本文讲解对称加密、非对称加密、消息摘要、MAC、数字签名、公钥证书的用途、不足和解决的问题。

0.概述

当发送方 A 向接收方 B 发送数据时，需要考虑的问题有：

1.数据的安全性，即数据加密。

2.数据的完整性，即数据不被篡改。

3.数据的真实性，即数据确实来自于发送方，传输过程中没有被替换。

4.数据的不可否认性，即验证发送方确实发送了数据。

本文的整体结构见下图。

基本概念：

密码：按特定法则编成，用以对通信双方的信息进行明密变换的符号。

密钥：在现代密码学中，秘钥指的是一组特定的秘密数据，在加密时，它控制密码算法按照指定的方式将明文变换为相应的密文，并将一组信源标识信息变换不可伪造的签名；在解密时，它控制密码算法按照指定的方式将密文变换为相应的明文，并将签名信息变换成不可否认的信源证据。

1.数据传输的安全

保证数据传输安全的方法就是对数据进行加密了，常用的加密算法有对称加密和非对称加密。

1.1 对称加密

又称共享加密，加解密使用相同的密钥。

常见算法： DES 3DES AES RC5 RC6

例：

1).为了安全，A 将数据加密发送给 B。

2).密文即使在传送过程中被截获，因为不知道密钥也无法解密。

3).B 接收到密文之后，需要使用加密相同的密钥来解密。

4).需要 A 将密钥传给 B，但保证密钥传输过程中的安全又成了问题。

优点： 计算速度快，算法简单、密钥较短。

缺点： 规模复杂，需要保存的密钥数量多。为了传送数据的安全，将数据加密后进行传输，但是对称加密需要发送方将密钥安全地传给接收方以便接收方解密，因此密钥如何安全传送又成了一个问题。

问题： 如何保证密钥的安全性？

1.2 非对称加密

也称公钥加密，这套密钥算法包含配套的密钥对，分为加密密钥和解密密钥。加密密钥时公开的，又称为公钥；解密密钥时私有的，又称为私钥。数据发送者使用公钥加密数据，数据接收者使用私钥进行数据解密。

常见算法： RSA ECC DSA Diffe-Hellmen

例：

1).B 生成密钥对，将公钥传给 A，私钥自己保留。公钥即使被其他人获得也没有关系。

2).A 用 B 传过来的密钥将要发送的明文数据加密，然后将密文发送给 B。其他人即使获得密文也无法解密，因为没有配对的用来解密的私钥。

3).B 接收到 A 传送过来的密文，用自己保留的私钥对密文解密，得到明文。

优点： 解决了密钥的安全性问题。密钥数量少。可以用于数字签名。

缺点： 一是计算速度慢；二是无法保证公钥的合法性，因为接收到的公钥不能保证是 B 发送的，如，攻击者截获 B 的消息，将公钥替换。

这里先留下一个问题，后面叙述解决办法：如何保证公钥是合法的？

2.保证数据完整性

消息摘要

消息摘要函数是一种用于判断数据完整性的算法，也称为散列函数或哈希函数，函数的返回值就散列值，散列值又称为消息摘要或者指纹。

这种算法是不可逆的，即无法通过消息摘要反向推导出消息，因此又称为单向散列函数。

常见算法： MD5 SHA MD2 MD4 SHA-1

例：

当我们使用某一软件时，下载完成后需要确认是否是官方提供的完整版，是否被人篡改过。通常软件提供方会提供软件的散列值，用户下载软件之后，在本地使用相同的散列算法计算散列值，并与官方提供的散列值向对比。如果相同，说明软件完整，未被修改过。

优点：可以保证数据的完整性。

缺点：无法保证数据的真实性，即不能确定数据和散列值是来自发送方的，因为攻击者完全可以将数据和散列值一起替换。

问题： 如何验证发送的数据确实来自于发送方？

3.保证数据的真实性

要保证数据来自发送方，即确认消息来自正确的发送者，称为消息认证。认证函数有三类：消息加密函数、消息认证码、散列函数。

3.1 消息认证码（MAC = C(K , M)有 3 种基本模式）

消息认证码（Message Authentication Code，简称 MAC）是一种可以确认消息完整性并进行认证的技术。消息认证码可以简单理解为一种与密钥相关的单向散列函数。核心：使用密钥来认证，第三方没有密钥就无法冒充发送方。

