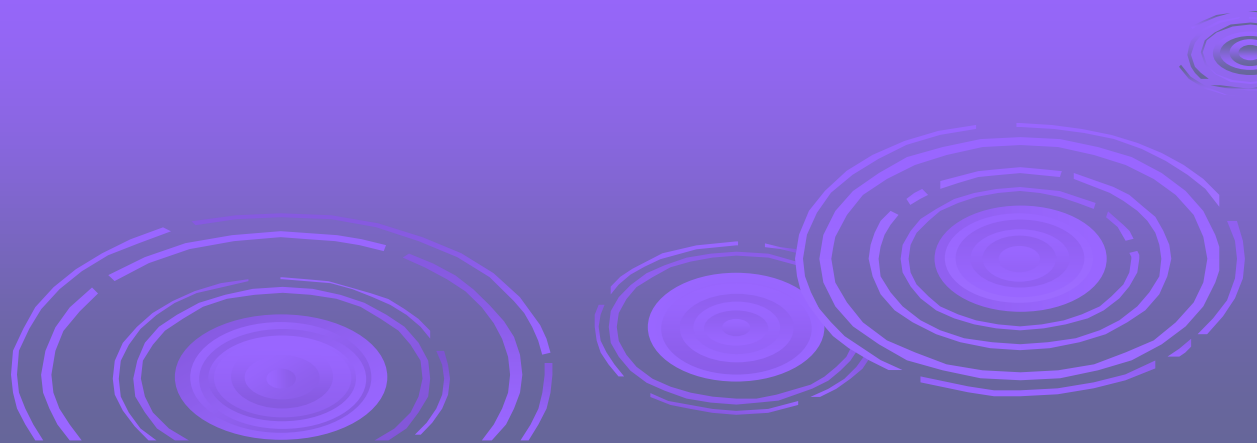
Cryptography and Network Security Chapter 14

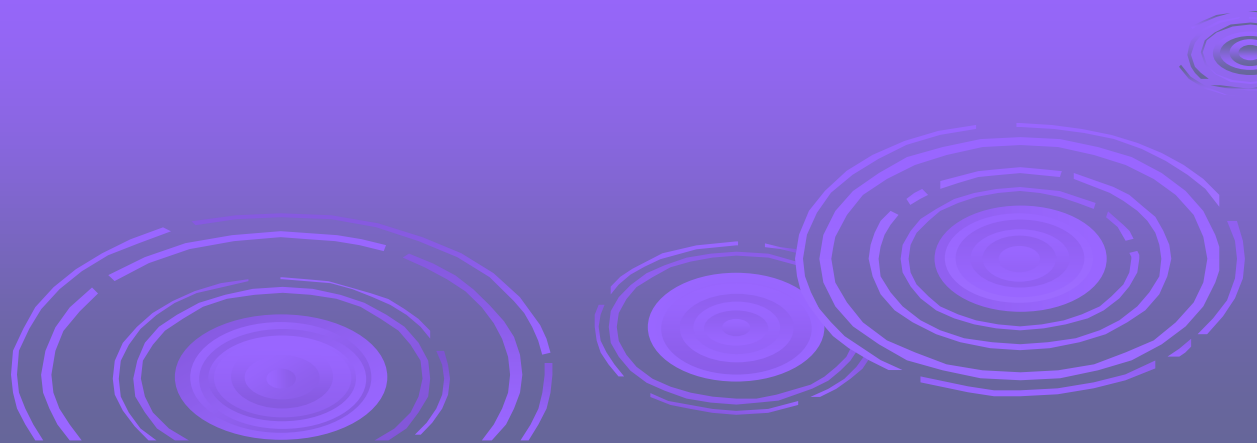


Key Management and Distribution

- topics of cryptographic key management / key distribution are complex
  - cryptographic, protocol, & management issues
- symmetric schemes require both parties to share a common secret key
- public key schemes require parties to acquire valid public keys
- have concerns with doing both

Road Map

- symmetric key distribution using symmetric encryption
- symmetric key distribution using public-key encryption



Key Distribution

- symmetric schemes require both parties to share a common secret key
- issue is how to securely distribute this key
- whilst protecting it from others
- frequent key changes can be desirable
- often secure system failure due to a break in the key distribution scheme

