College of Computer and Information Science
Northeastern University
CS4740/6740: Network Security

Fall 2012

Problem Set 4 (due March 17th, 2013 @ 11:59pm) [100 points]
Presentations on March 13, 2012

This problem set is to be done in teams. Late submissions will result in a 10% penalty per day (e.g., 2.5 days late result in 25% penalty).

1. Propose an architecture and design for a secure instant messaging system. The architecture can have a server (but not necessarily, it is up to you to make the decision). If a server is used the messages between the users should not go through the server (with the exception of the first discovery messages).
2. Your protocols should satisfy the following constraints:
    a. The users only needs to remember a **single** password. The client application should not remember the users passwords.
    b. If you are using public/private keys, the client cannot remember the private key of the users. You can however assume that the client knows the public key of the server (if you are using public/private keys).
    c. Your system should provide mutual authentication (user and server), message integrity and confidentiality.
3. Describe in detail the security protocols that you are proposing for the whole system. Remember that you are not allowed to use an SSL library or a complete existing protocol. You are allowed to use cryptographic libraries (e.g., jce) that provide encryption, hashing, etc.
4. Discuss the following issues:
    a. Does your system protect against the use of weak passwords? Discuss both online and offline dictionary attacks.
    b. Is your design resistant to denial of service attacks?
    c. To what level does your system provide end-points hiding, or perfect forward secrecy?
    d. If the users do not trust the server can you devise a scheme that prevents the server from decrypting the communication between the users without requiring the users to remember more than a password? Discuss the cases when the user trusts (vs. does not trust) the application running on his workstation.

You will get more points for core security, addressing weak passwords, resiliency to denial of service attacks. Bonus points can be obtained for additional services such as identity hiding.

Notes:

1. Remember that later on you will have to implement your design. You can propose two designs, one design that you will be implementing and a second design for a system that might be more complex and that will provide more security against the discussed attacks but that you will not implement. Read the secure IM application implementation guidelines on:
   http://www.ccs.neu.edu/home/noubir/Courses/CS6740/F12/problems/app-guidelines.txt

2. You will have to present your design to the class on March 13th. Please prepare slides. You have to submit your final design document on March 17th and will not be allowed to change your design afterwards.

Notes for the presentations:

The duration of each presentation is 10 minutes. Remember that you have to prepare a set of slides. During the presentations you should take notes of other teams potential flaws. You can expect some short questions.

Your presentation should focus on the design that you will be implementing. The goal is for every team to know about other teams design. This includes:

- The setup:
    - architecture (server?, clients?),
    - assumptions (e.g., type of keys, who knows what),
    - services
- Protocols (messages and exchanges): examples of protocols you might be using are
    - authentication protocol (login),
    - key establishment protocol (and authentication with peers),
    - messaging protocol,
    - logout protocol
- Discussion of the choices you made and how they help you provide the claimed services.

Try to make clear slides (not too dense but self-contained). For each phase/sub-protocol clearly describe all the details of the message: Registration (does not have to be automated can be pre-configured), login, commands (e.g., list), client-2-client, etc. Use the notation of the textbook for signing/encrypting messages (the same notation we used in class). The design should be clear enough that one could take it and implement your system.

Discuss which services you are providing (e.g., PFS, identity hiding, DoS protection) and why you believe they are "secure". See PS4 questions. You can also discuss additional services that you are not planning to implement if time permits but it is not necessary.