CSEC
CYBER SECURITY
EDUCATION CONSORTIUM

0100101010010101010101010011001010101010000101010101010101010100010100100

Cryptographic Protocols

Sujeet Shenoi
Tandy School of Computer Science
University of Tulsa, Tulsa, OK 74104
sujeet@utulsa.edu

ADVANCED
TECHNOLOGICAL
EDUCATION

Cryptographic Protocols

0100101010010101010101010011001010101010000101010101010101010100010100100

Cryptographic Protocol

- Orderly sequence of steps to achieve certain security properties
- Established in advance
- Mutually subscribed
- Unambiguous
- Complete

Kinds of Protocols

01001010100101010101010011001010101010000101010101010101010100010100100

Arbitrated Protocols

- Trusted third party participates in each transaction
- Expensive, slow, vulnerable

Adjudicated Protocols

- Third party judges fairness after the fact
- Address disadvantages of arbitrated protocols
- Detect failure only after the fact

Self-Enforcing Protocols

- Guarantee fairness; cheating is immediately obvious
- May not exist for every situation

Applications

010010101001010101010101001100101010101000010101010101010101010100010100100

- Key Distribution
- Certificates
- Digital Signatures
- Clipper Key Exchange
- Mental Poker
- Oblivious Transfer
- Contract Signing
- Certified Mail

Key Distribution Protocols

1. Symmetric Key Exchange (without Server)
2. Symmetric Key Exchange (with Server)
3. Asymmetric Key Exchange (without Server)
4. Asymmetric Key Exchange (with Server)

Symmetric Key Exchange (No Server)

Requirements

- A & B share K_M (Master Key)

Protocol

1. A: Generates session key: K_S
2. $A \rightarrow B$: $\{ K_S \}_{K_M}$

Symmetric Key Exchange (Server)

0100101010010101010101010011001010101010000101010101010101010100010100100

Needham-Schroeder Protocol

1. $A \rightarrow S: \quad A \bullet B \bullet I_A$ (I_A: Unique session ID)
2. $S \rightarrow A: \quad \{ I_A, B, K_{AB}, \{ K_{AB}, A \}_{K_B} \}_{K_A}$ (new K_{AB})
3. $A \rightarrow B: \quad \{ K_{AB}, A \}_{K_B}$

Asymmetric Key Exchange (No Server)

Important Point

- Asymmetric key encryption is very expensive
- Never encrypt message; transmit encrypted symmetric key

Protocol

1. $A \rightarrow B$: $\{ \{ K_{AB} \}_{K_A^{priv}} \}_{K_B^{pub}}$

2a. $B \rightarrow A$: $\{ n \}_{K_{AB}}$ (n: Random number)

2b. $A \rightarrow B$: $\{ n + 1 \}_{K_{AB}}$

Asymmetric Key Exchange (Server)

Protocol

1. $A \rightarrow S$: $A \bullet B$
2. $S \rightarrow A$: $\{ K_B^{pub}, B \}_{K_S^{priv}}$
3. $A \rightarrow B$: $\{ A, I_A \}_{K_B^{pub}}$ (I_A: message reference)
4. $B \rightarrow S$: $B \bullet A$
5. $S \rightarrow B$: $\{ K_A^{pub}, A \}_{K_S^{priv}}$
6. $B \rightarrow A$: $\{ I_A, I_B \}_{K_A^{pub}}$ (I_B: message reference)
7. $A \rightarrow B$: $\{ \underline{K_{AB}}, I_B \}_{K_B^{pub}}$ ($\underline{K_{AB}}$: message)

Digital Certificates

010010101001010101010101001100101010101000010101010101010101010100010100100

Binding an Individual's Identity and Public Key

- A: President and CEO
- B, C: Vice Presidents
- C supervises D and E

- B's Certificate: $\{ ID_B, Pos_B, K_B^{pub}, H(.) \}_{K_A{}^{priv}}$
- C's Certificate: $\{ ID_C, Pos_C, K_C^{pub}, H(.) \}_{K_A{}^{priv}}$
- D's Certificate: $\{ ID_D, Pos_D, K_D^{pub}, H(.) \}_{K_C{}^{priv}} \bullet$ C's Certificate
- Always need a top-level authority

Digital Signature Protocols

0100101010010101010101010011001010101010000101010101010101010100010100100

Goals

- Non Forgeable [M, sig(M, P)]
- Authentic
- Non Alterable; Non Reusable
- Non Repudiation

- Symmetric Key Protocol (Arbiter (A) needed)
- Cryptographic Sealing (Arbiter (A) needed)
- Asymmetric Key Protocol (Self-Enforcing)

Digital Signature (Symmetric Key)

0100101010010101010101010011001010101010000101010101010101010100010100100

Symmetric Key Protocol

(Non Forgeability, Authenticity, Non Repudiation)

