1.**OSI 安全架构** **安全攻击:**任何危及信息系统安全的行为。**安全机制:**用来检测阻止攻击或从攻击状态恢复到正常状态的过程(或实现该过程的设备)。**安全服务:**加强数据处理系统和信息传输的安全性的一种处理过程或通信服务，目的在于利用一种或多种安全机制进行反攻击。

2.**安全服务的定义及种类** X.800 定义：在通信开放系统中，为系统或数据传输提供足够安全的协议层服务。RFC2828 定义:安全服务是一种由系统提供的对系统资源进行特殊保护的处理或通信服务；安全服务通过安全机制来实现安全策略 。种类：数据机密性、数据完整性、消息认证、用户认证、不可抵赖性、访问控制。

3.**安全机制的定义及种类** 特定安全机制:在特定协议层实现：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；普遍安全机制:可信功能、安全标签、事件检测、安全审计跟踪、安全恢复。

4.**对称加密方案有五个基本成分:**明文 加密算法 密钥 密文 解密算法

5.**对加密信息的攻击类型：**唯密文攻击、已知明文攻击、选择明文攻击、选择密文攻击、选择文本攻击

6.**古典密码类型及攻击方式：**凯撒密码 穷举；单表代替 字母频率统计； 多表代替(playfair 密码)；流加密 一次一密才能保证安全(按位异或)；

7.**流加密与分组加密：**流密码每次加密数据流的一位或一个字节；分组密码将一个明文分组作为整体加密，通常得到与明文等长的密文分组，典型的分组大小是 64 位或 128 位。

8.**Feistel 密码:**交替使用代替(明文元素唯一替换为密文元素)和置换(明文元素顺序改变)；是对香农提出的交替使用混淆和扩散的乘积密码的实际应用，代替置换网络(SP)。

扩散和混淆：扩散就是使明文的统计特征消散在密文中，可以通过让每个明文数字尽可能地影响多个密文数字获得，等价于每个密文数字被许多明文数字影响 。混淆：尽可能使密文和加密密钥间的统计关系更加复杂，以阻止攻击者发现密钥。

Feistel 结构参数：分组长度(传统 64 位，AES128 位)、密钥长度、迭代轮数、子密钥产生算法、轮函数、快速软件加解密、简化分析难度。加解密本质一样逆过程，异或特性。

9.**DES(数据加密标准)：**feistal 基础上，采用 64 位分组长度和 56 位密钥(64 位，8 位一个奇偶校验码)，64 位的输入经 16 轮变换得 64 位输出，解密使用相同步骤和相同密钥。

细节：①.IP 置换 输入的 64 位数据块按位重新组合，输出 L0、R0 两部分，各长 32 位。②.密钥置换 由 56 位密钥产生 48 位子密钥。 ③.E 扩展置换 IP 置换后获得右半部分 R0，将 32 位输入扩展为 48 位输出。目的：生成与密钥相同长度的数据进行异或运算；提供更长的结果，在后续的替代运算中可以进行压缩。④.S 盒代替 压缩后的密钥与扩展分组异或后得到 48 位的数据，送入 S 盒，进行替代运算。代替过程产生 8 个 4 位的分组，组合在一起形成 32 位数据。S 盒代替是 DES 算法的关键步骤，所有的其他的运算都是线性的，而 S 盒是非线性的，相比于其他步骤，提供了更好安全性。⑤.P 盒置换 该置换把输入的每位映射到输出位，P 盒置换的结果与最初的 64 位分组左半部分 L0 异或，然后左、右半部分交换，接着开始另一轮。⑥.IP-1 末置换 初始置换的逆过程，DES 最后一轮后，左、右两半部分并未进行交换，而是两部分合并形成一个分组做为末置换的输入。

DES 算法特点 分组加密算法：64 位为分组，64 位明文输入，64 位密文输出。对称算法：加解密使用同一秘钥，有效秘钥长度为 56 位 代替和置换：DES 算法是混乱和扩散加密技术的组合,先替代后置换。易于实现:算法只使用标准算术逻辑运算，作用数最多只有 64 位，硬件技术很容易实现,算法重复特性使得可以非常理想地用在一个专用芯片中。

