目录

1 Introduction **2**

1.1 计算机安全 2

1.2 安全攻击 2

1.2.1 被动攻击 2

1.2.2 主动攻击 3

1.2.3 网络攻击 3

1.3 安全服务 4

1.4 安全机制与模型 6

2 Cryptography **7**

2.1 密码体制 7

2.2 密码分析 8

2.3 密码体质的安全性 8

3 DES **9**

3.1 DES 描述 9

3.1.1 初始置换 9

3.1.2 轮结构 10

3.1.3 子密钥的产生 11

3.2 DES 的工作模式 12

3.3 常规加密 13

4 PKCS & RSA **14**

4.1 公钥密码体制原理 14

4.2 RSA 算法 14

4.2.1 原理 14

4.2.2 应用 15

1 Introduction

1.1 计算机安全

定义: 对某个自动化信息系统的保护措施，其目的在于实现信息系统资源的完整性、可用性以及机密性。

- 机密性: 维持施加在数据访问和泄露上的授权限制，保护个人隐私和私有信息。
  - 数据机密性: 保证私有的或机密的信息不会泄露给未经授权的个体
  - 隐私性: 保证个人可以控制和影响相关的信息，这些信息可能被收集、存储和泄露
  - 损失: 指非授权的信息泄露
- 完整性: 防范不当的信息修改和破坏，保证信息的认证与授权。
  - 数据完整性: 保证只由特定的、已授权的方式来更改信息和代码。
  - 系统完整性: 保证系统正常实现预期功能，而不会被故意或偶然的非授权操作控制
  - 损失: 指未经授权的信息修改和破坏
- 可用性: 保证系统及时运转，及时且可靠的获取和使用信息，不会拒绝已授权的用户。
  - 损失: 指对信息、信息系统访问或使用的中断。

还有两个追加的定义:

- 真实性: 可以被验证和信任。
- 可计量性: 每个实体的行为可以被唯一追踪。

1.2 安全攻击

1.2.1 被动攻击

定义: 通过窃听或监视数据传输来收集、利用数据信息，但不会影响系统资源和正常访问，用户一般也不会察觉到。主要分为信息收集和流量分析。

- 信息收集: 使通信设施中的消息泄露并获取。
- 流量分析: 当消息内容被加密之后，通过观察消息的模式、频道和长度推测双方的位置和身份。

对付被动攻击方法的一般是用加密来防范而不是检测。

1.2.2 主动攻击

定义: 对数据流的改写和错误数据流的添加，一但会改变系统资源和影响系统运作。分为伪装、重放、消息篡改和拒绝服务。

- 伪装: 攻击者捕获认证信息，利用认证信息重放，使一个具有较少特权的经过认证的实体，通过模仿一个具有其他特权的实体而得到这些额外的特权。
- 重放: 获取数据单元并按照它之前的顺序重新传输，以此来产生一个非授权的效应。
- 消息篡改: 合法消息的某些部分被篡改，或者消息被延迟、重排，从而产生非授权效应。
- 拒绝服务: 阻止或禁止对通信设备的正常使用或管理。

主动攻击方法难以防范，一般通过检测主动攻击并恢复主动攻击造成的损坏和延迟。

1.2.3 网络攻击

(1) 口令窃取

- 猜测攻击
  - 利用已知或假定的口令尝试登陆
  - 根据窃取的口令文件进行猜测
  - 窃取某次合法终端之间的会话，并记录所使用的口令
- 抵御方式
  - 阻止选择低级口令
  - 对口令文件严格保护
  - 彻底解决方案: 使用令牌机制，如一次性口令方案

(2) 欺骗攻击

- DNS 欺骗: 解析域名的时候返回一个虚假 IP
- Email 欺骗: 冒充和管理员相同的邮件地址给用户发邮件要求修改口令/在附件中添加病毒和木马
- Web 欺骗: 通过伪造站点拷贝，使 Web 入口进入到攻击者的服务器

