Midterm Exam

Computer Security

Due 10/24/17 at 6:10pm (bring to class)

Name:

Undergraduate/Graduate:

Question 1 (5 points) (1/2 each)

T/F (Clearly circle only one choice)

1) T F Data integrity countermeasures are reactive mechanisms that protect against active attacks

2) T F Data confidentiality countermeasures are proactive mechanisms that protect against passive attacks

3) T F 3-DES is three times faster than DES

4) T F Password is an example of an authentication mechanisms that is based on "what an entity has".

5) T F Man in the middle is not possible on the following transmission from Alice to Bob: Alice ---------- m||SHA-256(m) ----------> Bob

6) T F The size of the output of SHA-512 depends on the size of the input.

7) T F The factorization problem addresses how to break a given number into the product of two prime numbers

8) T F The first halves of the hash values generated by SHA-256 and SHA- 512 are the same

9) T F End-to-end encryption encrypts data on a per-link basis

10) T F Finger prints falls under the category of dynamic biometrics

Question 2 (5 points)

Multiple Choice Questions (1 points each)
(Clearly select only one choice)

1. Consider a Diffie-Hellman system that uses a prime q=11 and a primitive root α=2, If Alice has a private key XA = 5, what is Alice's public key XA? private key XB = 4, what is the shared key?
   a. 7 and 4
   b. 8 and 3
   c. 9 and 2
   d. 10 and 1

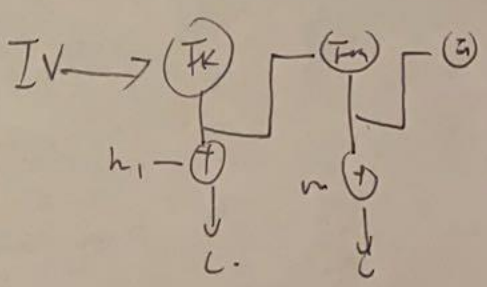
   Show your work:

2. How many blocks will be generated out of a 320 bits plaintext message when encrypted using DES? What is the size of the final ciphertext message?
   a. 5 and 256
   b. 4 and 320
   c. 5 and 320
   d. 4 and 256

3. Determine if each of the 3 properties of hash functions below is true or false:
   a. if x != x' then H(x) != H(x').
   b. It should be easy to generate H(x) an give message x
   c. It should be easy to generate a message x from a given hash H(x)

4. AES in CTR (counter) mode with a counter value that does not change per block is the same as AES in _____ mode.
   a. ECB mode
   b. CFB mode
   c. CBC mode

5. A message consisting of 15 blocks is encrypted using DES in CBC mode. The message is then transmitted from Alice to Bob. If a bit error occurs in the 2nd and 4th blocks of the message during transmission, the following blocks will not be correctly deciphered:
   a. 2nd and 3rd
   b. 3rd and 4th
   c. 4th and 5th
   d. 2nd through the 4th
   e. 2nd through the 5th
   f. rd through the 5th
   g. All 15 blocks



Question 3 (10points)

Short Answer Questions (10 points)

1) (2.5pts) Alice wants to send Bob an encrypted file. She communicated with Bob that she will use DES in CBC mode. She verbally (but secretly) shared the key and the IV with Bob (no one intercepted). The file had 10 blocks, all of which were encrypted successfully by Alice (using DES in CBC as mentioned), sent to Bob, received without any errors, and decrypted successfully by Bob. All the blocks were decrypted correctly except for the first block. Where is the problem?

Answer:

The problem is with the initialization vector (IV). If a file is decrypted with **incorrect initialization vector**, then this will corrupt the first block but all other blocks will be intact. So, the initialization vector shared by Alice to Bob may be incorrect.

The block of ciphertext depends on all plaintext blocks that has already been processed up to that point. To make sure that each message is unique, we need to use an **initialization vector (IV)** in the first block.

The encryption in CBC mode is sequential. So, the message needs to be filled with the multiple of the block size of the cipher.

If a file is being decrypted with an incorrect initialization vector, it may cause the initial first block to be corrupt. But all other blocks of plaintext will be correct. The reason is that each block of the ciphertext is XORed with the last blocks of the ciphertext rather than the plaintext.

This means that recovery of a plaintext block can be done by two blocks of ciphertext that are adjacent. So, the decryption will be parallelized. A single-bit change in the ciphertext may corrupt that block of plaintext. These corresponding bits can be inverted in the following block of plaintext, but other subsequent blocks remain intact.