- $S \rightarrow A$: $\{ M \}_{K_S}$
- $A \rightarrow R$: $\{ M, S, \{ M \}_{K_S} \}_{K_R}$
- R: Unlocks with K_R
  Saves M and $\{ M \}_{K_S}$

Cryptographic Sealing (Digital Signature)

0100101010010101010101010011001010101010000101010101010101010100010100100

Cryptographic Sealing Function

- $f(M) \rightarrow$ unique value
- S and R register f_S and f_R with Arbiter A

Protocol (No Secrecy)

1. $S \rightarrow A$: $M \bullet f_S(M)$
2. A: Recomputes $f_S(M)$
   Compares with $f_S(M)$ received from S
3. $A \rightarrow R$: $M \bullet S \bullet f_S(M) \bullet f_R(M, S)$

Digital Signature (Asymmetric Key)

Protocol 1
(Non Forgeability, Authenticity, Non Repudiation)

- $S \rightarrow R$: $\{ M \}_{K_S^{priv}}$
- R: $\{ \{ M \}_{K_S^{priv}} \}_{K_S^{pub}} \equiv M$ (saves $\{ M \}_{K_S^{priv}}$)

Protocol 2 (Double Encryption: Privacy)

- $S \rightarrow R$: $\{ \{ M \}_{K_S^{priv}} \}_{K_R^{pub}}$
- R: $\{ \{ \{ \{ M \}_{K_S^{priv}} \}_{K_R^{pub}} \}_{K_S^{pub}} \}_{K_R^{priv}} \equiv M$

Diffie-Hellman Key Exchange

S & R: Agree on a large prime p (1024+ bits)

S & R: Agree on a generator g mod p

S & R: Choose private numbers x (S) & y (R)

<u>Step 1</u>

$S \rightarrow R$: $g^x \bmod p$

<u>Step 2</u>

$R \rightarrow S$: $g^y \bmod p$

<u>Step 3</u>

S: Computes $K = (g^y)^x \bmod p$

R: Computes $K = (g^x)^y \bmod p$

Intruder cannot compute K even with p, g, g^x, g^y

Exponentiation (easy); Discrete Logarithm (hard)

Clipper Key Exchange

0100101010010101010101010011001010101010000101010101010101010100010100100

Seven Step Protocol

- Three steps for key exchange
- Four steps for mutual authentication
- S & R share secret key K_P, symmetric algorithm and asymmetric algorithm

Clipper Key Exchange (contd.)

0100101010010101010101010011001010101010000101010101010101010100010100100

Step 1

$S \rightarrow R$: $\{ K_S^{pub} \}_{K_P}$

Step 2

R: Uses K_P to obtain K_S^{pub}

R: Chooses random session key K_k

$R \rightarrow S$: $\{ \{ K_k \}_{K_P} \}_{K_S^{pub}}$

Step 3

S : Uses K_S^{priv} and K_P to obtain K_k

Clipper Authentication (contd.)

0100101010010101010101010011001010101010000101010101010101010100010100100

Step 4

$S \rightarrow R$: $\{M\}_{K_k}$ M: Random string

Step 5

R: Uses K_k to obtain M

$R \rightarrow S$: $\{M, N\}_{K_k}$ N: Random string

Step 6

S : Uses K_k to obtain M & N; Checks M

$S \rightarrow R$: $\{N\}_{K_k}$

Step 7

R : Uses K_k to obtain N; Checks N

Mental Poker Protocol (Symmetric Key)

01001010100101010101010100110010101010100001010101010101010101000101001 00

- $A \rightarrow B$: $\{ m_1 \}_{K_A}$ … $\{ m_{10} \}_{K_A}$
- B: Locks 5 messages: $\{ \{ m_j \}_{K_A} \}_{K_B}$ …
- $B \rightarrow A$: $\{ m_i \}_{K_A}$ … & $\{ \{ m_j \}_{K_A} \}_{K_B}$ …
- A: Unlocks all 10 messages with K_A
  Keeps 5 messages: $\{ \{ m_i \}_{K_A} \}_{K_A}$ … $= m_i$ …
- $A \rightarrow B$: $\{ \{ \{ m_j \}_{K_A} \}_{K_B} \}_{K_A}$ … $= \{ m_j \}_{K_B}$ …
- B: Unlocks all 5 messages with K_B
  Keeps 5 messages: $\{ \{ m_j \}_{K_B} \}_{K_B}$ … $= m_j$ …

Mental Poker Protocol (Asymmetric Key)