(3) 缺陷和后门攻击

- 网络蠕虫传播: 蠕虫向“读”缓冲区诸如大量数据；解决: 向守护程序发送新的代码

- 缓冲器溢出攻击: 扰乱程序，使堆栈粉碎；解决: 改进设计，消除缺陷
- 缺陷攻击: 程序中某些代码不能满足特定需求；解决：采取步骤降低缺陷发送可能性

(4) 认证失效

通过某种方式使得认证机制失效，导致服务器被攻击者欺骗。通过修改认证方案消除这种缺陷来组织这类攻击。

(5) 协议缺陷

协议本身的缺陷导致的攻击，例如攻击者可以对 TCP、DNS 等协议发起序列号攻击，通过改进设计或避免在堆栈上执行代码来消除缺陷。

(6) 信息泄露

协议丢失的信息 (如 DNS 有丰富的数据来源) 易被攻击者利用，攻击者借助这些信息攻破系统 (如口令猜测、欺骗攻击等)

(7) 指数攻击

攻击者在网络上放置病毒，用户浏览时被病毒入侵，导致计算机中毒，信息被窃取。病毒需要依附其他程序，而蠕虫是一种更为特殊的病毒，可以自行传播。当计算机被感染后，病毒会以此为宿主继续扫描其他机器进行传播，这种递归式的传播速度是指数级的。

(8) 拒绝服务

过度使用服务，使软件、硬件过度运行，网络连接超出容量，最终导致关机或系统瘫痪，降低服务质量。

其中有一种分布式拒绝攻击，攻击者使用一台主傀儡机控制多个傀儡机，然后同时向某个目标发起攻击，导致其拒绝服务。

1.3 安全服务

(1) 鉴别/认证

鉴别/认证的功能就是向接受者保证消息是他所要求的源，保证通信的真实性。

- 对等实体鉴别/对等实体认证: 在数据交互连接建立 (如 TCP) 中确认对等实体的身份，保证实体没有被假冒，对等实体是不同系统中应用相同协议的两个实体。
- 数据源鉴别/数据源认证: 对数据单元的来源提供确认，向接收方保证所接受到的数据单元来自要求的源，但不提供对数据单元的复制和改写的保护。

(2) 访问控制

防止未授权的用户非法使用系统资源，每个试图获取访问权限的实体必须要先认证。

(3) 数据保密性

保护传输中的数据不会遭受被动攻击，防止流量数据遭受窃听分析。

连接保密性 保护一次连接中所有的用户数据

无连接保密性 保护单个数据块里的所有用户数据

选择域保密性 对一次连接或单个数据块里选定的数据部分提供保密性

流量保密性 保护那些可以通过观察流量而获得的信息

(4) 数据完整性

确保被认证实体发送的数据与接收到的数据完全相同，防止非法实体对数据进行修改、插入、删除以及数据交换中的数据丢失。

具有恢复功能的连接完整性 提供一次连接中所有用户数据的完整性、检测整个数据序列内存在的修改、插人、删除或重放，且试图恢复之。

无恢复的连接完整性 同上，但仅提供检测，无恢复

选择域连接完整性 提供一次连接中传榆的单个数据块用户数据中选定部分的数据完整性，并判断选定域是否有修改、插入、删除或重放

无连接完整性 为单个无连接数据块提供完整性保护，并检测是否有数据修改。另外，提供有限的重放检测

选择域无连接完整性 为单个无连接数据块内选定域提供完整性保护; 判断选定域是否被修改

(5) 抗否认性/不可抵赖性

防止发送方在发送数据后否认发送和接收方在收到数据后否认收到或伪造数据的行为。

- 具有源点证明的不能否认：为数据接收者提供数据源证明，防止发送者以后任何企图否认发送数据或它的内容。
- 具有交付证明的不能否认：为数据发送者提供数据交付证明，防止接收者以后任何企图否认接收数据或它的内容。

