only took 20ms to commit, which is < 250ms.

13. [2 points]: In the worst case, about how long will 100 transactions, each of which does a single write, take?

(Circle the BEST answer)

A. < 250 ms

B. < 500 ms, but more than 250 ms

C. < 25 sec, but more than 500 ms

D. >= 25 sec

D is correct. 100 writes with a 500ms max on the interval length means 100 * 500ms = 50,000 ms = 50sec in the worst case.

14. [2 points]: In the worst case, about how long will 100 transactions issued at the same time (concurrently) take, if each transaction does a single write?

(Circle the BEST answer)

A. < 250 ms

B. < 500 ms, but more than 250 ms

C. < 25 sec, but more than 500 ms

D. >= 25 sec

D is correct. Even when writes are done concurrently, commit-wait will force them to go sequentially.

V SSL/TLS

Refer to the following diagram when answering the following question about SSL/TLS.

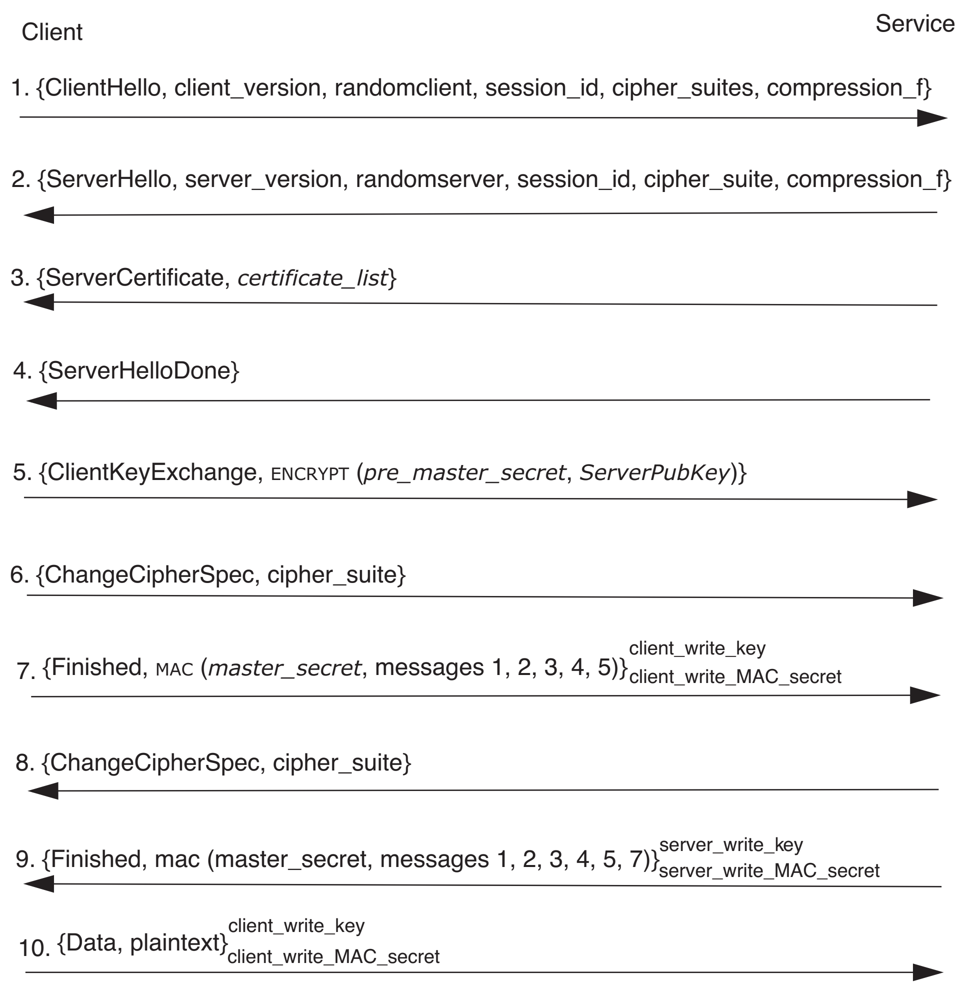


FIGURE 11.10

Typical TLS exchange of handshake protocol messages.

The previous page contains a reproduction of the typical TLS handshake, reproduced from Chapter 11 of the textbook.

15. [10 points]: Which of the following statements about this protocol are correct?
(Circle True or False for each choice.)

A. True / False The pre_master_secret prevents replay.
Answer: True or False. We accepted either answer because it could be interpreted differently.

B. True / False The client and server exchange the master_secret, in addition to the random numbers that each generates for randomclient and randomserver.
Answer: False. They do not exchange the master_key. Each generates it separately and uses it.

C. True / False The master_secret, in conjunction with randomclient and randomserver, are used to generate the keys that the client and server use for message authentication, message confidentiality, and as initialization vectors for block chaining.
Answer: True.

D. True / False Because message 8 is not included in the MAC computation in message 9, the order of their delivery does not matter.
Answer: False. Message 8 must precede message 9, because both ends have to agree that they have changed the cypher suite, for 9 to be useful.

