

**Problem Set 2 (due February 20, 2007 at 11:59pm).
[100 points + 20 bonus points]**

Important Notes:

1. No team work is allowed for this problem set.
2. Late submissions will result in a 10% penalty per day (e.g., 2.5 days late result in 25% penalty).
3. You can use the internet to get some help for problem 1, but you should use your own words and examples when answering the questions.

Problem 1 [50 points + 20 bonus points]: Broadcast Authentication.

Assume that a node wants to broadcast a real-time stream of data. Each receiver wants to verify that the data is generated by the source and was not modified, or replayed (i.e., integrity/data origin protection).



1. Why can't the source just use any message authentication code mechanism based on symmetric key crypto (e.g., HMAC) if we assuming that all the receivers share the same secret with the source (i.e., what is the threat)?
2. If the receiving nodes do not have the capability to do a signature verification of each packet. Propose a technique to amortize signature verification and trades-off the computation with a delay in verification. Discuss potential DoS attacks on your approach.
3. Lamport hash technique allows protecting against both eavesdropping and database reading. First, briefly describe how Lamport hash can be used to provide broadcast authentication.
4. Conventional Lamport hash requires $O(n)$ computations or $O(n)$ memory storage. Propose a scheme that requires $O(\log n)$ computations and $O(\log n)$ memory storage for each authentication. In addition to the algorithm description, explain through a detailed example how the scheme works. *Hint: google for hash chains and carefully read "the" important paper(s) on this topic!*
5. Assume that a set of nodes are loosely synchronized (i.e., we know a bound on the maximum time difference between the source and all other receiving nodes: T). How can the scheme described in (4) be used to provide data origin authentication of broadcast messages?
6. **(BONUS: 20 points if submitted before 4/17):** Implement the broadcast authentication scheme, and the proposed algorithm.

Problem 2 [25 points]:

Consider the following authentication protocol, to achieve a mutual authentication between A , and B (without session key establishment). Assume that the public keys of then involved entities are known beforehand.

- Step 1. $A \rightarrow B: \{N_A, A\}_{KB}$
 Step 2. $B \rightarrow A: \{N_A, N_B\}_{KA}$
 Step 3. $A \rightarrow B: \{N_B\}_{KB}$

1. Show that this protocol has a flaw.

Hint: consider an intruder I who wants to impersonate A to B . Intruder I first waits for A to try to authenticate him, then...

2. If this protocol is used to establish a session key $K = N_A \oplus N_B$, can I succeed in obtaining K .
3. Propose a modification of the protocol that prevents this attack.

Problem 3 [15 points]:

Assume that A and B share a secret key of length 8 bytes (K_1, K_2, \dots, K_8). Does the following protocol have any major vulnerability? If yes describe an attack in detail and show how many steps it would take an adversary to identify the secret key.

1. $A \rightarrow B: \text{I am } A, (R_{11}, R_{12}, \dots, R_{18})$
2. $B \rightarrow A: (R_{21}, R_{22}, \dots, R_{28})$
3. $A \rightarrow B: \text{Hash}(K_1 + R_{21} \bmod 2^8, K_2 + R_{22} \bmod 2^8, \dots, K_8 + R_{28} \bmod 2^8)$
4. $B \rightarrow A: \text{Hash}(K_1 \oplus R_{11}, K_2 \oplus R_{12}, \dots, K_8 \oplus R_{18})$

Assume that *Hash* is a cryptographically secure hashing function. Note that K_i and R_{ij} are of length 1 byte.

Problem 4 [10 points]:

Page 302, problem 14.