1.4 安全机制与模型

特定安全机制	普适的安全机制
为提供 OSI 安全服务，可能合并到适当的协议层中 **加密** 使用数学算法将数据转换为不能轻易理解的形式。这种转换和随后的数据恢复都依赖于算法本身以及零个或者更多的加密密钥 **数字签名** 为了允许数据单元接收方证明数据源和数据单元的完整性，并且防止数据伪造（例如，通过接收方）而将数据附加到数据单元中或者对数据单元进行密码变换 **访问控制** 强制执行对资源的访问权限的各种机制 **数据完整性** 确保数据单元或者数据单元流完整性的各种机制 **认证交换** 通过信息交换以确保一个实体身份的一种机制 **流量填充** 通过填充数据流空余位的方式来干扰流量分析 **路由控制** 支持对某些数据的特定物理安全通道的选择，并且允许路由改变，特别是当安全性受到威胁时 **公证** 使用可信第三方以确保某种数据交换的属性	没有指定特定 OSI 安全服务或者协议层的机制 **可信功能** 相对于某个标准而言正确的功能（例如，由安全策略建立的标准） **安全标签** 绑定在资源（可能是数据单元）上的记号，用来命名或者指定该资源的安全属性 **事件检测** 与安全相关事件的检测 **安全审计跟踪** 收集可能对安全审计有用的数据，它对系统记录和活动进行单独的检查和分析 **安全恢复** 处理来自机制的请求，例如事件处理和管理功能，并且采取恢复措施

图 1: 安全机制

服　务	加密	数字签名	访问控制	数据完整性	认证交换	流量填充	路由控制	公证
对等实体认证	Y	Y			Y			
数据源认证	Y	Y						
访问控制			Y					
机密性	Y						Y	
流量机密性	Y					Y	Y	
数据完整性	Y	Y		Y				
不可抵赖性		Y		Y				Y
可用性				Y	Y			

