Network Security (TIE-30406)
Coursework

Antonis Michalas
antonis.michalas@tut.fi

November 27, 2018

SUBMISSION DEADLINE: 05.12.2018 AT 16:00

Code	Description
$i : a$	Entity i executes the action a
$x \leftarrow y$	Assign value y to x
$\mathsf{Rand}()$	Random number generator
$x == y$	Check if x is equal to y
$\mathsf{pk_i}$	Public Key of i (known to all other hosts)
$\mathsf{sk_i}$	Private Key of i
Gen_{K_s}	Generate Session key
$\mathsf{Enc(K, m)}$	Symmetrically encrypt m with the key K
$\mathsf{Dec(K, c)}$	Symmetrically decrypt c with the key K
$\mathsf{E_{pk_i}(m)}$	Encrypt m with the public key of i
$\mathsf{D_{sk_i}(c)}$	Decrypt c with the private key of i
$h = H(m)$	Hash for m with the common hash function H
$(a, b, \ldots) \rightarrow i$	Send a message to i containing a, b, ...

Table 0.1: Primitives for the cryptographic tools.

EXERCISES

EXERCISE 1 – XOR ENCRYPTION (3 MARKS)

You are given a message m and its OTP encryption c. Can you compute the OTP key from m and c?

EXERCISE 2 – XOR ENCRYPTION (3 MARKS)

Alice and Bob wish to exchange 2 messages, m_1 and m_2. Assume that Alice wishes to send m_1 to Bob, and Bob responds with m_2. They encrypt the messages using the XOR operation, using a key of the same length with the message. The encrypted messages are sent over the channel:

$$c_1 = m_1 \oplus K_1$$

$$c_2 = m_2 \oplus K_2$$

Unfortunately, Bob forgets to use the second key K_2 after the first transmission and reuses K_1 ($K_1 == K_2$). Assume Eve is listening the channel and reads:

$$c_1 = 001101000100110001101111$$

$$c_2 = 001010010100110001111000$$

Assume that Eve is able to understand the content of m_1, e.g., she guessed that it is the binary representation of the ASCII characters net. Write the binary and ASCII representation of m_1, m_2, and K_1 and explain how Eve can obtain m_2 and K_1 (to convert ASCII text to binary you can use the following site: `http://www.roubaixinteractive.com/PlayGround/Binary_Conversion/Binary_To_Text.asp`).

EXERCISE 3 – KEY ESTABLISHMENT (2 MARKS)

Alice and Bob want to establish a two-way secure channel. Please answer the following two questions for the case that (a) they use symmetric key cryptography, and (b) they use asymmetric key cryptography:

1. How many keys do they need?
2. Who needs to know what key?

EXERCISE 4 – SYMMETRIC ENCRYPTION (8 MARKS)

Consider the following protocol (Figure 0.1) which Alice and Bob use in order to *mutually authenticate* each other, i.e., convince each other that "they are who they say they are". Assume that Alice and Bob share a secret key K.

In this protocol, Alice first sends an unpredictable random number R_A. In the second step, Bob encrypts this message to prove knowledge of the key K and also sends a random number R_B. In the third step, Alice decrypts $E(K, R_A)$. If the result is not her original number she aborts the protocol otherwise she encrypts R_B and sends it to Bob. Bob performs a similar check and if everything is OK, he's convinced he's talking to Alice. Find two attacks in which an attacker can impersonate some of them to the other.

(*Assume that the key is not compromised, so nobody can use it to create fake messages.*)

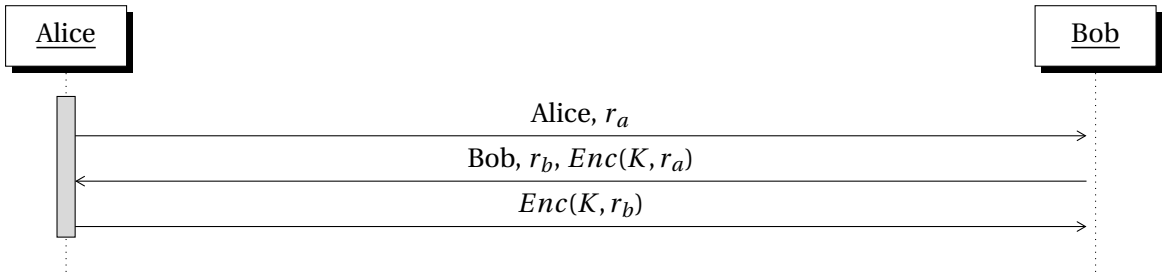

Figure 0.1: Mutual Authentication protocol

EXCERSICE 5 – CRYPTOGRAPHIC PROTOCOLS (8 MARKS)

