**第6章 高级加密标准AES**

**1、Review Questions（课后思考题）：**

**6.3 What is the difference between Rijndael and AES?**

**Rijndael和AES有何不同？**

Rijndael allows for block lengths of 128, 192, or 256 bits. AES allows only a block length of 128 bits.

**6.14 What is the difference between the AES decryption algorithm and the equivalent inverse cipher?**

**AES解密算法和其等价逆算法之间有何不同？**

For the AES decryption algorithm, the sequence of transformations for decryption differs from that for encryption, although the form of the key schedules for encryption and decryption is the same. The equivalent version has the same sequence of transformations as the encryption algorithm (with transformations replaced by their inverses). To achieve this equivalence, a change in key schedule is needed.

**第7章 分组加密的工作模式：**

**1、Review Questions（课后思考题）：**

**7.1 What is triple encryption?**

**什么是三重加密？**

With triple encryption, a plaintext block is encrypted by passing it through an encryption algorithm; the result is then passed through the same encryption algorithm again; the result of the second encryption is passed through the same encryption algorithm a third time. Typically, the second stage uses the decryption algorithm rather than the encryption algorithm.

**7.2 What is a meet-in-the-middle attack?**

**什么是中间相遇攻击？**

This is an attack used against a double encryption algorithm and requires a known (plaintext, ciphertext) pair. In essence, the plaintext is encrypted to produce an intermediate value in the double encryption, and the ciphertext is decrypted to produce an intermediation value in the double encryption. Table lookup techniques can be used in such a way to dramatically improve on a brute-force try of all pairs of keys.

**7.3 How many keys are used in triple encryption?**

**在三重加密中用到多少个密钥？**

Triple encryption can be used with three distinct keys for the three stages; alternatively, the same key can be used for the first and third stage.

**7.4 List and briefly define the block cipher modes of operation.**

**列出分组密码的5种工作模式并简短地给出各自的定义。**

**ECB, 分组后分别加密。**

**CBC, 加密时把上一个分组的密文和当前分组的明文异或后输入给AES算法，输出作为密文。**

**CFB, 当s和block相同时：把上一个分组的密文输入给AES算法，用输出和当前分组的明文异或得到密文。当s小于block时：引入寄存器，把aes的出取s位和明文异或，得到s位的密文，并把这s位的密文存到寄存器中，并挤出最旧的s位。 寄存器的值作为aes的输入。**

**OFB，当s和block相同时：把上一个分组的AES输出输入给AES算法，用输出个当前分组的明文异或得到密文。当s小于block时：也是用跟cfb类似的寄存器方式，区别是aes输出里的s位存到寄存器。**

**CTR，把计数器的值输入给AES算法，用输出和当前分组的明文异或得到密文。**

**7.5 Why do some block cipher modes of operation only use encryption while others use both encryption and decryption?**

In some modes, the plaintext does not pass through the encryption function, but is XORed with the output of the encryption function. The math works out that for decryption in these cases, the encryption function must also be used.

**为什么3DES的中间部分采用了解密而不是加密？**

Encrypt-decrypt-encrypt (EDE) is the preferred method because if a single key is used for all 3 operations it is equivalent to regular 56-bit DES. That is, a 56-bit DES implementation can decrypt that message. This makes this version of 3DES backwards compatible with DES.

Encrypt-encrypt-encrypt (EEE) is also a valid method though. It is no more or less valid than EDE. However, EDE is usually preferred for the reasons mentioned above.

**2、Problems：**

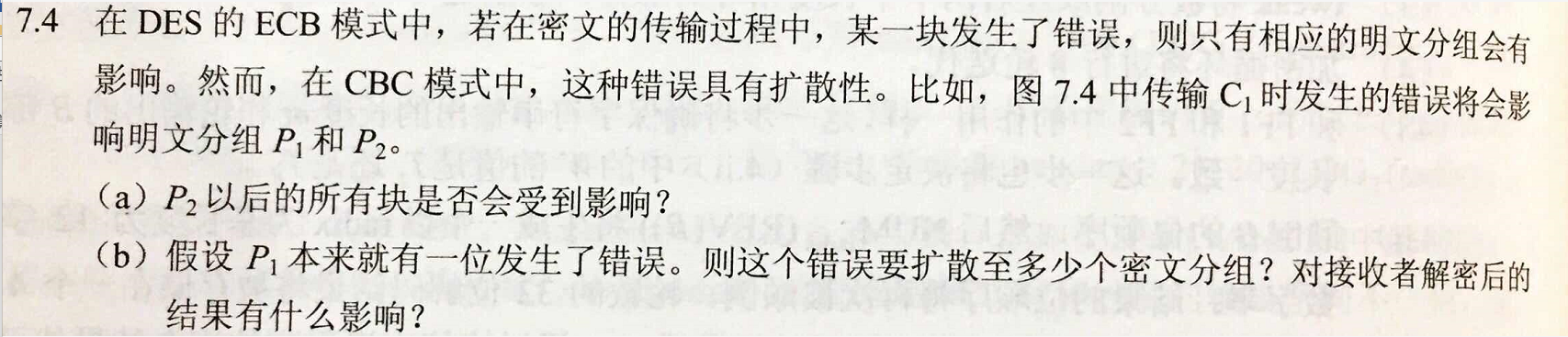
**7.4 With the ECB mode, if there is an error in a block of the transmitted ciphertext, only the corresponding plaintext block is affected. However, in the CBC mode, this error propagates. For example, an error in the transmitted C1(Figure 7.4) obviously corrupts P1 and P2.**

**a. Are any blocks beyond P2 affected?**

No. For example, suppose C1 is corrupted. The output block P3 depends only on the input blocks C2 and C3.

**b. Suppose that there is a bit error in the source version of P1. Through how many ciphertext blocks is this error propagated? What is the effect at the receiver?**

An error in P1 affects C1. But since C1 is input to the calculation of C2, C2 is affected. This effect carries through indefinitely, so that all ciphertext blocks are affected. However, at the receiving end, the decryption algorithm restores the correct plaintext for blocks except the one in error. You can show this by writing out the equations for the decryption. Therefore, the error only effects the corresponding decrypted plaintext block.



**7.8 If a bit error occurs in the transmission of a ciphertext character in 8-bit CFB mode, how far does the error propagate?**

**在8位的CFB模式中，若传输中一个密文字符发生了一位错，这个错误将传播多远？**

Nine plaintext characters are affected. The plaintext character corresponding to the ciphertext character is obviously altered. In addition, the altered ciphertext character enters the shift register and is not removed until the next eight characters are processed.

**第9章 公钥密码学与RSA**

**1、Review Questions（课后思考题）：**

**9.1 What is a public key certificate?**

**公钥密码体制的主要成分是什么？**

A public-key certificate contains a public key and other information, is created by a certificate authority, and is given to the participant with the matching private key. A participant conveys its key information to another by transmitting its certificate. Other participants can verify that the certificate was created by the authority.

**9.2 What are the roles of the public and private key?**

**公钥和私钥的作用是什么？**

A user's private key is kept private and known only to the user. The user's public key is made available to others to use. The private key can be used to encrypt a signature that can be verified by anyone with the public key. Or the public key can be used to encrypt information that can only be decrypted by the possessor of the private key.

**9.3 What are three broad categories of applications of public-key cryptosystems?**

**钥密码体制的三种应用是什么？**

**Encryption/decryption:** The sender encrypts a message with the recipient's public key. **Digital signature:** The sender "signs" a message with its private key. Signing is achieved by a cryptographic algorithm applied to the message or to a small block of data that is a function of the message. **Key exchange:** Two sides cooperate to exchange a session key. Several different approaches are possible, involving the private key(s) of one or both parties.

**9.4 What requirements must a public-key cryptosystems fulfill to be a secure algorithm?**

**为得到安全算法，公钥密码体制应满足哪些要求？**

**1.** It is computationally easy for a party B to generate a pair (public key *PUb*, private key *PRb*).

**2.** It is computationally easy for a sender A, knowing the public key and the message to be encrypted, *M*, to generate the corresponding ciphertext:

*C* = E(*PUb*, *M*)

**3.** It is computationally easy for the receiver B to decrypt the resulting ciphertext using the private key to recover the original message:

*M* = D(*PRb*, C) = D(*PRb*, E(*PUb*, *M*))

**4.** It is computationally infeasible for an opponent, knowing the public key, *PUb*, to determine the private key, *PRb*.