例：

1).A 把消息发送给 B 前，先把共享密钥发送给 B。

2).A 把要发送的消息使用共享密钥计算出 MAC 值，然后将消息和 MAC 发送给 B。

3).B 接收到消息和 MAC 值后，使用共享密钥计算出 MAC 值，与接收到的 MAC 值对比。

4).如果 MAC 值相同，说明接收到的消息是完整的，而且是 A 发送的。

这里还是存在对称加密的密钥配送问题，可以使用公钥加密方式解决。

优点： 可以保证数据的完整性和真实性。

缺点： 接收方虽然可以确定消息的完整性和真实性，解决篡改和伪造消息的问题，但不能防止 A 否认发送过消息。

例：

假如 A 给 B 发送了消息，B 接收到之后，A 否认自己发送过消息给 B，并抵赖说，“因为我和 B 都能计算处正确的 MAC 值，所以可能是 B 的密钥被攻击者盗取了，攻击者给 B 发的消息。”

问题： 如何让发送方无法否认发送过数据？

3.2 数字签名

数字签名（Digital Signature）可以解决发送方否认发送过消息的问题。

数字签名的重点在于发送方和接收方使用不同的密钥来进行验证，并且保证发送方密钥的唯一性，将公钥算法反过来使用可以达到此目的：A 发送消息前，使用私钥对消息进行签名，B 接收到消息后，使用配对的公钥对签名进行验证；如果验证通过，说明消息就是 A 发送的，因为只有 A 采用配对的私钥；第三方机构也是依据此来进行裁决，保证公正性。

例：

1).A 把消息用哈希函数处理生成消息摘要，并报摘要用私钥进行加密生成签名，把签名和消息一起发送给 B。

2).数据经过网络传送给 B，当然，为了安全，可以用上述的加密方法对数据进行加密。

3). B 接收到数据后，提取出消息和签名进行验签。采用相同的哈希函数生成消息摘要，将其与接收的签名用配对的公钥解密的结果对比，如果相同，说明签名验证成功。消息是 A 发送的，如果验证失败，说明消息不是 A 发送的。

问题： 依然是，如何确保公钥的合法性？

4.公钥证书

我们看到，上面的公钥加密，数字签名的问题都在于如何保证公钥的合法性。

解决办法是将公钥交给一个第三方权威机构——认证机构（Certification Authority）CA 来管理。接收方将自己的公钥注册到 CA，由 CA 提供数字签名生成公钥证书（Public-Key Certificate）PKC，简称证书。证书中有 CA 的签名，接收方可以通过验签来验证公钥的合法性。

例：

1).接收方 B 生成密钥对，私钥自己保存，将公钥注册到 CA。

2).CA 通过一系列严格的检查确认公钥是 B 本人的。

3).CA 生成自己的密钥对，并用私钥对 B 的公钥进行数字签名，生成数字证书。证书中包含 B 的公钥和 CA 的签名。这里进行签名并不是要保证 B 的公钥的安全性，而是要确定公钥确实属于 B。

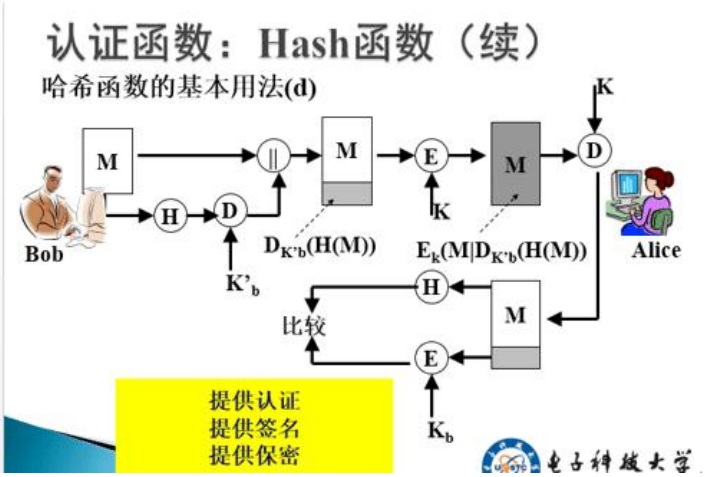
4).发送方 A 从 CA 获取 B 的证书。

5).A 使用 CA 的公钥对从 CA 获取的 B 的证书进行验签，如果成功就可以确保证书中的公钥确实来自 B。

6).A 使用证书中 B 的公钥对消息进行加密，然后发送给 B。

7).B 接收到密文后，用自己的配对的私钥进行解密，获得消息明文。

例题：



假设明文用 M 表示，H( )为 hash 函数。Ekx（ ）表示为用户 x 的私钥签名函数，表示密钥为 K 的对称加密函数。Alice 为发送方，Bob 为接收方。试结合对称密码体制和公钥密码体制的优缺点，运用对称密码体制，公钥密码体制和 hash 算法，设计一个涵盖保密、认证、数字签名和数字信封的通信模型。