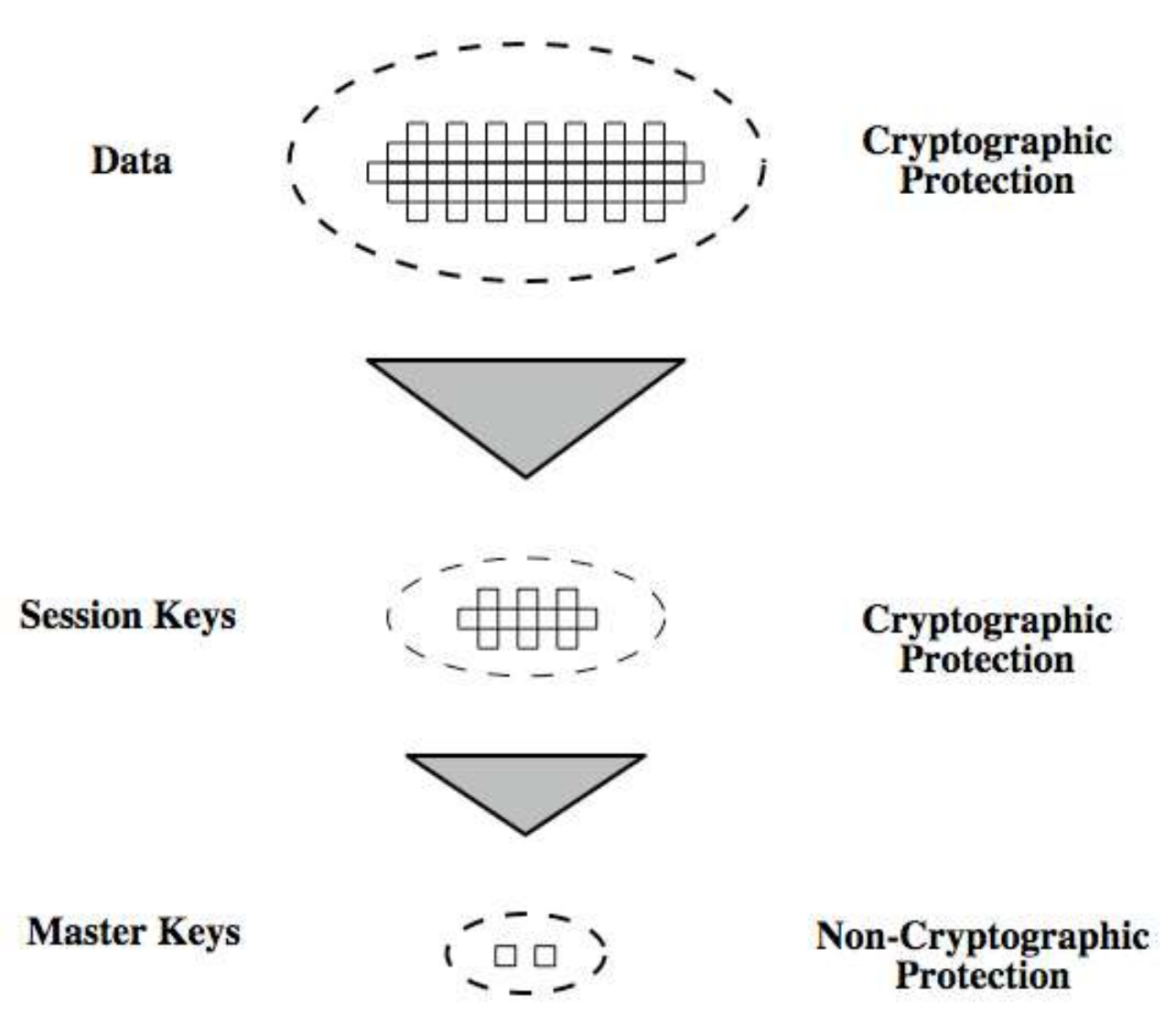
Key Distribution

- given parties A and B have various **key distribution** alternatives:
  1. A can select key and physically deliver to B
  2. third party can select & deliver key to A & B
  3. if A & B have communicated previously can use previous key to encrypt a new key
  4. if A & B have secure communications with a third party C, C can relay key between A & B