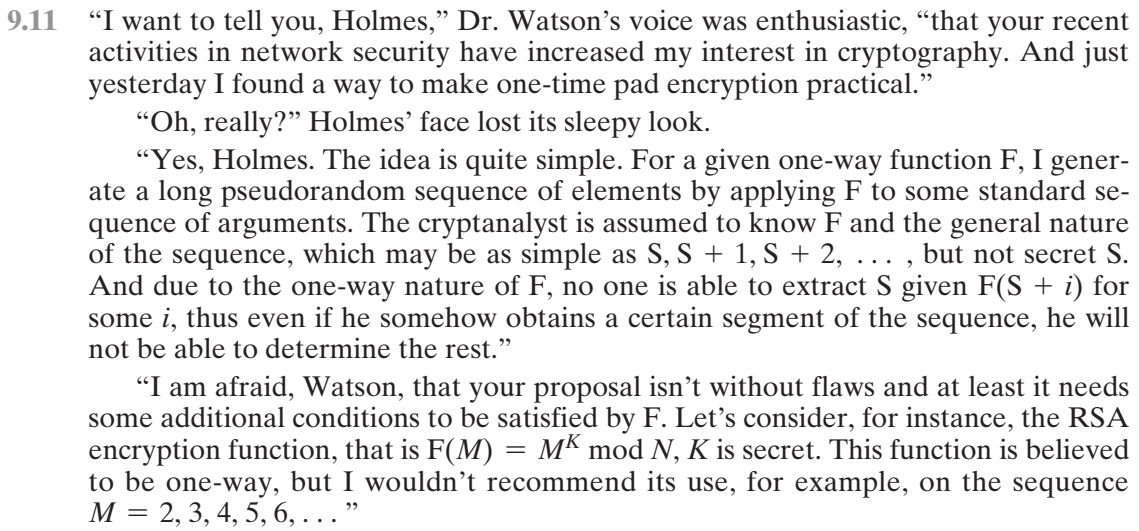
**5.** It is computationally infeasible for an opponent, knowing the public key, *PUb*, and a ciphertext, *C*, to recover the original message, *M*.

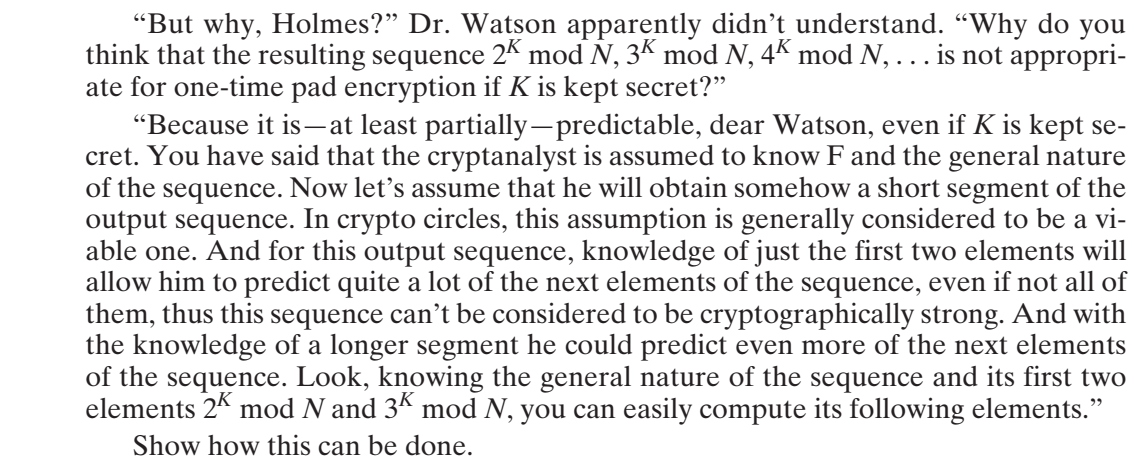
**2、什么是单向函数？什么是单向陷门函数？**

A **one-way function** is one that maps a domain into a range such that every function value has a unique inverse, with the condition that the calculation of the function is easy whereas the calculation of the inverse is infeasible:

A **trap-door one-way function** is easy to calculate in one direction and infeasible to calculate in the other direction unless certain additional information is known. With the additional information the inverse can be calculated in polynomial time.

**3、选做：课后习题（Problems）9.11,9.15,9.18**



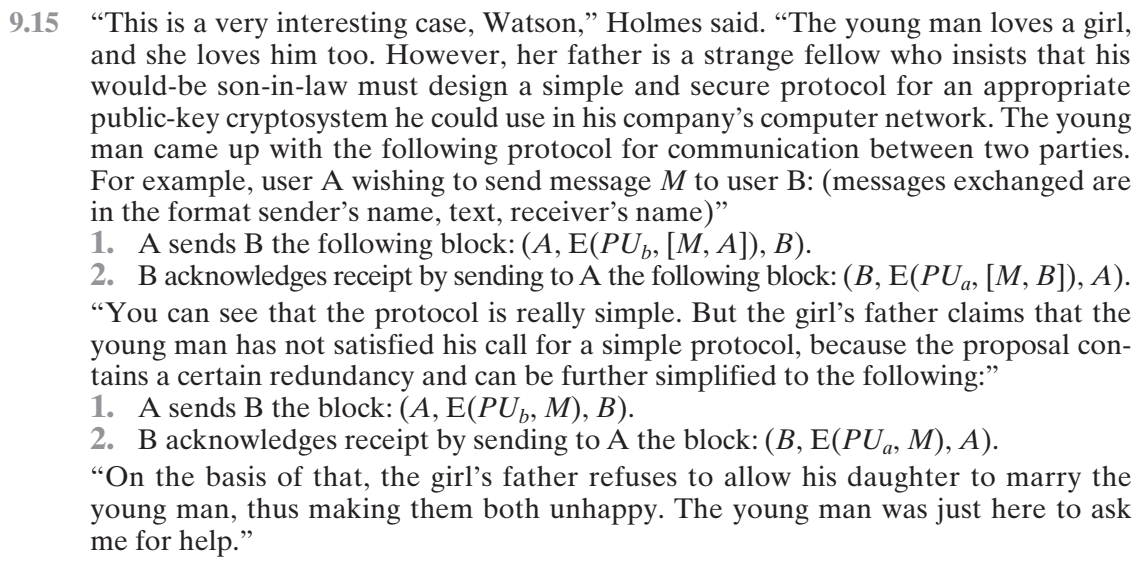


**9.11** 3rd element, because it equals to the 1st squared,

5th element, because it equals to the product of 1st and 2nd

7th element, because it equals to the cube of 1st,

etc.

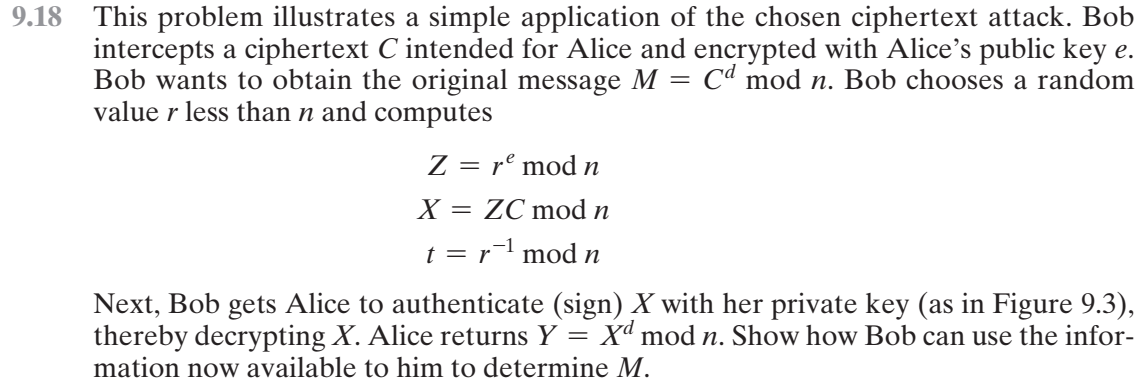


**9.15** 1) Adversary X intercepts message sent by A to B, i.e. [A, E(PUb, M), B]

2) X sends B [X, E(PUb, M), B]

3) B acknowledges receipt by sending X [B, E(PUx, M), X]