DES 强度：密钥长度 56 位，目前穷举攻击可以实现；分析攻击：差分密码攻击、线性密码攻击，针对密钥；计时攻击:本质上通过观察算法的一个既定实现对多种密文解密的所需时间来获得关于密钥或明文信息。

10.**雪崩效应：**明文或密钥的某一位发生变化会导致密文的很多位(接近一半)发生变化。

11.**分组密码设计:**Feistel 强度来源：迭代轮数、子密钥产生算法、轮函数；迭代越多，密码分析越困难；核心是函数 F,非线性，提供混淆功能。

12.**不可约多项式 m(x)**：当且仅当该多项式不能表示为次数小于 m(x)的次数的两个多项式的乘积。

13. **AES(高级加密标准)：**AES 在 8 位的字节上运算。字节加定义为对位异或操作。字节乘定义为有限域 GF(2^8）内的乘法，其中不可约多项式 m(x)为 $m(x) = x^8+x^4+x^3+x+1$。明文分组的长度为 128 位即 16 字节，密钥长度可以为 16 字节(运算 10 轮)，24 字节(12 轮)或 32 字节(14 轮)。

总体结构：①.它不是 Feistel 结构，Feistel 结构中数据分组中的一半被用来修改数据分组中的另一半，然后交换这两部分。AES 算法在每一轮都使用代替和混淆将整个数据分组作为一个单一的矩阵处理；②.对于 10 轮运算，输入的密钥被扩展成由 44 个 32 位字所组成的数组,每轮有四个不同的字(128 位)作为该轮的轮密钥。③.由 4 个不同的阶段组成，包括一个置换和 3 个代替：字节代替 行移位 列混淆 轮密钥加；④.仅在轮密钥加阶段中使用密钥。故该算法以轮密钥加开始，以轮密钥加结束。如果将其他不需要密钥的运算用于算法开始或结束的阶段，不知道密钥的情况下就能计算其逆，不能增加算法的安全性。⑤.轮密钥加实质上是 Vernam 密码形式,即按位异或。另外三个阶段一起提供了混淆、扩散以及非线性功能。⑥.每个阶段均可逆 对字节代替，行移位和列混淆，在解密算法中用与它们相对应的逆函数，轮密钥加的逆就是用同样的轮密钥和分组相异或。⑦.加密执行顺序：(第 0 轮)轮密钥加；(第 1 轮)(字节代替 行移位 列混淆 轮密钥加)；(最后 1 轮)(字节代替 行移位 轮密钥加) 解密顺序：(第 0 轮)轮密钥加；(第 1 轮)(逆向行移位 逆向字节代替 轮密钥加 逆向列混淆)；(最后 1 轮)(逆向行移位 逆向字节代替 轮密钥加)。

14.**Rijndeal 和 AES 有何不同：**Rijndael 允许块长度为 128、192 或 256 位，AES 仅允许一个块长度为 128 位。AES 是由 Rijndael 产生的。

15.**为什么 2-DES 不安全**：①.使用不同的密钥进行两次加密，效果可以约化为单次加密。②.可以采用中间相遇攻击破解，以空间换取时间。

16.**3-DES**:为了对付中间相遇攻击，提出 3-DES 加密，实际用两个密钥进行三次加密，即加密-解密-加密(EDE)。中间用解密，是为了使用三重 DES 的用户可以利用该算法解密单 DES 加密的数据。

17.**分组密码五种模式(可用于块模式或流模式，DES、AES 均可)：①块模式输入：可能需要对最后一个块填充。②.流模式：将分组密码当流密码使用，加解密函数均相同。**

	模式	描述	应用
块模式	电码本模式（ECB）	用相同的密钥分别对明文分组独立加密	单个数据安全传输
	密文分组链接(CBC)	加密算法的输入是上一个密文组和下一个明文组的异或	面向分组通用传输认证
流模式	密文反馈（CFB）	一次处理S位，上一块密文作为加密算法的输入，产生伪随机数输出与明文异或作为下一单元的密文	面向数据流通用传输认证
	输出反馈（OFB）	与CFB类似，只是加密算法的输出是上一次加密的输出，且使用整个分组	噪声信道数据流传输
	计数器（CTR）	每个明文分组都与一个经过加密的计数器相异或，对每个后续分组计数器递增	面向分组通用传输高速传输