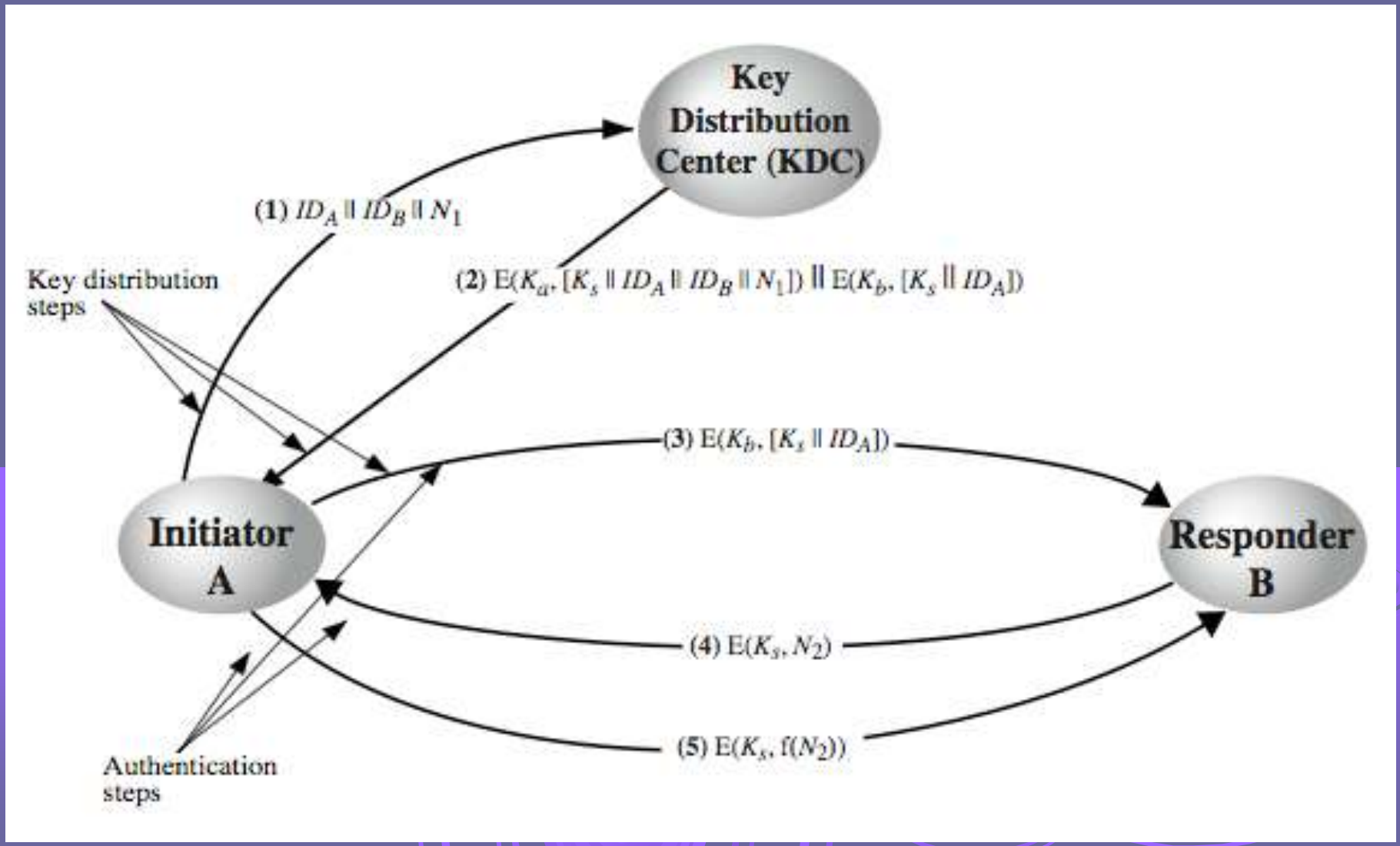
Key Hierarchy

- typically have a hierarchy of keys
- session key
  - temporary key
  - used for encryption of data between users
  - for one logical session then discarded
- master key
  - used to encrypt session keys
  - shared by user & key distribution center

Key Hierarchy



Key Distribution Scenario



Key Distribution Issues

- hierarchies of KDC's required for large networks, but must trust each other
- session key lifetimes should be limited for greater security
- use of automatic key distribution on behalf of users, but must trust system
- use of decentralized key distribution
- controlling key usage

Hierarchical Key Control

- It is not necessary to limit the key distribution function to a single KDC.
- a hierarchy of KDCs can be established.
- there can be local KDCs, each responsible for a small domain of the overall internetwork
- A hierarchical scheme minimizes the effort involved in master key distribution, because most master keys are those shared by a local KDC with its local entities.

