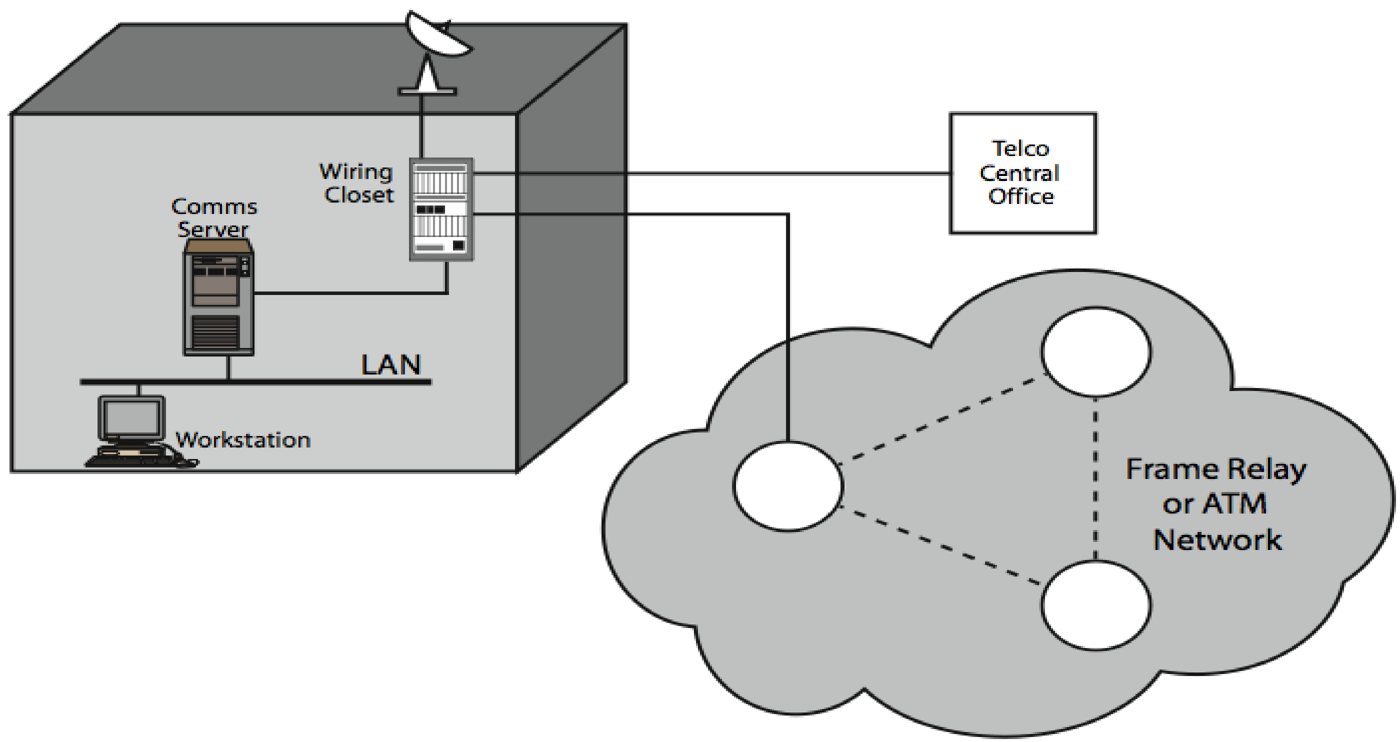
# Confidentiality using Symmetric Encryption

