Computer and Network Security – Exercise no. 1

Submit in Pairs/Single to mailbox 19 by 5/12/11, 2:00 p.m.

1. In this question, we shall see how an adversary who has access to the memory of a process can find secret information. We shall demonstrate this on the mine sweeper game.

   In mine sweeper, the player has to identify all the empty cells and clear them. Opening a cell which contains a mine leads to losing the game. In the game version we look at (the one that comes with Windows XP), a cell may be either open or close. If it is open, it shows how many of the neighboring cells contain a mine (or if it contains a mine — the player loses). If the cell is not open, then it may be marked by two symbols: A flag (indicating that the player thinks this cell contains a mine) or a question mark (the player is uncertain). Both symbols can appear on a closed cell (either with a mine or without), but not at the same time.

   On the course's website, you can find a few game boards (30 columns and 16 rows), each containing 99 mines, named Board1 to Board5. Additionally, you will find on the website the memory dump of the stack of the corresponding processes. The format of the memory files is

   ```
   Address <16 bytes in hexadecimal notation>
   ```

   For example, when the line is

   ```
   108870 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
   ```

   Then the byte value at address 108870 is 00, and in address 10887E is 0x0E.

   (a) Find the memory area where the game board is stored.
   Hint: Assume that the board game is kept in memory, where each cell has its own byte. Additionally, you can assume that the memory of two different games is mostly the same (up to the board).

   (b) Find what is the representation (hexadecimal value) of
   - An open cell with a number,
   - A closed cell which does not contain a mine,
   - A closed cell which contains a mine,
   - A cell marked with a flag that contains a mine,
   - A cell marked with a flag that does not contain a mine,
   - A cell marked with a question mark that contains a mine,
   - A cell marked with a question mark that does not contain a mine,

   Explain in your question what were your actions.

(c) If you ever played Windows' minesweeper, you are probably used to an astonishing fact: Your first click never lands on a mine. On the course's website you will find two boards (SpecialCase1 and SpecialCase2) and their corresponding memory dumps. Explain what happened after clicking on the board SpecialCase1 in the 5th cell from the left in the top row (obtaining the board SpecialCase2).

2. This question deals with a variant of AES.
Recall that in AES (Rijndael), a round of encryption consists of the following four operations:

- SubBytes
- ShiftRows
- MixColumns
- AddRoundKey

For each of the following changes to AES, determine whether they are weaker than AES, or as secure as AES. In the case of weaker variants, describe an attack that breaks the new cipher. For similar security, explain why it is as secure as the original AES.

(a) If all the MixColumns operations are omitted from the cipher.

(b) If all ShiftRows operations are omitted from the cipher.

(c) If all operations of the same time are put together, i.e., the encryption is changed to:

- 10 SubBytes, followed by
- 10 ShiftRows, followed by
- 9 MixColumns, followed by
- 11 AddRoundKey

(d) the ShiftRows operation is changed such that the rotation is to the right (instead of to the left).

When describing an attack, give the expected time, memory, and data complexities.

3. (a) Sun-Tzu wishes to send a message M to Machiavelli. M is to be transmitted in a plaintext form (i.e., not encrypted), but authenticated using AES-CBC-MAC (i.e., CBC-MAC instantiated with AES). They met once to exchange a key k securely (i.e., the key is not available to the adversary). As Sun-Tzu did not wish to set the IV in advance, they decided to send along the message the IV used for the authentication, i.e., the message is going to be $IV||M||\text{AES-CBC-MAC}_k(M)$.

Can Genghis Khan alter the sent message such that the authentication process will work, despite the change in the message?

(b) Explain why CTR-MAC, defined like CBC-MAC (but based on the counter mode of operation), is a bad idea. Compare the security of the proposal you have discussed with the one of CBC-MAC.

(c) Napoleon has decided to invent the mode of operation "FrenchCBC". To encrypt a message in FrenchCBC, the plaintext is **decrypted** using CBC, and to decrypt a message in FrenchCBC, the ciphertext is **encrypted** using CBC. First check that the new mode,

FrenchCBC, indeed works (i.e., encryption and decryption are inverse operations). Then, define FrenchCBC-MAC, which returns the last block of encryption under FrenchCBC. Explain why this MAC (FrenchCBC-MAC) is not to be used with messages of more than two blocks.

(d) Sun-Tzu wishes to send to Machiavelli an **encrypted** message M with AES-CBC-MAC authentication. To save the amount of keying material and IVs, they decided to send the encrypted message using AES-CBC with the same key and IV. Hence, the message format is: AES-CBC$_k(M)||$AES-CBC-MAC$_k(M)$.

i. Show how Machiavelli decrypts the message and authenticates it.

ii. Assuming that Machiavelli's authentication check succeeds, does that mean that the message was not altered. Specifically, can Khan alter the sent message such that the authentication process will work, despite the change in the message?

4. Recall the definition of RSA signatures:

- **Initialization:** The user picks two large prime numbers p and q, and computes $n = pq$, and finds e, d such that $e \cdot d \equiv 1 \bmod (p-1)(q-1)$. The public key is (n, e) and the secret key is d.
- **Signature:** The user takes m, and computes $sig = m^d \bmod n$.
- **Verification:** The recipient takes m, sig, and checks whether $sig^e \stackrel{?}{\equiv} m \bmod n$.

(a) Show that the signatures on the messages $0, 1, -1$ are fixed independent of the public key.

(b) Show how given two valid (message,signature) pairs, (m_1, sig_1) and (m_2, sig_2), it is possible to create a third valid (message,signature) pair with $m_3 = m_1 \cdot m_2 \bmod n$.

(c) A student claimed that he can generate valid (message,signature) pairs, given just the public key. Show how the student can do so. (Hint: first pick the signature, and then find the corresponding message).

(d) Explain why your answers to the above questions change if the signature is computed as $sig = [h(m)]^d \bmod n$. Also, show how to verify this sort of signature.