Session Key Lifetime

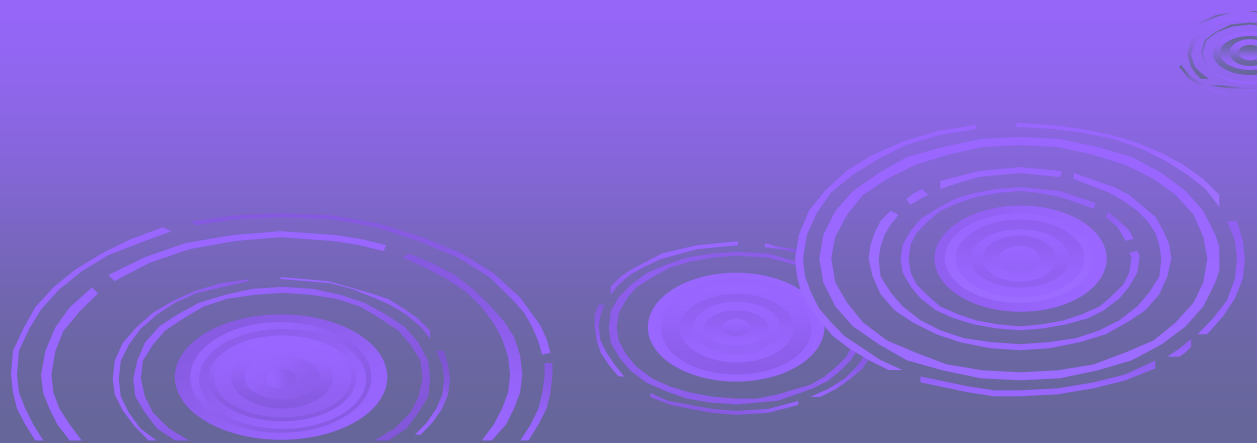
- For connection-oriented protocols, one obvious choice is to use the same session key for the length of time that the connection is open, using a new session key for each new session.
- For a connectionless protocol, such as a transaction-oriented protocol, there is no explicit connection initiation or termination. Thus, it is not obvious how often one needs to change the session key.

Road Map

- symmetric key distribution using symmetric encryption
- symmetric key distribution using public-key encryption
- distribution of public keys
  - announcement, directory, authrority, CA
- X.509 authentication and certificates
- public key infrastructure (PKIX)

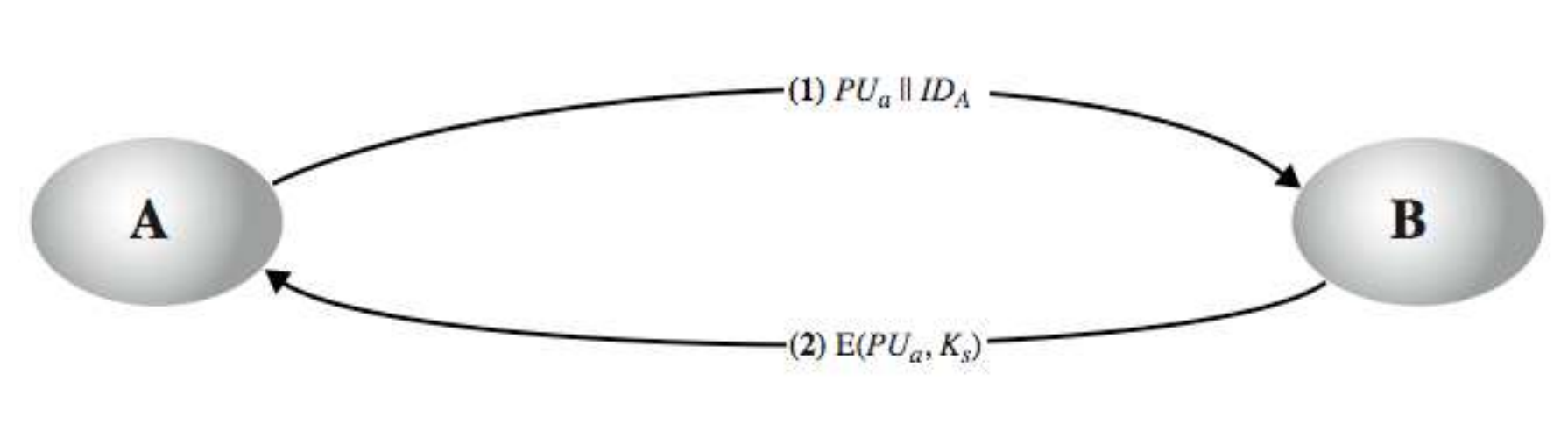
Symmetric Key Distribution Using Public Keys