图 2: 安全服务与安全机制的关系

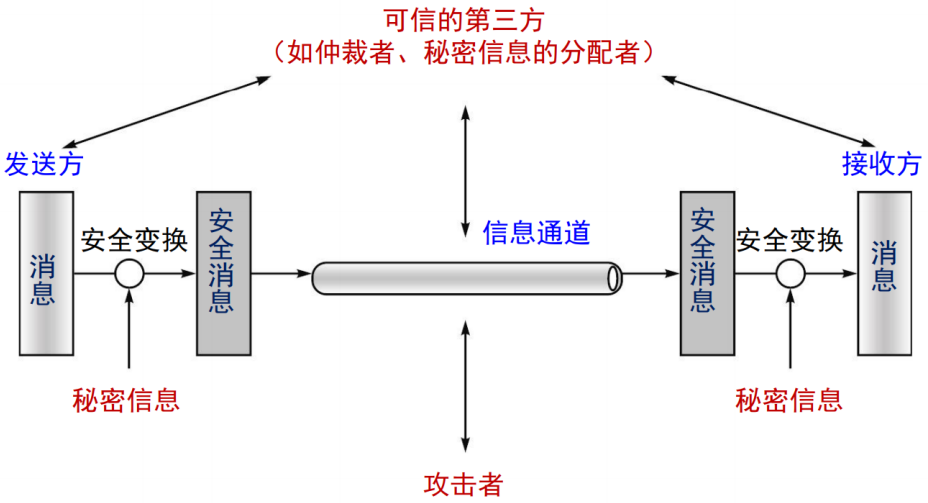

图 3: 网络安全模型

2 Cryptography

2.1 密码体制

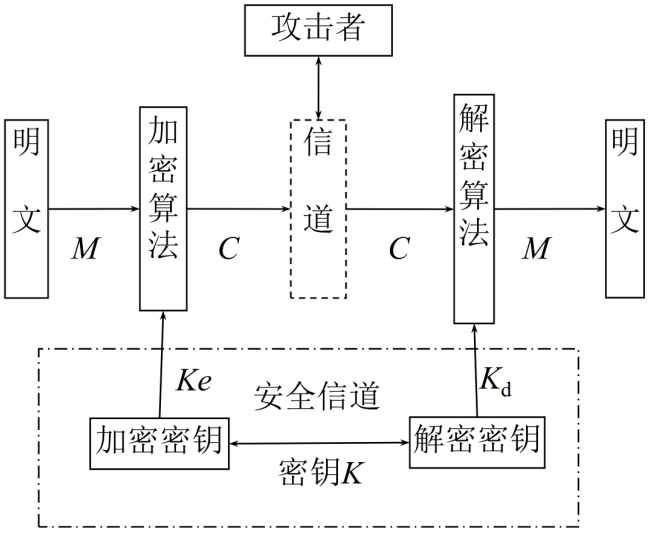

图 4: 密码体制

如果一个密码体制的 $K_d = K_e$ 或由其中一个很容易推出另一个密码体制则称为单密钥密码体制或对称密码体制或传统密码体制的分类，否则称为双密钥密码体制。

如果在计算上 K_d 不能由 K_e 推出, 这样将 K_e 公开也不会损害 K_d 的安全，于是便可将 K_e 公开，这种密码体制称为公开密钥密码体制。

根据对明文和密文的处理方式和密钥的使用不同，可将密码体制分为分组密码体制和序列密码体制。

2.2 密码分析

定义: 通过分析, 从截获的密文中推断出原来的明文。

攻击密码系统的方法有

- 穷举攻击: 枚举所有可能的密钥
- 统计分析攻击: 分析统计规律
- 数学分析攻击: 针对加解密算法的数学基础和某些密码学特性，通过数学求解的方法来破译密码

破译密码的类型有

- 唯密文攻击: 分析者掌握了密码算法及明文统计特性，并截获一个或多个用同一密钥加密的密文，通过对这些密文进行分析求出明文或密钥。
- 已知明文攻击: 分析者掌握了若干个明文和对应的密文，以此来分析求出加密算法或密钥。
- 选择明文攻击: 攻击者事先选择一定数量的明文，让加密算法加密，得到相应的密文，通过分析密文求出加密算法或密钥。
- 自适应选择明文攻击: 选择明文攻击的特殊情况。密码分析者不仅能选择被加密的明文，而且也能基于以前加密的结果修正这个选择。
- 选择密文攻击: 攻击者掌握对解密机的访问权限，可构造任意密文所对应的明文。在此种攻击模型中，密码分析者事先任意搜集一定数量的密文，让这些密文透过被攻击的加密算法解密，透过未知的密钥获得解密后的明文。

2.3 密码体质的安全性

- 无条件安全性: 如果密码分析者具有无限的计算能力，密码体制也不能被攻破，那么这个密码体制就是无条件安全的。
- 计算安全性: 如果攻破一个密码体制的最好的算法用现在或将来可得到的资源都不能在足够长的时间内破译，这个密码体制被认为在计算上是安全的。
- 可证明安全性: 密码体制的安全与一个问题是相关的，并没有证明密码体制是安全的，可证明安全性也有时候被称为归约安全性。

