University of Oslo
INF3510 Information Security
Spring 2011
Workshop Questions & Answers

Lecture 5: Cryptography

Question 1

In which situations can cryptography be used to protect information? Justify your answer.

Answer

Cryptography can be used to protect information when information is transmitted through unprotected networks or stored in places where there is no physical or logical access control.

Question 2

Alice wants to send a message to Bob, without Carol (or other eavesdroppers) observing it. Alice and Bob have agreed to use a symmetric cipher. Key exchange has already occurred, and so they share a key K. Outline the steps that Alice and Bob must follow when they encrypt and decrypt, respectively.

Answer

Encryption:
i) Alice prepares the message *M*.
ii) Alice encrypts the message using the symmetric cipher algorithm in encryption mode E and the key *K*, to produce the ciphertext *C*, where *C* = E(*M,K*).
iii) Alice transmits the message to Bob.

Decryption:
i) Bob receives the ciphertext *C*.
ii) Bob decrypts the ciphertext using the symmetric cipher algorithm in decryption mode D and the key *K*, to recover *M*, where *M* = D(*C,K*).
iii) Alice reads/interprets the message.

Question 3

Alice wants to send a message *M* with a digital signature *Sig*(*M*) to Bob. Alice and Bob have an authentic copy of each other's public keys. Outline the steps that Alice must follow when signing *M*, and the steps that recipient Bob must follow for validating the signature *Sig*(*M*).

Answer

Digital signature generation: by Alice

i. Alice prepares message *M*.
ii. Alice applies a secure hash algorithm H to produce hash value H(*M*).
iii. Alice uses her private key $K_{priv}(A)$ with the asymmetric algorithm in signature mode S (equivalent to encryption mode E in case of the RSA algorithm) to produce signature *Sig*(*M*) = S(H(*M*), $K_{priv}(A)$).
iv. Alice transmits message *M* and signature *Sig*(*M*) to Bob, together with her unique name and specification of the hash algorithm and the asymmetric algorithm she used.

Digital signature validation by Bob:

i. Bob receives message *M'* (denoted as *M'*, not *M*, because its origin is uncertain) and the signature *Sig*(*M*).
ii. Bob applies the secure hash algorithm H on *M'* to produce hash value H(*M'*).
iii. Bob recovers hash value H(*M*) from received signature *Sig*(*M*) by using Alice's public key $K_{pub}(A)$ with asymmetric algorithm in recovery mode R (equivalent to decryption mode D in case of the RSA algorithm) to produce H(*M*) = R(*Sig*(*M*), $K_{pub}(A)$).
iv. Bob checks that H(*M*) = H(*M'*). If TRUE, then the signature *Sig*(*M*) is valid so Bob can be convinced that Alice sent message *M'* i.e. that *M'* = *M*. If FALSE, then the signature *Sig*(*M*) is invalid, so Bob doesn't know who sent message *M'*. He might then decide to reject the message, or use it knowing that its origin is uncertain.

The reason why Bob − and anybody else who knows Alice's public key $K_{pub}(A)$ − can be convinced that Alice sent message *M* is that only she could have produced the corresponding digital signature *Sig*(*M*) because only she has the private key $K_{priv}(A)$. The semantic interpretation of the digital signature is still not totally clear. It could e.g. mean

a) Alice sent the message but doesn't necessarily agree with the content of the message
b) Alice agrees with the content of the message.

In the paper-and-pen world, interpretation (b) is normally assumed. In the digital communications world, interpretation (a) is normally assumed.

Question 4

Suppose that a binary additive stream cipher (such as the one time pad) has been used to encrypt an electronic funds transfer. Assuming that no other cryptographic processing is used, show that an attacker can change the amount of the funds transfer without knowing anything about the key used. (You may assume that the attacker knows the format of the plaintext message used for the funds transfer.)

Answer

The part of the message in which the amount is recorded remains in the same position in the ciphertext as in the plaintext. Therefore the attacker can change the bits in that position which will alter the amount transferred. In general there is no way that the attacker can know whether the alteration will increase or decrease the value of the amount. In practice the attacker may know that the amount is likely to be small and therefore only change the digits in the high value positions. This illustrates that a binary stream cipher provides no message integrity even though it can provide unconditional confidentiality if the one time pad is used.