4) X decrypts E(PUx, M) using his secret decryption key, thus getting M



**9.18** Note that because *Z* = *re* mod *n*, then *r* = *Zd* mod *n*. Bob computes:

*tY* mod *n* = *r*–1*Xd* mod *n* = *r*–1*ZdCd* mod *n* = *Cd* mod *n* = *M*

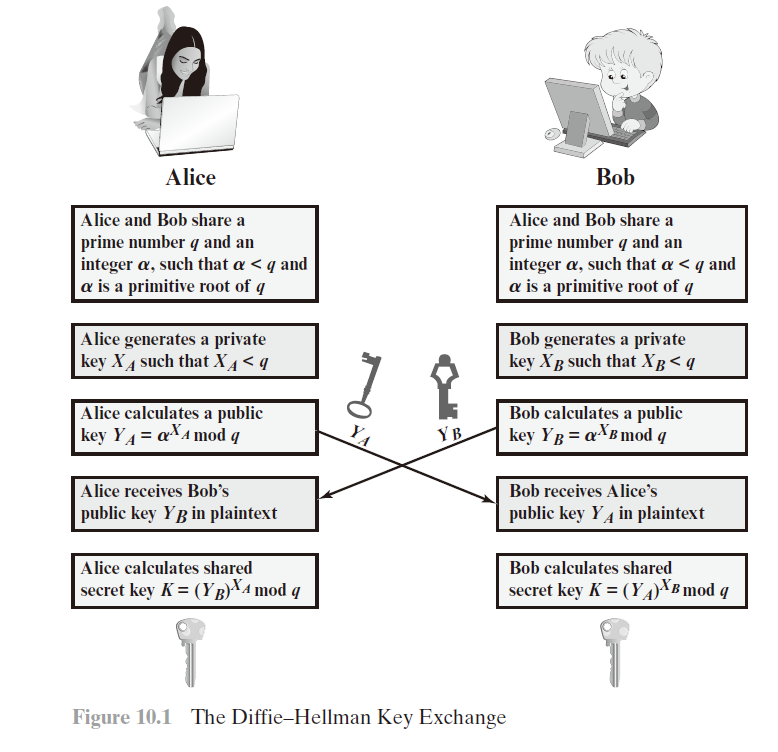
**第10章 密钥管理和其他公钥密码体制**

**1、Review Questions（课后思考题）：**

**10.1 Briefly explain Diffie–Hellman key exchange.**

**简要说明Diffie-Hellman密钥交换**

Two parties each create a public-key, private-key pair and communicate the public key to the other party. The keys are designed in such a way that both sides can calculate the same unique secret key based on each side's private key and the other side's public key.



**10.2 What is an elliptic curve?**

**什么是椭圆曲线？**

An elliptic curve is one that is described by cubic equations, similar to those used for calculating the circumference of an ellipse. In general, cubic equations for elliptic curves take the form

*y*2 + *axy* + *by*= *x*3 + *cx*2 + *dx* + *e*

where *a*, *b*, *c*, *d*, and *e* are real numbers and *x* and *y* take on values in the real numbers

**10.3 What is the zero point of an elliptic curve?**

Also called the point at infinity and designated by *O*. This value serves as the additive identity in elliptic-curve arithmetic.

**10.4 What is the sum of three points on an elliptic curve that lie on a straight line?**

**椭圆曲线上同在一条直线上的三个点的和是什么？**

If three points on an elliptic curve lie on a straight line, their sum is *O*.

**第11章 密码学Hash函数**

1、**Review Questions（课后思考题）：**

**11.1 What characteristics are needed in a secure hash function?**

安全散列函数需要具有哪些性质？

**1.** H can be applied to a block of data of any size.

**2.** H produces a fixed-length output.

**3.** H(*x*) is relatively easy to compute for any given *x*, making both hardware and software implementations practical.

**4.** For any given value *h*, it is computationally infeasible to find *x* such that H(*x*) = *h*. This is sometimes referred to in the literature as the **one-way** property.

**5.** For any given block *x*, it is computationally infeasible to find *y* ≠ *x* with H(*y*) = H(*x*).

**6.** It is computationally infeasible to find any pair (*x*, *y*) such that H(*x*) = H(*y*).

**11.2 What is the difference between weak and strong collision resistance?**

抗弱碰撞和抗强碰撞之间的区别是什么？

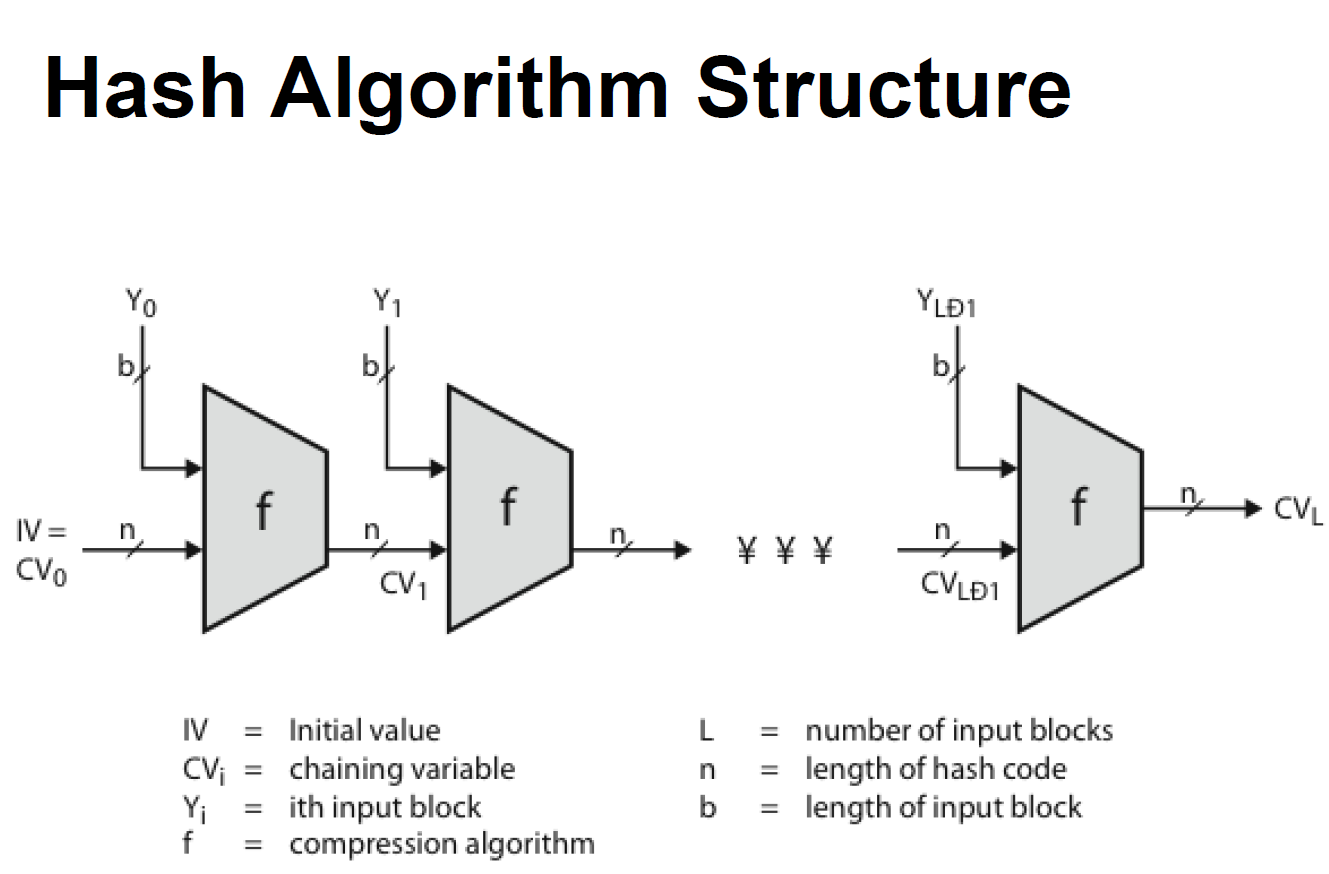
**weak collision resistance：** For any given block *x*, it is computationally infeasible to find *y* ≠ *x* with H(*y*) = H(*x*).

**strong collision resistance:** It is computationally infeasible to find any pair (*x*, *y*) such that H(*x*) = H(*y*).

**11.3 What is the role of a compression function in a hash function?**

散列函数中的压缩函数的作用是什么？

A typical hash function uses a compression function as a basic building block, and involves repeated application of the compression function.



**第12章 消息认证码**

1. **Review Questions（课后思考题）：**

12.1 What types of attacks are addressed by message authentication?

消息认证是为了对付哪些类型的攻击？  
**Masquerade:** Insertion of messages into the network from a fraudulent source. This includes the creation of messages by an opponent that are purported to come from an authorized entity. Also included are fraudulent acknowledgments of message receipt or nonreceipt by someone other than the message recipient.

**Content modification:** Changes to the contents of a message, including insertion, deletion, transposition, and modification.

**Sequence modification:** Any modification to a sequence of messages between parties, including insertion, deletion, and reordering.

**Timing modification:** Delay or replay of messages. In a connection-oriented application, an entire session or sequence of messages could be a replay of some previous valid session, or individual messages in the sequence could be delayed or replayed. In a connectionless application, an individual message (e.g., datagram) could be delayed or replayed.