块长度处理方式：①.密文窃取 更改后两个明文块的处理方式的技术，可导致对后两个密文块进行重新排序的传输，并且不会扩展密文。通过从最后一个密文块到最后一个明文块（可能是不完整的）填充高阶位（从第二个到最后一个块窃取密文）来实现的。最后一个（现在已满）的块被加密，然后与倒数第二个密文块交换，然后将其截断为最终明文块的长度，删除被盗的比特，从而得到与密文长度相同的密文。②.字节填充：很多标准，补充块长度。

ECB：相同的明文块输入导致相同的密文块输出，对于长消息的输入不利。单个分组中有一个或多个比特错误只会影响该分组明文。

CBC：密文块依赖于前面所有明文块，有重复明文组，也不会被发现。第一块明文和初始向量 IV 异或，需要安全性保证。

CFB：使用密文单元填充移位寄存器，错误出现后，传播几个块后消失；不能重复使用相同的密钥流序列。**OFB:**用加密函数的输出填充移位寄存器，出现比特错误不会传播，虽然对明文进行分组，但不需要填充，只取最左和最后块长度相同位进行计算，其它丢弃。抗消息流篡改攻击能力不如 CFB；具有典型流密码结构，但 OFB 一次加密一个明文分组(64 位或 128 位)，流密码一次加密一个字节。**CTR:**计数器首先被初始化为某值，后随消息块增加计数器值加 1(模 2^b , b 为分组长度)。加密时，计数器加密后与明文分组异或得到密文分组没有链接。解密使用具有相同值的计数器序列，用加密后的计数器的值与密文分组异或来恢复明文分组，故解密时必须知道初始计数器值。需要并确保 key 和计数器值唯一。优点：能够并行处理多块明文(密文)的加密(解密)；可以充分利用能够支持并行运算的各类处理器；加密算法不依赖明密文，可以预处理加密盒，极大提高吞吐量。

18.某些分组密码的操作模式仅使用加密算法而其他的模式既使用加密算法又使用解密算法：某些模式下，明文不进行加密，而是与加密函数的输出进行异或。这可以在数学上证明，想要进行解密，必须使用加密函数，比如异或操作。

19.**随机数的使用：**密钥分发和相互认证方案；会话密钥产生；RSA 公钥加密算法密钥产生；对称流密码加密位流产生。**流密码：**按位异或，类似一次一密，但一次一密使用真正随机数流，流密码使用伪随机数流，他们都不能重复使用。**RC4 算法**：可变密钥长度、面向字节操作的流密码，算法以随机置换作为基础，获得的随机数非线性很好。

20.**对称加密目的：**数据保密，包括安全通信及安全存储；身份验证，CBC-MAC。

21.**密钥层次使用：**①.会话密钥，短期连接使用，断开即丢弃。②.主密钥：用于加密会话密钥，由用户和密钥分发中心长时间共享，可通过物理传递分发。

22.**临时交互号：**可以是一个时间戳、计数器或者随机数，最低要求是每次请求的临时交互号是不同的，而且为了防止伪造还要求敌手猜出该临时交互号是困难的。目的是为了防止重放攻击，通信双方记住所用的交互号，如果发现报文中有以前使用过的交互号，就认为是重放攻击。

24.**KDC(密钥分发中心)：**密钥分发中心是一种被授权将临时会话密钥发送给主体的系统。使用密钥分发中心与目标主体共享的主密钥，以加密形式传输每个会话密钥。

25.**Kerberos 认证过程**：客户机向认证服务器（AS）发送请求，要求得到某服务器的证书，然后 AS 响应包含这些用客户端密钥加密的证书。证书的构成为：a.服务器 “ticket”；b.会话密钥 “session key”。客户机将 ticket （包括用服务器密钥加密的客户机身份和一份会话密钥的拷贝）传送到服务器上。会话密钥可以（现已经由客户机和服务器共享）用来认证客户机或认证服务器，也可用来为通信双方以后的通讯提供加密服务，或通过交换独立子会话密钥为通信双方提供进一步的通信加密服务。