3 DES

3.1 DES 描述

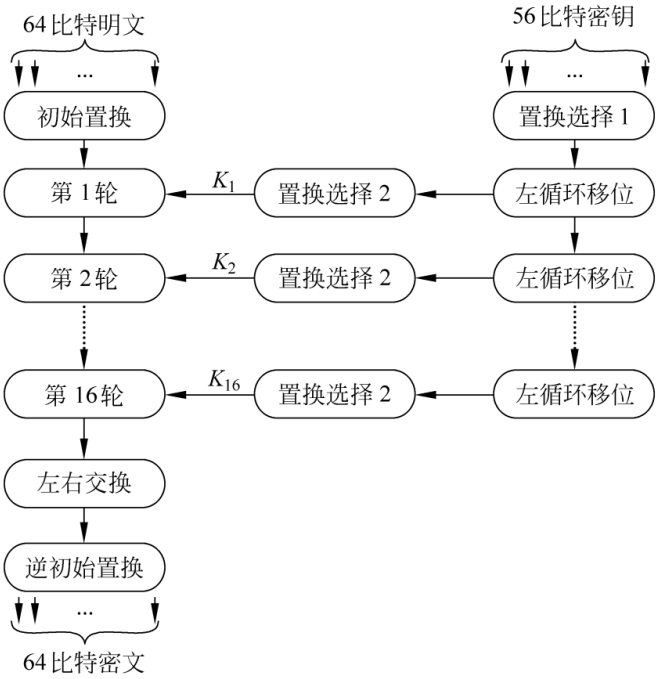

图 5: DES 加密算法框图

3.1.1 初始置换

通俗地说，置换盒上的 A_i=k 表示将明文上的第 k 个比特放在 A_i 处，例如 A_1=58 表示将明文的第 58 个比特置换在 A_1 处。逆初始置换盒同理，且 $X = IP^{-1}(IP(X))$

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

图 6: 初始置换 IP 盒与逆初始置换 IP 盒

3.1.2 轮结构

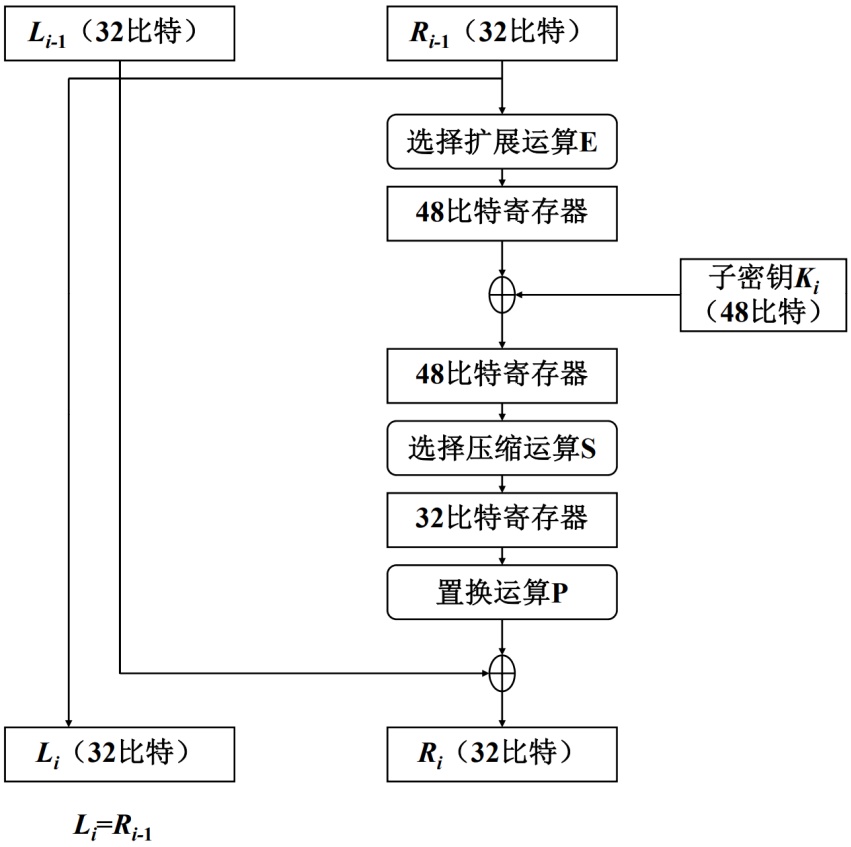

图 7: DES 的一轮迭代

1. 将 64 比特分成 L_{i-1} 和 R_{i-1} 两部分
2. 对 R_{i-1} 进行扩展运算 E: 在每一行的首尾按顺序补充两个数字 (见图 8)
3. 扩展后得到的 48 位 R_{i-1} 与当前轮的子密钥 K_i 做异或运算
4. 压缩运算 S:48 位分成 8 个 6 比特，假设 B=110011，令令 $B_1B_6 = 11_2 = 3$ 为行号，$B_2B_3B_4B_5 = 1001_2 = 9$ 为列号，B = bin(S[3][9]) = bin(11) = 1011，S 盒构造见图 9
5. 置换运算 P: 和初始置换 IP 一样，P_i=k 表示将 k 个比特置换在 P_i，P 盒构造见图 8
6. $L_i = R_{i-1}$,$R_i = L_{i-1}$ 异或 R_{i-1}

