

Principles of Information Security

CS 5323 Lecture 13

Prof. William Winsborough
November 4, 2008

Business

- Questions from previous lectures?
- IDEA Survey will be Thursday at the end of the class (so you can leave when you're done)
- Remainder of slides are ©2004 Matt Bishop

4 November 2008

Winsborough CS 5323 Lecture 13

2

Chapter 9: Key Management

- Session and Interchange Keys
- Key Exchange
- Cryptographic Key Infrastructure
- Storing and Revoking Keys
- Digital Signatures

4 November 2008

Winsborough CS 5323 Lecture 13

3

Overview

- Key exchange
 - Session vs. interchange keys
 - Classical, public key methods
- Cryptographic key infrastructure
 - Certificates
- Key storage
 - Key revocation
- Digital signatures

4 November 2008

Winsborough CS 5323 Lecture 13

4

Notation

- $X \rightarrow Y: \{ Z || W \} k_{X,Y}$
 - X sends Y the message produced by concatenating Z and W enciphered by key $k_{X,Y}$, which is shared by users X and Y
- $A \rightarrow T: \{ Z \} k_A || \{ W \} k_{A,T}$
 - A sends T a message consisting of the concatenation of Z enciphered using k_A , A 's key, and W enciphered using $k_{A,T}$, the key shared by A and T
- r_1, r_2 nonces (nonrepeating random numbers)

4 November 2008

Winsborough CS 5323 Lecture 13

5

Session, Interchange Keys

- Alice wants to send a message m to Bob
 - Assume public key encryption
 - Alice generates a random cryptographic key k_s and uses it to encipher m
 - To be used for this message *only*
 - Called a *session key*
 - She enciphers k_s with Bob's public key k_B
 - k_B enciphers all session keys Alice uses to communicate with Bob
 - Called an *interchange key*
 - Alice sends $\{ m \} k_s \{ k_s \} k_B$

4 November 2008

Winsborough CS 5323 Lecture 13

6

Benefits

- Limits amount of traffic enciphered with single key
 - Standard practice, to decrease the amount of traffic an attacker can obtain
- Prevents some attacks
 - Example: Alice will send Bob message that is either "BUY" or "SELL". Eve computes possible ciphertexts $\{ \text{"BUY"} \}_k$ and $\{ \text{"SELL"} \}_k$. Eve intercepts enciphered message, compares, and gets plaintext at once

4 November 2008

Winsborough CS 5323 Lecture 13

7

Key Exchange Algorithms

- Goal: Alice, Bob get shared key
 - Key cannot be sent in clear
 - Attacker can listen in
 - Key can be sent enciphered, or derived from exchanged data plus data not known to an eavesdropper
 - Alice, Bob may trust third party
 - All cryptosystems, protocols publicly known
 - Only secret data is the keys, ancillary information known only to Alice and Bob needed to derive keys
 - Anything transmitted is assumed known to attacker

4 November 2008

Winsborough CS 5323 Lecture 13

8

Classical Key Exchange

- Bootstrap problem: how do Alice, Bob begin?
 - Alice can't send it to Bob in the clear!
- Assume trusted third party, Cathy
 - Alice and Cathy share secret key k_A
 - Bob and Cathy share secret key k_B
- Use this to exchange shared key k_s

4 November 2008

Winsborough CS 5323 Lecture 13

9

Simple Protocol

Alice $\xrightarrow{\{ \text{request for session key to Bob} \}_k$ Cathy

Alice $\xleftarrow{\{ k_s \}_k k_A \parallel \{ k_s \}_k k_B}$ Cathy

Alice $\xrightarrow{\{ k_s \}_k k_B}$ Bob

4 November 2008

Winsborough CS 5323 Lecture 13

10

Problems

- How does Bob know he is talking to Alice?
 - Replay attack: Eve records message from Alice to Bob, later replays it; Bob may think he's talking to Alice, but he isn't
 - Session key reuse: Eve replays message from Alice to Bob, so Bob re-uses session key
- Protocols must provide authentication and defense against replay

4 November 2008

Winsborough CS 5323 Lecture 13

11

Needham-Schroeder

Alice $\xrightarrow{\text{Alice} \parallel \text{Bob} \parallel r_1}$ Cathy

Alice $\xleftarrow{\{ \text{Alice} \parallel \text{Bob} \parallel r_1 \}_k k_s \parallel \{ \text{Alice} \parallel k_s \}_k k_B \}_k k_A}$ Cathy

Alice $\xrightarrow{\{ \text{Alice} \parallel k_s \}_k k_B}$ Bob

Alice $\xleftarrow{\{ r_2 \}_k k_s}$ Bob

Alice $\xrightarrow{\{ r_2 - 1 \}_k k_s}$ Bob

4 November 2008

Winsborough CS 5323 Lecture 13

12

Argument: Alice talking to Bob

- Second message
 - Enciphered using key only she, Cathy knows
 - So Cathy enciphered it
 - Response to first message
 - As r_1 in it matches r_1 in first message
- Third message
 - Alice knows only Bob can read it
 - As only Bob can derive session key from message
 - Any messages enciphered with that key are from Bob