# Confidentiality using Symmetric Encryption

- traditionally symmetric encryption is used to provide message confidentiality



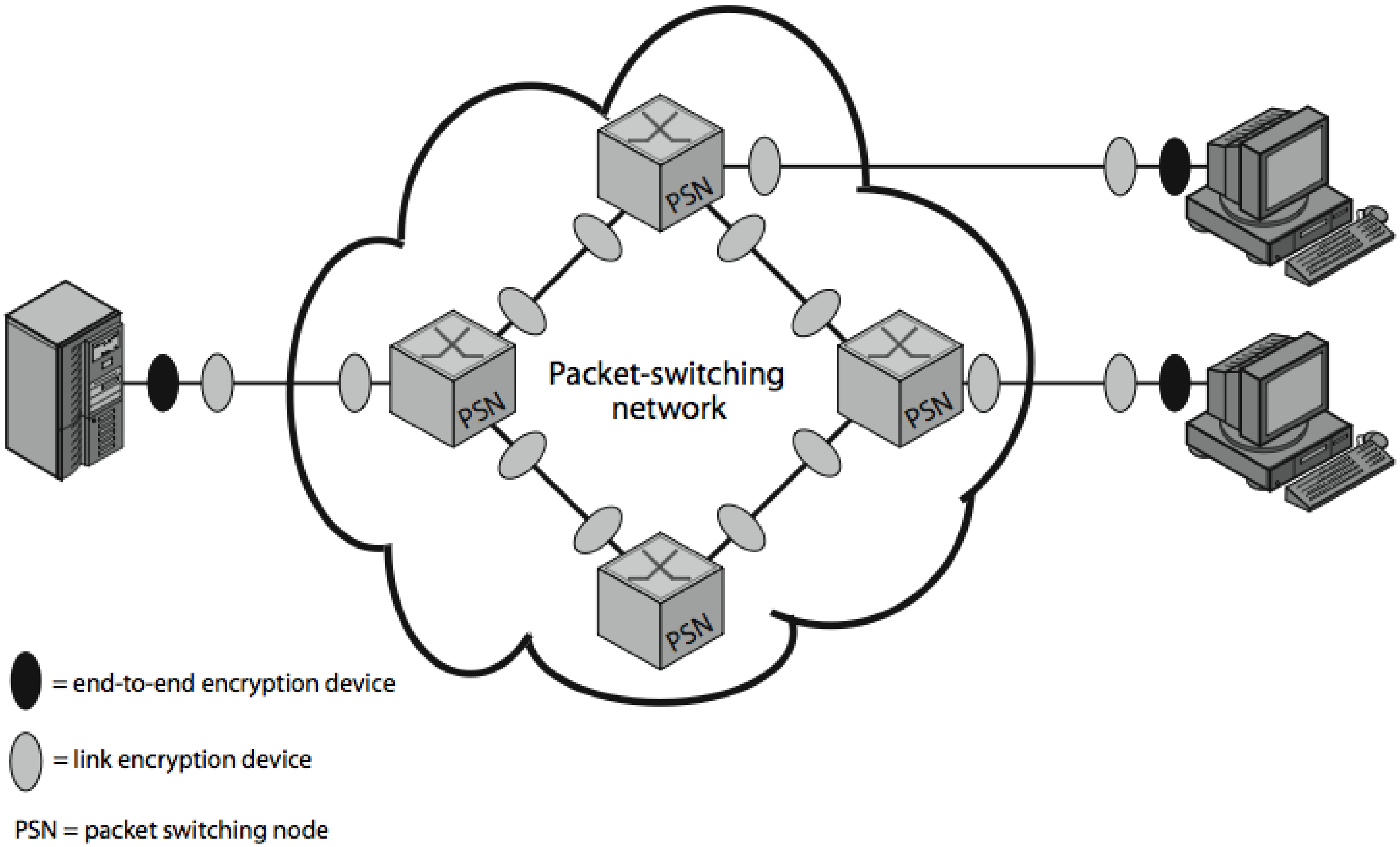
# Points of vulnerability

- Snooping from another workstation
- Reprogram switches or router to capture data from the network
- Use dial in to LAN or server to snoop
- Physically tap lines in wiring closet
- External router links to enter or snoop
- In wireless links risk of eavesdropping is greater

# Placement of Encryption

- have two major placement alternatives
- **link encryption**
  - encryption occurs independently on every link
  - implies must decrypt traffic between links
  - requires many devices, but paired keys
- **end-to-end encryption**
  - encryption occurs between original source and final destination
  - need devices at each end with shared keys