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

图 8: 扩展运算 E 和置换运算 P

行＼列		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S_1	0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
	1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
	2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
	3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

图 9: S 盒构造 (共 8 个)

3.1.3 子密钥的产生

1. 通过 PC-1 盒置换，然后分成两个 28 位的 C_{i-1}, D_{i-1}
2. 循环对 C_{i-1}, D_{i-1} 进行循环左移得到 C_i, D_i，每一轮移的位数不同 (见图 10)
3. C_i, D_i 合并后通过 PC-2 盒置换得到第 i 轮迭代用的子密钥 K_i

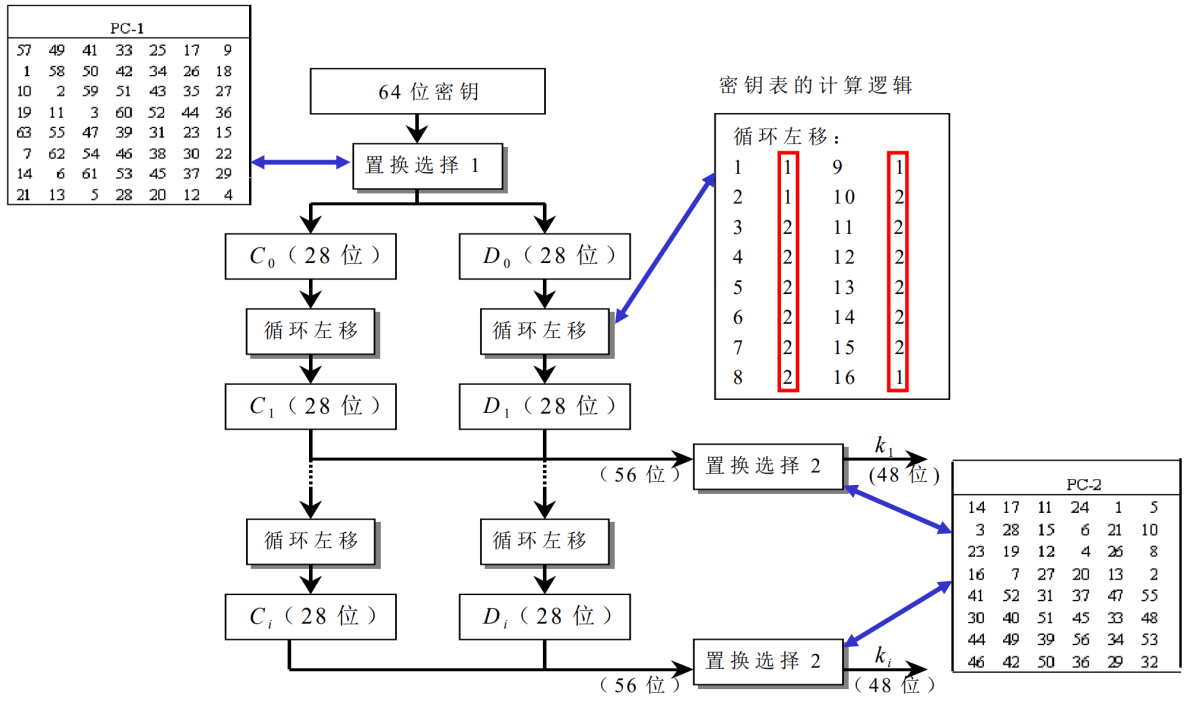

图 10: 子密钥流程图

3.2 DES 的工作模式

(1) 电子密码本 ECB: 将明文按 64 比特分割，对每一段单独使用 DES 加密

- 可以并行实现
- 不能隐藏明文模式：即相同明文生成相同密文
- 误差传递: 密文块损坏-> 仅对应明文块损坏

(2) 密码分组链接 CBC: 将明文按 64 比特分割得到 $P_1, P_2, P_3...$，用初始化向量 IV 与 P_1 异或后再进行对 P_1 加密，得到的密文 C_1 与 P_2 异或后再进行对 P_2 的加密，以此类推。解密时，将 C_1 解密出的结果与 IV 异或得到 P_1，将 P_2 解密出的结果与 C_1 异或得到 P_2，以此类推。