4 November 2008

Winsborough CS 5323 Lecture 13

13

Argument: Bob talking to Alice

- Third message
 - Enciphered using key only he, Cathy know
 - So Cathy enciphered it
 - Names Alice, session key
 - Cathy provided session key, says Alice is other party
- Fourth message
 - Uses session key to determine if it is replay from Eve
 - If not, Alice will respond correctly in fifth message
 - If so, Eve can't decipher r_2 and so can't respond, or responds incorrectly

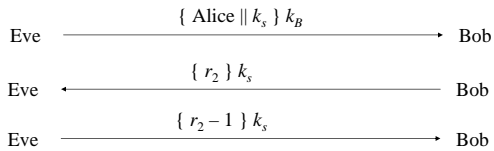
4 November 2008

Winsborough CS 5323 Lecture 13

14

Denning-Sacco Modification

- Assumption: all keys are secret
- Question: suppose Eve can obtain session key. How does that affect protocol?
 - In what follows, Eve knows k_s



4 November 2008

Winsborough CS 5323 Lecture 13

15

Solution

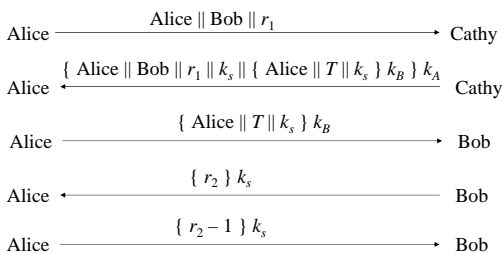
- In protocol above, Eve impersonates Alice
- Problem: replay in third step
 - First in previous slide
- Solution: use time stamp T to detect replay
- Weakness: if clocks not synchronized, may either reject valid messages or accept replays
 - Parties with either slow or fast clocks vulnerable to replay
 - Resetting clock does *not* eliminate vulnerability

4 November 2008

Winsborough CS 5323 Lecture 13

16

Needham-Schroeder with Denning-Sacco Modification



4 November 2008

Winsborough CS 5323 Lecture 13

17

Otway-Rees Protocol

- Corrects problem
 - That is, Eve replaying the third message in the protocol
- Does not use timestamps
 - Not vulnerable to the problems that Denning-Sacco modification has
- Uses integer n to associate all messages with particular exchange

4 November 2008

Winsborough CS 5323 Lecture 13

18

The Protocol

Alice $\xrightarrow{n \parallel \text{Alice} \parallel \text{Bob} \parallel \{ r_1 \parallel n \parallel \text{Alice} \parallel \text{Bob} \} k_A}$ Bob

Cathy $\xleftarrow{n \parallel \text{Alice} \parallel \text{Bob} \parallel \{ r_1 \parallel n \parallel \text{Alice} \parallel \text{Bob} \} k_A \parallel \{ r_2 \parallel n \parallel \text{Alice} \parallel \text{Bob} \} k_B}$ Bob

Cathy $\xrightarrow{n \parallel \{ r_1 \parallel k_s \} k_A \parallel \{ r_2 \parallel k_s \} k_B}$ Bob

Alice $\xleftarrow{n \parallel \{ r_1 \parallel k_s \} k_A}$ Bob

4 November 2008

Winsborough CS 5323 Lecture 13

19

Argument: Alice talking to Bob

- Fourth message
 - If n matches first message, Alice knows it is part of this protocol exchange
 - Cathy generated k_s because only she, Alice know k_A
 - Enciphered part belongs to exchange as r_1 matches r_1 in encrypted part of first message

4 November 2008

Winsborough CS 5323 Lecture 13

20

Argument: Bob talking to Alice

- Third message
 - If n matches second message, Bob knows it is part of this protocol exchange
 - Cathy generated k_s because only she, Bob know k_B
 - Enciphered part belongs to exchange as r_2 matches r_2 in encrypted part of second message

4 November 2008

Winsborough CS 5323 Lecture 13

21

Replay Attack

- Eve acquires old k_s , message in third step
 - $n \parallel \{ r_1 \parallel k_s \} k_A \parallel \{ r_2 \parallel k_s \} k_B$
- Eve forwards appropriate part to Alice
 - Alice has no ongoing key exchange with Bob: n matches nothing, so is rejected
 - Alice has ongoing key exchange with Bob: n does not match, so is again rejected
 - If replay is for the current key exchange, and Eve sent the relevant part *before* Bob did, Eve could simply listen to traffic; no replay involved