- public key cryptosystems are inefficient
  - so almost never use for direct data encryption
  - rather use to encrypt secret keys for distribution



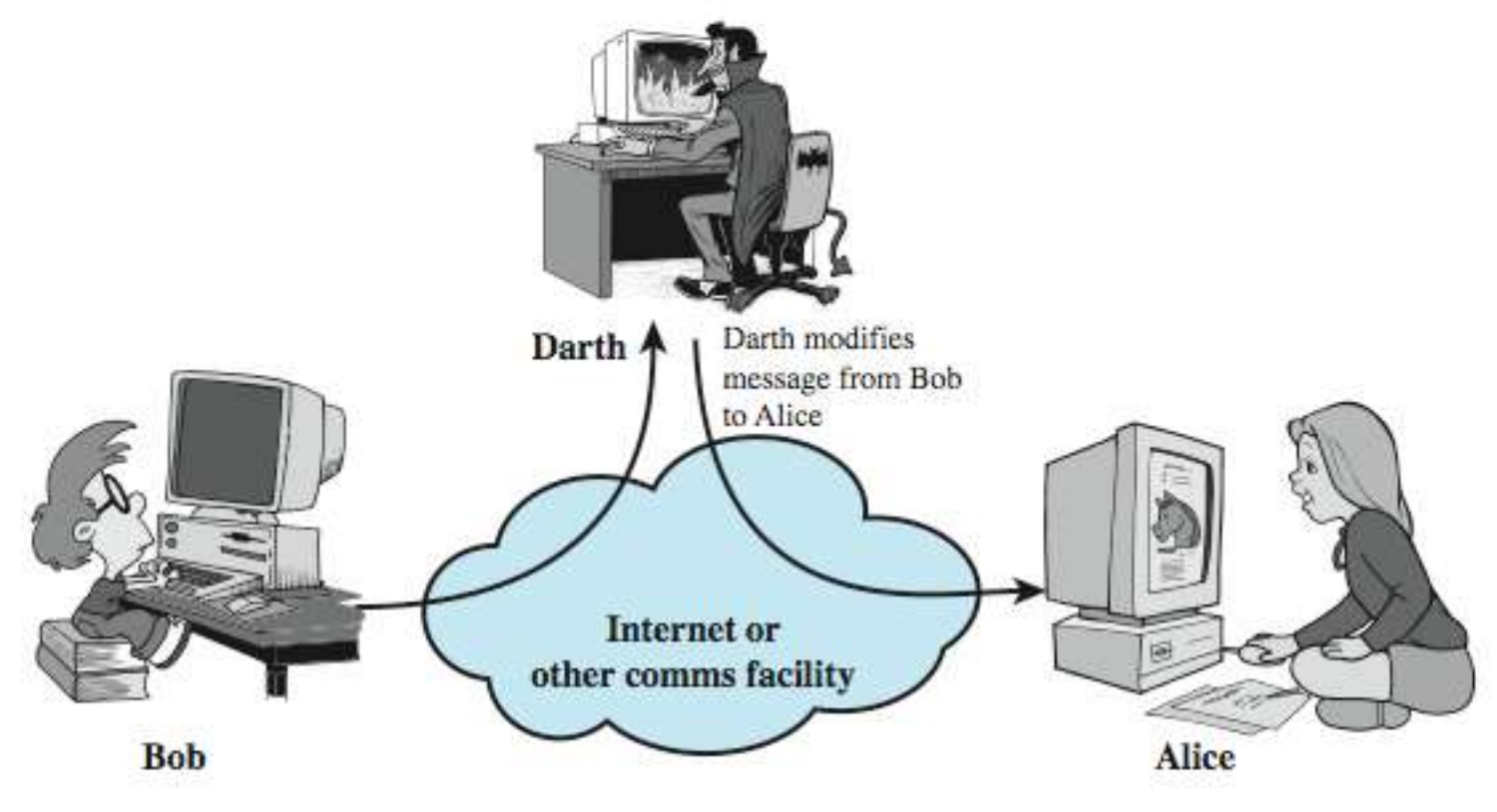
Simple Secret Key Distribution

- Merkle proposed this very simple scheme
  - allows secure communications
  - no keys before/after exist

