SENG2250 System and Network Security
School of Electrical Engineering and Computing
Semester 2, 2020

Assignment 2 (100 marks, 15%) - Due: 9 October, 23:59

Aims

This assignment aims to establish a basic familiarity with secure authentication and system security via analysing, demonstrating and designing solutions.

Questions

1. Perfect Forward Secrecy (15 marks)

The following Key Exchange Algorithm (KEA) was designed by the NSA of the US government. It is a variant of the Diffie-Hellman key exchange protocol.

$$A \rightarrow B: g^{r_A} \quad (mod\ p)$$
$$B \rightarrow A: g^{r_B} \quad (mod\ p)$$

A computes the shared key: $K_{AB} = (g^{r_B})^{x_A}(g^{x_B})^{r_A} \quad (mod\ p)$

B computes the shared key: $K_{BA} = (g^{r_A})^{x_B}(g^{x_A})^{r_B} \quad (mod\ p)$

Notations

A, B	Identity of users A and B, respectively.
p	A large and safe prime number.
g	A generator of the underlying multiplicative cyclic group of order p.
r_A	A random number selected by user A for one session, it is never reused.
r_B	A random number selected by user B for one session, it is never reused.
x_A, x_B	Long-term private keys of users A and B, respectively.
g^{x_A}, g^{x_B}	Long-term public keys of user A and B, respectively, they are already known to each other.

Does the KEA protocol provide the perfect forward secrecy? Please justify your answer, otherwise, **zero** marks will be given. (**15 marks**)

2. Hash Chain (25 marks)

Alice designed a hash chain based authentication protocol as follows

Step 1: The authentication server (i.e., verifier) chooses a cryptographic hash function $h: \{0,1\}^* \rightarrow \{0,1\}^\ell$ and releases it to the public.
Step 2: A user chooses a random seed $s \in \{0,1\}^*$ and computes n times of hash of s, such that

$$H_1 = h(s),\ H_2 = h(H_1), \dots, H_n = h(H_{n-1})$$

Step 3: The user <u>securely</u> (suppose attack-free) shares H_n with the server and discards s.
Step 4: The user stores $H_1, \dots, H_n$ and the server stores H_n, in their local databases, respectively.
Step 5: In the ith authentication, the user interacts with the server as follows.

User → **Server**: $E(k_i;\ H_{i-1}, N_u, i), N_u, i$

Server: *retrieves the session key k_i (i.e H_i) from the database. If k_i does not exist, then the authentication failed. Otherwise, the server decrypts the ciphertext that obtains H'_{i-1}, N'_u, i'. The user is authenticated if the following equation holds:*

$$H_i = h(H_{i-1}), \qquad N_u = N'_u, \qquad i = i'$$

Finally, the server sets $k_{i-1} = H_{i-1}$ and adds k_{i-1} to the database.

<u>Notations</u>

i – the index of the session key;

E – a secure symmetric-key encryption;

k_i – the ith session key that $k_i = H_i$;

N_u – a user-selected nonce.

Alice claims that the above authentication protocol provides:

- Replay attack resistance.
- Forward security. (It is not the perfect forward secrecy, see below)

Forward security: If a session key was compromised, previous (uncompromised) sessions remain secure, even if the adversary captures all previous messages.

For example, assume k_1 and k_2 are session keys used in sessions s_1 and s_2, respectively. Assume s_1 happened later than s_2. If k_1 was compromised, the authentication messages encrypted by using k_2 are still secure.

Your task: Analyse if this authentication protocol achieves the security requirements:

- Replay attack resistance. (**10 marks**)
- Forward security. (**15 marks**)

If yes, justify your answer, otherwise, modify the protocol to satisfy the security requirements.

