东南大学网络空间安全学院

密码学与安全协议

第四讲 对称密码的使用方法

黄　杰

信息安全研究中心

本讲内容

- 对称加密算法实现的保密性
- 对称加密算法实现的完整性

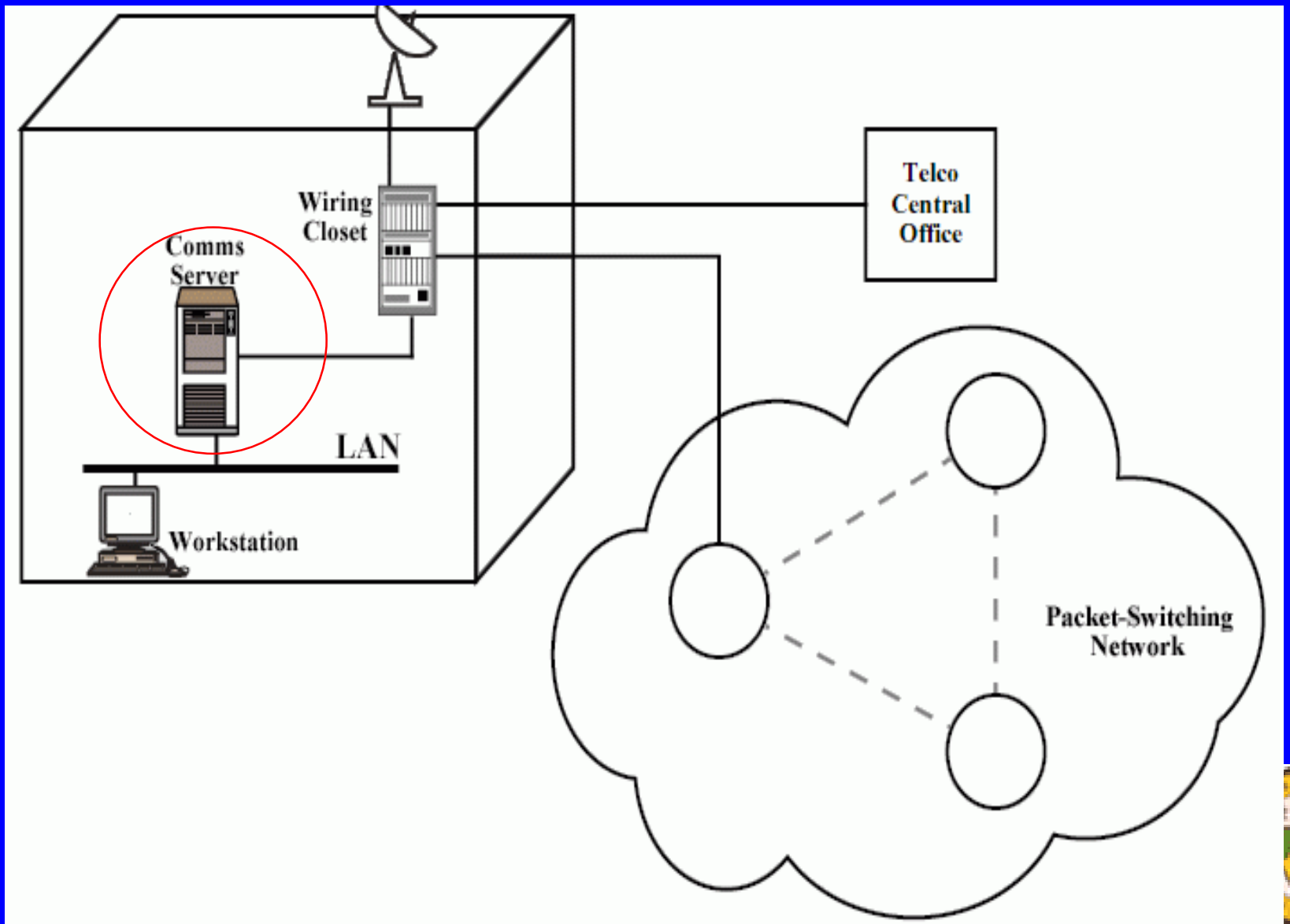
知识点

1、链路加密和端到端加密

2、消息完整性认证方法

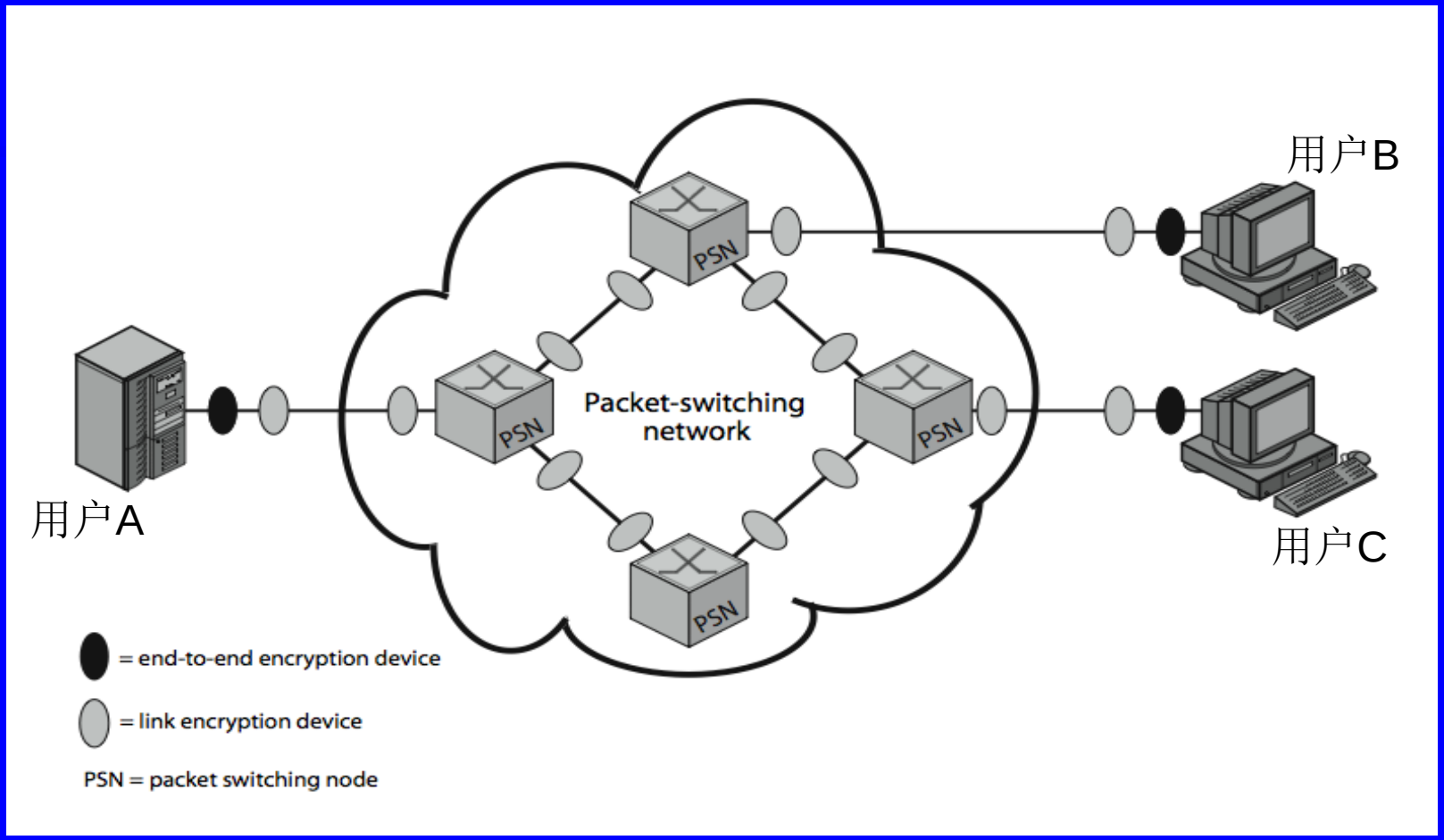
对称加密算法实现的保密性

案　例


Wiring
Closet
Comms
Server
LAN
Workstation
Telco
Central
Office
Packet-Switching
Network

- 攻击：一切皆有可能！
  - 外部网络并不受终端用户的控制。
- 保护通信机密性的最有效办法之一是对通信数据进行加密。
- 如何加密？
  - 何处加密？
  - 加密什么？

- 保密需求：
  - 连接保密性
  - 无连接保密性
  - 选择字段保密性
  - ……



- 问题：
  - 实现用户A和用户B、C之间保密通信的方法？
  - 分析它们之间的优缺点。

链路加密与端对端加密

- 分组交换网络中的消息
