COMP7120/8120 Cryptography and Data Security

Authentication Protocols
Authentication Handshakes

• Secure communication almost always includes an initial authentication handshake.
  - Authenticate each other
    • Based on cryptographic primitives
  - Establish session keys
  - *This process is not trivial; flaws in this process undermine secure communication*
    • Cryptographic primitives being secure is not equivalent to the design based on them being secure.
Authentication with Shared Secret

- Assumptions:
  - Shared key: $K_{\text{Alice-Bob}}$
  - $R$ is a random number

- Questions:
  - Can Alice prove to Bob that she is indeed Alice?
  - Can Bob prove that he is indeed Bob?
Authentication with Shared Secret (Cont’d)

- Questions:
  - Can Alice prove to Bob that she is indeed Alice?
  - Can Bob prove that he is indeed Bob?
  - Other potential vulnerability?
Authentication with Public Key

- Alice and Bob know each other’s public key

- Questions:
  - Can Alice prove to Bob that she is indeed Alice?
  - Can Bob prove that he is indeed Bob?
Authentication with Public Key (Cont’d)

- Questions:
  - Can Alice prove to Bob that she is indeed Alice?
  - Can Bob prove that he is indeed Bob?
  - Other potential vulnerability?
Mutual Authentication

Alice

I’m Alice

\[ R_1 \]

\[ H(K_{Alice-Bob}, R_1) \]

\[ R_2 \]

\[ H(K_{Alice-Bob}, R_2) \]

Bob

Optimize

Alice

I’m Alice, \[ R_2 \]

\[ R_1, H(K_{Alice-Bob}, R_2) \]

\[ H(K_{Alice-Bob}, R_1) \]

Bob
Mutual Authentication (Cont’d)

• Reflection attack

Step 1:
Trudy: I’m Alice, \( R_2 \)

\[ R_1, H(K_{\text{Alice-Bob}}, R_2) \]

Bob

Step 3:
\[ H(K_{\text{Alice-Bob}}, R_1) \]

Trudy

Step 2:
Trudy: I’m Alice, \( R_1 \)

\[ R_3, H(K_{\text{Alice-Bob}}, R_1) \]

Bob
Reflection Attacks (Con’t’d)

• Lesson: Don’t have Alice and Bob do exactly the same thing
  - Different keys
    • Totally different keys
    • $K_{\text{Alice-Bob}} = K_{\text{Bob-Alice}} + 1$
  - Different Challenges: Alice and Bob’s challenges cannot be the same
  - The initiator should be the first to prove its identity
    • Assumption: initiator is more likely to be the bad guy
Mutual Authentication (Cont’d)

I’m Alice, \( R_2 \)

\( R_1, H(K_{Alice-Bob}, R_2) \)

\( H(K_{Alice-Bob}, R_1) \)

Bob

Countermeasure: Alice proves herself first

I’m Alice

\( R_1 \)

\( H(K_{Alice-Bob}, R_1), R_2 \)

\( H(K_{Alice-Bob}, R_2) \)

Bob
Mutual Authentication (Cont’d)

- Public keys
  - Authentication of public keys is a critical issue

- Question:
  - will reflection attack still work?
  - Other potential vulnerability?
Better Design

- Provide mutual authentication
- Make two parties do different things
- Challenge the initiators first
- Avoid reflection attacks
- Avoid message decryption
Integrity/Encryption for Data

- Communication after mutual authentication should be cryptographically protected as well
  - Require a **session key** established during mutual authentication
Establishment of Session Keys

- Secret key based authentication
  - Assume the following authentication happened.
  - Can we use $K_{Alice-Bob}\{R\}$ as the session key?
  - Can we use $K_{Alice-Bob}\{R+1\}$ as the session key?
  - Can we use $K_{Alice-Bob+1}\{R\}$ as the session key?
  - In general, modify $K_{Alice-Bob}$ and encrypt $R$. Use the result as the session key.

![Diagram](image-url)
Establishment of Session Keys

- Public key based authentication
  - RSA based key negotiation

Alice

1) Get R1,
2) Generate random number R2,
3) get key as $K = H(R1 \oplus R2)$,

Send R1 encrypted using $K_b,p$

Generate random number R1

Public key: $K_a,p$
Private key: $K_a,i$

Send R2 encrypted using $K_a,p$

Bob

1) get R2
2) get symmetric key as $K = H(R1 \oplus R2)$,

Public key: $K_b,p$
Private key: $K_b,i$

1) get R1,
Establishment of Session Keys

- Public key based authentication
  - Diffie-Hellman negotiation
    - Alice and Bob signs the quantity they send

Alice
- Generate random number $S_A$
- Compute, sign and send $T_A = g^{S_A} \mod p$
- Compute $T_B^{S_A} \mod p = g^{S_A S_B} \mod p$

Bob
- Generate random number $S_B$
- Compute $T_A^{S_B} \mod p = g^{S_A S_B} \mod p$
- Compute $T_B = g^{S_B} \mod p$

Public key: $K_a, p$
Private key: $K_a, i$

Public key: $K_b, p$
Private key: $K_b, i$
Two-Way Public Key Based Authentication

• Approach I
  - Alice chooses and encrypts $R_1$ with Bob’s public key
  - Bob chooses and encrypts $R_2$ with Alice’s public key
  - Session key is $H(R_1 \oplus R_2)$
  - Trudy will have to compromise both Alice and Bob

• Approach II
  - Alice and Bob establish the session key with Diffie-Hellman key exchange
  - Alice and Bob signs the quantity they send
Summary

• Design a perfect authentication protocol requires non-trivial efforts
  - Can be based on symmetric or public key systems
  - The initiators should authenticate themselves first
  - Need asymmetric challenge-response, be aware of reflection attacks

• Design based on public key:
  • RSA key negotiation
  • Diffie-Hellman with authentication