26.**传统密码和公钥密码对比：**传统密码 加解密使用相同的密钥和相同的算法；收发双方必须共享密钥；密钥必须保密；没有其他消息，解密消息不可行；知道算法及部分密文不足以确定密钥；公钥密码 同一算法用于加密和解密，但加解密使用不同密钥；发送方和接收方拥有的密钥不同；两个密钥必须保密一个；后面两条同传统密码最后两条。

27.**单向陷门函数：**一个函数，若计算函数值很容易，并且在缺少一些附加信息时计算函数的逆是不可行的，但是已知这些附加信息时，可在多项式时间内计算出函数的逆。

28.**RSA 算法攻击：**穷举攻击；数学攻击 试图分解两个素数的乘积；计时攻击依赖于解密算法的运行时间；基于硬件故障的攻击；选择密文攻击 这种攻击利用了 RSA 算法性质。

29.**公钥分发：**公开发布 发布简单，易被冒充；公开可访问目录 可信实体负责公开目录维护和分配，缺点是如果获得该实体私钥，可任意修改信息。公钥授权；公钥证书(CA)。

30.**公钥密码体制应用：**①加解密、②数字签名、③密钥交换。算法：RSA(①②③)、椭圆曲线(①②③)、Diffie-Hellman(③)、DSS 数字签名标准(②)。

31.**椭圆曲线安全性：**依据椭圆曲线对数难以计算；安全性相同情况下，ECC 使用位数更短；密钥长度相同时和 RSA 计算量差不多；类似的安全性，ECC 提供了显着的计算优势。

32.**公钥证书：**证书包含公钥和公钥所有者的标识，整个数据块由第三方进行签名。通常，第三方是证书管理员，被用户信任。一个用户以一种安全的方式将他的公钥交给管理员，从而获得一个证书，接着用户就可以公开证书。任何需要该用户公钥的人都可以获得该证书，并通过查看附带的可信签名来验证证书的有效性。通信一方也可以通过传递证书的方式将他的密钥信息传达给另一方。其他通信各方可以验证该证书确实是证书管理员生成的。

33.**公钥密码体制主要成分：**明文、加密算法、公钥和私钥、密文、解密算法。

34.**为满足安全算法，公钥体制要求：**①.B 产生一对密钥(公钥 PU_b，私钥 PR_b)在计算上是容易的。②.已知公钥和要加密的消息 M，发送方 A 产生相应的密文在计算上是容易的 $C = E(PU_b, M)$ ③.接收方 B 使用其私钥对接收的密文解密以恢复明文在计算上是容易的 $M=D(PR_b,C)$ ④.已知公钥 PU_b 时,攻击者要确定私钥 PR_b 在计算上是不可行的。⑤.已知公钥 PU_b 和密文 C，攻击者要恢复明文 M 在计算上是不可行的。⑥.加密和解密函数的顺序可以交换。

35.**单向函数：**将一个域映射到一个范围内的函数，以使每个函数值都具有唯一的逆，条件是该函数的计算很容易而反函数的计算是不可行。

36.**关于选择密文攻击的例题：**想求 $M=C^d \bmod n$。选择一个小于 n 随机数 r，并计算 $Z = r^e \bmod n$　$X = ZC \bmod n$　$t = r^{-1} \bmod n$　$Y = x^d \bmod n$。已知 $Z = r^e \bmod n$，有 $r = Z^d \bmod n$, $tY \bmod n = ((r^{-1} \bmod n)(x^d \bmod n)) \bmod n = (r^{-1} Z^d C^d) \bmod n = C^d \bmod n = M$

37.**简要说明 D-H 密钥交换：**两方分别创建一个公钥和私钥对，并将公钥传达给另一方。密钥的设计方式是，双方可以根据双方的私钥和另一方的公钥计算相同的唯一秘密密钥。

38.**椭圆曲线及零点：**三次方程式描述的曲线，类似于用于计算椭圆周长的曲线，形如：$y^2+axy+by=x^3+cx^2+dx+e$；零点在椭圆曲线算法中也称为无穷大点，并由加法恒等式指定。