  - 信息头，包括源地址，目的地址，采用协议等
  - 用户数据
- 链路加密
  - 在通信链路两端加上加密设备。
  - 每次分组交换都需要将消息解密，在交换节点将信息解密。
  - 用户对分组交换节点的安全性不能保证。
  - 整个网络需要维护密钥的数量庞大。
  - 接收者无法确定消息的来源。

链路加密与端对端加密

- 端对端加密
  - 由源主机或终端加密用户数据，信息头以明文方式传送。
  - 密文经由网络传到目的主机或终端，目的主机与源主机，共享一个密钥以便解密。
  - 缺点：信息头在传输过程中为明文，易受流量分析攻击。

链路加密与端对端加密的特点

	链路加密	端对端加密
末端系统与中间系统的安全性	消息在发送主机时是明文	消息在发送主机时是密文
	消息在中间节点时是明文	消息在中间节点时是密文
用户角色	与发送主机交互	与发送过程交互
	加密过程对用户透明	用户自己使用加密
	主机含加密设备	用户决定加密算法
	所有用户使用一个加密设备	用户选择加密体制
	可以硬件实现	软件实现
	所有消息被加密或都不被加密	用户决定每条消息是否加密
实现时要注意的	每对中间主机节点需要一个密钥	每对用户需要一个密钥
	提供主机认证	提供用户认证