3. Two-Factor Authentication Protocol Analysis (35 marks)

Multi-factor user authentication mechanisms require a user to possess multiple authentication factors, such as a knowledge factor ("something the user knows"), a possession factor ("something the user has"), and an inherence factor ("something the user is"), in order to login a computer system. One commonly used two-factor user authentication mechanism is based on smart-card (something the user has) and password (something the user knows). Such a mechanism should ensure that an adversary cannot pass the authentication even if he/she has obtained one authentication factor. Consider the following two-factor authentication protocol:

User Setup. Let x denote a 128-bit secret key of a remote web server, and $h(\cdot)$ a secure cryptographic hash function. Each legitimate client C with identity ID_C shares a 6-digit password pwd with the server. In addition, C has a smart-card issued by the server, which has the information (ID_C, B, p, g) stored in the Read Only Memory (ROM) of the card, where $B = h(pwd) \oplus h(x||ID_C)$, p is a large prime number, g is a generator of $\mathbb{Z}_p^*$, and $||$ denotes concatenation of two bit-strings.

User Login.

1. In order to login the server, the client first attaches the smart-card to a card reader which is connected to a computer, and then types in the password pwd. The computer retrieves the values of (ID_C, B, p, g) from the smart-card via the card reader, and computes
$$Z = B \oplus h(pwd).$$
After that, the computer chooses a random number $u \in \{1, \dots, p-1\}$ and computes
$$N_C = g^u \bmod p,$$
and sends a login request (ID_C, N_C) to the remote server.
2. Upon receiving the request, the web server first checks if ID_C belongs to a legitimate client. If the server cannot find ID_C in the database, then the request is rejected. Otherwise, the server chooses a random number $v \in \{1, \dots, p-1\}$, computes
$$N_S = g^v \bmod p, K = N_C^v \bmod p, Z' = h(x||ID_C), \text{ and } T_S = h(Z', N_C, N_S, K).$$
The server sends (N_S, T_S) to the client.
3. After receiving (N_S, T_S) from the server, the client's computer computes
$$K' = N_S^u \bmod p, T_S' = h(Z, N_C, N_S, K'),$$
and verifies if $T_S' = T_S$. If the equation holds, then the server is authenticated. The client's computer generates $T_C = h(Z, N_S, N_C, K')$ and sends T_C to the web server.
4. The web server computes $T_C' = h(Z', N_S, N_C, K)$ and verifies if $T_C' = T_C$. If the equation holds, then the client is authenticated; otherwise, the client authentication fails. If the client has three consecutive authentication failures, then the client's account will be locked by the web server, and the client needs to contact the administrator in order to unlock the account.

Your Task: Analyse the above authentication protocol. Does the protocol achieve two-factor user authentication? If your answer is yes, justify your answer by giving a security analysis for the protocol; otherwise, if your answer is no, show an attack against the protocol. When doing the analysis, consider the situation that <u>one of the two authentication factors is compromised and known by the adversary</u>.

- Does the protocol achieve two-factor user authentication, i.e. is it secure? (**5 marks**)
- Justify your answer. (**30 marks**)

4. Multilevel Security (25 marks)

Given the following access control matrix and security labels, answer the questions to find the capabilities of subjects.

	O1	**O2**	**O3**	**O4**
A	-	rw	w	rw
B	r	-	w	r
C	w	w	rw	w
D	rw	r	-	r

Access control matrix: Subjects (**A, B, C, D**); Objects (**O1, O2, O3, O4**);
r: read; w: write; -: no permission.

	Security Label
A	Top-Secret
B	Secret
C	Unclassified
D	Unclassified

Clearances of subjects: Top-Secret > Secret > Unclassified.

	Security Label
O1	Top-Secret
O2	Secret
O3	Secret
O4	Unclassified

Classifications of objects: Top-Secret > Secret > Unclassified.

i). Apply the **BLP** model, fill out the following table to specify readable and writable objects of subjects. (**12.5 marks**)

	Readable Objects	**Writable Objects**
A		
B		
C		
D		

ii). Apply the **Biba** model, fill out the above table to specify readable and writable objects of subjects. (**12.5 marks**)

Submission

All assignments must be submitted via Blackboard (Assessment tab for SENG2250). If you submit more than once, then only the latest will be graded. Your submission should be one ZIP file containing:

- Assessment item cover sheet.
- A PDF file which contains answers to all questions.

The mark for an assessment item submitted after the designated time on the due date, without an approved extension of time, will be reduced by 10% of the possible maximum mark for that assessment item for each day or part day that the assessment item is late. Note: this applies equally to week and weekend days.

Plagiarism

A plagiarised assignment will receive ZERO marks (and be penalised according to the university rules).