- $A \rightarrow B$: $\{ m_1 \}_{K_A}{}^{pub}$ … $\{ m_{10} \}_{K_A}{}^{pub}$
- B: Locks 5 messages: $\{ \{ m_j \}_{K_A}{}^{pub} \}_{K_B}{}^{pub}$ …
- $B \rightarrow A$: $\{ m_i \}_{K_A}{}^{pub}$ … & $\{ \{ m_j \}_{K_A}{}^{pub} \}_{K_B}{}^{pub}$ …
- A: Unlocks all 10 messages with $K_A{}^{priv}$
  Keeps 5 msgs: $\{ \{ m_i \}_{K_A}{}^{pub} \}_{K_A}{}^{priv}$ … = m_i …
- $A \rightarrow B$: $\{ \{ \{ m_j \}_{K_A}{}^{pub} \}_{K_B}{}^{pub} \}_{K_A}{}^{priv}$ … = $\{ m_j \}_{K_B}{}^{pub}$ …
- B: Unlocks all 5 messages with $K_B{}^{priv}$
  Keeps 5 msgs: $\{ \{ m_j \}_{K_B}{}^{pub} \}_{K_B}{}^{priv}$ … = m_j …

Oblivious Transfer Protocol

CSEC CYBER SECURITY EDUCATION CONSORTIUM

01001010100101010101010011001010101010000101010101010101010100010100100

Eight Step Protocol (Flipping a coin at a distance)

1. A: Picks asymmetric key pairs: (K_I^{priv}, K_I^{pub}) (K_J^{priv}, K_J^{pub})
2. B: Picks symmetric key: K_B
3. $A \rightarrow B$: $K_I^{pub} \bullet K_J^{pub}$
4. B: Picks one key at random: K_H^{pub}
   $B \rightarrow A$: $\{K_B\}_{K_H^{pub}}$
5. A: Picks I or J at random (say: J)
   Computes: $K_A = \{\{K_B\}_{K_H^{pub}}\}_{K_J^{priv}}$ $(K_A = K_B \text{ if } H = J)$
6. $A \rightarrow B$: $\{A \text{ loses}\}_{K_A} \bullet J$
7. B: $M = \{\{A \text{ loses}\}_{K_A}\}_{K_B}$
   $B \rightarrow A$: $M \bullet H$ (B loses if $H \neq J$)
8. $A \rightarrow B$: $K_I^{priv} \bullet K_J^{priv}$ (for verification)

ADVANCED TECHNOLOGICAL EDUCATION

Contract Signing Protocol

1. A: Selects 2n symmetric keys: $C_1 .. C_{2n}$
   Arranges them in pairs: (C_i, C_{n+i}) $i = 1 .. n$
2. $A \rightarrow B$: $X_i = \{ S \}_{C_i}$ $i = 1 .. 2n$ (S = Std Msg; X_i: S-puzzle)
3. A: Agrees to contract if B produces a pair (C_i, C_{n+i}) for any i (S-puzzle solution)
4. B: Repeats Steps 1-3: keys: D_i and S-puzzles: Δ_i
5. $A \rightarrow B$: Exchange (C_i, C_{n+i}) … by Oblivious Transfer Protocol
   $B \rightarrow A$: Exchange (D_i, D_{n+i}) … by Oblivious Transfer Protocol
6. For $j = 1 .. \text{keylength}$:
   $A \rightarrow B$: j^{th} bit of C_i $i = 1 .. 2n$
   $B \rightarrow A$: j^{th} bit of D_i $i = 1 .. 2n$

Certified Mail Protocol

CSEC CYBER SECURITY EDUCATION CONSORTIUM

0100101010010101010101010011001010101010000101010101010101010100010100100

1. A: Selects n + 1 symmetric keys: g_0 .. g_n

   Computes: $g_{n+i} = g_0 \oplus g_i$ $i = 1 .. n$

2. $A \rightarrow B$: $G = \{ M \}_{g_0}$ ($g_0 = g_{n+i} \oplus g_i$ for all i)

3. $A \rightarrow B$: $G_i = \{ SA \}_{g_i}$ $i = 1 .. 2n$ (SA = Std Msg)

4. B: Selects 2n symmetric keys: h_1 .. h_{2n}

   $B \rightarrow A$: $H_i = \{ SB \}_{h_i}$ $i = 1 .. 2n$ (SB = Std Msg)

5. B: Agrees to acknowledge receipt of plaintext of G if A can produce any one of (h_i, h_{n+i}) and all g_j ($j = 1 .. 2n$)

6. $A \rightarrow B$: Exchange (g_i, g_{n+i}) … by Oblivious Transfer Protocol

   $B \rightarrow A$: Exchange (h_i, h_{n+i}) … by Oblivious Transfer Protocol

7. For $j = 1$ .. keylength:

   $A \rightarrow B$: j^{th} bit of g_i $i = 1 .. 2n$

   $B \rightarrow A$: j^{th} bit of h_i $i = 1 .. 2n$

NSF

ADVANCED TECHNOLOGICAL EDUCATION