12.2 What two levels of functionality comprise a message authentication or digital signature mechanism?

消息认证或数字签名方法有哪两层功能？

At the lower level, there must be some sort of function that produces an authenticator: a value to be used to authenticate a message. This lower-level function is then used as primitive in a higher-level authentication protocol that enables a receiver to verify the authenticity of a message.

12.3 What are some approaches to producing message authentication?

产生消息认证有哪些方法？

Message encryption, message authentication code, hash function.

12.4 When a combination of symmetric encryption and an error control code is used for message authentication, in what order must the two functions be performed?

对称加密和错误控制码一起用于消息认证时，这两个函数必须以何种顺序执行？

Error control code, then encryption.

12.5 What is a message authentication code?

什么是消息认证码？

An authenticator that is a cryptographic function of both the data to be authenticated and a secret key.

12.6 What is the difference between a message authentication code and a one-way hash function?

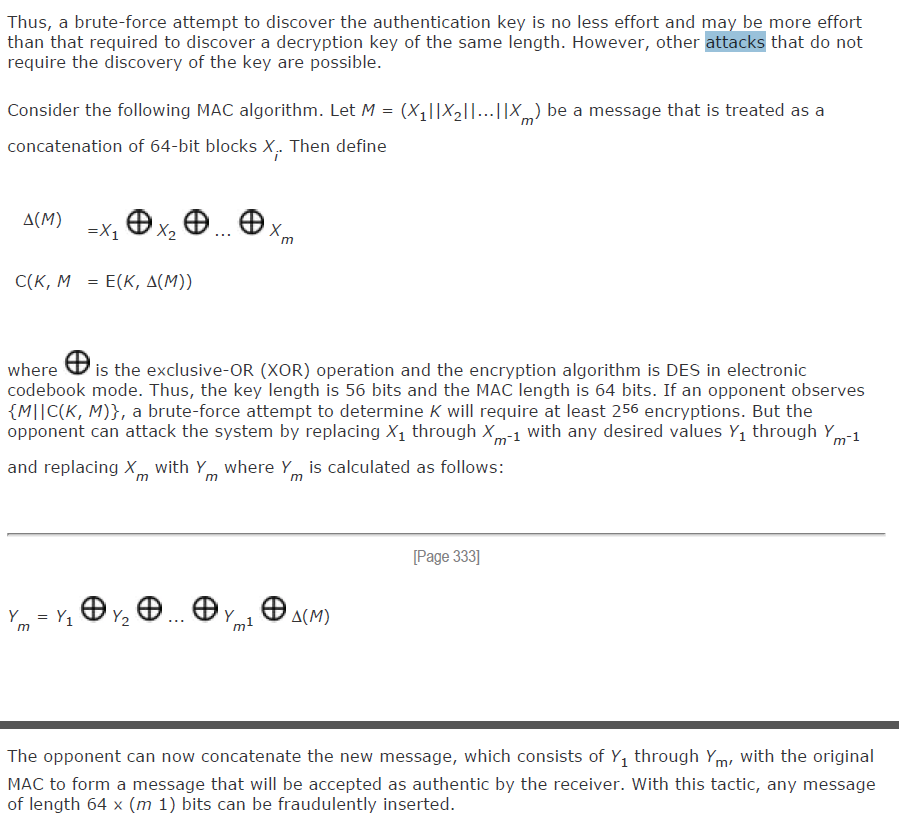
消息认证码和单向Hash函数之间的区别是什么？

A hash function, by itself, does not provide message authentication. A secret key must be used in some fashion with the hash function to produce authentication. A MAC, by definition, uses a secret key to calculated a code used for authentication.

12.8 Is it necessary to recover the secret key in order to attack a MAC algorithm?

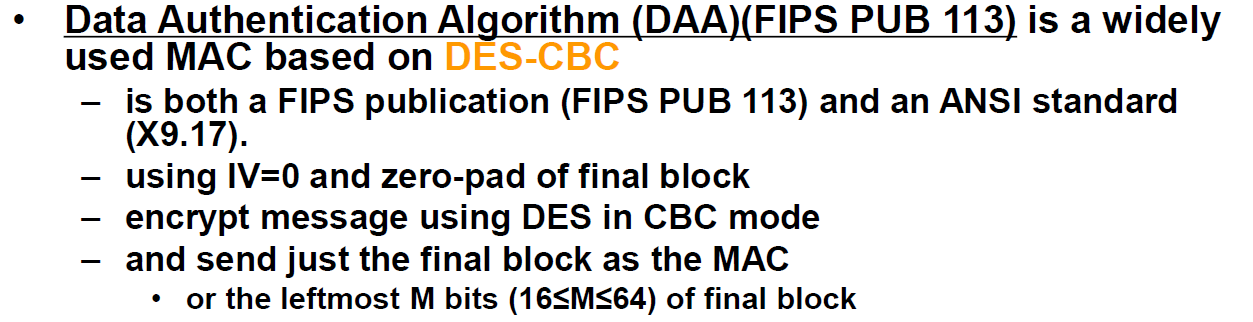
为了攻击MAC算法，必须要恢复密钥吗？

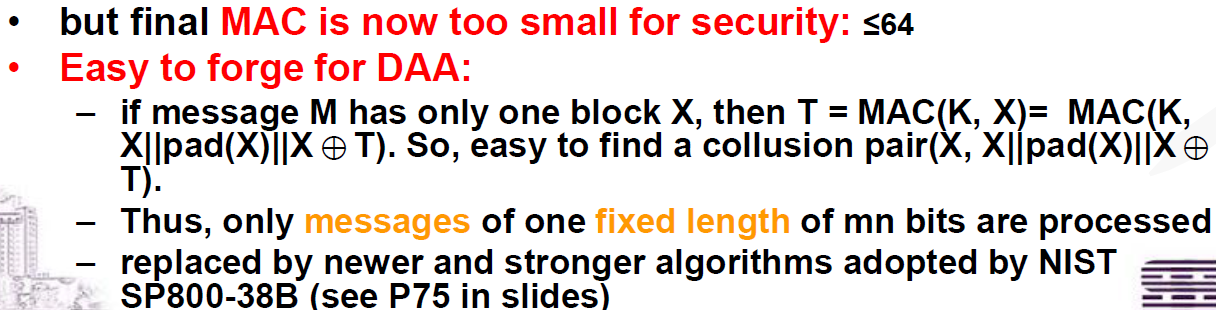
No. Section 11.3 outlines such attacks.

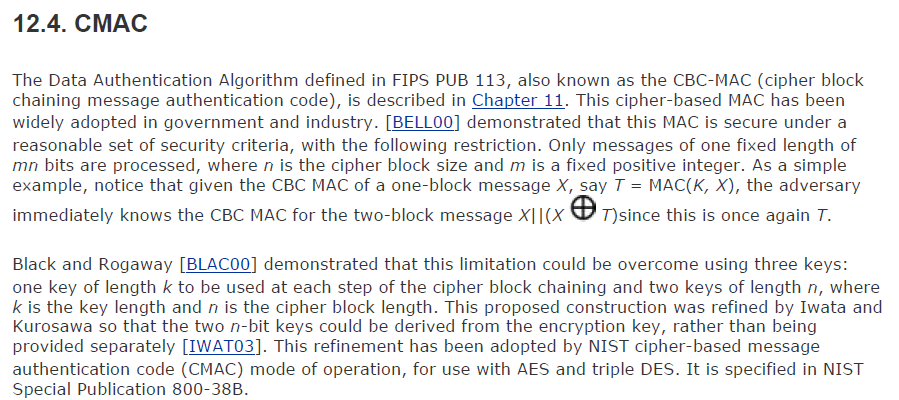


说的是，有些情况下，可以不需要知道密钥，利用MAC算法的性质构造出MAC值一样的数据。

2.利用CBC-MAC（即FIPS PUB 313定义的数据认证算法）来生成消息认证码，安全吗？为什么？



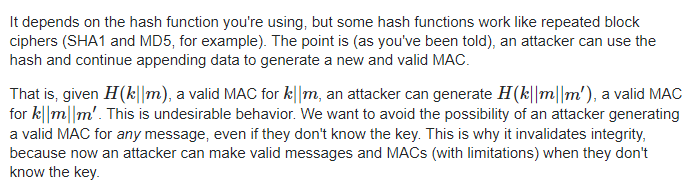




3. （选做）利用Hash(Message||Pad(M)||key）生成消息认证码，安全吗？为什么？

跟pad方式有关。 比如如果pad是填充0。那X,和X000的hash的mac就相同了，不安全。

4. （选做）利用Hash(key||Message）生成消息认证码，安全吗？为什么？



**第13章 数字签名**

1、**Review Questions（课后思考题）：**

**13.1 List two disputes that can arise in the context of message authentication.**

列出消息认证中出现的两种争议。

Suppose that John sends an authenticated message to Mary. The following disputes that could arise: **1.** Mary may forge a different message and claim that it came from John. Mary would simply have to create a message and append an authentication code using the key that John and Mary share. **2.** John can deny sending the message. Because it is possible for Mary to forge a message, there is no way to prove that John did in fact send the message.

