Computer Security (COM-301)

Applied cryptography II

ECB properties

To encrypt a series of plaintext blocks p1, p2, . . . pn using a block cipher E operating in electronic code book (ECB) mode, each ciphertext block c1, c2, . . . cn is computed as ci = Ek(pi).

Which of the following is **not** a property of this block cipher mode?

a) Any repeated plaintext blocks will result in identical corresponding ciphertext blocks.
b) Decryption can be fully parallelized.
c) If a ciphertext block is modified or corrupted, then after decryption the corresponding plaintext block and all the following plaintext blocks will be affected.
d) None of the above; that is, (a), (b), and (c) are all properties of the ECB block cipher mode.

The correct answer is (c). In ECB, altering a ciphertext block only affects a single plaintext block, see diagram after with the red marker.
(a) and (b) are in the slides

ECB properties

To encrypt a series of plaintext blocks $p_1, p_2, \ldots p_n$ using a block cipher E operating in electronic code book (ECB) mode, each ciphertext block $c_1, c_2, \ldots c_n$ is computed as $c_i = E_k(p_i)$.

Which of the following is **not** a property of this block cipher mode?

a) Any repeated plaintext blocks will result in identical corresponding ciphertext blocks.
b) Decryption can be fully parallelized.
c) If a ciphertext block is modified or corrupted, then after decryption the corresponding plaintext block and all the following plaintext blocks will be affected.
d) None of the above; that is, (a), (b), and (c) are all properties of the ECB block cipher mode.

The correct answer is (c). In ECB, altering a ciphertext block only affects a single plaintext block, see diagram after with the red marker.
(a) and (b) are in the slides

Lausanne-Bern Direct line

The police forces in Lausanne and Bern want to build a new messaging system that allows them to exchange reports about crimes in real time so that suspects can no longer escape to the other city to avoid law enforcement. To achieve its goal, the system needs to relay messages without long delays. The police thus decides to build their messaging system based on a symmetric stream cipher: Every morning, the main precinct in Lausanne sends a policeman in disguise to Bern with a fresh secret key. This key is used during the whole day for all messages sent between the two cities.
Messages have the following format and headers:

```
Date: <date of crime>
Crime: <type of crime>
Suspect name: <name>
Case description: <free text describing what happened>
```

Théo the Thief, that often operates in Lausanne, reads about the new messaging system in the newspaper Le Temps and thinks "Oh no! Now I cannot use my hideout in Bern because the Bern police will have all reports"

Do you agree or disagree with Théo's statement?
If you agree provide a security argument why the new messaging system is secure.
If you disagree, describe a vulnerability in the messaging system and suggest an alternative that would address the problems.

1) depending on the communication channel, messages to Bern police can be dropped on the go
2) if they cannot just delete the messages (e.g. ACK system) then tamper with messages since no integrity -> bad
3) key exchange can be tampered with (strong assumption but if justified, ok)

Conclusion: this system is not secure and hideout is safe.

PGP

The following picture explains how PGP (Pretty Good Privacy) is used to encrypt emails.

a) Why does this scheme provide confidentiality?

(a) This scheme provides confidentiality with respect to third parties that can see the "Encrypted Message" on the wire.

The confidentiality is provided by the encryption with the Random Key. Without knowledge of the Random Key, no-one can decrypt the message. At that point in time, we have confidentiality, but just this encryption is not enough as then the message cannot be read.
To enable the receiver to read the message, PGP uses key transport. The sender encrypts the Random Key with the RSA Public Key of the receiver.

Both encrypted message and encrypted key from the receiver.

PGP (class proposals)

b) If you also need to provide integrity of the message, what would you need to add? If you think an option is bad, explain why.

- option 1: Add a hash of the message

  `Enc(k, data||H(data)), Enc(PKrec, k)`

- option 2: Add a signature of the data and the key

  `Enc(k, data), Enc(PKrec,k),`

  `Sig(SKsen,data||k)`

(b) We know from the lectures that encryption by itself does not provide integrity.

In the class, it was proposed to use a hash function to prove integrity so option 1 could seem like the right solution. This is not sufficient, as it does not prevent an adversary from modifying the message. Note that the key is sent encrypted with the public key of the receiver. This means that **anyone**, including an adversary, can produce the part of the message `Enc(PKrec, k)`. Also, hashes are keyless functions. This means that **anyone** with access to a message m can produce the hash of the message H(m)
This means that the adversary can intercept the original message `Enc(k, data||H(data)), Enc(PKrec, k)` throw it away, and compute a new combination for a message data': `Enc(k' , data' || H(data' ) ), Enc(Pkrec, k')`
Given this combination, the receiver has no means to know whether this comes from the original sender, or from an adversary that has changed the message.

The way to ensure integrity, is to add a signature (opt 2), which ensures that the adversary **cannot** create one valid signature for the message as they do not have access to the signing secret key of the sender SKsen.

We can assume safe delivery of the public keys, for ex though a PKI.

Destination Fakeland

A group of security researchers traveling to Fakeland learn that, upon arrival at the airport, Fakeland's border authorities will require their laptops for inspection. Fakeland authorities are famous for installing spying software during the inspection, so the researchers decide to take a snapshot of the laptops' state to make sure that they can detect changes.
For this purpose they plan to hash the content of the laptops' hard drive and write this hash on a paper. This way when they receive their laptops back, they can compute the hash of the content again and compare it to the value in their notes.

What property or properties must the hash function have in order to prove that no new software was installed (by comparing the hash on the piece of paper with the hash computed after crossing the border)? (Justify your answer)

Second pre-image resistance. In order to escape the detection mechanism of the researchers but install the spyware, Fakeland has to tweak such that the hash is the same. To make that hard, second pre-image resistance is needed.

Geletram

Alice uses the Geletram application to send messages to Bob. Alice and Bob share a secret symmetric key K. This key K is also known by Geletram.
For each message msg Alice wants to send to Bob through Geletram, Geletram does the following:

It generates a fresh symmetric key KGeletram, it sends
packet = {c = Encrypt(K, msg), m = MAC(KGeletram, c), KGeletram}
to Bob's Geletram to be decoded, where Encrypt is a symmetric encryption scheme, and MAC stands for Message Authentication Code.

Eve is an adversary that controls the channel in between Alice's Geletram and Bob's, i.e., Eve can read and modify any packet before it reaches Bob's Geletram.

Does Geletram provide confidentiality and integrity of the message msg with respect to Eve? If yes, justify; if not, propose a fix.

No integrity: Eve can replace Ci by block_i xor Δ(if block mode) and recompute the MAC.
Fix: encrypt-then-mac using K. Asymmetric crypto.