（1）发方 A 将原文信息进行哈希运算，得一哈希值即数字摘要 MD；

（2）发方 A 用自己的私钥 PVA，采用非对称 RSA 算法，对数字摘要 MD 进行加密，即得数字签名 DS；

（3）发方 A 用对称算法 DES 的对称密钥 SK 对原文信息、数字签名 SD 及发方 A 证书的公钥 PBA 采用对称算法加密，得加密信息 E；

（4）发方用收方 B 的公钥 PBB，采用 RSA 算法对对称密钥 SK 加密，形成数字信封 DE，就好像将对称密钥 SK 装到了一个用收方公钥加密的信封里；

（5）发方 A 将加密信息 E 和数字信封 DE 一起发送给收方 B；

（6）收方 B 接受到数字信封 DE 后，首先用自己的私钥 PVB 解密数字信封，取出对称密钥 SK；

（7）收方 B 用对称密钥 SK 通过 DES 算法解密加密信息 E，还原出原文信息、数字签名 SD 及发方 A 证书的公钥 PBA；

（8）收方 B 验证数字签名，先用发方 A 的公钥解密数字签名 DS 得数字摘要 MD；

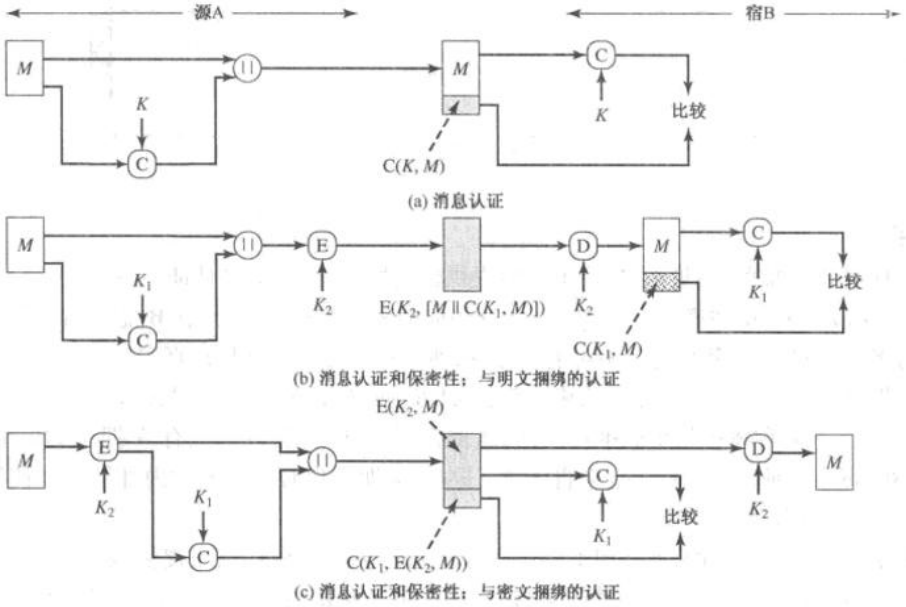
（9）收方 B 同时将原文信息用同样的哈希运算，求得一个新的数字摘要 MD’；

（10）将两个数字摘要 MD 和 MD’进行比较，验证原文是否被修改。如果二者相等，说明数据没有被篡改，是保密传输的，签名是真实的；否则拒绝该签名。

答：

发送方 A：使用 hash 处理原文 M 得到数字摘要——>使用 A 的私钥将数字摘要加密（RSA）得到数字签名——>使用对称密钥将原文 M、A 的公钥、数字签名加密（DES）得到加密信息 E——>使用 B 的公钥将对称密钥加密（RSA）得到数字信封——>将加密信息 E 和数字信封统一发送给 B

消息认证码(MAC)　MAC = C(K , M　)M 是输入消息；C 是 MAC 函数；K 是共享密钥;



用户数据的安全性和隐私保护是制约云计算发展和应用的主要障碍之一，试结合本课程分析信息安全技术说明如何解决云计算环境中的数据安全和隐私保护问题？

答：（1）加强云终端的安全控制，加强移动网络传输与接入的安全性。（2）加强对云服务业务的访问控制，针对业务系统制定一套安全统一的策略管理模式，以免业务流程被非法控制。（3）对互联网运营环境进行优化，对用户身份及 IP 地址严格管理，提高云服务下用户数据信息安全。（4）阅读隐私声明使用过滤器，使用过滤器对数据进行邮箱检测，确保数据在流失时可被及时发现。（5）云计算是将大量计算资源、存储资源和软件资源连接在一起，形成大规模虚拟共享资源地，在存储数据的时候，应该把数据交给公共数据中心统一加密处理。