# Placement of Encryption



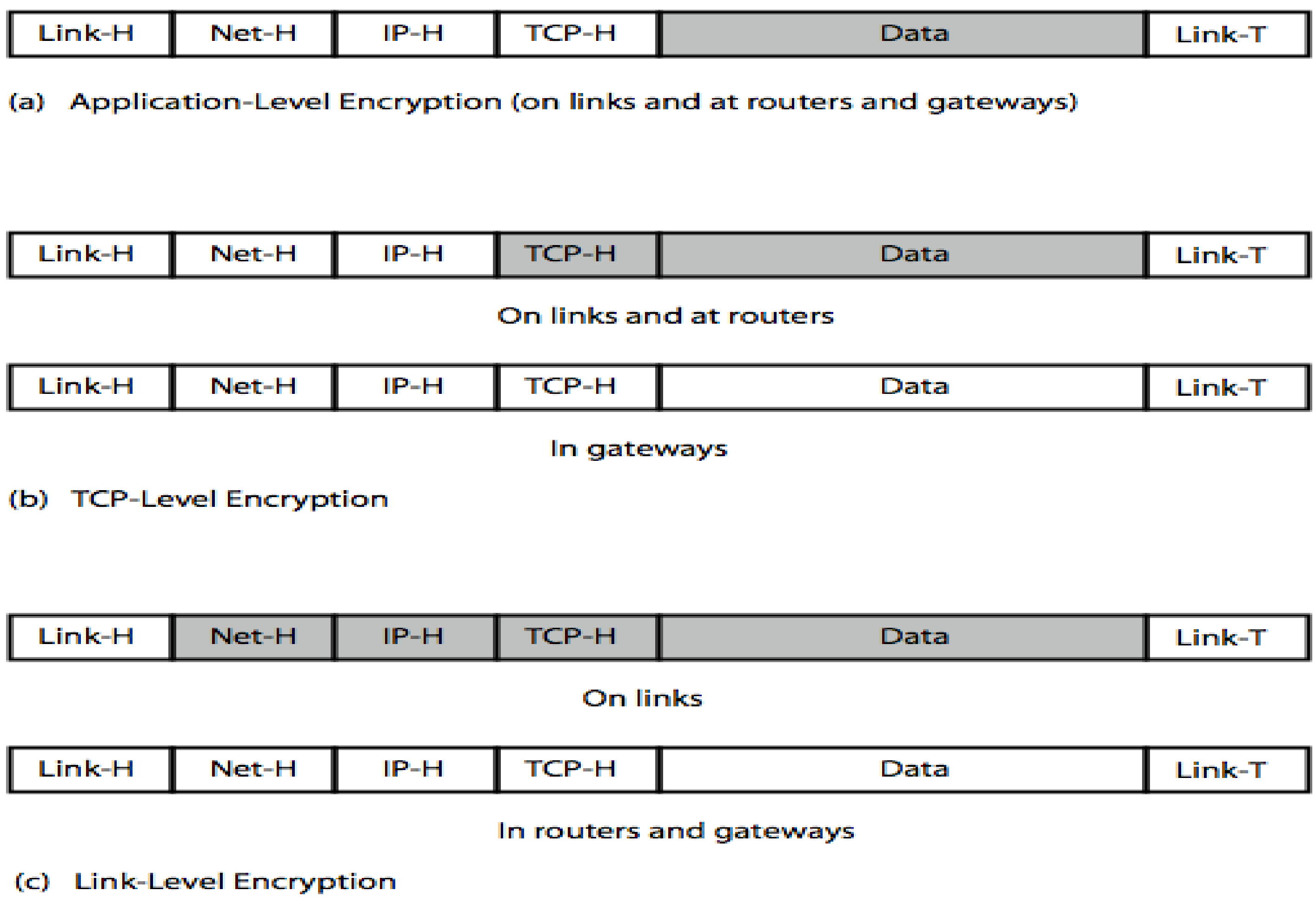
# Placement of Encryption

- when using end-to-end encryption must leave headers in clear
  - so network can correctly route information
- hence although contents protected, traffic pattern flows are not
- ideally want both at once
  - end-to-end protects data contents over entire path and provides authentication
  - link protects traffic flows from monitoring