E. True / False The protocol shown above authenticates both the client to the server and the server to the client.
Answer: False. This protocol only authenticates the server.

Initials:

VI Stack Smashing

Consider the following C program fragment:

```
void foo(void *arg, size_t len) {
  char buff[100];
  memcpy(buff, arg, len);
  /* ... real work happens here ...*/
  return;
}
```

Attackers have developed an exploit for this function, which they have posted on the Internet. The crucial lines for our purposes are in the following fragment:

```
void attack() {
  char magic[370];
  /* ... */
  build_magic_value(magic);
  /* ... */
  foo(magic, 370);
}
```

16. [4 points]: Using your knowledge of the general outline of how a buffer overrun attack typically works, choose True or False for each of the following statements.

(Circle True or False for each choice.)

A. True / False The first 100 characters of `magic` could be code, addresses, irrelevant padding, or any combination of these elements.
Answer: True. The whole buffer must be filled with something before the attacker can write beyond into the stack, but it doesn't matter what it is. In general we can't be sure whether any part of the filled buffer is code, addresses, or just some meaningless bytes.

B. True / False The last 270 characters of `magic` are carefully chosen to overwrite part of the call stack of routine `foo`.
Answer: True. The point of a buffer overflow attack is to replace what comes immediately after the buffer. In the case of a buffer that is a stack-allocated local variable, what is overwritten is the call stack. We don't necessarily know exactly what is in the computed value "magic" but we can be pretty sure that the part after filling the buffer is designed to replace the executing program's stack in some cunning way.

C. True / False The last 270 characters of `magic` are C source code.
Answer: False. The values written into the call stack will most likely be some combination of machine addresses and machine instructions. Overwriting the call stack with characters that represent C source code is very unlikely to do anything useful for an attacker.

Initials:

D. True / False At least one of the instructions included in `magic` must be a multiply.
Answer: False. While it is possible for "magic" to include a multiply instruction, there is nothing about a buffer overflow attack that intrinsically requires it.

In response to the posted attack, the programmer responsible for `foo` writes a new version where the relevant lines are as posted below:

```
void foo(void *arg, size_t len) {
  char * buff = malloc(100);
  if (buff == NULL) return;
  memcpy(buff, arg, len);
  /* ... */
  return;
}
```

The programmer tests the new `foo` with the posted version of `attack()` and discovers that although `attack()` causes a core dump, it no longer works to break into the system.

17. [5 points]: Again using your knowledge of the general outline of how a buffer overrun attack typically works, indicate which of the following statements are true.

(Circle True or False for each choice.)

A. True / False The `memcpy` call should have been replaced by a `printf` call.
Answer: False. We don't know what exactly the program is doing with buf in the body, but we can't simply cut out the memcpy call in favor of printing out something.

B. True / False The `foo` function is still vulnerable to a buffer overrun attack.
True. Moving the buffer from the stack to the heap just changes the details of how an attack can take advantage of overflowing the buffer... it doesn't eliminate the risk.

C. True / False Adding bounds-checking on use of `buff` would have been a better change than moving the buffer to the heap.
True. Checking the bounds on the usage of buff would have eliminated the risk of this buffer overflow, rather than just changing the flavor of buffer overflow problem.

D. True / False Simply changing the declared size of `buff` might have also stopped the specific attack code, but without eliminating the vulnerability.
True. The value "magic" is probably tailored to a very specific data layout and machine architecture, and even a single-byte change of alignment may well be enough to violate the assumptions it's depending on. But stopping that specific value of "magic" doesn't eliminate the vulnerability; it won't take an attacker long to devise a "magic2" that works for the new version.

E. True / False The `memcpy` call should have been split into two calls: the first call to copy the first 100 characters, and then a second call to copy any additional characters to a second buffer.
False. If we assume that the program was operating correctly except for the buffer overflow vulnerability, then it doesn't ever expect more than 100 characters of input. There is no point in copying excess characters to some second buffer, especially because that second buffer will still be vulnerable to a buffer overflow attack unless some kind of bounds checking is implemented. If we are implementing bounds checking, we might as well apply it to the first buffer (as in part C).

Initials:

VII GFS

18. [7 points]: Which of the following statements about files in the Google File System (GFS) are true? **(Circle True or False for each choice.)**

A. True / False File locking is used to ensure consistency of file content.

B. True / False It is designed to handle frequent hardware failures during operation.

C. True / False File naming/renaming is atomic.

D. True / False From the point of view of clients, it provides single-copy consistency over its replicas.

E. True / False Time-limited leases are used to reduce the need for network traffic to the master.

F. True / False To detect possible corruption of data, GFS compares contents of a chunk on multiple chunk-servers.

G. True / False GFS uses Paxos to allow the group of chunk-servers to elect a new master in case of master failure.

Answer: F, T, T, F, T, F, F.

End of Quiz II

Please double check that you wrote your name on the front of the quiz,
and circled your recitation section number.