二　填空题（每空 1 分，共 20 分）

1. ISO 7498-2 确定了五大类安全服务，即鉴别、访问控制、数据保密性、数据完整性和不可否认。同时，ISO 7498-2 也确定了八类安全机制，即加密机制、数据签名机制、访问控制机制、数据完整性机制、认证交换、业务填充机制、路由控制机制和公证机制。
2. 古典密码包括代替密码和置换密码两种，对称密码体制和非对称密码体制都属于现代密码体制。传统的密码系统主要存在两个缺点：一是密钥管理与分配问题；二是认证问题。在实际应用中，对称密码算法与非对称密码算法总是结合起来的，对称密码算法用于加密，而非对称算法用于保护对称算法的密钥。
3. 根据使用密码体制的不同可将数字签名分为基于对称密码体制的数字签名和基于公钥密码体制的数字签名，根据其实现目的的不同，一般又可将其分为直接数字签名和可仲裁数字签名。
4. DES 算法密钥是 64 位，其中密钥有效位是 56 位。RSA 算法的安全是基于分解两个大素数的积的困难。
5. 密钥管理的主要内容包括密钥的生成、分配、使用、存储、备份、恢复和销毁。密钥生成形式有两种：一种是由中心集中生成，另一种是由个人分散生成。
6. 认证技术包括站点认证、报文认证和身份认证，而身份认证的方法主要有口令、磁卡和智能卡、生理特征识别、零知识证明。
7. NAT 的实现方式有三种，分别是静态转换、动态转换、端口多路复用。
8. 数字签名是笔迹签名的模拟，是一种包括防止源点或终点否认的认证技术。

2. P^2DR^2 模型的策略域分析

网络系统是由参与信息交互的各类实体元素构成，可以是独立计算机、局域网络或大规模分布式网络系统。实体集合可包括网络通信实体集、通信业务类型集和通信交互时间集。

通信实体集的内涵表示发起网络通信的主体，如：进程、任务文件等资源；对于网络系统，表示各类通信设备、服务器以及参与通信的用户。网络的信息交互的业务类型存在多样性，根据数据服务类型、业务类型，可以划分为数据信息、图片业务、声音业务；根据 IP 数据在安全网关的数据转换服务，业务类型可以划分为普通的分组；根据 TCP/IP 协议传输协议，业务类型可以划分为 ICMP、TCP、UDP 分组。信息安全系统根据不同安全服务需求，使用不同分类法则。通信交互时间集则包含了通信事件发生的时间区域集。

安全策略是信息安全系统的核心。大规模信息系统安全必须依赖统一的安全策略管理、动态维护和管理各类安全服务。安全策略根据各类实体的安全需求，划分信任域，制定各类安全服务的策略。

在信任域内的实体元素，存在两种安全策略属性，即信任域内的实体元素所共同具有的有限安全策略属性集合，实体自身具有的、不违反 S_a 的特殊安全策略属性 S_{pi} 。由此我们不难看出，$S=S_a+\Sigma S_{pi}$.

安全策略不仅制定了实体元素的安全等级，而且规定了各类安全服务互动的机制。每个信任域或实体元素根据安全策略分别实现身份验证、访问控制、安全通信、安全分析、安全恢复和响应的机制选择。

（1）服务器在口令文件中存储用户 ID、口令 PWD 及一个随机数 r（即盐）的散列值：H=h(pwd+r)；（2 分）

（2）用户输入用户名登录服务器，服务器根据用户 ID 查询是否是合法用户，如果是则从口令文件中取出随机数 r 并发送给用户；（1 分）

（3）用户将自己的口令和收到的 r 生成散列值并发送给服务器：H'= h(pwd+r)；（2 分）

（4）服务器比较 H=H'？如果相同则认证通过，否则认证失败；（2 分）

（5）服务器更新随机数 r=r'。（1 分）

参考答案二：S/KEY 方法

认证系统组成：用户、服务器、口令文件和认证协议；（各 0.5 分，小计 2 分）

认证协议及过程：

（1）服务器在口令文件中存储用户 ID、登录次数 n 以及口令 PWD 的 n 次散列值：$H=h^n(pwd)$；（2 分）

（2）用户输入用户名登录服务器，服务器根据用户 ID 查询是否是合法用户，如果是则从口令文件中取出 n 并发送给用户；（1 分）

（3）用户将自己的口令的 n-1 散列值发送给服务器：$H'= h^{n-1}(pwd)$；（2 分）

（4）服务器计算 h(H')=H？如果相同则认证通过，否则认证失败；（2 分）

（5）服务器更新随机数 n=n-1 以及 $H= h^{n-1}(pwd)$。（1 分）