# Placement of Encryption

- can place encryption function at various layers in OSI Reference Model
  - link encryption occurs at layers 1 or 2
  - end-to-end can occur at layers 3, 4, 6, 7
  - as move higher less information is encrypted but it is more secure though more complex with more entities and keys

# Encryption vs Protocol Level




Shading indicates encryption.

| | | |
|---|---|---|
| TCP-H | = | TCP header |
| IP-H | = | IP header |
| Net-H | = | Network-level header(e.g., X.25 packetheader,LLC header) |
| Link-H | = | Data link control protocolheader |
| Link-T | = | Data link control protocoltrailer |

# Traffic Analysis

- is monitoring of communications flows between parties
  - useful both in military & commercial spheres
  - can also be used to create a covert channel
- link encryption obscures header details
  - but overall traffic volumes in networks and at end-points is still visible
- traffic padding can further obscure flows
  - but at cost of continuous traffic

# Key Distribution

- symmetric schemes require both parties to share a common secret key
- issue is how to securely distribute this key
- often secure system failure due to a break in the key distribution scheme

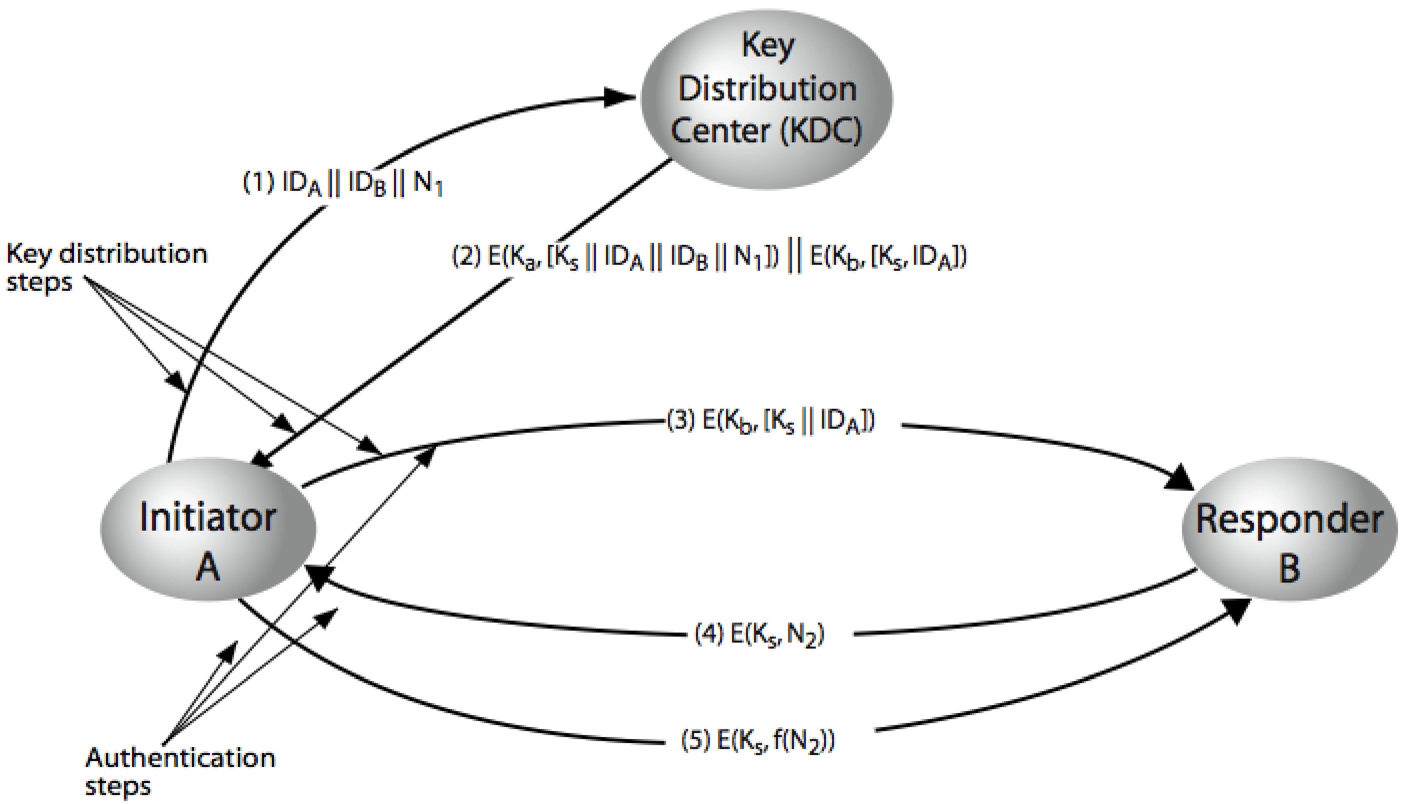
# Key Distribution

- 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 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

# Public-Key Cryptography – General Characteristics

- public-key/two-key/asymmetric cryptography
  - A concept, there are several such cryptosystems
- uses 2 keys
  - public-key
    - may be known by anybody, and can be used to encrypt messages, and verify signatures
  - private-key
    - known only to the recipient, used to decrypt messages, and sign (create) signatures
- keys are related to each other but it is not feasible to find out private key from the public one

# Public-Key Cryptography – General Characteristics

- Keys are related to each other but it is not feasible to find out private key from the public one
- It is computationally easy to en/decrypt messages when the relevant keys are known
- Trap-door one-way function

