CS 556 – Computer Security
Fall 2012

Dr. Indrajit Ray
Email: indrajit@cs.colostate.edu

Department of Computer Science
Colorado State University
Fort Collins, CO 80523, USA
Public Key Cryptosystem

- Traditional secret key cryptography uses a single key shared by both sender and receiver
  - Problem – How to share the key?
- Does not protect sender from the receiver forging a message & claiming that message is sent by sender, the two parties being equal
- Solution - public key / asymmetric key cryptosystem
Goal of Public Key Crypto

● “... two parties communicating solely over a public channel and using only publicly known techniques can create a secure connection.”

Public Key Cryptosystem

- Public-key (or two-key or asymmetric key) cryptography involves the use of two keys:
  - a public-key, which may be known by anybody, and can be used to encrypt messages, and verify signatures
  - a private-key, known only to the recipient, used to decrypt messages, and create signatures
Public Key Cryptosystem

- First conceived by Diffie and Hellman in 1976
- Rivest, Shamir and Adleman were first to describe a public key system in 1978
- Merkle and Hellman published a different solution in 1978
- Many proposals have been broken (including the Merkle-Hellman proposal)
- Serious candidates today
  - RSA
  - El Gamal
Three important classes of public-key algorithms:

- Public-Key Distribution Schemes (PKDS) - used to securely exchange a single piece of information that is then used as a session key for a secret-key scheme
- Public Key Schemes (PKS) - used for encryption, where the public-key encrypts and the private-key decrypts messages
- Signature Schemes - used to create a digital signature, where the private-key creates and the public-key verifies signatures
● Any public-key scheme can be used as a PKDS, just by selecting a message which is the required session key

● Many public-key schemes are also signature schemes (provided encryption & decryption can be done in either order)
Public Key Cryptosystem

Alice
Message Source → Encrypt M with Bob’s Public Key
C = E[M, K_E]

Insecure Communications Channel
C → Decrypt C with Bob’s Private Key
M = D[C, K_D]

Bob
Message Destination

Key Source Provides Bob’s Public Key

Reliable Key Channel

Cryptanalyst

Key Source Generates Key Pair

Dr. Indrajit Ray, Computer Science Department
Properties of Public/Private Keys

- Public / Private key pairs, \((K_E, K_D)\) are easy to generate (a polynomial time (P-time) problem)
- \(D[E[M, K_E], K_D] = M\)
- \(D[E[M, K_D], K_E] = M\)
- \(K_E \neq K_D\)
- \(D[E[M, K_E], K’] = M \rightarrow K_D = K’\)
Properties of Public/Private Keys

- The keys $K_E$ and $K_D$ are mathematically related.
- A plaintext encrypted with $K_E$ can be decrypted only with $K_D$ and vice versa.
  - Does not matter which one you choose to call public key / private key.
Property of the Keys

- Knowing the public-key and public description of the cipher, it is
  - Computationally infeasible to compute the private key \((K_D)\)
    (an NP-time problem)
  - Thus the public-key may be distributed to anyone wishing to communicate securely with its owner
  - Although proper distribution of the public-key is a non-trivial problem - the key distribution problem
USING PUBLIC KEY CRYPTOGRAPHY
Confidentiality For Small Payload (One Way)

- Look up the recipient’s public key
- Encrypt the payload with the recipient’s public key and send on an unprotected network
- The recipient will receive and decrypt the message with their private key
Confidentiality For Small Payload (Two Way)

- Look up the recipient’s public key
- Encrypt the following with the recipient’s public key and send on an unprotected network
  - The payload
  - Your public key
- The recipient will receive and decrypt the message and your public key with their private key and encrypt their response with your public key
- When you receive the response, decrypt it with your private key

1Could have recipient look up originator's public key
Confidentiality For A Session

- Look up the recipient’s public key
- Encrypt the following with the recipient’s public key and send on an unprotected network:
  - Your public key
  - Your part of a session key
- The recipient will decrypt the message, combine your session key part with their session key part, and encrypt this with your public key
- When you receive the response, decrypt it with your private key and begin the private key session
Digital Signatures

Alice

Message Source -> Decrypt M with Alice’s Private Key: $C = D[M, K_D]$

Reliable Communications Channel

C -> Encrypt C with Alice’s Public Key: $M = E[C, K_E]$

Bob

Key Source Provides Alice’s Public Key

Message Destination

Key Source Generates Alice’s Private Key

$K_D$ -> Reliable Key Channel

$K_E$ -> Reliable Key Channel

Source

Message

Destination
A signature is a non repudiable proof assuming that a one-to-one mapping can be established between a public key and its owner.

In general non-repudiation requires a notarized signature, involving a third party, that vouches for the one-to-one mapping between a public key and its owner.

In large systems this can involve hierarchies of notarization.