**13.2 What are the properties a digital signature should have?**

数字签名应该具有哪些性质？

**1.** It must be able to verify the author and the date and time of the signature. **2.** It must be able to authenticate the contents at the time of the signature. **3.** The signature must be verifiable by third parties, to resolve disputes.

**13.5 In what order should the signature function and the confidentiality function be applied to a message, and why?**

签名函数和保密函数应以何种顺序作用于消息，为什么？

It is important to perform the signature function first and then an outer confidentiality function. In case of dispute, some third party must view the message and its signature. If the signature is calculated on an encrypted message, then the third party also needs access to the decryption key to read the original message. However, if the signature is the inner operation, then the recipient can store the plaintext message and its signature for later use in dispute resolution.

**13.6 What are some threats associated with a direct digital signature scheme?**

直接数字签名方法中会遇到哪些威胁？

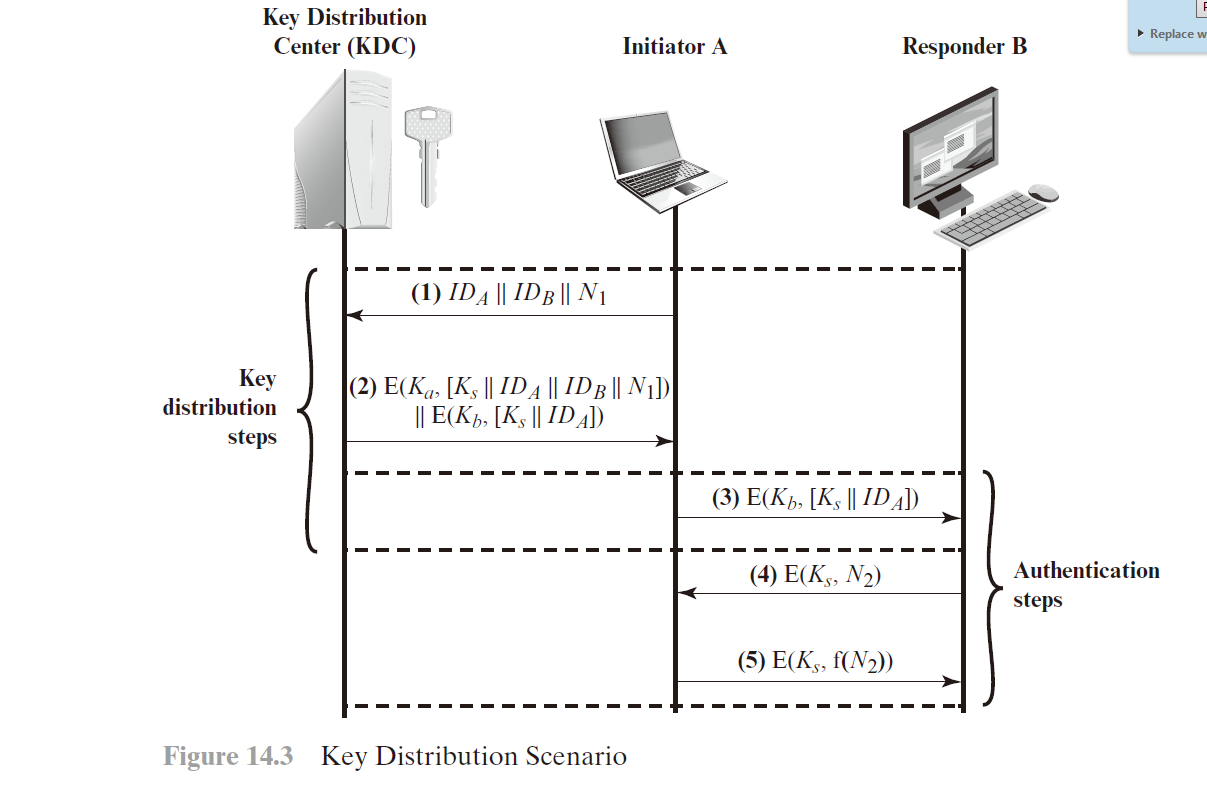
**1.** The validity of the scheme depends on the security of the sender's private key. If a sender later wishes to deny sending a particular message, the sender can claim that the private key was lost or stolen and that someone else forged his or her signature. **2.** Another threat is that some private key might actually be stolen from X at time T. The opponent can then send a message signed with X's signature and stamped with a time before or equal to T

**第14章 密钥管理和分发**

**1、Review Questions（课后思考题）：**

**14.1 Explain why man-in-the-middle attacks are ineffective on the secret key distribution protocol discussed in Figure 14.3.**

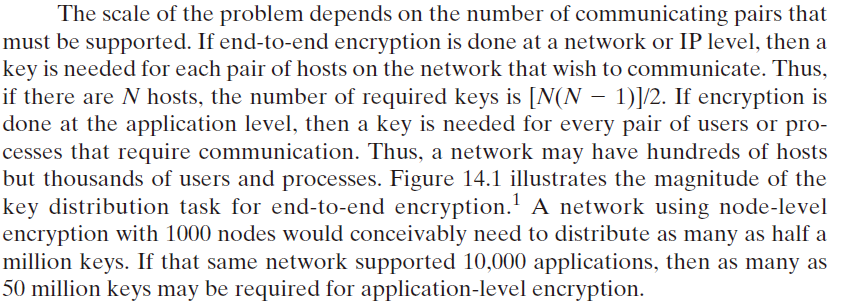
**对于教材的图14.3中的秘密密钥分发协议进行中间人攻击是无效的，请解释其原因。**



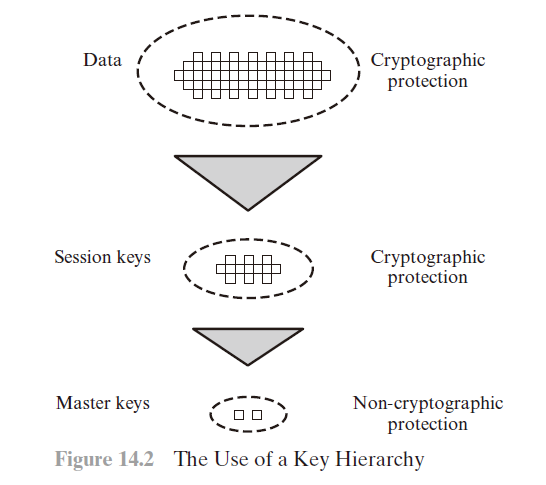
**中间人没有Ka,也就没法解密出服务器发给他的****，他不能取出****发给B，B也就不相信他。**

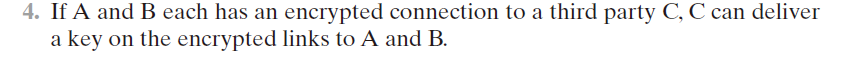
**14.2 What is the major issue in end to end key distribution? How does the key hierarchy concept address that issue?**

**列出在两个通信方之间分发密钥的方法。**



**主要的问题是，key的数量非常庞大。对于N个节点的网络，需要N\*（N-1）/2对key。这些key如何分发和管理成为了问题。**





**用key hierarchy配合 option 4的方法， 所有节点和一个中间节点c共享密钥，然后任意两个节点间a b的密钥，通过c来协商和分发。 这样，不通信的节点间就不需要密钥，需要通信时候可以通过c按需协商和分发密钥。**

**14.3 What is a nonce?**

**什么是临时交互号？**

A nonce is a value that is used only once, such as a timestamp, a counter, or a random number; the minimum requirement is that it differs with each transaction.

**14.6 List four general categories of schemes for the distribution of public keys.**

**列出四种公钥分发模式的类型。**

Public announcement. Publicly available directory. Public-key authority. Public-key certificates

**14.8 What is a public-key certificate?**

**什么是公钥证书？**

A public-key certificate contains a public key and other information, is created by a certificate authority, and is given to the participant with the matching private key. A participant conveys its key information to another by transmitting its certificate. Other participants can verify that the certificate was created by the authority.

**14.9 What are the requirements for the use of a public-key certificate scheme?**

**1.** Any participant can read a certificate to determine the name and public key of the certificate's owner. **2.** Any participant can verify that the certificate originated from the certificate authority and is not counterfeit. **3.** Only the certificate authority can create and update certificates. **4.** Any participant can verify the currency of the certificate.