Based on the primitives defined in Table 0.1, suppose you want to connect two hosts A and B over a WLAN system. Assume the following two scenarios:

a) Host B wants to authenticate host A based on a **challenge-response** protocol. Someone designed the following protocol:
   a) $B : challenge \leftarrow \mathsf{Rand}()$
   b) $B : (challenge) \rightarrow A$

c) $A : response \leftarrow \mathsf{E}_{\mathsf{sk}_A}(challenge)$

d) $A : (response) \rightarrow B$

e) $B : \mathsf{E}_{\mathsf{pk}_A}(response) == challenge$

Does it offer mutual authentication? If no, how would you change the protocol?

b) The nodes want to exchange large volumes of data ensuring Integrity, Sender Authenticity and Confidentiality. Design the communication scheme using notation from the Table 0.1 to achieve the desirable security in a efficient manner.

EXCERSICE 6 – ATTACKING A SYSTEM (3 MARKS)

I created a really simple login form that requires the user to provide a username and a password. In order to protect the login procedure as well as the password that is stored in the database (DB), I used a hash function. The problem is that I can't remember which exactly hash function I used. So, the first thing you need to do is to find out which hash function I used in order to hash the credentials. Furthermore, my memory is not working well (or not working at all) since I moved to London. As a result, I realized that I also forgot my password! In a desperate try to solve this issue I connected to my local DB and found out the following hash value that corresponds to my password:

h(password) = b3746dd06becc0d83eb520f64111cbb542e03e09

Having this information can you successfully bypass the login authentication and help me log in back to my account? Can you also remind me my password? If so, please explain how this is possible.

The link to the login form is here: `http://amichalas.com/UoW/ECSF705/Login_2/`. For the username...you can use anything!

EXERCISE 7 – CRYPTOGRAPHIC PROTOCOLS (10 MARKS)

Please use the given and your defined notation and tools and design security protocols for the following settings:

a) A host, A, communicates with a host B across the Internet. Both hosts have each just its own public and private key pair. Host A needs to "ping" host B in order to make sure it is "alive". Assume that the two hosts just synchronized themselves with a Time Server and thus $t^A_{clock} = t^B_{clock}$. Provide the best possible solution under the conditions. Please explain if the problem is solved.

b) If yes, demonstrate how the assumed adversary is countered by your protocol. If no, please explain what additional steps are required and provide two ways to address the problem.

c) For either case you demonstrated a sufficient solution in (b), consider that A will repeatedly and frequently "ping" B. Rather than using public key cryptography, consider that A and B wish to use a symmetric key scheme. First, please explain what can be the motivation, why can a symmetric key protocol be a better choice. Second, please design a protocol that allows A to "transport" a symmetric key to B.

d) Explain how your protocol in the previous setting allows (or augment here your protocol as necessary) A and B to mutually authenticate each other and be sure that only the other end (that is, for A only B and vice versa) got and now knows the new key.

EXERCISE 8 – LOGIN (15 MARKS)

Consider the following method for Alice logging into a server S.
Setup: Alice picks a password P and a number N and then computes $x_1 = f(P,1), x_2 = f(x_1,2), x_3 = f(x_2,3), ..., x_N = f(x_{N-1},N)$, where $f()$ is some easy to compute function but hard to invert. She then stores the pair (x_N, N) with the server to whom she wishes to login later. Similarly, the server maintains for each user the (different) value x_N and the index N.

Authentication: When Alice wants to have access to the host, she types her username and the host looks up her entry and sends $N-1$ to her. She then responds back with the value x_{N-1}, which the host verifies by computing $f(x_{N-1}, N)$ and comparing against the stored value x_N. If the two values match, the server gives Alice access to her account and replaces the values (x_N, N) with the values $(x_{N-1}, N-1)$.

a) What are the advantages (if any) of this scheme over ordinary passwords?

b) What are some attacks (if any) that can be applied to this scheme?