应用层端到端加密

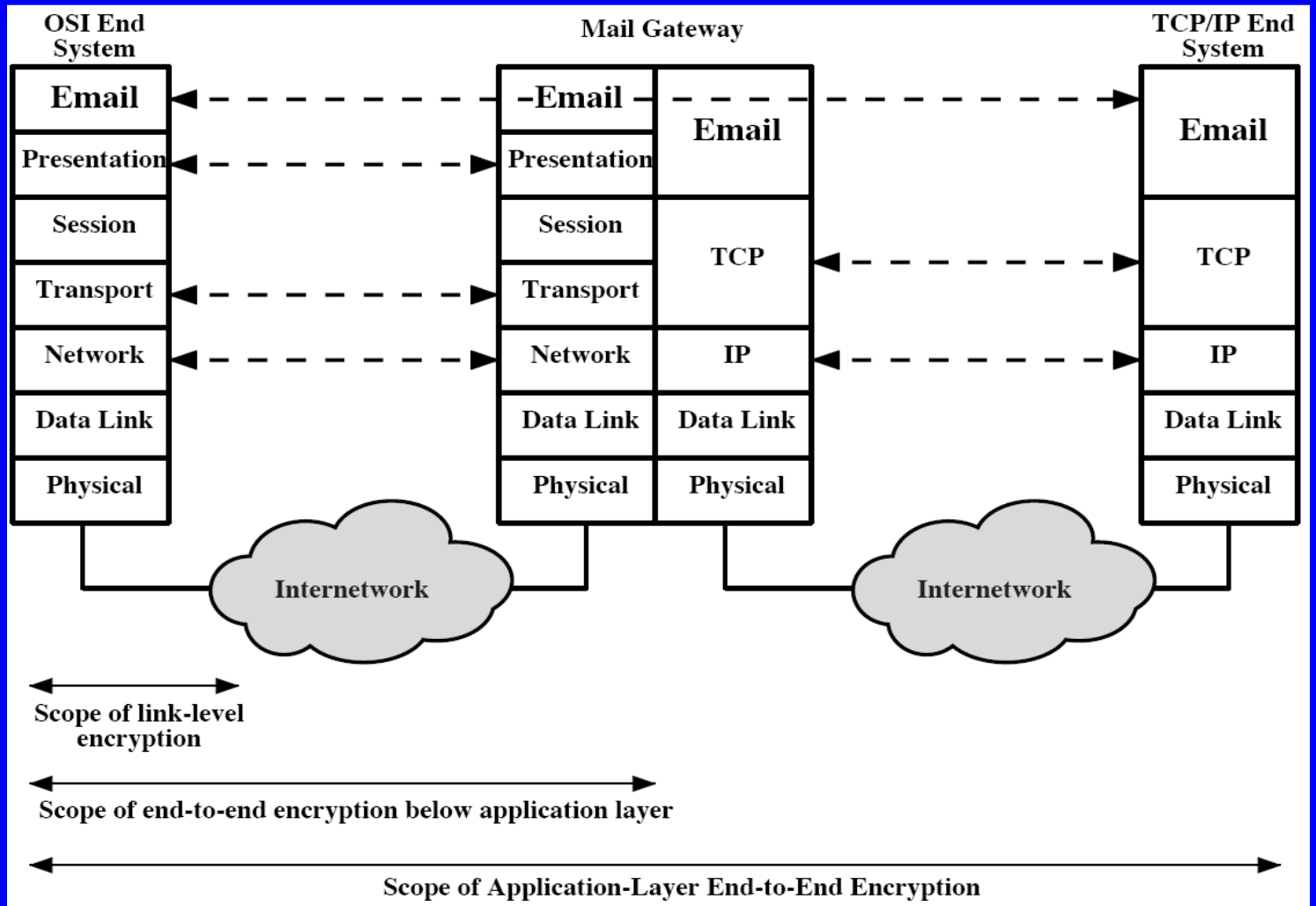
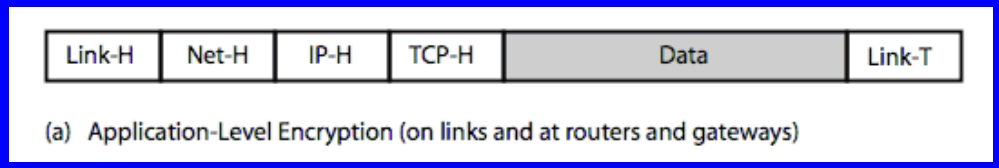