2) (2.5pts) Using Caesar cipher we encrypted the string "helloworld" iteratively 26 times. Each encryption used the output of the previous encryption as its input (like DES rounds). The keys used (in the exact order) are as follows [0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25]. This means that the first encryption used key=0, its output was used as input to the second encryption which used key=1, etc. What is the final ciphertext after all 26 encryptions? Briefly justify your answer.

Answer:

I would like to answer this as follows ...

Let's say the plain text is "ABC" as of now.

And on that, we will apply the keys in the exact order as given.

Caesar cipher is used here. So, in plain words, if a plain text character is 'A', and the key is 3 i.e. greater than 0, then 'A' is shifted right by 3 places i.e. 'A' + 3 = 'D' in cipher text.

(In other way, for A-Z, the values are 0-25. So, A = 0 which means A+3 = 3 which is the numeric value for D.)

Here, the first key is 0 which means it won't shift the plain text by any places.

After that, key = 1 is used, say the plain-text is "ABC" then cipher text for key = 1 will become "BCD" i.e. adding 1 to every letter.

Now, this encrypted text "BCD" will become input for key = 2 and so the cipher text after key = 2 will be "DEF"

Till here we used keys 0,1 and 2.

Adding them all we get 3.
What will happen if we directly add this 3 to original plain text "ABC" ? We'll get the same result "DEF" right.

So, we can say that if we add up all the keys and shift the plain text by that number of places, it will give same final cipher text.

One thing to remember is, this is not applicable for every algorithm.

Moreover, as it is Caesar cipher, if the key is greater than or equal to 26, then we will divide it with 26 and the remainder will become new key because we have 0 to 25 elements only. If we add 25 to 'A' it becomes 'Z'. So, if we add 26, then it will be 25 + 1 = 'Z' + 1 = 'A' again.

So, if the key is greater than or equal to 26 i.e. greater than 25, then the values will be repeated that's why we do
key = key % 26 this will ensure key value will never be more than 25.

This will help when we add all the keys.

So, as I highlighted earlier, we can add up all the keys and shift the plain text by that number of places, it will give the final cipher text.

So, what is the total of 0 to 25 ? It is 325.

So, key = 325 which is greater than 26.

So, key = 325 % 26 = 13

So, we got a key = 13 here.

Now, plain text is "helloworld"
Shifting every letter by 13 places, we get **uryybjbeyq**

Because, 'H' = 7 so 7 + 13 = 20 = 'U' and so on.

So, this how, at the end the cipher text will be uryybjbeyq.

And we can say that the resultant cipher text is similar like the one generated by using 13 as a key.

This is how I will try to find and justify the solution for the same.

Go through the concept and the calculation. Why % 26 and such.
After going through this, if you've any doubt, then surely comment about the same.
I will address it and try to resolve it.

3) (2.5) The password "to12G=!_" is to be stored in a password file using the salting string "Gr()1" and SHA512. Show what exactly gets stored in the password file.

4) (2.5) If the size of a message is 5120 bits, how many blocks will you end up with if using SHA-512 to hash it? What will be the HEX of the last block?

Answer:
There will be 6 blocks each being 1024 bits wide.
The contents of the last block (in bits) would be:

1_<895 zeros>_<length of message being 128-bits wide>
1_000..0_000..0001010000000000

The SHA-512 algorithm works on message blocks which are 1024 bits wide.
Therefore, for a message length of 5120 bits, we would have (5120/1024)=5 message blocks plus the last block.

The last block contains the bit 1 followed by the (896 - 1 - L) zeros, where L is the length of the message which overspills from the previous 1024 block and then the last part of the message contains the actual length of the message represented in 128-bits.
For our case:

Actual length: 5120 bits
The length perfectly fits into 5 1024-bit wide blocks, hence for us L = 0
As a result, the number of zeros we get is 895 zeros
Then 5120 can be represented by the following binary: 0001010000000000

Midterm Exam

Computer Security

Name: Yingda Fan

2.5/5

Question 1 (5 points) (0.5 point each)

T/F (Clearly circle only one choice)

T ✓ 1) (T) F Ensuring messages are received in order is a key objective of the Integrity component of the CIA triad

F ✓ 2) T (F) If MD5("x") = 0x9dd4e461268c8034f5c8564e155c67a6 and MD5("y") = 0x415290769594460e2e485922904f345d the MD5("xy") = 0x9dd4e461268c8034f5c8564e155c67a6415290769594460e2e485922904f345d

F ✗ 3) T (F) The S-Box in AES changes as a function of the symmetric key used — ✓ change

F ✗ 4) T (F) If two messages X and Y exist where H(X) = H(Y), then the hash function H is not secure.

