Name

CWID

Exam

Thursday, Nov 15, 2018

CS458 - Fall 2018 - Exam 2-A
Solutions

Please leave this empty!

1.1 1.2 1.3 1.4 1.5 1.6 Sum

Instructions

- **You have to hand in the assignment to the instructor after the exam**.
- **This is an individual and not a group assignment. Fraud will result in `0` points**.
- Things that you are **not** allowed to use:
  - Textbook
  - Printed lecture notes
  - Phone
- A single sheet of notes will be allowed. You can write on both sides of the note page.
- Calculators are allowed.
- The exam is **90** minutes long

Question 1.1 True or False (6 Points)

1. (**1 pts.**) [True/False] If Alice has a message to send to Bob and she wants to encrypt the message using asymmetric cryptography so that no one other than Bob can read it, she does so by using Bob's public key.
2. (**1 pts.**) [True/False] Properly used, a MAC provides both confidentiality and integrity.
3. (**1 pts.**) [True/False] The security of the Diffie-Hellman key exchange protocol depends on the difficulty of solving the Diffie-Hellman problem, and therefore indirectly depends on the difficulty of solving discrete logarithms
4. (**1 pts.**) [True/False] A key transport protocol securely transfers a secret key to other parties.
5. (**1 pts.**) [True/False] Asymmetric key establishment protocols do not scale well to networks with large numbers of users and they provide typically no perfect forward secrecy
6. (**1 pts.**) [True/False] One main drawback of the Kerberos system is that the Key Distribution Center is a single-point of failure.

Question 1.2 Fill in the blanks (9 Points)

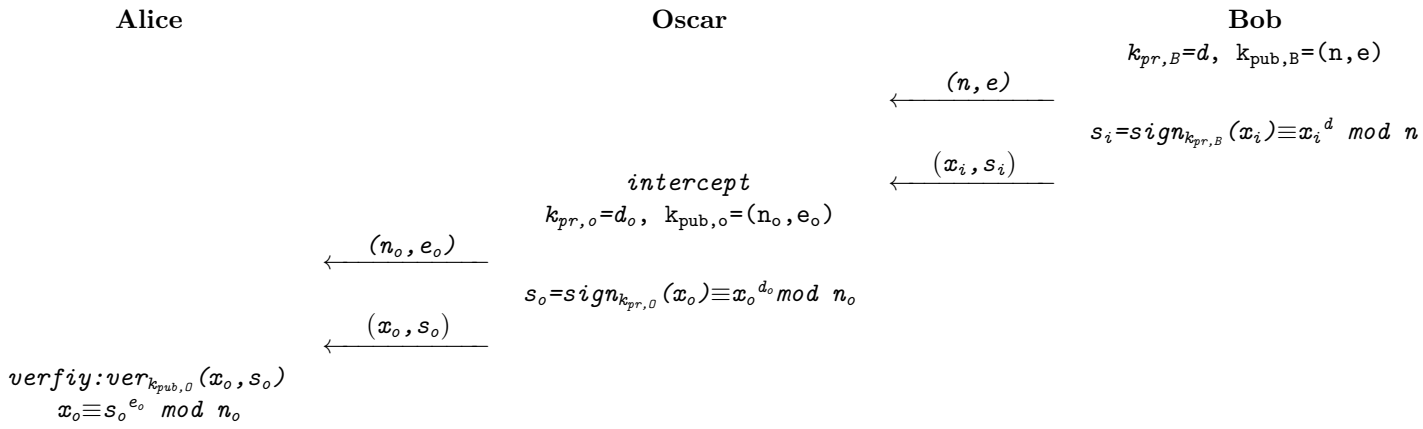
1. (**1 pts.**) `Message integrity` is the security goal of ensuring that a communication arrives at the recipient in a form identical to what the sender transmitted.
2. (**1 pts.**) `Forward security`is the following property of key agreement protocols: Even if the private key associated to one of the party's public keys is compromised, past session keys cannot be recovered from the contents of the exchanged messages.
3. (**1 pts.**) `MAC` is a symmetric-key algorithm for ensuring that a message has not been tampered with.
4. (**1 pts.**) `Revocation list.` is a way of checking whether the private key matching the public key in a certificate has been compromised and so the certificate should no longer be accepted.
5. (**3 pts.**) The three security requirements for hash functions are `preimage resistance/one-way`, `second preimage resistance` and `collision resistance`.
6. (**1 pts.**) All asymmetric protocols require that the public keys are authenticated, e.g., with certificates. Otherwise `man-in-the-middle` attacks are possible.
7. (**1 pts.**) In `Rainbow table` attack, an attacker uses a table that contains all possible passwords already in a hash format.

Question 1.3 (15 Points)

In an `RSA digital signature scheme`, Bob signs messages x_i and sends them together with the signatures s_i and his public key to Alice. Bob's public key is the pair (n,e); his private key is d.