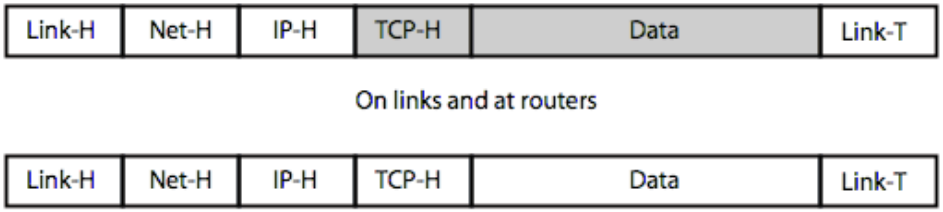
Figure 7.4 Encryption Coverage Implications of Store-and-Forward Communications

协议和加密方法

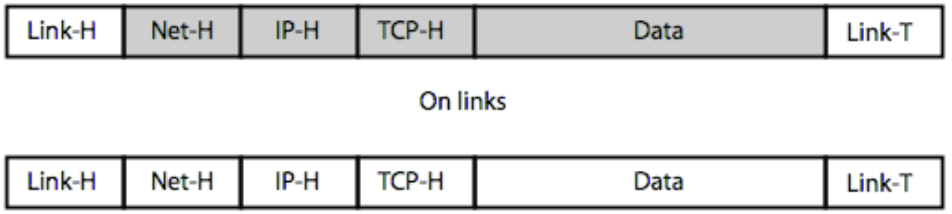
- 随着层数的上升
  - 要加密的信息变少，机密性更高
  - 需要的密钥增加



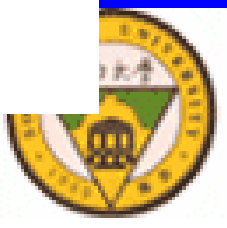
(a) Application-Level Encryption (on links and at routers and gateways)



(b) TCP-Level Encryption



(c) Link-Level Encryption

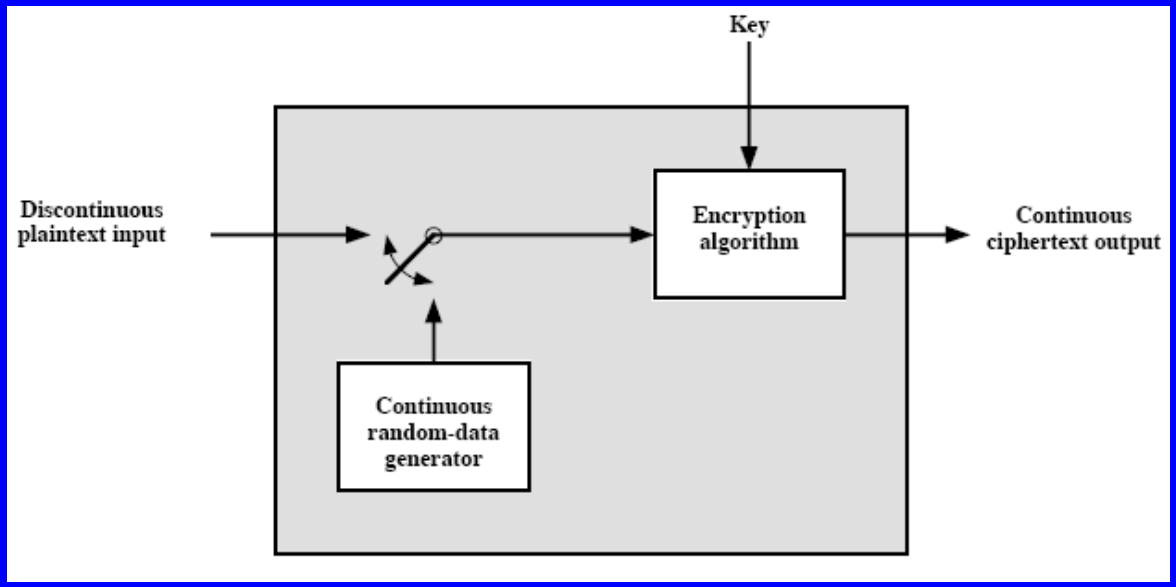


传输保密性

- 数据传输分析可以得到的信息类型
  - 谁与谁通信——传输双方的标识
  - 信息的重要程度——传输双方的联系频率、消息格式、长度或者频繁交换
  - 发送信息与现实事件的相关性。

    如：中途岛之战。AF
- 隐通道
  - 指采用某种方式进行通信，而这种方式是通信设备的设计者所不知道的。

- 链路加密方法
  - 消息头被加密，减少传输分析的机会，但无法抵御流量分析攻击。
  - 验证消息的来源难度加大。接收方拿到的消息是最后一段链路解密后的明文。
  - 传输填充。



- 端对端加密方法
  - 无法抵抗流量分析攻击。
  - 按照统一长度的数据填充传输层和应用层传输的消息。
  - 可随机插入空消息到消息流中。

两种方法结合在一起使用

密钥的使用方法

- 定义若干种类型的会话密钥
  - 数据加密密钥
  - PIN加密密钥
  - 文件加密密钥
- 定义、标识不同的密钥
  - 如：DES密钥　其中的8位效验位