F ✓ 5) T (F) A digital signature is the encryption of the hash of a message using a Diffie-Hellman private key.

F ✗ 6) T (F) A digital envelop distributes RSA public keys by encrypting them using symmetric keys

F ✓ 7) T (F) Similar to CBC, decryption using the Cipher Feedback (CFB) Mode can be done in parallel but encryption cannot.

F ✗ 8) T (F) Passwords is an example of "something that you have" means of authentication — know

T ✓ 9) (T) F The owner of a file that has the permission 532 won't be able to write to it but any other user can

T ✗ 10) (T) F 3DES in EEE mode cannot be backward compatible with the single DES — EDE 才 backward

Question 2 (10 points)

Multiple Choice/Fill in the blanks Questions (1 points each)
(Clearly circle your choice(s))

X 1) Risk is the intersection of three concepts: loss or damage to an information system as a result of unauthorization exploiting a modification.

✓ 2) If a machine is capable of doing 10^4 decryptions/sec, then the worst-case scenario for breaking a key that has 10^{12} possibilities is 108 seconds and the best case scenario is 10^{-4} seconds.

3) Assume a system that mandates users to select exactly a 10-character password. Passwords are drawn from a list of 45 printable characters. An attacker can test 5.5 million passwords per second. If we want to set the guessing probability to 0.001, what would be the lifetime of a password in years (round to 1 decimal digit)?

a. 5.1
(b.) 0.1
X c. 1.1
d. 2.1
e. Other _____

X 4) How many blocks will be generated out of a 512 bits plaintext message when encrypted using AES? What is the size of the final ciphertext message in bits?

(a.) 5 and 128
b. 3 and 128
c. 5 and 512
d. 3 and 512
e. Other 4 and 512

✓ 5) AES in CTR (counter) mode with a counter value that does not change per block is the same as AES in ECB mode.

6) In S/Key, when key K_i (where $1 < i < n$) is compromised, the following keys will also be compromised:

X (a.) K_i through K_n
(b.) K_1 through K_i
c. K_i and K_{i+1}
d. K_i and K_{i-1}
e. Other _____

7) When using Bloom Filters to assess the "goodness" of a password, which of the following are possible (FP: good password rejected, FN: bad password accepted, TP: bad password rejected, TN: good password accepted). Choose only one answer:

a. FP
b. FN
c. TP
d. TN
e. FP and FN
f. FP and TN
g. FP and TP
h. FN and TN
i. FN and TP
j. TN and TP
k. FP and FN and TN
l. FP and FN and TP
m. FP and TN and TP
n. FN and TN and TP
o. FP and FN and TN and TP

8) Using SHA512 and an 8-bit Salt, an attacker would have to pre-compute _____ hashes for the password "password" to be able to recognize its hash in a stolen password file:

a. 2^{512}
b. 512
c. 256
d. 128
e. 64
f. 32
g. 16
h. 8

9) I belong to group Awesome. To enable everyone in my group to list the content of a directory and add subdirectories to it, but only allow the content of the directory to be listed for other non-Awesome users, I should set the permissions to:

a. 664
b. 564
c. 464
d. 364
e. 264
f. 164
g. 064
h. All of the above
i. Other 764

10) For a message whose size is 897, SHA512 will add 1023 padding bits. The bits will be in the following format:

a. 00000...
b. 11111...
c. 10101...
d. 01111...
e. 10000...

4/5

Question 3 (5points)

1) (2pts) Assuming keys range from 0-25 (i.e. key a = 0, b = 1, ... z = 25), the encryption of:

arsenalgoalkeeperwojciechszczesnyisavailableagainfollowingaribinjurytheowalcottwi llbeassessedaftermissingenglandsmidweekdrawwithukrainebecauseofaviruswh ileaboudiabyisadoubtwithahipproblemsouthamptonpairgastonramirezandmay ayoshidacouldmaketheirdebutsdespitehavingonlytrainedwiththeirnewteamma tessincethursday

using Vigenere key "eas", followed by an encryption of the result using Caesar key "y", followed by the encryption using Vigenere key "yck" is:

The same as plaintext.

Because the position of each letter just transport two round after three encryption

2) (1pts) Why is 2^{64} the maximum message size that can be hashed by SHA-256?

The result of SHA-256 is a 256-bits hash value, which could seem as a 32-byte array, which is normally represented by a length of 64-bits array in hexadecimal.

I can't make sense of this answer

X