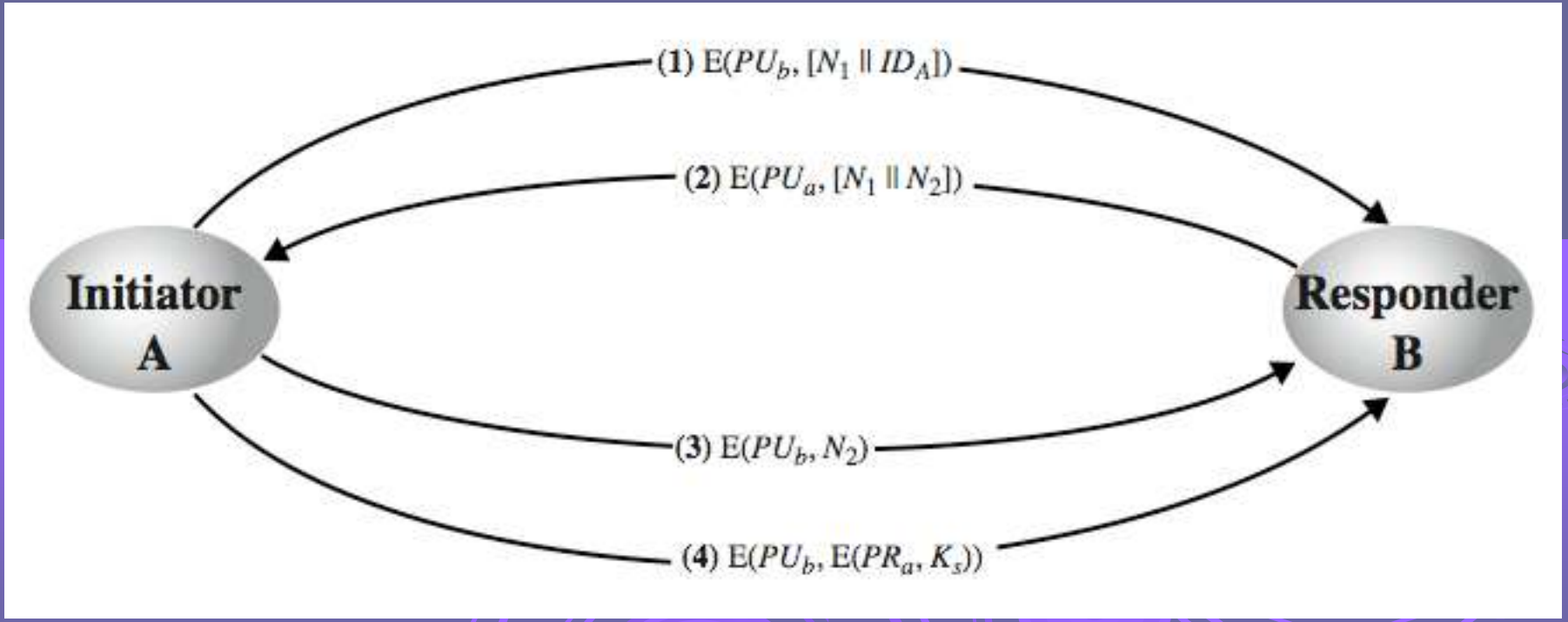


Man-in-the-Middle Attack

- this very simple scheme is vulnerable to an active man-in-the-middle attack



Secret Key Distribution with Confidentiality and Authentication

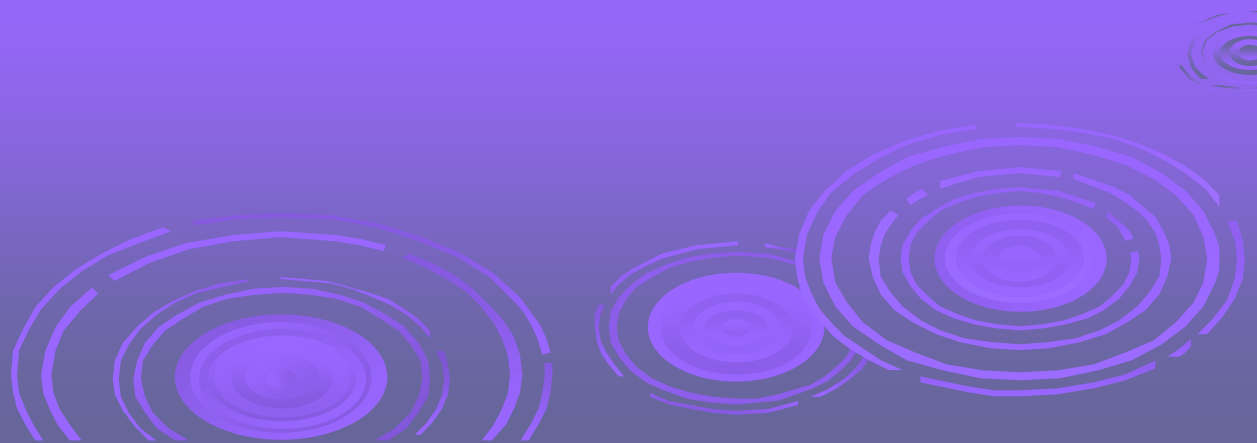


Hybrid Key Distribution

- retain use of private-key KDC
- shares secret master key with each user
- distributes session key using master key
- public-key used to distribute master keys
  - especially useful with widely distributed users
- rationale
  - performance
  - backward compatibility

Summary

- have considered:
  - symmetric key distribution using symmetric encryption
  - symmetric key distribution using public-key encryption



MODULE 4

Key Management and Distribution

* Key Distribution

- Symmetric Schemes require both parties to share a common secret key.
