SENG2250 System and Network Security
School of Electrical Engineering and Computing
Semester 2, 2018

Assignment 3 (25 marks, 25%) - Due: Friday, 9 November, 16:59

Aims

This assignment aims at 1) understanding, designing, and analysing security systems under specific application scenario; 2) be familiar with different types of security protocols and show their use in secure system design; 3) implementing fundamental security protocols for further understanding the process of secure handshake work flow in practice.

Tasks 1: Secure Cloud-Based RFID Supply Chain System Design (13 marks)

Radio Frequency Identification (RFID) has been widely adopted for object identification. An RFID system comprises three essential components, namely RFID tags, readers and a backend server. An RFID tag is associated with a unique identifier which is allocated by the backend server. A typical RFID system is established by a single party who initiates the secret keys. To identify a tag, a reader communicates with the tag and sends the tag's response to the backend server. The server checks the tag's identity by using the shared keys and informs the reader whether the tag is valid. Conventionally, the system is considered to be controlled by a single party who maintains all the secret information. But, in some practical scenarios, RFID tags, readers and servers could be operated by different parties.

Supply chain is a popular application of RFID techniques. Roughly speaking, each of goods will be attached by an RFID tag which has a unique identifier. By using (mutual) **authentication** protocols between RFID **reader**, **tag** and **server**, it allows user to identify and trace the location of particular object. Note that the reader, server and tag might be owned by different parties.

Cloud-based RFID supply chain system provides various services. We describe entities of this system as following.

- **User**: It is the owner of goods. A user firstly registers to the cloud system. Then, he/she can add information of goods into the cloud system. Specifically, in this supply chain system, user will attach RFID tag to each of goods and input (tag, goods) information to the cloud's database. For example, a stored record could express the relationship between tag's identifier and goods information. User does not necessarily need to know the secret key of each tag, instead he/she will delegate this key management task to the cloud system.
- **Cloud system**: It is a trusted authority which provides services for both users and couriers. Basically, it has the following functionalities.
  - It can register a new user and provides secure user authentication for login.

- It can register a courier as a client and provides secure client authentication. Note that courier does **NOT** own RFID tag or its information, such as tag's identifier and the tag's location.
- It allows a user to authorise a courier to manage/access particular group of RFID tags when the courier is delivering the goods. That is, courier will be allowed to gather tag information.
- It provides key management for tags. Cloud system may generate, transfer and maintain tag secret keys, but it is not a mandatory requirement. Whether cloud system knows tag's secret (keys) or not will be determined by underlying cryptographic techniques.
- It can help authorised client (courier) to identify (authenticate) RFID tags. Cloud system will involve RFID tag authentication then responds the result to auhorised client.

- **Courier**: It is an organisation which provides delivery services to users. It provides an application that users can trace and collect current goods information from it. A courier is the entity who has RFID readers.
  - **RFID reader**: It can interrogate tags, but it **cannot** decide whether a tag is valid or not. Instead, reader typically transfer the information to backend server who has capability to conduct tag authentication.
  - **RFID tag**: We consider lightweight RFID tags which have limited memory and computational power. For example, the memory size of a tag is **1KB**. This memory will be used for all storage, including tag identifier, system parameters, keys, intermediate results and others. More powerful tags usually mean higher cost to users. **Note that a lightweight tag does not have time system and it cannot access network.**

Your Tasks. According to the above cloud-based RFID supply chain system, answer the following questions.

1. Analyse potential security threats and issues of this system. Justify your answer for each. (**3 marks**)

2. What technology can be used to provide client authorisation in this system? Describe the architecture in detail. (**2 marks**)

3. Design a **public-key based** mutual authentication protocol for tag authentication. It should provide at least the following properties. (**3 marks**)

   - Perfect forward secrecy.
   - Tag anonymity - only the authorised entity will be able to know whether the tag is valid or nor.

   Can we use PKI to verify the public keys (for tag and server) during the protocol execution? Why? (**1 mark**)

   Note: You cannot use RSA based encryption/signatures, because it would be unaffordable to lightweight RFID tags.

4. Design a symmetric-key based mutual authentication protocol to satisfy the following requirement. (**4 marks**)

   - Tag anonymity - only the authorised entity will be able to know whether the tag is valid or nor.

- Secure key update - after each **successful** mutual authentication, the secret key of tag must be updated. Consider how to ensure the shared secret is updated consistently.
  Hint: When a tag is somehow been desynchronised with server, i.e their shared key become different, is there any solution can help to "re-synchronise" the tag?

Please describe the protocols as below:

$$A \rightarrow B : E(sk_b, a_0)$$
$$A \leftarrow B : E(sk_a, b_0)$$
$$\vdots$$

Tasks 2: Programming (12 marks)

Alice and Bob intend to do message exchange. They will use the following method to establish a secure channel and exchange messages then.

- Alice and Bob uses STS protocol to establish a session key.
- Once session key is created, they use 3-DES encryption to protect message confidentiality.
- To enhance the security, they also apply the Counter Mode with 3-DES encryption for each message.

Your task: Please implement the above mechanism using C++ or Java under the following requirement.

- Any public key encryption or digital signature scheme needed in this method will be based on **RSA**.
- Any symmetric key encryption applied in this method will be **3-DES**.
- Any hash function used in this method will be **SHA-256**. You can use its implementation from external libraries.
- Implement STS protocol. (**3 marks**)
- Implement RSA encryption and signature, because you cannot use it directly from external libraries. (**2 marks**)
- Implement 3-DES encryption. You **can** use DES implementation from any cryptographic libraries. (**2 marks**)
- Assume that Alice and Bob know each other's public key at the beginning.
- Implement the **Counter (CTR) Mode**. (**2 marks**)
- Show that Alice and Bob can send/receive messages by using 3-DES with CTR mode after the secure session established. Assume that each message will be in **64 bytes**. (**3 marks**)
- You can use socket programming or simulate message sending/receiving by using function calls.
- You MUST use BigIntegers (Java) or NTL (C++) to handle large number computation. Note that, the RSA key size must be at least **1024**-bit.

Notes

- Submit the source code and provide a screen shot in your report to show the program execution.

- Provide instructions to show how the program will be compiled and executed.
- Provide name (and installation instructions if needed) of external cryptographic libraries used for the implementation. In this case, you should specify what method/package was used for the assignment.
- Uncompilable or unexecutable program may receive **zero** mark.

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.
- Report (PDF file): answers of Task 1, program compilation and execution instructions, screen shot of program run.
- Source code of Task 2.

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 a zero mark (and be penalised according to the university rules).