对称加密算法实现的完整性

——消息认证码

- 完整性指网络信息未经授权不能进行改变的特性，即：信息在存储或传输过程中保持不被偶然或蓄意地删除、修改、伪造、乱序、重放、插入等破坏和丢失的特性。

- 消息认证的分类
  - 消息内容的认证　　消息认证码
  - 消息的序号和操作时间的认证
  - 消息的信源和信宿　　收发双方身份认证

消息认证码

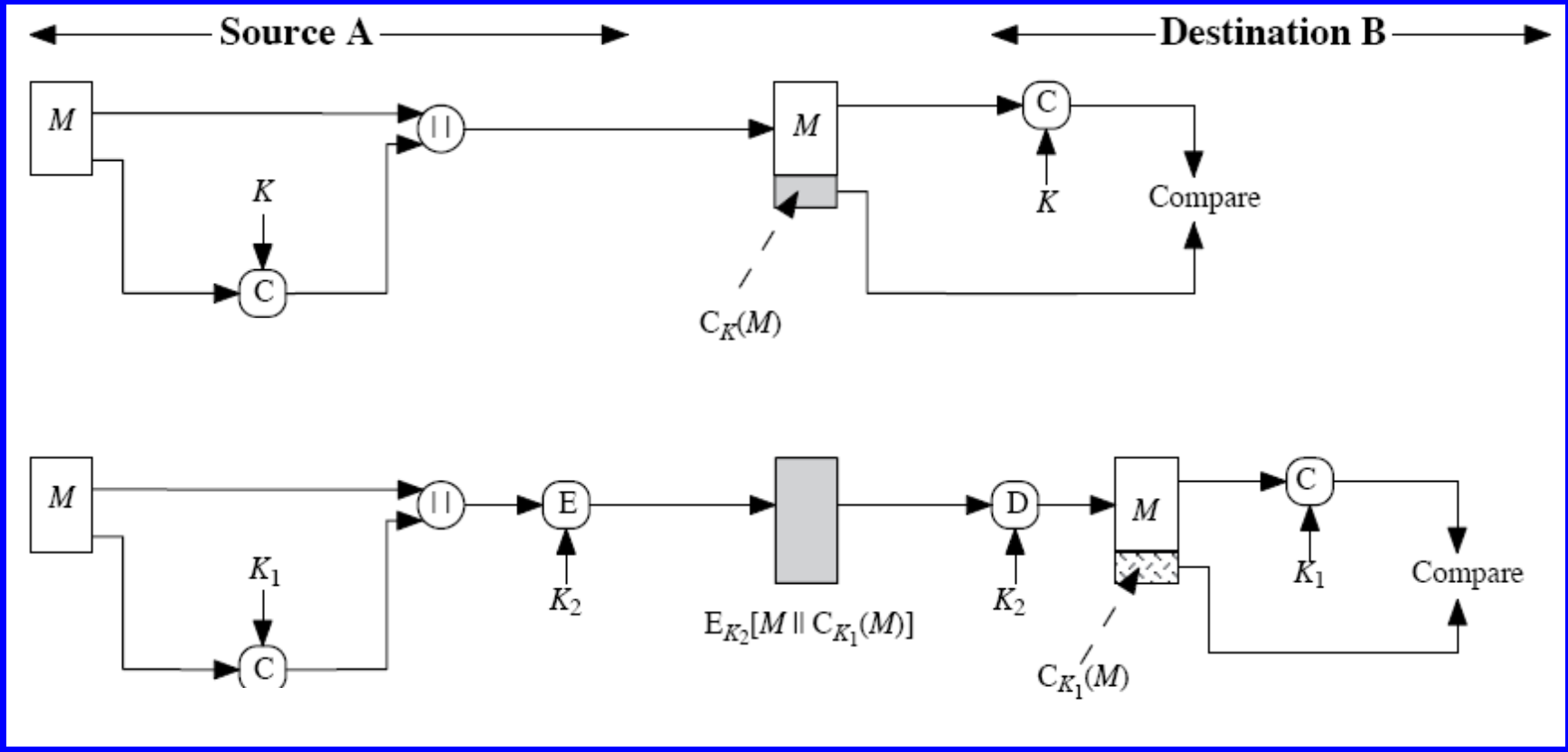
- 原因
  - 有许多应用将同一消息广播给很多接受者；
  - 通信某一方处理的负荷很大，只能验证消息；
  - 明文形式的计算机程序进行认证；