**14.10 What is the purpose of the X.509 standard?**

**X.509标准的用途是什么？**

X.509 defines a framework for the provision of authentication services by the X.500 directory to its users. The directory may serve as a repository of public-key certificates. Each certificate contains the public key of a user and is signed with the private key of a trusted certification authority. In addition, X.509 defines alternative authentication protocols based on the use of public-key certificates.

**14.11 What is a chain of certificates?**

**什么是证书链？**

A chain of certificates consists of a sequence of certificates created by different certification authorities (CAs) in which each successive certificate is a certificate by one CA that certifies the public key of the next CA in the chain.

**14.12 How is an X.509 certificate revoked?**

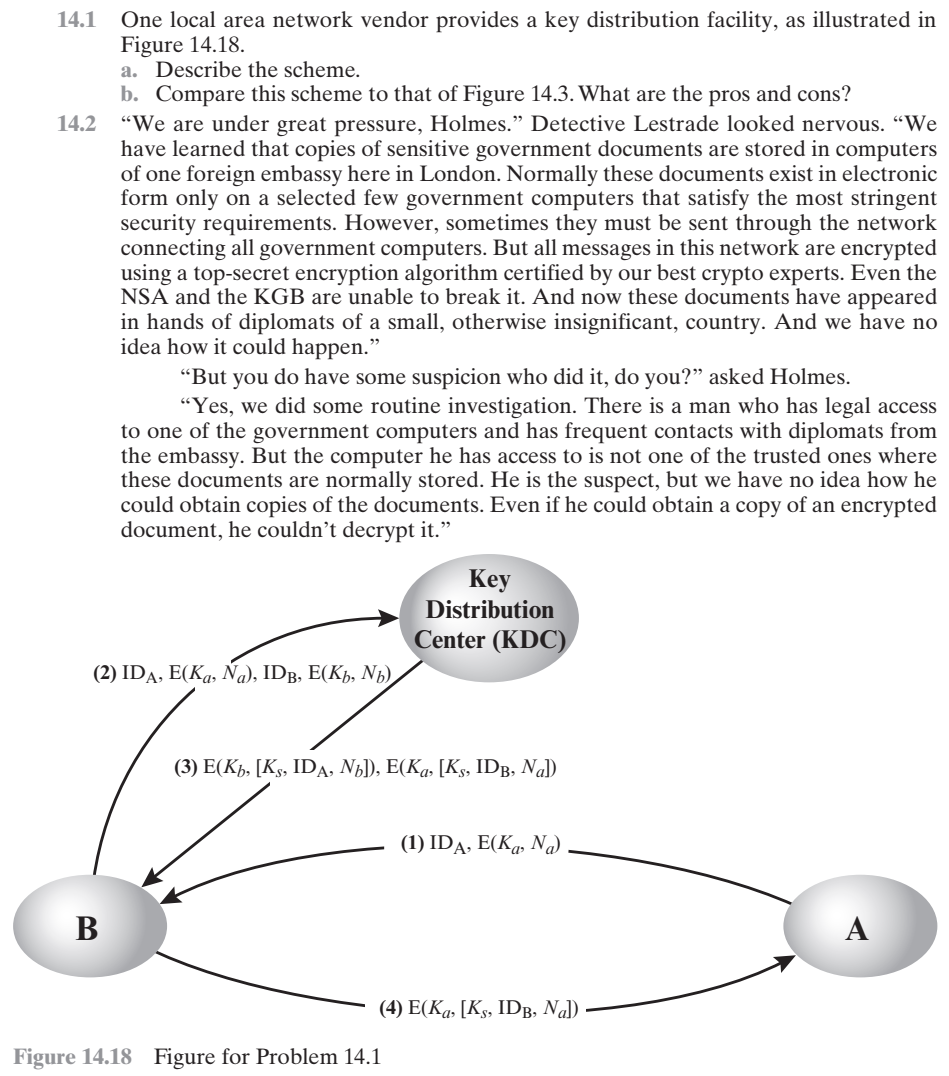
**X.509证书如何撤销？**

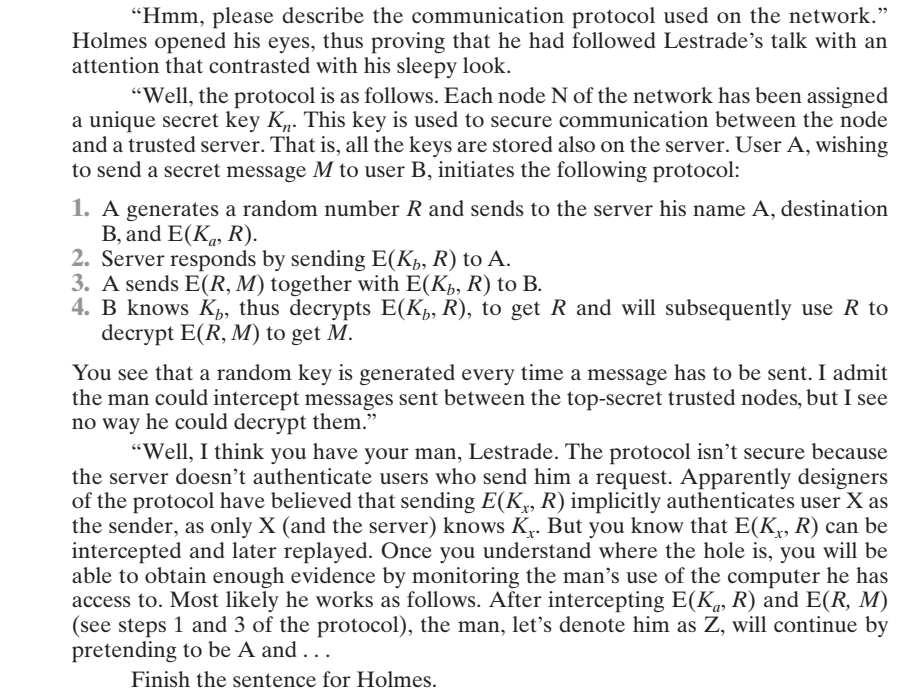
The owner of a public-key can issue a certificate revocation list that revokes one or more certificates.

**2、会话密钥和主密钥有什么不同？**

A **session key** is a temporary encryption key used between two principals. A **master key** is a long-lasting key that is used between a key distribution center and a principal for the purpose of encoding the transmission of session keys. Typically, the master keys are distributed by noncryptographic means.

**3、（选做）课后习题(Problem)：**





**14.1**

**7.3** **a.** A sends a connection request to B, with an event marker or nonce (Na) encrypted with the key that A shares with the KDC. If B is prepared to accept the connection, it sends a request to the KDC for a session key, including A's encrypted nonce plus a nonce generated by B (Nb) and encrypted with the key that B shares with the KDC. The KDC returns two encrypted blocks to B. One block is intended for B and includes the session key, A's identifier, and B's nonce. A similar block is prepared for A and passed from the KDC to B and then to A. A and B have now securely obtained the session key and, because of the nonces, are assured that the other is authentic.

**b.** The proposed scheme appears to provide the same degree of security as that of Figure 7.9. One advantage of the proposed scheme is that the, in the event that B rejects a connection, the overhead of an interaction with the KDC is avoided.

**14.2**

**7.4 i)** Sending to the server the source name A, the destination name Z (his own), and E(*Ka*, *R*), as if A wanted to send him the same message encrypted under the same key R as A did it with B

**ii)** The server will respond by sending E(*Kz*, *R*) to A and Z will intercept that

**iii)** because Z knows his key *Kz*, he can decrypt E(*Kz*, *R*), thus getting his hands on R that can be used to decrypt E(*R*, *M*) and obtain *M*.