- Issue is how to securely distribute this key.
- whilst protecting it from others
- frequent key changes can be desirable
- often secure system failure due to a break in the key distribution scheme

* Key Hierarchy

- typically have a hierarchy of keys
- **Session key**
  - temporary key
  - used for encryption of data b/w users
  - for one logical session then discarded
- **Master key**
  - used to encrypt session keys
  - shared by user & key distribution center

[Dig refer ppt]

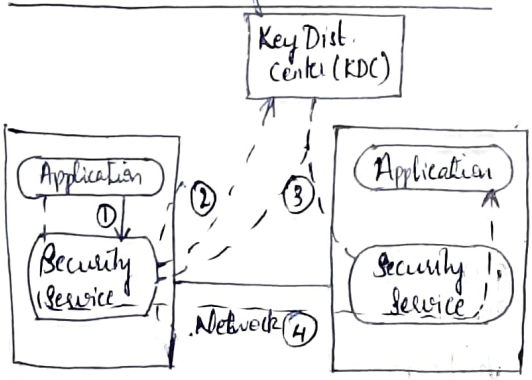
* Key Distribution Scenario

Refer ppt (dig)

* Key Distribution Issues

- trust on KDC
- session key lifetimes should be limited

* A Transparent Key Control Scheme



① Host sends packet requesting connection.
② Security service buffers packet; asks KDC for session key.
③ KDC distributes session key to both hosts.
④ Buffered packet transmitted.