Oscar can perform `man-in-the-middle` attacks, i.e., he can replace Bob's public key by his own on the channel. His goal is to alter messages and provide these with a digital signature which will check out correctly on Alice's side. Show everything that Oscar must do for a successful attack.

Oscar receives the message x_i, alters it and signs it with his own private key d_o.
Then he sends the new message x_o together with the signature s_o and the putatively appropriate public key (n_o, e_o) of Alice (which is instead the one of Oscar).



Question 1.4 (10 Points)

Suppose that `m > 2` users want to communicate securely and confidentially. Suppose further that each of the `m` users wants to be able to communicate with every other user without the remaining `m-2` users being able to listen on their conversation. How many distinct keys are needed if we are using:

- `A symmetric key cryptosystem`, where two users use a shared secret key to communicate,
- `A public key cryptosystem`, where every user has a public key, K_{pub} and a private (secret) key, K_{pr}.

How many keys are needed for each type of cryptosystems if `m = 1000`?

Solution

- Case 1: A symmetric key cryptosystem
  - Every user has to posses `m-1` distinct encryption/decryption keys to be able to communicate with every other user. Since two communicating users share a common key, the total number of cryptographic keys is equal to: $N = \frac{m(m-1)}{2}$.
  - $\Rightarrow$ for $m = 1000$, $N = \frac{1000 \times 999}{2}$ distinct keys are needed when A symmetric key cryptosystem is used.
- Case 2: Public Cryptosystem
  - If `m` users are using a public key cryptosystem, then in total $N = 2 \times m$ distinct cryptographic keys are needed to make communication secure since every user is assigned one encryption key and one decryption key.
  - $\Rightarrow$ the total number of public key cryptographic keys $K = (K_{pub}, K_{pr})$ is equal to: $N = 2 \times m$. For $m = 1000$, $N = 2 \times 1000$ distinct keys are needed when public key cryptosystem is used.

Question 1.5 (10 Points)

The `26` lower-case letters of the alphabet and the digits `0,1,2,...,9` are used to make four character long computer passwords.

i. How many passwords are possible if repetition of characters within a password is not allowed?

No. of password $= 36 \times 35 \times 34 \times 33$

ii. How many passwords are possible if repetition of characters within a password is allowed, BUT passwords must contain at least one letter and at least one digit?

- Let:
- p = No. of passwords containing at least 1 letter and 1 digit
- q = No. of passwords without restruction
- r = No. of passwords without digits (all letters)
- s = No. of passwords without letters (all digits)

$\Rightarrow p = q - r - s = 36^4 - 26^4 - 10^4 = 1{,}212{,}640$

Question 1.6 (20 Points)

SuperMail is a company designing a secure email system for protecting emails using cryptography. They have hired you to advise them on the best way to use cryptographic algorithms for this purpose. Their system generates two public/private keypairs for each user of the system (one keypair for signing, one for encryption) and provides a way for each user to securely learn the public keys of all of their contacts. In the following, you can assume that digital signatures are computed using RSA and public-key encryption is performed using ElGamal.

SuperMail wants every email to be authenticated and protected from modification or tampering while it is transit from the sender to the receiver. Suppose Alice is sending an email M to Bob. Given SuperMail's design constraints, which of the following options would be a secure way to protect the authenticity and integrity of her email? Circle all secure choices. Illustrate/Justify why for any insecure choice.

i. Alice's software should encrypt M under Bob's public key. In other words, Alice's software should send $E_{K_{pub,B}}(M)$ to Bob.

Not a secure way to protect the authenticity and integrity of Alice's email: Encryption does not provide authenticity/integrity. Anyone can send such a ciphertext

ii. Alice's software should send M along with a digital signature on M using Alice's private key. In other words, Alice should send $\{M, Sign_{K_{pr,A}}(M)\}$.

A secure way to protect the authenticity and integrity of Alice's email

iii. Alice's software should choose a new symmetric key k for this email. Then it should send four pieces of information: the message M, a MAC on M under the key k, an encryption of k under Bob's public key, and a digital signature on k using Alice's private key. In other words, Alice should send $\{M, MAC_k(M), E_{K_{pub,B}}(k), Sign_{K_{pr,A}}(k)\}$

Not a secure way to protect the authenticity and integrity of Alice's email:

- Once Bob receives one such message, he can send forged messages to Carol and make think Alice sent them. For instance, Bob can send $\{M', MAC_k(M'), E_{K_{pub,C}}(k), Sign_{K_{pr,A}}(k)\}$
- Also, the signature on k might reveal the value of k. Digital signature schemes are not guaranteed to provide confidentiality protection for the message that was signed. (In some signature schemes, the signature reveals the message that was signed; in others, it does not. Both possibilities are allowed by the definition of security for digital signatures.) Consequently, a man-in-the-middle might be able to recover k from $Sign_{K_{pr,A}}(k)$ and then modify M and recompute a new MAC that will be valid (using his knowledge of k).

This page left blank intentionally. There are no more questions.