4 November 2008

Winsborough CS 5323 Lecture 13

22

Kerberos

- Authentication system
 - Based on Needham-Schroeder with Denning-Sacco modification
 - Central server plays role of trusted third party (“Cathy”)
- Ticket
 - Issuer vouches for identity of requester of service
- Authenticator
 - Identifies sender

4 November 2008

Winsborough CS 5323 Lecture 13

23

Idea

- User u authenticates to Kerberos server
 - Obtains ticket $T_{u,TGS}$ for ticket granting service (TGS)
- User u wants to use service s :
 - User sends authenticator A_u , ticket $T_{u,TGS}$ to TGS asking for ticket for service
 - TGS sends ticket $T_{u,s}$ to user
 - User sends A_u , $T_{u,s}$ to server as request to use s
- Details follow

4 November 2008

Winsborough CS 5323 Lecture 13

24

Ticket

- Credential saying issuer has identified ticket requester
- Example ticket issued to user u for service s

$$T_{u,s} = s \parallel \{ u \parallel u\text{'s address} \parallel \text{valid time} \parallel k_{u,s} \} k_s$$
 where:
 - $k_{u,s}$ is session key for user and service
 - Valid time is interval for which ticket valid
 - u 's address may be IP address or something else
 - Note: more fields, but not relevant here

4 November 2008

Winsborough CS 5323 Lecture 13

25

Authenticator

- Credential containing identity of sender of ticket
 - Used to confirm sender is entity to which ticket was issued
- Example: authenticator user u generates for service s

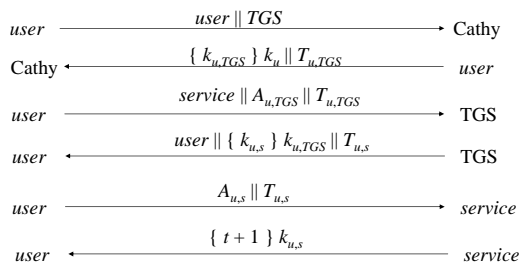
$$A_{u,s} = \{ u \parallel \text{generation time} \parallel k_t \} k_{u,s}$$
 where:
 - k_t is alternate session key
 - Generation time is when authenticator generated
 - Note: more fields, not relevant here

4 November 2008

Winsborough CS 5323 Lecture 13

26

Protocol



4 November 2008

Winsborough CS 5323 Lecture 13

27

Analysis

- First two steps get user ticket to use TGS
 - User u can obtain session key only if u knows key shared with Cathy
- Next four steps show how u gets and uses ticket for service s
 - Service s validates request by checking sender (using $A_{u,s}$) is same as entity ticket issued to
 - Step 6 optional; used when u requests confirmation

4 November 2008

Winsborough CS 5323 Lecture 13

28

Problems

- Relies on synchronized clocks
 - If not synchronized and old tickets, authenticators not cached, replay is possible
- Tickets have some fixed fields
 - Dictionary attacks possible
 - Kerberos 4 session keys weak (had much less than 56 bits of randomness); researchers at Purdue found them from tickets in minutes

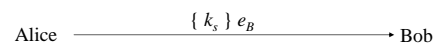
4 November 2008

Winsborough CS 5323 Lecture 13

29

Public Key Key Exchange

- Here interchange keys known
 - e_A, e_B Alice and Bob's public keys known to all
 - d_A, d_B Alice and Bob's private keys known only to owner
- Simple protocol
 - k_s is desired session key



4 November 2008

Winsborough CS 5323 Lecture 13

30

Problem and Solution

- Vulnerable to forgery or replay
 - Because e_B known to anyone, Bob has no assurance that Alice sent message
- Simple fix uses Alice's private key
 - k_s is desired session key

Alice $\xrightarrow{\{\{k_s\} d_A\} e_B}$ Bob

4 November 2008

Winsborough CS 5323 Lecture 13

31

Notes

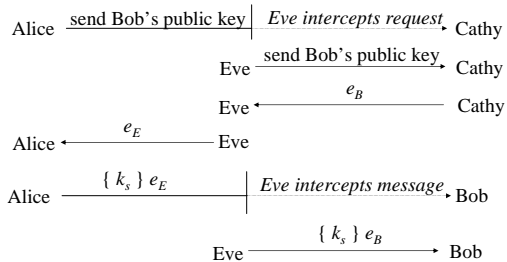
- Can include message enciphered with k_s
- Assumes Bob has Alice's public key, and *vice versa*
 - If not, each must get it from public server
 - If keys not bound to identity of owner, attacker Eve can launch a *man-in-the-middle* attack (next slide; Cathy is public server providing public keys)
 - Solution to this (binding identity to keys) discussed later as public key infrastructure (PKI)

4 November 2008

Winsborough CS 5323 Lecture 13

32

Man-in-the-Middle Attack



4 November 2008

Winsborough CS 5323 Lecture 13

33