- 不能并行
- 相同明文生成不同密文
- 密文块损坏-> 两明文块损坏

(3) 密码反馈 CFB: 不再限定明文长度为 64 比特，而是 1-64 的任意值 s，为了表达方便，假定 s=8，然后声明一个初始化向量 IV，则 CFB 的加密流程为:

- 首先对 IV 进行 DES 加密，得到 64 比特密文 C

- 将 8 比特的明文 P_1 与 C[0:8] 进行异或得到密文 C_1
- 更新 IV = IV[8:]+C_1，继续进行对 P_2 的加密

CFB 的解密流程为:

- 首先对 IV 进行 DES 加密，得到 64 比特密文 C
- 将 8 比特的密文 C_1 与 C[0:8] 进行异或得到明文 P_1
- 更新 IV = IV[8:]+C_1(注意这里仍为 C_1 而非 P_1)，继续进行对 P_2 的解密

特点:

- 分组密码-> 流密码，解除长度限制
- 不能并行
- 相同明文生成不同密文
- 一个单元损坏影响多个单元

(4) 输出反馈 OFB: 与 CFB 稍有不同

- 首先对 IV 进行 DES 加密，得到 64 比特密文 C
- 将 8 比特的明文 P_1 与 C[0:8] 进行异或得到密文 C_1
- 更新 IV = IV[8:]+C[0:8]，继续进行对 P_2 的加密

OFB 解密流程为:

- 首先对 IV 进行 DES 加密，得到 64 比特密文 C
- 将 8 比特的密文 C_1 与 C[0:8] 进行异或得到明文 P_1
- 更新 IV = IV[8:]+C[0:8]，继续进行对 P_2 的解密

相比于 CFB，一个单元损坏只影响对应单元

3.3 常规加密

(1) 链路加密: 对所有消息进行加密，到达节点后解密，然后用下一个链路层的密钥再加密进行传输。优点: 掩盖了消息源点、终点、长度、频率。缺点: 密钥分配难以管理，对网络性能有影响。