$Y=f_{ku}(X)$ easy, if ku and X are known

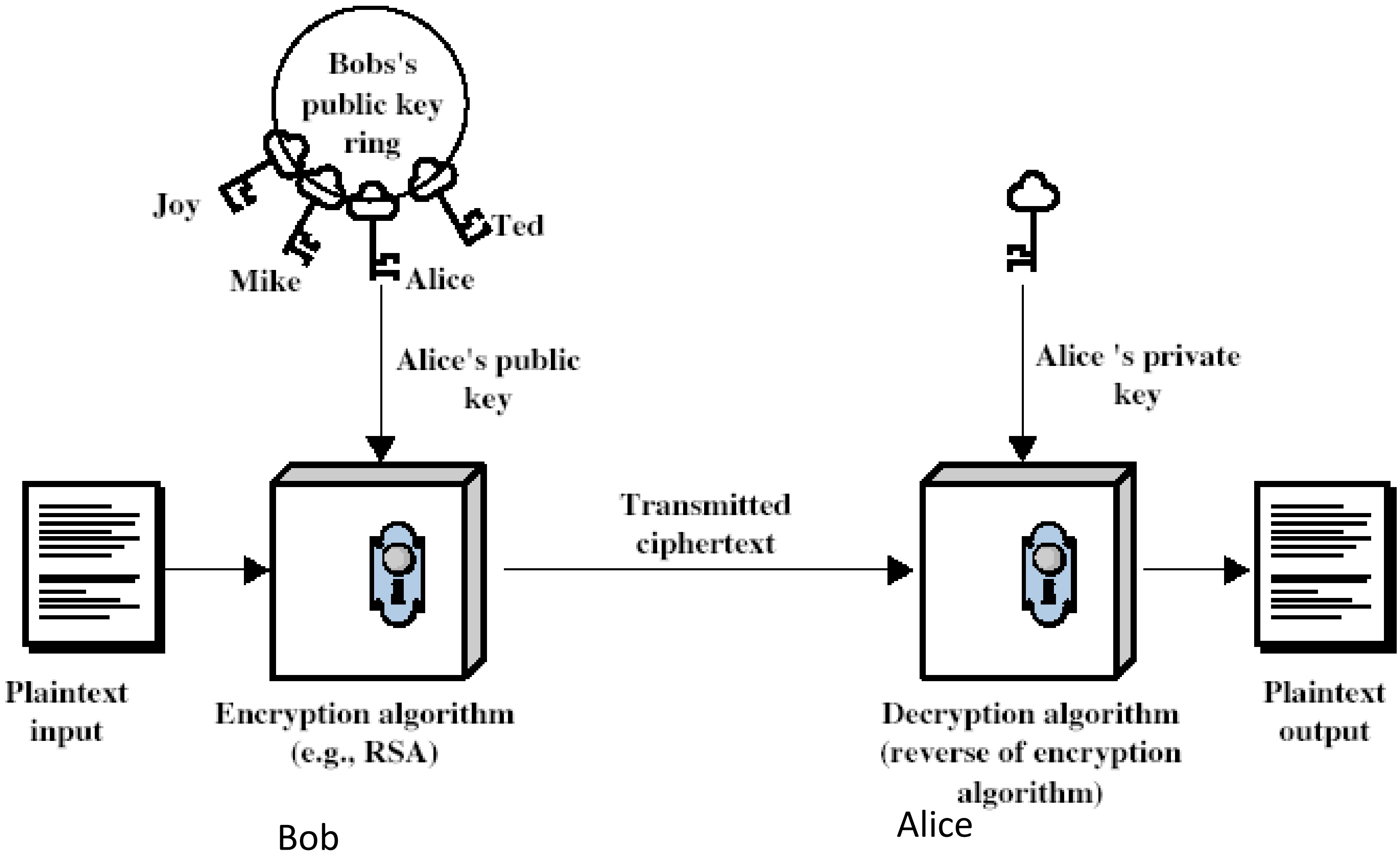
$X=f_{kr}^{-1}(Y)$ easy, if kr and Y are known, but infeasible if Y is known but kr is not known