EXERCISE 9 – SYMMETRIC KEY SECURITY PROTOCOLS (9 MARKS)

Please use the given and your defined notation and tools and design security protocols for the following settings:

a) A wireless sensor node, A, is required to provide periodically measurements to a device within range, B. The two share a symmetric key. Each message sent by A contains a single measurement. B needs to verify the origin authenticity of each message.

b) In the same setting as above, consider a period over which multiple measurements are sent by A. Now, B needs to verify the authenticity and integrity of this sequence of measurements. At first, assume that $t^A_{clock} = t^B_{clock}$ throughout this period.

c) In the previous setting, assume that the clock of A can**not** be synchronized with that of B. Again, B needs to verify the authenticity and integrity of this sequence of measurements.

d) As an additional requirement, B needs to ensure that all measurements of A are confidential.

e) In the previous setting, now B needs A to use different symmetric keys for ensuring confidentiality and authenticity. Let those be $\mathsf{K}^{\mathsf{c}}_{\mathsf{AB}}$ and $\mathsf{K}^{\mathsf{a}}_{\mathsf{AB}}$ respectively and assume that those are available at A and B.

f) Without re-writing the protocol you devised for the previous setting: How can you handle a situation that A and B have only one shared key for authentication but they decide to use a second one for confidentiality?

EXERCISE 10 – ASYMMETRIC KEY SECURITY PROTOCOLS (9 MARKS)

Please use the given and your defined notation and tools and consider the following intelligent transportation setting: A vehicle A, e.g., owned by a private individual, equipped with an On-Board Unit (OBU), a second vehicle B, and a Road-Side Unit (RSU) C, a special-purpose device, e.g., owned by a highway/city authority. Each of the A, B, and C is equipped with a public/private key pair and a certificate provided by the same authority.

a) A transmits periodically the so-called safety beacons, short messages that provide anyone within range A's velocity and direction. Safety beacons must be authenticated by every device that receives them and their freshness should be verified. Please design and present your protocol, stating your assumptions.

b) Please explain briefly how your above protocol achieves the sought security properties.

c) When B deems it is within range of C, notably when it receives a periodically transmitted "hello" message by C: *(i)* B needs to authenticate C, *(ii)* C needs to authenticate B, and *(iii)* B needs to download a rather large file (e.g., a map with travel info attached) from C that should be authenticated and fresh but not secret. Please design and present your protocol, stating your assumptions. Please note that you are not constrained in not using symmetric key cryptography at all; nonetheless, you *cannot* assume an *a priori* existence of symmetric shared keys.

d) Please explain briefly how your above protocol achieves core security properties (e.g. integrity, authenticity).

EXERCISE 11 – KEY DISTRIBUTION (15 MARKS)

Consider the following key distribution protocol in which two users Alice and Bob wish to establish a shared key K_{AB} with the help of a trusted server S. Assume Alice and Bob share secret keys K_{AS} and K_{BS} with S and that nonces are 64 bits long and keys are 128 bits long.

1. $A \rightarrow B : Alice, N_A$
2. $B \rightarrow S : Bob, N_B, \mathsf{Enc}(K_{BS}, \langle A, N_A, N_B \rangle)$
3. $S \rightarrow A : \mathsf{Enc}(K_{AS}, \langle K_{AB}, B, N_A, N_B \rangle), \mathsf{Enc}(K_{BS}, \langle A, K_{AB} \rangle)$
4. $A \rightarrow B : \mathsf{Enc}(K_{BS}, \langle A, K_{AB} \rangle), \mathsf{Enc}(K_{AB}, N_B)$

a) Describe at least two attacks that can be applied to this protocol (If your attack reduces to simple forwarding, it will not count...).

b) For each attack, give a countermeasure that renders the attack useless.

EXERCISE 12 – SSL (15 MARKS)

In the class we briefly described the steps that are involved in the SSL protocol. However, in the description and in order to make the protocol simpler we omitted many important steps. More precisely, we did not describe how we ensure the freshness of the exchanged messages and we did not provide any mechanism to guarantee their integrity. Redesign the protocol we described in the class in order to prevent any kind of replay attacks and also to ensure the integrity of the messages. Discuss/prove why the protocol you designed is considered as secure. To do so, you also need to define the adversarial model you will consider. Finally, make sure that all the assumptions you will take regarding the capabilities of the attacker are considered as realistic.