(2) 节点加密: 与链路加密不同，解密再加密的过程是在节点上的一个安全模块进行，且不对报头和路由信息加密，因此安全性较差。

(3) 端到端加密: 仅在源点加密一次，在到达终点之前始终不解密。优点: 便宜，易于实现。缺点: 不对源点终点加密，因此安全性也较差。

4 PKCS & RSA

4.1 公钥密码体制原理

公开公钥 PK_B，任何人想对 B 发送信息，用公钥对消息进行加密发送，只有 B 知道密钥，因此只有 B 能进行解密。

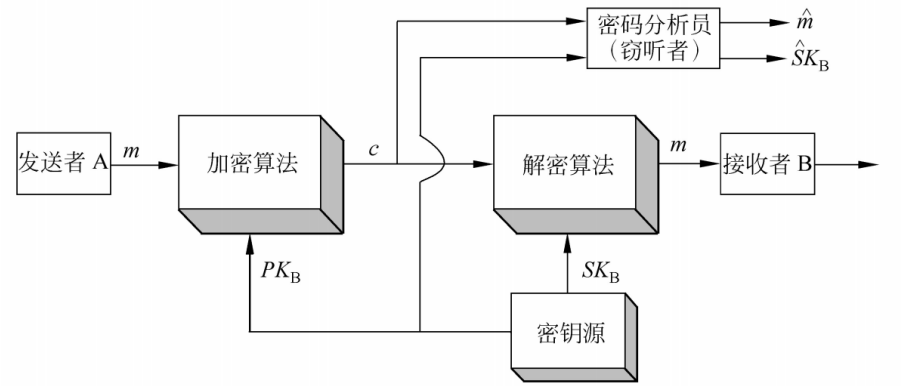

图 11: 公钥密码体制加密框图

公开公钥 PK_A，A 用密钥加密 m 得到 c 充当自己的数字签名，其他人可以用公钥尝试解密来验证是否是 A 的签名。

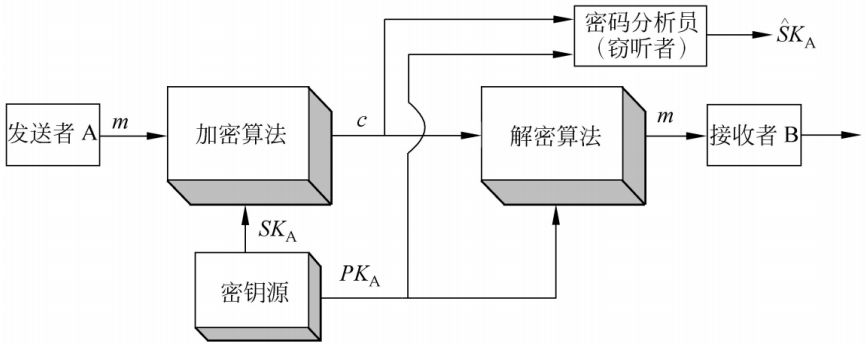

图 12: 公钥密码体制认证框图

4.2 RSA 算法

4.2.1 原理

1. 选取两个质因数 p,q, 假定 p = 47,q = 71, 则公钥 n = 47*71 = 3337, ψ(n) = (p-1)*(q-1) = 3220
2. 随机选取一个与 ψ(n) 互质的数 e, 假定 e=79, 公开 e
3. 计算密钥 d 使得 (d*e)%ψ(n)=1, 这里 d=1019, 保密 d
4. 加密: C=$(P)^e$%n, 假定明文 P=688, C = $(688)^{79}$%3337=1570
5. 解密: P=$(C)^d$%n, P=$(1570)^{1019}$%3337=668, 明文恢复

4.2.2 应用

除了加密和数字签名之外，还有:

(1) 双重加解密:A 有 (e_1, n^1, d^1),B 有 (e_2, n^2, d^2),A 想发消息 m 给 B, 首先用 B 的公钥加密确保只有 B 能解密, 然后再用自己的私钥加密让 B 知道这真的是 A 发的,$c = ((m^{e_2})\%n_2)^{d_1}\%n_1$,B 解密时先用 A 的公钥解密再用自己的私钥解密,$m = ((c^{e_1})\%n_1)^{d_2}\%n_2$。

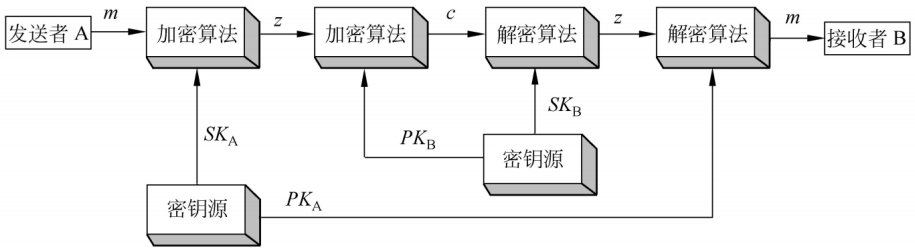

图 13: 公钥密码体制认证框图

(2) 密钥交换: 假定 A 与 B 使用 DES 进行通讯，那么应用加密密钥 K_1= 解密密钥 K_2，现在采用如下方案:B 随机的产生一个 K_1, 然后用 A 的 RSA 公钥 (e,n) 进行加密得到 K_2，然后直接将 K_2 公布，而只有 A 能通过 K_2 解出 K_1，再用 K_1 去解密 DES，这样仅需用 RSA 加密 DES 的密钥而无需加密信息本身 (因为 DES 的执行速度是 RSA 的 100-1000 倍)。