Question 5

Hash functions are commonly used for checking message integrity.

a. List four basic properties of hash functions
b. Use the internet to locate an SHA-1 demonstration tool — there's an interactive one written by Eugene Styer that can be used at http://people.eku.edu/styere/Encrypt/JS-SHA1.html. Investigate the four properties by examining the SHA-1 hashes for the following messages:
   (i) Take $100 from my account
   (ii) Take $1000 from my account
   (iii) Take $100 from your account
   (iv) Investigate other hashes for both longer and shorter messages

Answer

a. The four basic hash function properties are:
   H1: Fixed length output for arbitrary length input
   H2: One-way - given *M* it is easy to compute ***H***(*M*), but given ***H***(*M*) it is hard to find *M*.
   H3: Collision resistant - hard to find *M* and *M'* so that ***H***(*M*) = ***H***(*M'*)
   H4: A small change in *M* produces a major change in ***H***(*M*).
b. You should find that even though the input messages are very close, the output hash values are completely different. Note that with for any message length, both long and short, the output length is 160 bits, or 40 hexadecimal digits.

Question 6

Alice wants to send a message to Bob. Alice wants Bob to be able to ensure that the message did not change in transit. Briefly outline the cryptographic steps that Alice and Bob must follow to ensure the integrity of the message by creating and verifying a *MAC*.

Answer

• Alice
i) Generates message *M*
ii) Generates *MAC* = H(*M,K*) with *M* and *K* as input parameters
iii) Sends {*M, MAC*} to Bob
• Bob
i) Receives {*M', MAC*} (message denoted as M' because it's integrity is uncertain)
ii) Generates *MAC'* from *M'*.
iii) Compares *MAC'* and *MAC*
iv) If *MAC* = *MAC'* then Bob knows message was unchanged in transit

Question 7

a. What is a MAC (message authentication code)?
b. What is the difference between a MAC and a hash function?
c. In what ways can a hash value be secured so as to provide message authentication?
d. Imagine that a specific hash function is used for HMAC, and that vulnerabilities have been found in the hash functions so that the HMAC is insecure. Which changes in the HMAC are required in order to make it secure again?

Answer

a. An authenticator that is a cryptographic function of both the data to be authenticated and a secret key.
b. A hash function, by itself, does not provide message authentication. A secret key must be used in some fashion with the hash function to produce authentication. A MAC, by definition, uses a secret key to calculated a code used for authentication.
c. Hash codes can be used in various ways: HMAC, CBC-MAC and CMAC are examples
d. To replace a given hash function in an HMAC implementation, all that is required is to remove the existing hash function module and drop in the new module.

Question 8

a. Explain why message authentication alone is insufficient as proof of message origin in general, and to settle disputes about whether messages have been sent.
b. What security service is provided by digital signatures, and explain how this service relates to message authentication.
c. In what order should the signature function and the encryption function be applied? Also explain why that order makes sense.
d. How can a sender give a plausible reason for repudiating a signed message?

Answer

a. Suppose that John sends an authenticated message to Mary. The following disputes that could arise: 1. Mary may forge a different message and claim that it came from John. Mary would simply have to create a message and append an authentication code using the key that John and Mary share. 2. John can deny sending the message. Because it is possible for Mary to forge a message, there is no way to prove that John did in fact send the message.
b. Digital signatures provide non-repudiation, which is stronger than authentication because it provides proof to third parties of message origin.
c. It is important to perform the signature function first and then an outer confidentiality function. In case of dispute, some third party must view the message and its signature. If the signature is calculated on an encrypted message, then the third party also needs access to the decryption key to read the original message. However, if the signature is the inner operation, then the recipient can store the plaintext message and its signature for later use in dispute resolution.
d. The sender can claim that the private signature key was stolen.

Question 9

The Diffie-Hellman key agreement algorithm achieves key agreement by allowing two hosts to create a shared secret.

a. Clearly explain the operation of the Diffie–Hellman key exchange protocol.
b. Clearly explain why the basic Diffie–Hellman protocol does not provide any assurance regarding which other party the protocol is run with.

Answer

a. They share a base *g* and a modulo *m*. Let *a* and *b* be the secret keys of A and B respectively. Then:
   - A → B : g^a (mod m)
   - B → B : g^b (mod m)
   - *A* computes $(g^b)^a = g^{ab}$ (mod m)
   - *B* computes $(g^a)^b = g^{ab}$ (mod m)

   A,B now share the symmetric key g^{ab}

b. The basic Diffie-Hellman protocol includes no authentication of the messages exchanged. Therefore Alice has no assurance that she is running the protocol with Bob and Bob has no assurance that he is running the protocol with Alice.