**第15章 用户认证**

1. **Review Questions（课后思考题）：**

15.2 List three general approaches to dealing with replay attacks.

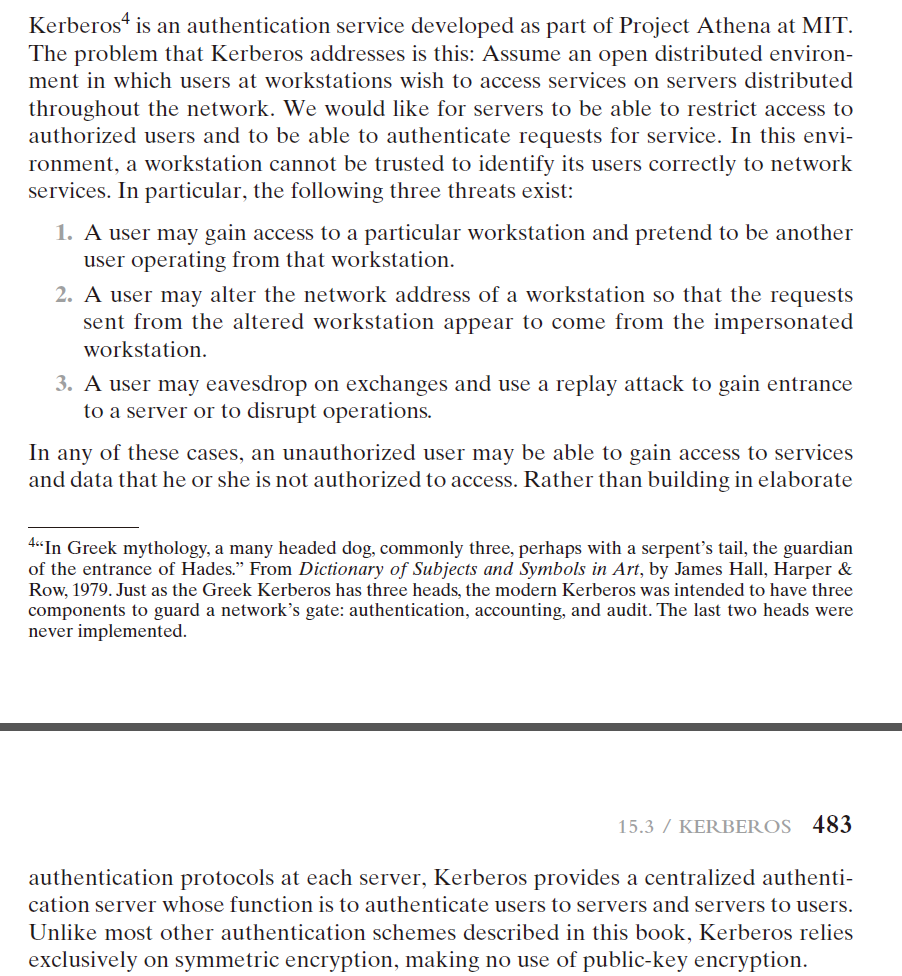
列出三个常用的防止重放攻击的方法

**13.8** **1.** Attach a sequence number to each message used in an authentication exchange. A new message is accepted only if its sequence number is in the proper order. **2.** Party A accepts a message as fresh only if the message contains a timestamp that, in A's judgment, is close enough to A's knowledge of current time. This approach requires that clocks among the various participants be synchronized. **3.** Party A, expecting a fresh message from B, first sends B a nonce (challenge) and requires that the subsequent message (response) received from B contain the correct nonce value.

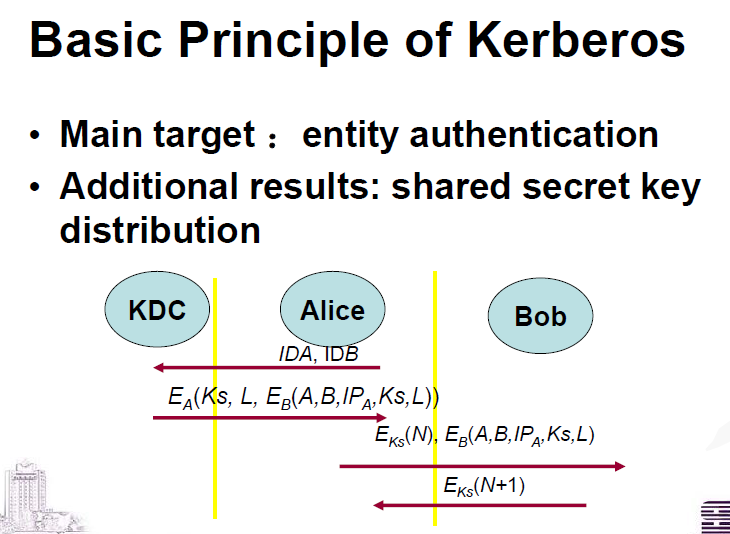
15.4 What problem was Kerberos designed to address?

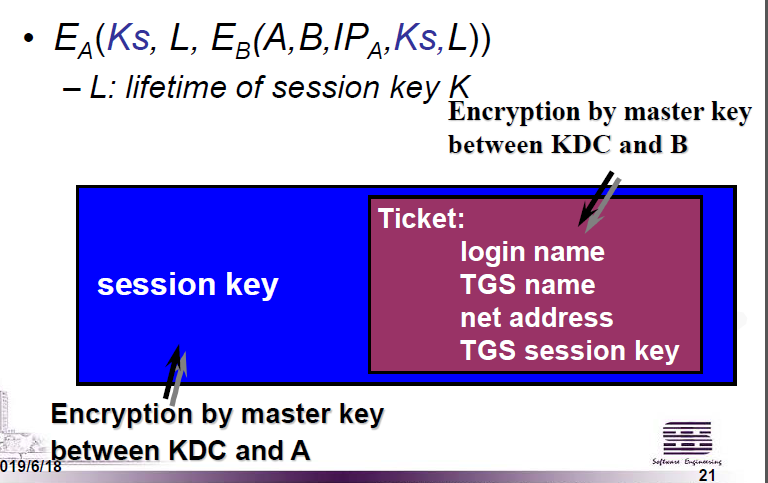
Kerberos主要处理什么问题？

The problem that Kerberos addresses is this: Assume an open distributed environment in which users at workstations wish to access services on servers distributed throughout the network. We would like for servers to be able to restrict access to authorized users and to be able to authenticate requests for service. In this environment, a workstation cannot be trusted to identify its users correctly to network services.



1. **Problems（课后习题）：**





15.10 In Kerberos, when Bob receives a Ticket from Alice, how does he know it is not genuine?

如果Alice是假的，他就不能用KA解密出KDC发来的，也就不能有用Bob的KB加密的消息。 对于一个假的消息，Bob解密后发现消息格式里的A B L这些信息不合法，就知道Alice是假的了。

15.11 In Kerberos, how does Bob know that the received token is not corresponding to Alice’s?

在Kerberos，当Bob收到一个来自于Alice的票据，如何得知其是否真实？如何得知其确实来自于Alice？

是否真实见上一题。

如果消息来源不是Alice，那 里面的A就一定不是alice，而且这个消息没法伪造，因为只有KDC才知道KB.

15.12 In Kerberos, how does Alice know that a reply to an earlier message is from Bob?

在Kerberos，当Alice收到一个回复，她如何得知该消息来自于Bob（且是Bob最新的回复）



Alice发给Bob的信息是用KS加密的，这个KS是只有Alice和Bob和KDC 才知道的。 只有BOB才能成功解密，并且发出挑战的应答。

Alice知道N 和L，L里有时间戳信息。 Alice收到N+1后，减1也就得到了N, 然后查找N对应的L，即可知道时间戳信息。

也许这题是想问如果有多个Bob: bob1 bob2，怎么知道来自哪个bob?

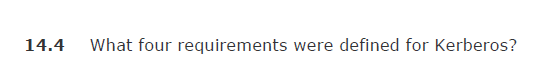
如果bob1 bob2的ip不同，那就好办。

否则就尝试用bob1的ks 和bob2的ks解密，看哪个解密出来的跟某一个N匹配就可以了。

1. 列出身份识别所基于的4个常用工具。

貌似有问题，书上找不到、课件上找不到直接对应的话。

可能问的是这个？



但是这个也没讲。

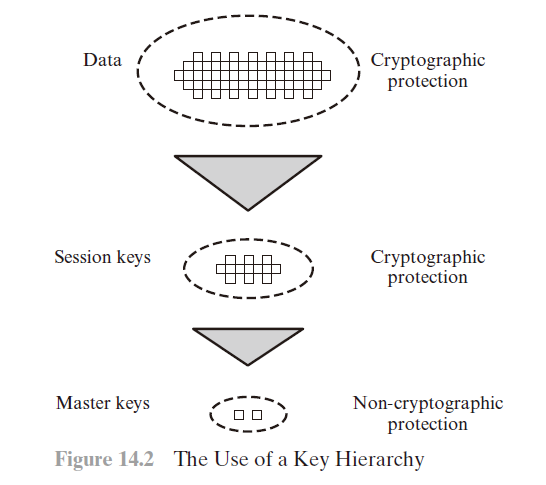
4、（选做）在服务端存储带盐的口令的HASH值，比直接存储口令的HASH值，要更加安全。为什么？

1.不带salt，相同的密码的hash值是相同的，会泄露一部分信息。

2.不带salt，容易被事先build好的彩虹表攻击。

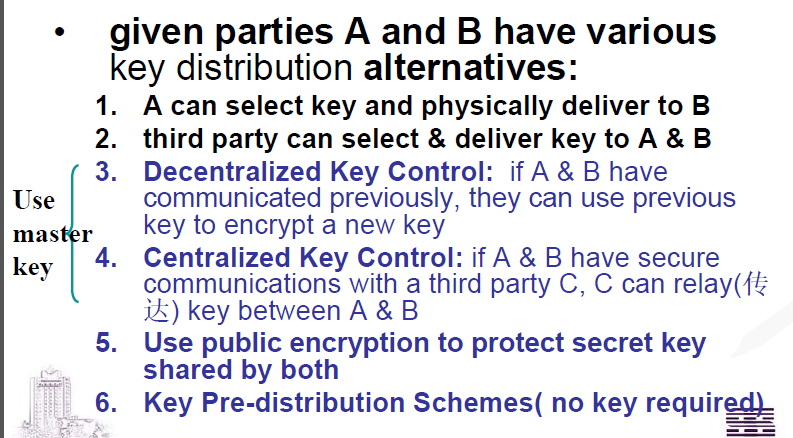
**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*补充：选做\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**1. 层次式密钥中涉及到哪两类密钥？他们的区别是什么？**



A **session key** is a temporary encryption key used between two principals. A **master key** is a long-lasting key that is used between a key distribution center and a principal for the purpose of encoding the transmission of session keys. Typically, the master keys are distributed by noncryptographic means.