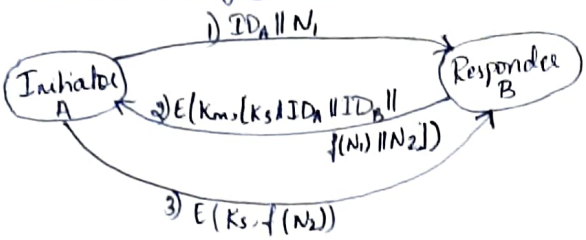
Automatic Key Distribution for Connection-Oriented Protocol

- This scheme is useful for providing end-to-end encryption at a network or transport level in a way that is transparent to the end users.
- The approach ~~at a network or transport level~~ assumes that communication makes use of a connection-oriented end-to-end protocol, such as TCP.
- The noteworthy element of this approach is a session security module (SSM), which may consists of functionality at one protocol layer, that performs end-to-end encryption and obtains session keys on behalf of its host or terminal.

Process

- When one host wishes to set up a connection to another host, it transmits a connection-request packet.
- The SSM saves that packet & applies to the KDC for permission to establish the connection.
- The communication between the SSM & the KDC is encrypted using a master key shared only by the SSM and the KDC.
- If the KDC approves the connection request, it generates the session key and delivers it to the two appropriate SSMs, using a unique permanent key for each SSM.
- The requesting SSM can now release the connection request packet, and a connection is set up between the two end systems.
- All user data exchanged between the two end systems are encrypted by their respective SSMs using the one-time session key.
- This automated key distribution approach provides the flexibility & dynamic characteristics needed to allow a no. of terminal users to access a no. of hosts and for the hosts to exchange data with each other.

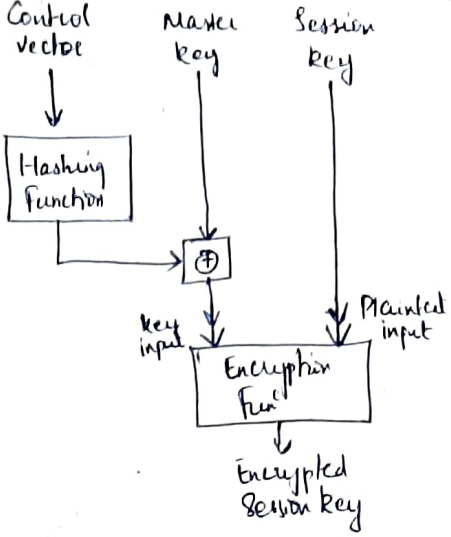
Decentralized Key Control



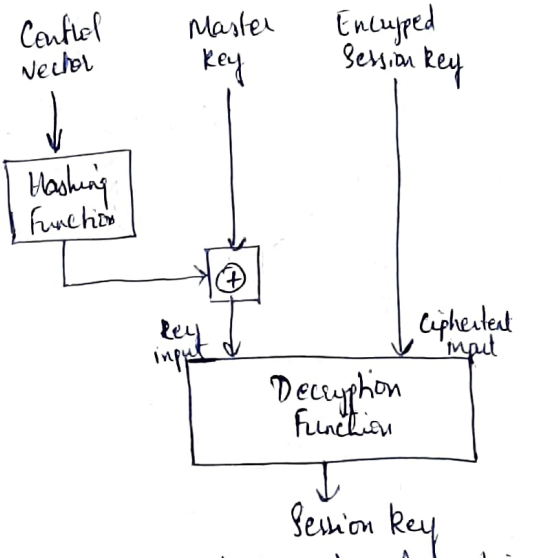
- Use of key distribution center imposes the requirement that the KDC be trusted and be protected from subversion.
- This requirement can be avoided if key distribution is fully decentralized.
- A fully decentralized is not practical for larger networks using Symmetric encryption only
- A decentralized approach requires that each end system be able to communicate in a secure manner with all potential partner end systems for purposes of session key distribution.
- Thus there may need to be as many as $[n(n-1)]/2$ master key for a configuration with n end systems.
- The message transferred using the master key are short, cryptanalysis is difficult.

Controlling Key Usage

- The concept of a key hierarchy and the use of automated key distribution techniques greatly reduce the no. of keys that must be manually managed and distributed.
- It also maybe desirable to impose some control on the way in which automatically distributed keys are used.
- For ex., in addition to seperating master keys from session keys, we may wish to define different types of session keys on the basic of use, such are
  - Data-encrypting key, for general communication across a network.
  - PIN-encryption key, for personal identification numbers (PINs) used in electronic funds transfer & point-of-sale applications.
  - File encryption key, for encrypting files stored in publicly accessible locations.



a) Control vector encryption



b) Control vector decryption

(Refer 447)