  – ku: public-key, kr: private key

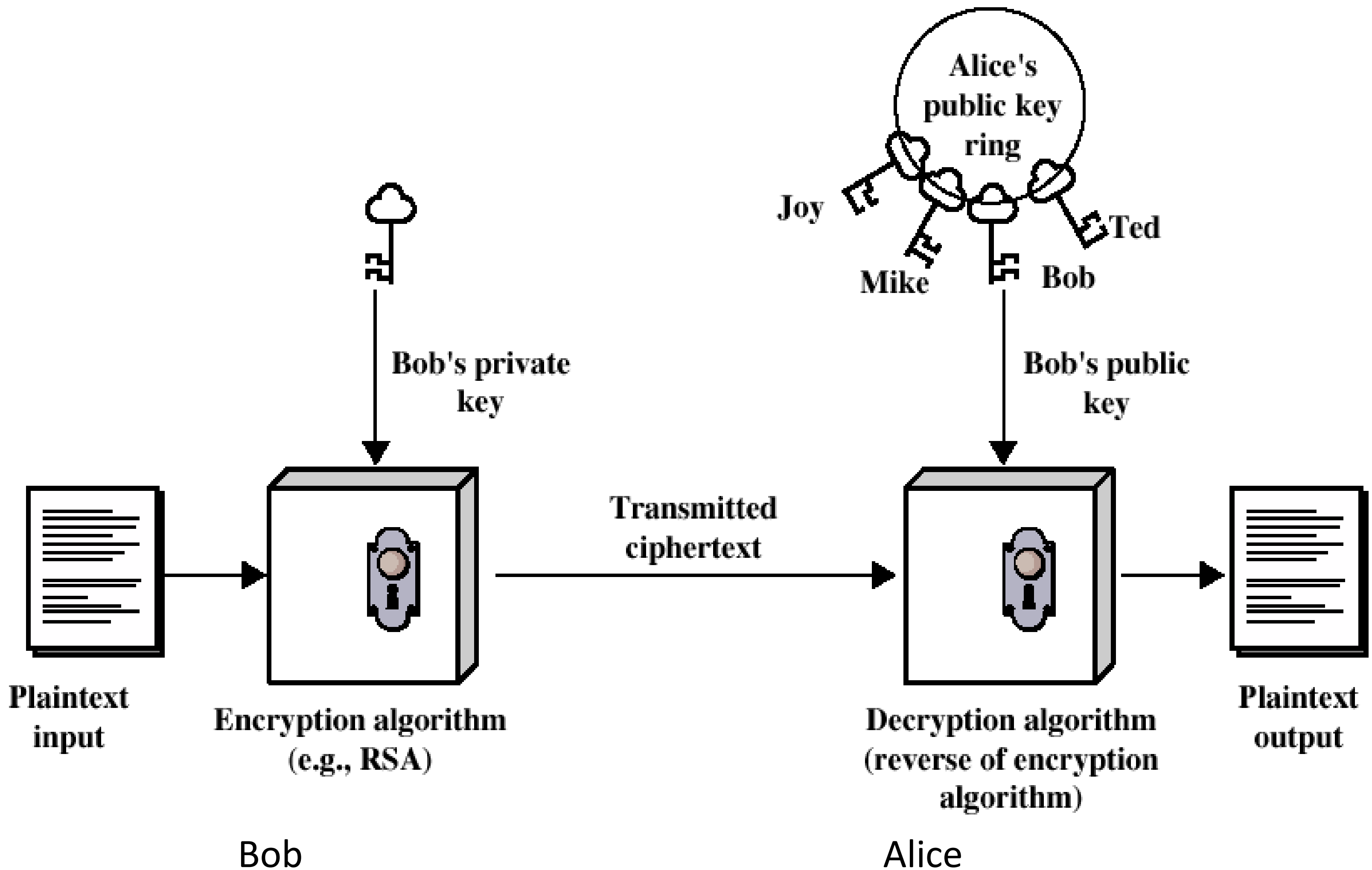
# Public-Key Cryptography – General Characteristics

- based on **number theoretic hard problems**
  - rather than substitutions and permutations
- 3 misconceptions about PKC
  - it replaces symmetric crypto
    - PKC rather complements private key crypto
  - PKC is more secure
    - no evidence for that, security mostly depends on the key size in both schemes
  - key distribution is trivial in PKC since public keys are public
    - making something public is not easy. How can you make sure that a public key belongs to the intended person?
    - key distribution is easier, but not trivial

# Public-Key Cryptography - Encryption



# Public-Key Cryptography - Authentication



# Why Public-Key Cryptography?

- Initially developed to address two key issues:
  - key distribution
    - symmetric crypto requires a trusted Key Distribution Center (KDC)
    - in PKC you do not need a KDC to distribute and know secret keys, but you need trusted third parties
  - digital signatures (non-repudiation)
    - not possible with symmetric crypto

