Homework 3
Introduction to Information Security (266-642)
Due Date: April 10, 2007 (Tuesday)

Note: You can talk to your classmates, instructor, and TA about the problems. However, unless stated otherwise, problems should be written up individually. University of Wisconsin rules for academic misconduct apply.
This is a long homework (230 points), so it has a higher weight. Please start early.
In the homework, "the Stallings book" refers to [2] and "the Handbook" refers to [1] (I have linked the Handbook to the class homepage. You can download it for free.) Unless otherwise stated each part of a question has equal weight.

Question 1 (Hash Algorithms [50 points]):
Part A [25 points] : Problem 11.4 from the Stalling's book.
Part B [25 points]: Problem 11.6 from the Stalling's book.

Question 2 (Specific hash algorithms [20 points]) :
Part A [15 points] : Problem 12.2 from the Stallings book.
Part B [5 points] : Problem 12.3 from the Stallings book.

Question 3 (Authentication Protocols [40 points]):
Part A Problem 13.1 from the Stallings book.
Part B: Problem 13.2 from the Stallings book.

Question 4 (DSA [45 points]):
Part A: Assume that Alice uses the same random number k to sign two messages M and M'. Demonstrate that if Oscar knows the two signatures, he can derive the private key x.
Part B: Show that knowing the random number x is *equivalent* to knowing the random number k, i.e., if Oscar knows x, he can find k and vice-versa.
Part C: Problem 13.14 from the Stallings book.

Question 5 (Kerberos [45 points]): For this question you have to read the explanation of Kerberos version 5 and appendix 14A from the Stallings book. Moreover, please read my note on interrealm authentication in Kerberos version 5.
Part A: Problem 14.1 from the Stallings book.
Part B: Problem 14.2 from the Stallings book.
Part C: Suppose there is a "trust relationship" between realms in the CS and Biology department. Bob, who is a user in the CS realm, wants to access a server V in the Biology realm. Show the various steps required for Bob to authenticate himself to V.

Question 6 (X.509 [30 points]):
Part A: Problem 14.3 from the Stallings book.
Part B: Consider the CAs arranged in a hierarchy as shown in Figure 1. Show the various certificates used to "navigate" the hierarchy. Demonstrate the chain that "validates" the public key of `Alice` to `Bob` and vice-versa.

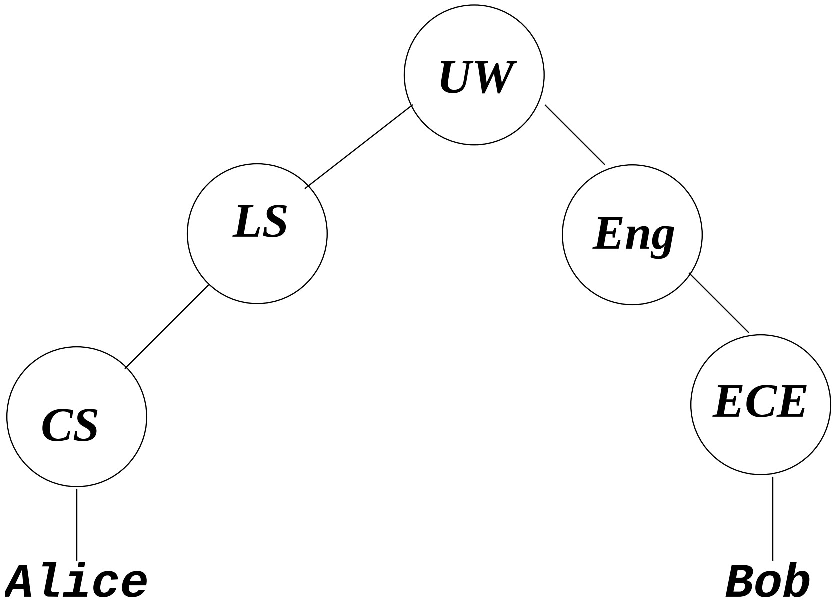

Figure 1: Hierarchy of certificate authorities.

1 Interrealm Authentication in Kerberos Version 5

Scenario: Assume that user U is in realm R_1 and wants to access the server V in realm R_k. There is a path $R_1 \rightarrow R_2 \rightarrow \cdots \rightarrow R_k$ from realm R_1 to R_k. Conceptually, each edge $R_i \rightarrow R_{i+1}$ (for $1 \leq i < k$) represents a trust relationship between realm R_i and R_{i+1}, which usually means that there is a shared key between the two realms.

Initial request: U requests a *ticket-granting ticket* or *TGT* from the KDC in realm R_1 (which we denote by $KDC[R_1]$) for realm R_k with the `FORWARDABLE` flag set.[1] Since R_1 does not have a trust relationship with R_k, it issues a TGT $TGT[R_1 \rightarrow R_2]$ for realm R_2 with the `FORWARDABLE` flag set. We are assuming that there is a mechanism for realm R_1 to discover that there is a path to realm R_k that goes through R_2. *Note:* I am also assuming that the servers only issue these tickets if their policy allows it. For example, $KDC[R_1]$ only issues the TGT with the `FORWARDABLE` flag on to U, if its policy allows it. This will be implicit throughout the document.

Walking the path: Using the TGT $TGT[R_1 \rightarrow R_2]$, U requests a TGT for realm R_3 from the *ticket granting server* or *TGS* (denoted by $TGS[R_2]$) in realm R_2. The TGT issued by $TGS[R_2]$ (denoted by $TGT[R_2 \rightarrow R_3]$) for R_3 has the `FORWARDABLE` and `FORWARDED` flags on. The $TGT[R_2 \rightarrow R_3]$ can have a different address than U (presumably an agent is handling this on behalf of the user U). This process is repeated until U "reaches" the realm R_k, i.e., it has a TGT $TGT[R_{k-1} \rightarrow R_k]$ issued by $TGS[R_{k-1}]$ for the realm R_k.

Accessing V: The TGT $TGT[R_{k-1} \rightarrow R_k]$ is presented to the TGS $TGS[R_k]$ to obtain a *service-granting ticket* or *SGS* $SGT[R_k, V]$ for server V. This SGS can then be used to access the server V.

[1] In general, an entity will be indexed by the realm that it pertains to, e.g., a ticket-granting ticket or TGT issued by realm R_i for realm R_j will be denoted by $TGT[R_i \rightarrow R_j]$.

References

[1] A.J. Menezes, P.C. Van Oorschot, and S.A. Vanstone. *Handbook of Applied Cryptography*. CRC press, 1997.

[2] William Stallings. *Cryptography and Network Security: Principles and Practice*. Prentice Hall, 2006.