39.**Hash 函数：**任意长度输入-定长输出，输出结果随机均匀，雪崩效应；**应用：**消息认证；数字签名；单向口令；保存密码；生成伪随机数；

安全性需求	描述
输入长度可变	应用于任意大小数据块
输出定长	定长输出
效率	对于给定x，计算H(x)容易
单向性	给定hash值h,计算H(y)=h的y在计算上不可行
抗弱碰撞	给定x,满足y!=x且H(x)=H(y)的y计算不可行
抗强碰撞	任何满足H(x)=H(y)的偶对(x,y)计算上不可行
伪随机性	满足为随机性测试标准

一个函数如果是抗强碰撞的，那么也同时是抗弱碰撞的，但反之则不一定成立。一个函数可以是抗强碰撞的，但不一定是抗原像攻击(单向性)的，反之亦然。一个函数可以是抗弱碰撞的，但不一定抗原像攻击。

压缩函数作用：如果压缩函数有抗碰撞能力，则迭代哈希也具有，典型的哈希函数使用压缩函数作为基本构建块，并涉及压缩函数的重复应用。

SHA1：输入最大长度小于 2^64，输出 160 位消息摘要，输入分组 512 比特；SHA2：输入最大长度小于 2^128，输出 512 位消息摘要，输入分组 1024 比特；

SHA 使用基本算数逻辑函数：模 2^64 或 2^32 的加法运算，循环移位，基于 AND，OR，NOT 和 XOR 的布尔运算。

40.**消息认证码(MAC)：**利用密钥生成固定长度短数据块，并将该数据块附加在消息后。MAC 算法不要求可逆性，而加密算法必须是可逆的；MAC 函数是多对一函数；

41.**消息认证：**保护数据完整性、验证发起人身份、不可否认来源(不会来自第三方)。**任何消息认证或数字签名机制两层功能：**下层中一定有某种产生认证符的函数 认证符是一个用来认证消息的值；上层协议中将该函数作为原语使接收方可以验证消息的真实性。产生认证符函数：Hash 函数、消息加密、MAC(收发双方共享密钥,不能提供数字签名)

42.**HMAC：**基于 Hash 的 MAC，HMAC 的经过验证的安全性与基础哈希算法的安全性有关 暴力攻击 生日攻击(但由于键入密钥，因此需要观察大量消息)

43.**消息认证是为了对付哪些类型的攻击？**伪装：从欺诈来源向网络中插入消息。内容修改：对消息内容的更改，包括插入，删除，转置和修改。序列修改：对双方之间的消息序列的任何修改，包括插入，删除和重新排序。定时修改：延迟或重放消息。在面向连接的应用程序中，整个会话或消息序列可以是某些先前有效会话的重播，或者序列中的单个消息可能会延迟或重放。在无连接应用中，单个消息（例如，数据报）可能被延迟或重放。

对消息认证码的攻击：重放攻击(加临时交互号) ；密钥推测攻击：暴力破解和生日攻击，要保证不能根据 MAC 值推测出通信双方使用的密钥

44.**MAC 和单向 Hash 函数区别** 散列函数本身不提供消息身份验证。必须以某种方式将密钥与哈希函数一起使用以进行身份验证。MAC 使用秘密密钥来计算身份验证的代码。

45.**消息认证和数字签名：**消息认证可以保护信息交换双方不受第三方的攻击 ，但是它不能处理通信双方自身发生的攻击；数字签名可以解决，具有认证功能：能验证签名者 、签名日期和时间、能认证被签的消息内容、签名应能由第三方仲裁，以解决争执。**(即数字签名性质)**

46.**数字签名安全性需求(要求)：**签名必须是与消息相关的二进制位串；签名必须使用发送方某些独有的信息．以防伪造和否认；产生数字签名比较容易；识别和验证签名比较容易；伪造数字签名在计算上是不可行的。无论是从给定的数字签名伪造消息，还是从给定的消息伪造数字签名，在计算上都是不可行的；保存数字签名的副本是可行的。