# Public-Key Cryptosystems

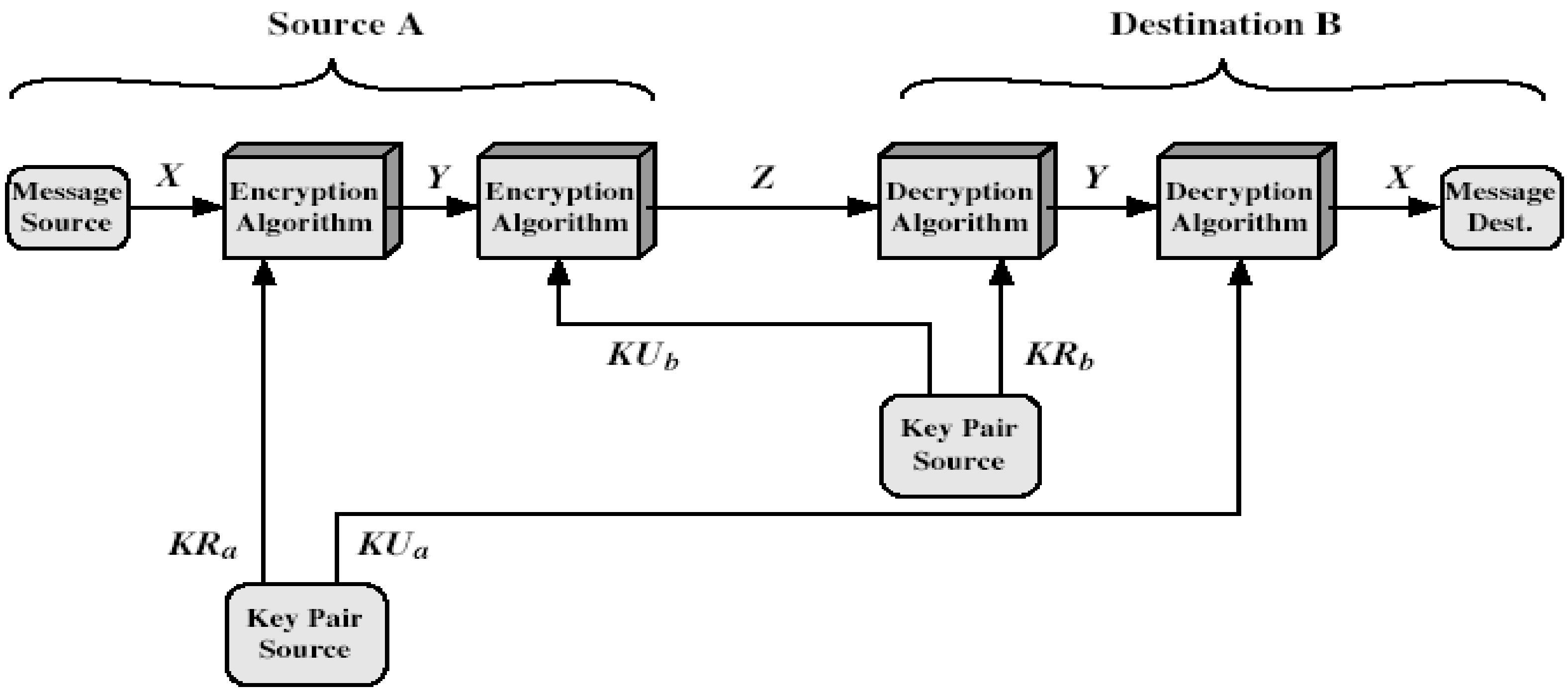



**Figure 9.4 Public-Key Cryptosystem: Secrecy and Authentication**

| | | | |
|---|---|---|---|
| $KU_a$ | A's Public Key | $KU_b$ | B's Public Key |
| $KR_a$ | A's Private Key | $KR_b$ | B's Private Key |

# Public-Key Applications

- 3 categories
  - encryption/decryption
    - to provide secrecy
  - digital signatures
    - to provide authentication and non-repudiation
  - key exchange
    - to agree on a session key
- some algorithms are suitable for all uses, others are specific to one

# Security of Public Key Schemes

- like private key schemes brute force attack is always theoretically possible
  - use large keys
  - consider the security / performance tradeoff
- due to public key / private key relationships number of bits in the key should be much larger than symmetric crypto keys
  - to do the hard problem really hard
  - 80-bit symmetric key and 1024-bit RSA key has comparable resistance to cryptanalysis
- a consequence of use of large keys is having slower encryption and decryption as compared to private key schemes
  - thus, PKC is not a proper method for bulk encryption

# Summary

- have considered:
  - the AES selection process
  - the details of Rijndael – the AES cipher
  - Confidentiality through symmetric encryption
  - Traffic analysis
  - Key distribution centre
  - Introduction to Public key cryptography