אבטחת מידע: תאוריה בראי המציאות

מבחן מועד א׳ סמסטר א׳ תשע״ב
20 בפברואר 2012
מרצה: ערן טרומר

(בגופן מוטה: הבהרות שנכתבו על הלוח)

הוראות

משך הבחינה שלוש שעות. לא תינתן הארכה.
משקל כל שאלה מצוין בתחילתה. יש לענות על כל השאלות.
תשובה ללא נימוק לא תזכה בניקוד.
ניתן לענות בעברית או באנגלית.
מותר שימוש בחומר עזר מודפס בלבד.

בהצלחה!

1. (11) In the following TPM question, please (unrealistically) assume there are no covert channels or implementation bugs in the system.
   a. (5) Achilles is using the Amazon EC2 cloud service to run Windows 7 in a virtual machine. He is concerned about his VM's vulnerability to hackers from faraway Troy who may have remotely broken into Amazon's infrastructure. While installing Windows, Achilles noticed the option to enable BitLocker, Microsoft's implementation of disk encryption using TPM sealing. After enabling the option, Windows reported that BitLocker with TPM sealing has been successfully activated.
      Can Achilles rest on his laurels, knowing that his data can be accessed only as permitted by the Windows access control mechanism?
   b. (6) Achilles decided to use a physical machine instead. He bought a trustworthy computer, carefully configured it with BitLocker TPM functionality, and then shipped it to a high-bandwidth data center operated by hosting service Crackspace.
      Late at night, a Trojan spy infiltrated the Crackspace data center carrying a screwdriver and a piece of wire of just the right size to short out the TPM "reset" pin. What can the spy do?

2. (10) A user downloaded from the Internet a program that is supposed to add bunnies to photos. He doesn't trust the program, so he would like to try running it on one of his image files without letting the program access any other file on the computer.
   a. (5) Can the user do this in a system based on capabilities? Explain.
   b. (5) Can the user do this in a system based on access control lists? Explain.

3. (10) In Android, the Contacts database is a file on the file system. It was suggested to implement contacts access control in the following manner: every read from contacts is an IPC to/from a dedicated Android application which is the only one having access to this file (has the appropriate UID/GID). We also want to ensure that if the Android IPC mechanism is trustworthy, and assuming the only interface between applications is the one defined in the Manifest, no application without contact read permissions can access any contact data.
   a. (5) Describe the Manifest.xml of this application. Which component(s) does it define? What are the attributes of these components? What else?
   b. (5) It was suggested to add component(s) to this application, so that the same application will be used in the same manner to implement read access control to the SMS database. Would you follow this suggestion? If not, what would you have done instead (having only security concerns in mind)?

4. (15) Recall that RC4 has a "key scheduling" algorithm that is ran first, in order to initialize the 256-byte state table S according to a key `key`. This is done prior to to running the stream generation algorithm (seen in Exercise 1) on the table S. Here is C++ code for the RC4 key scheduling algorithm:

```
char* schedule(string key) {
  char* S = malloc(256);
  for (i=0; i<256; ++i)
    S[i] = i;
  int j = 0;
  for (i=0; i<256; ++i) {
    j = (j + S[i] + key[i % length(key)]) % 256;
    swap(&(S[i]), &(S[j]));
  }
  return S;
}
```

   a. (5) Explain how to completely break RC4 (recover S at the end `schedule`) given a powerful cache side-channel attack that learns all memory accesses, as in Exercise 1.
   b. (10) The main cache side-channel shown in class is less powerful, and learns (at most) which cache sets each access belongs to.
      Assume that the programmer knows the parameters of the CPU's set-associative cache and the runtime memory mapping of its program. Assume a flat memory mapping (no difference between physical and virtual addresses).
      Describe an approach to implementing RC4 in a way that is resistantant to this cache side channel. Then, write short but accurate code implementing of the RC4 key scheduling algorithm, following your approach.

5. (12) HTTPS X.509 certificates are generated as follows (in a simplified version). The requester (i.e., web site owner) generates a key pair (k_{pri},k_{pub}) and sends the request pair (k_{pub},*domain*) to the certificate authority. The certificate authority verifies that that the requester owns *domain*, and computes the signature *sig* on the MD5 digest of "k_{pub};*domain*" (where ";" is a unique delimiter). The certificate is (k_{pub},*domain*,*sig*).
   Suppose it was discovered that the MD5 hash function has the following cryptographic vulnerability. There is an efficient procedure MD5COLLIDE that, given a pair (*x*,*y*), outputs another value *z* such that MD5("*x*;*y*")=MD5("*x*;*z*"), and moreover *z* is a short random-looking sequence of ASCII letters *(a new z is generated on every run)*.

   a. (4) Explain how to get two certificate for the price of one (for two different domains, doesn't matter which).
   b. (8) Explain how you can impersonate the Windows Update service (https://update.microsoft.com).

6. (22) Alice and Bob became close friends during their work in the security industry. For Bob's birthday tomorrow, Alice would like to securely perform several tasks. For each of the following goals, do you think it achievable or infeasible? Explain, and state your assumptions. Answer in 4-7 lines per item (depending on your handwriting).
   a. (4) Alice wishes to give Bob a message he will be able to read only 24 hours later. She implements this as software on a standard laptop and hands it to Bob, powered off.
   b. (4) Likewise, but Alice hands Bob a USB device.
   c. (4) Likewise *as in b*, but the message should be read no earlier than Bob's next birthday, next year.
   d. (5) Alice has just recalled that she wishes to buy Bob a book for his birthday. She chose a specific book and would like to know whether Bob already has that book, but without ruining the surprise: Bob shouldn't learn anything about Alice's choice. Bob has a list of all of his books on his smartphone. Unfortunately, the network is too slow to send the whole list to Alice. Fortunately, Bob's smartphone has a powerful cryptographic coprocessor.
   e. (5) Same as above, except Bob can only communicate a single yes/no answer to Alice. Bob can, of course, learn whether Alice's choice was already in his list, but can she make sure he learn no more?

7. (20) SignyThin provides a free Internet service which receives requests of the form (*name*,*weight*) and returns strings of the form
"On *current_date* I was told that *name* weighs *weight* kg".
Each message also carries a signature under SignyThin's signing key. The server code contains this snippet:

```
void sendmsg(char* name, int weight) {
     int lessweight = weight-1;  // let's be generous
     fprintf(socketfd, "On ");
     fprintf(socketfd, current_date_as_string());
     fprintf(socketfd, " I was told that ");
     fprintf(socketfd, name);
     fprintf(socketfd, " weighs %d kg", lessweight);
}
```

(Indeed, it turns out that the programmer is good-hearted but has never heard of "%s").
The signing key is stored in some fixed memory address KEYADDR. The stack and code addresses are well-randomized using ASLR. Assume a 32-bit x86 machine with the stack structure shown in class. *SignyThin's server code is known to the attacker.*

a. (5) Explain how an attacker can steal SignyThin's signing key.

The function was fixed by forbidding bad characters in *name*. However, the filtering code has a bug and still allows the string "%n". (Recall that "%n" in the printf() format string expects the next unused parameter to be an int*, and means "store the number of characters written so far into the integer pointed to by the next parameter".)

b. (5) Explain how an attacker can write to any memory address.
c. (5) Explain how an attacker can now steal SignyThin's key. (The attacker can afford heavy computations, but not exhaustive search of a whole key.)
d. (5) Describe a faster attack for the case where the signing scheme is RSA.