47.**DSA:**使用的是只提供数字签名功能的算法。与 RSA 不同，DSA 虽然是一种公钥密码方案,但不能用于加密或密钥交换。大概步骤： Hash 函数产生的 Hash 码和随机数 k 作为签名函数的输入,签名函数依赖于发送方的私钥 PRa 和全局公钥 PUg，接收方对接收到的消息产生 Hash 码和签名一起作为输入，验证函数依赖于 PUg 和 PUa，若验证函数的输出等于签名中的 r 成分，则签名是有效的。

48.**消息认证中的争议：**假设 A 向 B 发送了经过身份验证的消息。可能会发生以下争执：1.B 可能会伪造另一封邮件，并声称该邮件来自 A。B 只需创建一条消息并使用 A 和 B 共享的密钥附加验证码即可。 2. A 可以拒绝发送消息。由于 B 可能伪造一条消息，因此无法证明 A 实际上确实发送了该消息。

49.**签名函数和保密函数作用顺序：**重要的是**先执行签名功能**，然后再执行外部保密功能。如有争议，某些第三方必须查看该消息及其签名。如果签名是根据加密消息计算得出的，则第三方也需要访问解密密钥才能读取原始消息。但是，如果签名是内部操作，则接收者可以存储纯文本消息及其签名，以供以后在争议解决中使用。

50.**直接数字签名威胁：**方案的有效性取决于发送方私钥的安全性。如果发件人想否认以前发送过的消息，则发件人可以声称私钥已丢失或被盗，并且其他人伪造了他或她的签名。另一个威胁是，X 的私钥可能在 T 时刻被盗，攻击者可以用 X 的签名发送消息，并加盖 T 或 T 之前的时间戳。

51.**对称加密和错误控制码一起用于消息认证时，先错误认证，后加密**。由于攻击者很难产生密文，使得解密后其错误控制位是正确的，因此内部错误控制可以提供认证；如果放在外部，那么攻击者可以构造具有正确错误控制码的消息，虽然攻击者不知道解密后的明文是什么，但他可以造成混淆并破坏通信。

52.**DAA 生成消息身份验证代码是否安全：**不安全，DAA 采用 DES 运算的 CBC 工作模式，其初始向量为 0，需要认证的数据分成连续的 64 位分组，若最后分组不足 64 位，则在其后填充 0 补足；给定一个消息分组 X 的 CBC MAC 码，如 $T = MAC(K, X)$，则攻击者马上就知道对于两个消息分组 $X \| (T \oplus X)$ 的 CBC MAC 码还是 T。显然，对于短消息，攻击者可以很容易就找到符合条件的 M'使得 $MAC(K, M') = MAC(K, M)$。

53.**用户认证和消息认证区别：**用户认证:核实身份的方法是由或是对一个系统实体提出的(实时系统)。认证方法包括两步 鉴定阶段及核实阶段；消息认证允许通信双方验证收到的消息是否被更改以及资源是否可信。

用户认证和数字签名区别：用户认证的消息语义基本固定，会被实时证实或拒绝；数字签名消息是动态的，主要为了保证不可抵赖性。一定条件下用户验证可以转化为数字签名。

54.**固定密码的攻击及对策：**攻击：密码重放、详尽的密码搜索、密码猜测和字典攻击；对策：加长密码长度、加盐、限制猜测密码的次数等。

55.**零知识证明：**证明者能够在不向验证者提供任何有用的信息的情况下，使验证者相信某个论断是正确的。零知识证明实质上是一种涉及两方或更多方的协议，即两方或更多方完成一项任务所需采取的一系列步骤。证明者向验证者证明并使其相信自己知道或拥有某一消息，但证明过程不能向验证者泄漏任何关于被证明消息的信息。

56.**防止重放攻击的方法：**①.消息上附加序列号，仅当新消息的序列号顺序正确时，该消息才被接受。②.甲方仅在该消息包含时间戳与甲对当前时间的了解足够接近时，才接受该消息。这要求各个参与者之间的时钟同步。③.甲方期望从乙方收到新消息，首先向乙方发送一个随机数，并要求从乙方收到的后续消息（响应）包含正确的随机数值。

57.**抑制重放攻击：**当发件人的时钟早于预期收件人的时钟时，对手可以拦截来自发件人的消息，并在消息中的时间戳记在收件人所在站点成为当前时间戳之后，稍后重播。