**2. 请列出在通信双方之间进行密钥分配的六种方法。哪些方法用到了加密算法？哪些方法需要知道某些信息？**



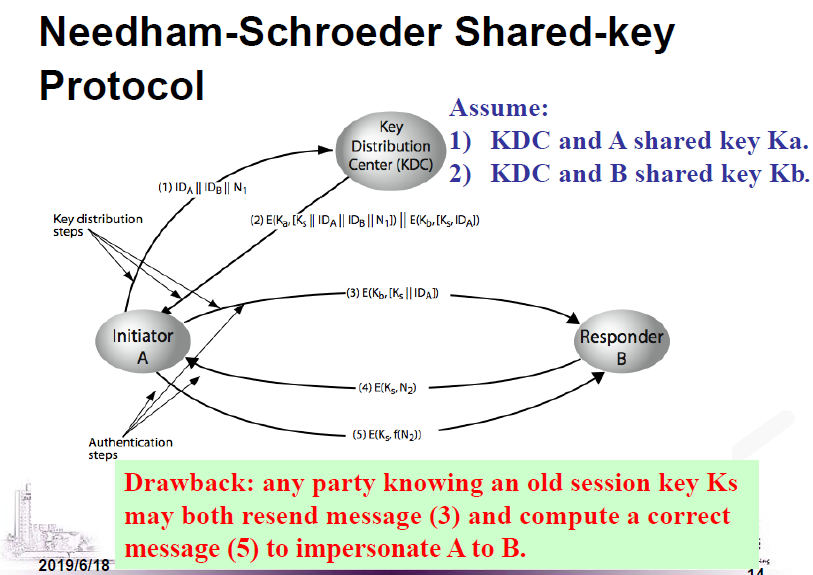
**哪些方法用到了加密算法？ 是加密算法还是密码学方法？**

**2.是什么意思？ 3.算不算用了加密算法？ 6.DH算不算加密算法？**

**哪些方法需要知道某些信息？ 怎么答？**

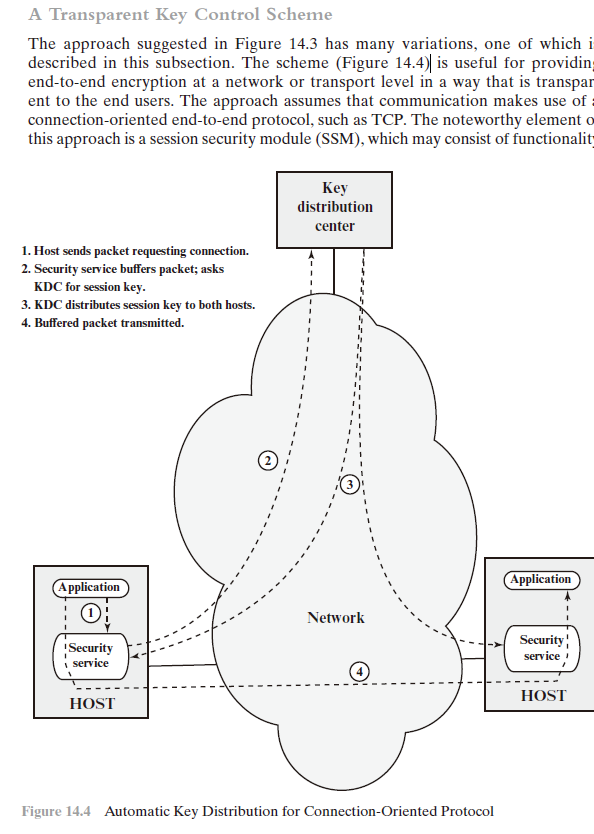
**3.需要提前知道一个key。 4.需要提前知道一个third party 和key?? 5.需要知道对方公钥，这个可以按需传输，但是算不算知道？考虑中间人攻击，需要知道证书，这个算不算？ 6.也可能有中间人攻击。需不需要知道证书？**

**3. 为什么基本的Needham-Schroeder密钥分配协议是不安全的？**



如果A截获了3，并知道了Ks（尽管可能很旧），就可以去骗B。

**4. 面向连接的自动密钥分配是采用链路加密？还是端对端加密？**

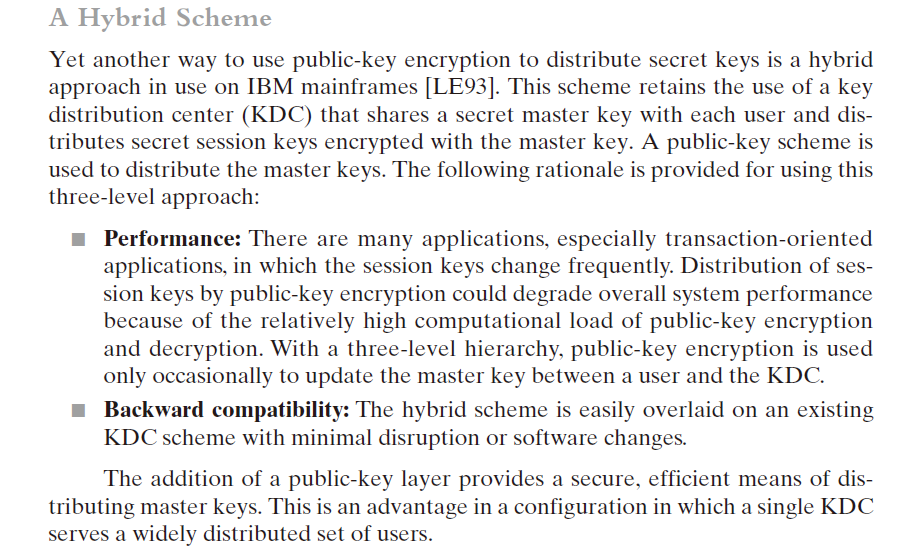


是端对端加密。

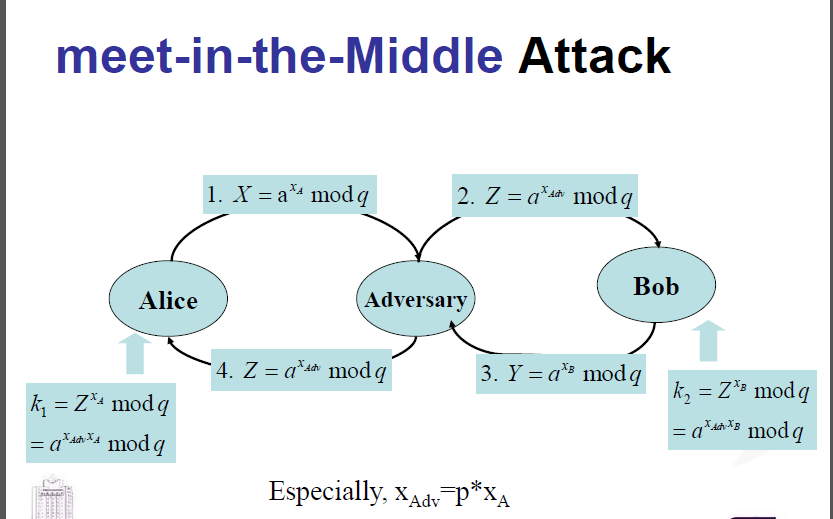
**5. 利用公钥进行密钥分配的方案中，简单密钥分配为什么不安全？**

不能知道公钥是中间人的公钥，还是对方的公钥。 需要证书才能安全。

**6. 什么是混合的密钥分配？**



**7. Diffie-Hellman密钥交换方案中的中间人攻击是指？**



中间人用自己的私钥分别和A B 进行DH密钥交换。