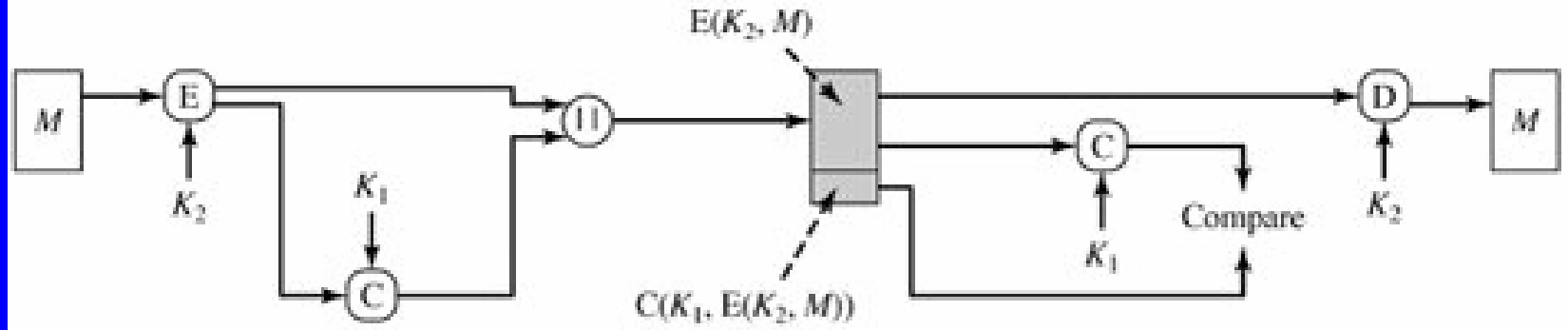
  - 有些应用不关心保密性，而关心消息认证；
  - 将认证和保密性分开；
  - 解密的消息不再受到保护，但MAC可以提供长期的认证。

消息认证码

- 利用密钥生成一个固定长度的短数据块，称为消息认证码MAC，并将MAC附加在消息之后。接收方通过计算MAC来认证该消息。
- 计算公式：$MAC=C_K(M)$
  - M：长度可变的消息
  - K：收发双方共享的密钥
  - $C_K(M)$：定长的认证符

消息认证码的使用方式


Source A
Destination B
M
K
C
||
M
C
K
Compare
$C_K(M)$
M
K1
C
||
E
K2
$E_{K_2}[M \| C_{K_1}(M)]$
D
K2
M
C
K1
Compare
$C_{K_1}(M)$


M
E
K2
K1
C
||
$E(K_2, M)$
$C(K_1, E(K_2, M))$
C
K1
Compare
D
K2
M

消息认证码的讨论

- 保密性与真实性是两个不同的概念
- 从根本上说，信息加密提供的是保密性而非真实性
- 加密代价大(公钥算法代价更大)
- 某些信息只需要真实性,不需要保密性
  - 广播的信息难以使用加密(信息量大)
  - 网络管理信息等只需要真实性
  - 政府/权威部门的公告

消息认证码的安全性

- 一般的加密算法：k位密钥，穷举攻击需要大约$2^{(k-1)}$次运算。
- 消息认证码：密钥位数比MAC长，因此许多密钥都会产生正确的MAC，而攻击者却不知哪一个是正确的密钥。
- 例如：80位密钥，32位的MAC。

  第一次穷举攻击：1个消息，产生1个MAC，可以由2^{48}个可能的密钥产生。

  第二次穷举攻击会得到2^{16}个可能的密钥。

  第三次穷举攻击会得到唯一一个密钥，即发送方所使用的密钥。

不需要密钥的攻击？

- 考虑以下的MAC算法：
- $M=(X_1 \| X_2 \| .... \| X_m)$是由64位分组$X_i$连接而成。
- 定义：
- $\Delta(M)= X_1 \oplus X_2 \oplus .... \oplus X_m$
- $C_K(M)=E_K(\Delta(M))$
- 其中加密算法为电子密码本方式DES，则密钥长为56位，MAC长为64位。确定K的穷举攻击需执行至少2^{56}次加密。

不需要密钥的攻击方法

- 设$M' = (Y_1 \| Y_2 \| .... \| Y_{m-1} \| Y_m)$
- $Y_m = Y_1 \oplus Y_2 \oplus .... \oplus Y_{m-1} \oplus \Delta(M)$
- 消息M'和$MAC = C_K(M) = E_K[\Delta(M)]$是一对可被接收者认证的消息。
- 为什么？
- 用此方法，任何长度为64×(m-1)位的消息可以作为欺骗性信息被插入！

对MAC函数的要求

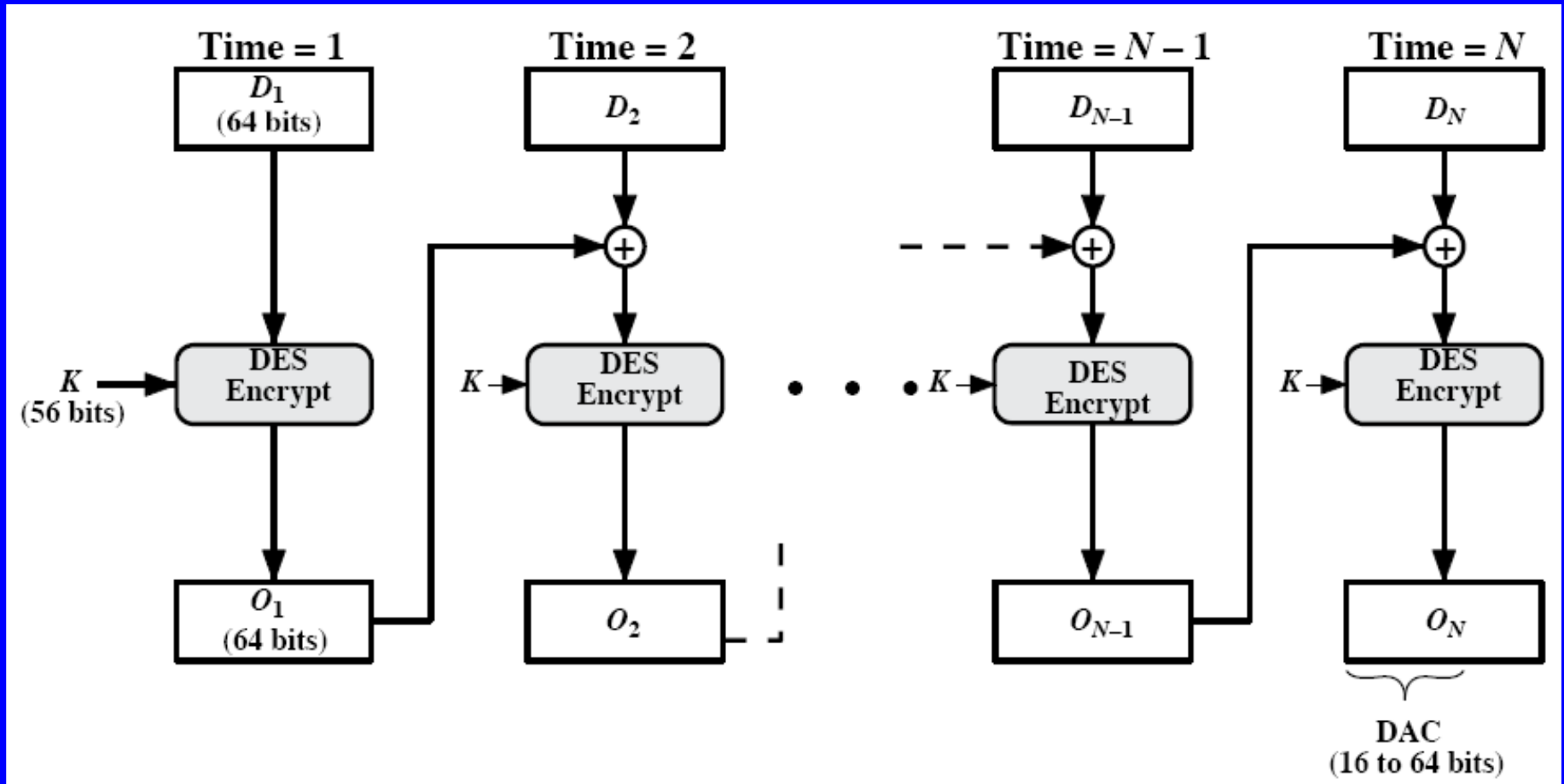
- 如果一个攻击者得到M和$C_K(M)$，则攻击者构造一个消息M′使得$C_K(M')=C_K(M)$应在计算上不可行。
- $C_K(M)$应均匀分布，即：随机选择消息M和M′，$C_K(M)= C_K(M')$的概率是2^{-n}，其中n是MAC的位数。即MAC函数具有均匀分布的特征。
- 令M′为M的某些变换，即：M′=f(M)，（例如：f表示逆转M中一位或多位），那么，$Pr[C_K(M)= C_K(M')] = 2^{-n}$。

例子：基于DES的消息认证码

- 数据认证算法DEA使用CBC (Cipher Block Chaining)方式，初始向量为0。它是使用最广泛的MAC算法之一。
- 将数据按64位分组，$D_1, D_2, \ldots, D_N$，必要时最后一个数据块用0向右填充。运用DES加密算法E，密钥为K。
- 数据鉴别码(DAC)的计算如下：

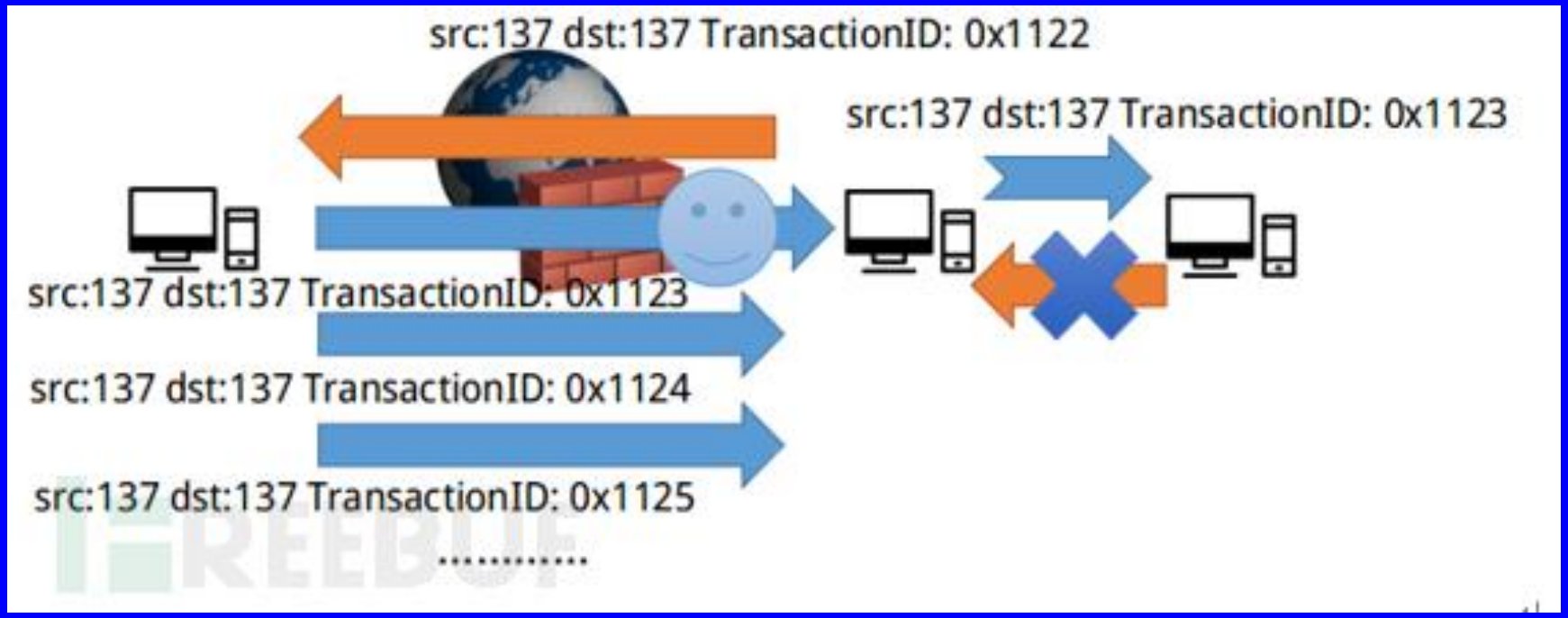
$$O_1 = E_K(D_1)$$
$$O_2 = E_K(D_2 \oplus O_1)$$
$$O_3 = E_K(D_3 \oplus O_2)$$
$$\ldots$$
$$O_N = E_K(D_N \oplus O_{N-1})$$

基于DES的消息认证码—CBC MAC



谢　谢

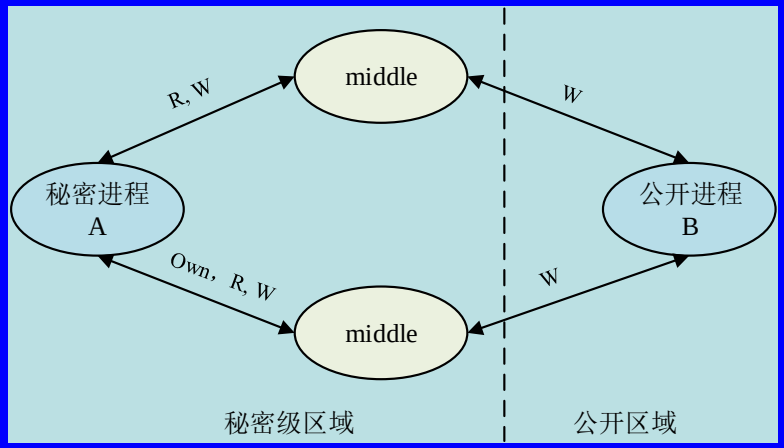
隐通道例子



1. 受害主机向另一台主机发送SMB请求时，由于默认共享端口445（SMB）和139（NetBIOS会话）不可达，则向端口137发送NetBIOS NB STATE（NetBIOS名称服务）。
2. 一旦防火墙允许这个请求通过，则在两台主机之间建立隧道。这个隧道将保持一段时间。
3. 攻击主机可以发送大量的RESPONSE包，每个包的ID依次增加，直至蒙对为止。

隐通道例子

要求：进程B的安全级别低于文件data



1）进程A创建秘密信息文件data

2）进程B打开秘密文件middle，并写入一个字符，内容为“0”或“1”。进程A一直监控文件middle，当它发现进程B写入文件时，说明进程B已经做好接收秘密信息的准备，此时开始利用隐蔽通道传送秘密信息

3）进程A可以改变文件data的DAC访问模式。A和B约定，若允许B“写”文件data，则表示进程A发送一个二进制比特“1”；否则，表示进程A发送比特“0”

4）进程B通过“写”方式打开文件data，如果返回成功打开的标识，则表示收到了一个比特“1”，否则为“0”

5）进程B每收到一个比特的信息，就将其写入文件middle。进程A可以通过检查文件middle的内容，确定信息是否发送正确

6）反复执行上述的（2）~（5）过程，直至进程A将所有秘密信息传送给进程B