58.**kerberos 主要处理问题**：Kerberos 解决的问题是：假设一个开放的分布式环境，工作站中的用户希望访问分布在整个网络中的服务器上的服务。 我们希望服务器能够限制对授权用户的访问并能够对服务请求进行身份验证。 在这种环境下，无法信任工作站以正确识别其用户以使用网络服务。

59.kerberos 定义的四个要求：安全：网络窃听者不应获得模仿用户的必要信息。可靠：对于所有依赖 Kerberos 进行访问控制的服务，缺少 Kerberos 服务的可用性就意味着缺少受支持服务的可用性。 因此，Kerberos 应该是高度可靠的，并且应该使用分布式服务器体系结构，并且一个系统能够备份另一个系统。透明：理想情况下，用户不应超出输入密码的要求而不会意识到正在进行身份验证。可扩展：该系统应能够支持大量客户端和服务器。

60.**列出验证用户身份的一般方法：**用户名/密码、PIN、动态口令、USB key

61.**加盐作用：**密码是由用户设定，实际用户设置的密码复杂度不够高，不同用户有可能使用相同密码，这些用户对应密文也会相同，如果存储用户密码的数据库泄露，攻击者很容易找到相同密码的用户，降低破解密码难度，故在对用户密码进行加密时，需要考虑对密码进行掩饰，即使是相同的密码，也应该要保存为不同的密文，即使用户输入的是弱密码，也需要考虑进行增强，从而增加密码被攻破的难度，而使用带盐的加密 hash 值便能满足该需求。

62.**习题 14.1** 第二问：安全等级相同，但如果 B 拒绝 A 连接，就不用通过 KDC，减小了开销。**习题 14.2：**按步骤来，A 向 KDC 发送的消息被 Z 截获，Z 将信息修改为(IDA,E(Ka,R),Z)，发给 KDC，KDC 将返回 E(Kz,R),被 Z 截获，由于 Z 知道主密钥 Kz,可以获得 R，当 A 向 B 发送 E(R,M)时，Z 截获便可解密获得 M。

63.**关于 kerberos 例题：**当 Bob 收到一个来自于 Alice 的票据，怎样判断是否真实？是否确实来自 Alice?若 Alice 收到一个回复，如何确定消息来自 Bob,且是最新消息?
①它包含由 KDC-Bob 密钥加密的 Alice 的 ID，Bob 的名字和时间戳。②它包含用 KDC-Bob 密钥加密的 Alice 的名字。③.它具有使用会话密钥加密的随机数（例如时间戳）。

64.**基于公钥加密的两种密钥分发方法：**①.直接以明文方式发布公钥 ②.使用公共密钥加密来分发秘密密钥

65.**公钥证书的使用有哪些要求？**①任何通信方可以读取证书并确定证书拥有者的姓名和公钥；②任何通信方可以验证该证书出自证书管理员，而不是伪造的；③.只有证书管理员可以产生并更新证书。④.任何通信方可以验证证书的时效性。

66.**X.509 用途：**X.509 定义了一个框架，用于通过 X.500 目录向其用户提供身份验证服务。该目录可以用作公共密钥证书的存储库。每个证书都包含用户的公钥并由一个可信的签证机构用私钥签名。

67.**证书链：**证书链由不同的证书颁发机构 CA 创建的一系列证书组成，其中每个后续证书都是一个 CA 颁发的证书，该证书用于证明链中下一个 CA 的公钥。

68.X.509 证书如何撤销：公钥的所有者可以发布证书吊销列表，以吊销一个或多个证书。

69.**公钥目录要求：**1.权威机构为每个参与者创建一个带有{name，public key}的目录。2.每个参与者都向目录授权机构注册了一个公共密钥。注册必须通过安全方式。3.参与者可以随时用新密钥替换现有密钥。4.权威定期发布整个目录或对该目录进行更新。5.参与者可以访问该目录。为此，从授权机构到参与者的安全，经过身份验证的通信是强制性的。

70.PKI:公钥基础设施，由硬件软件人策略和程序构成的整套体系，用来创建管理存储分发和撤销建立在非对称密码算法之上的数字证书。PKI 目的就是用来安全便捷高效地获得公钥。

一些数学知识: