信息安全基础知识

一. 保密性，完整性，端点认证

通信安全的特性或者说是分类有以下三个主要类别：保密性，完整性，端点认证

1.1 保密性<消息不能被泄露给其他无关的人>

大多数人认为安全就是考虑保密性。保密性的意思就是对无关的听众保密，通常听众是指窃听者。当政府监听你的电话就会对你的保密性构成威胁（同时这也是一种被动攻击，除非联邦调查人员开始试图在线路上模仿你的声音。）

显然，如果你有秘密的话，就会希望没有别人知道这些秘密，所以你最少也需要保密性。电影中常看到间谍在洗手间打开所有水龙头来扰乱窃听者，他们所寻求的特性就是保密性。

1.2 完整性<消息是完全正确的，不多不少不被篡改>

第二个重要的目标就是消息的完整性。这里的基本思想就是，我们想确认自己所收到的消息是从发送者那发送的消息。在纸制系统中会自动带有一的消息完整性。当你收到一封钢笔写的信时，由于钢笔印记很难从纸张上去除，所以你完全可以肯定没有抹掉的词语，然而，攻击者可以很容易地在信中增加一些笔记来彻底改变信息的原意。

另一方面，在电子世界中，由于所有的位看上去都是相似的，所以在传输过程中摆弄起信息来简直就是小菜一碟。你只需将信息从线路中去除，拷贝你想要的部分，随意增加什么数据进去，然后就可以产生一条由你挑选的新信息，而接收者却一无所知。这与攻击者拿到你写的信，买一个新的信纸，然后把消息修改后重新拷贝纸上是一样的。只是采用电子方式则要容易得多。

1.3 端点认证〈发送者和接收者必须是对的，但是认证可以单方认证，不一定是双方都需要，这得看情况〉

所关心的第三个特性是端点认证。通过它所要到达的意图就是要知道通信中的某个端点（通常为发送者）就是我们所指的那个端点。没有端点认证，要提供保密性和消息完整性就非常空难。例如，如果我们收到一份来自 alice 的消息，但无法确认该消息是由 Alice 而不是攻击者发送来的话，那么消息完整性对我们来说就不会有任何意义。与之类似，当我们想 Bob 发送一份机密消息。假如我们实际上是将机密消息发送给了攻击者的话，对我们来说也就没有什么意思可言了（但是如果做好了保密性，至少不会泄露消息的内容给攻击者，只是没有发送给正确的人，也给攻击者提供了一份可研究的加密资料）。

注意端点认证可以以非对称的方式提供。当你给某人打电话时，你可以确信接电话的人是谁——或者实际上至少也是处在你所拨叫的电话号码位置的人。另一方面，如果接电话者没有主叫识别，那么他们也不会知道是谁在给自己打电话。给某人打电话是一个接收方认证的例子，这里你知道接电话的人是谁（要想突破电话网的安全是困难的，但也不是不可能的），但是对方却不知道发送方是谁。

从另一个角度来讲，现金就是一个发送端认证的例子。一张美圆现钞就像是政府签名的消息。政府并不知道是谁拿到了给定的钞票，但是你却可以深信钞票实际上是由 US Mint 印刷的。原因就是货币很难伪造。

二. 消息摘要与数字签名

2.1 什么是消息摘要？

消息摘要（Message Digest）又称作数字摘要(Digital Digest)。它是一个唯一对应一个消息或文本的固定长度的值，它由一个单向 Hash 加密函数对消息进行作用而产生。如果消息在途中改变了，则接收者通过对收到消息的新产生的摘要

与原摘要比较，就可知道消息是否被改变了。因此消息摘要保证了消息的完整性。消息摘要采用单向 Hash 函数将需加密的明文"摘要"成一串 128bit 的密文，这一串密文亦称为数字指纹(Finger Print)。它有固定的长度，且不同的明文摘要成密文，其结果总是不同的，而同样的明文其摘要必定一致。HASH 函数的抗冲突性使得如果一段明文稍有变化，哪怕只更改该段落的一个字母，通过哈希算法作用后都将产生不同的值。而 HASH 算法的单向性使得要找到到哈希值相同的两个不同的输入消息，在计算上是不可能的。所以数据的哈希值，即消息摘要，可以检验数据的完整性。哈希函数的这种对不同的输入能够生成不同的值的特性使得无法找到两个具有相同哈希值的输入。因此，如果两个文档经哈希转换后成为相同的值，就可以肯定它们是同一文档。所以，当希望有效地比较两个数据块时，就可以比较它们的哈希值。例如，可以通过比较邮件发送前和发送后的哈希值来验证该邮件在传递时是否修改。

<总结：消息摘要可以理解为一种算法，可以将任意长度的信息输入后，输出一个固定长度的值，它是不可逆的。因为信息摘要与信息紧密联系，所以它保证了消息的完整性，如果信息被篡改，则信息摘要值与原来的信息摘要值 M 不等。发送方和接收方都有计算摘要值的函数，所以当接收方收到消息，就可以得到另一个摘要值 M ‘(M≠M）>

2.2 消息摘要的用途：

数字签名和消息人证码（MAC message authentication code）

签名：使用私用密钥对消息摘要进行加密，得到签名。

这里提到了私用密钥，那么就来解释一下公用密钥和私用密钥的区别：以发送方 A 为例，它想发送信息给 B（接收方），需要以下步骤：

1 A 用自己的私用密钥加密消息摘要（A 用要发送的信息及消息摘要函数计算出消息摘要）得到签名

2 A 用 B 的公用密钥加密传送的消息明文（未加密的消息）及签名

3 B 用自己的私用密钥来解密 A 发来的数据包，得到 A 的签名和消息明文

4 B 用消息摘要函数，计算消息明文，得到消息摘要 M'

5 如果消息摘要 M 和 M '相等，说明消息是完整的，未被篡改。

三 MAC 消息验证码及密钥管理问题

3.1 消息验证码

假设 Alice 和 Bob 共享一个密钥，Alice 想给 Bob 发送一条消息，而 Bob 将会知道它是 Alice 发的。如果她加密的话，非常简单，只需将他们的共享密钥用做加密密钥即可，但是如我们所讲的，这种方法并不能提供任何信息未被篡改的真正保证。我们需要一种新的工具，即消息验证码（message authentication codes,MAC）。MAC 类似于摘要算法，但是它在计算的时候还要采用一个密钥，因此 MAC 同时依赖于所使用的密钥以及要计算起 MAC 的信息。实际上 MAC 通常是根据摘要算法构造得出的。<可以理解为 MAC=密钥+消息摘要>

尽管存在许多基于各种摘要算法来构造 MAC 的尝试，但是因特网安全团体就一种构造方法达成了一致，它被称做 HMAC[Krawzyk1997]，这种方法描述了如何基于满足某种合理假定摘要来创建具有可证明的安全特性的 MAC。SSLv3 中使用的是一中 HMAC 的变种，而真正的 HMAC 在 TLS 中使用。

3.2 密钥管理的问题

Alice 拿到我们的消息，使用共享密钥对信息进行加密，添加一个也是基于该密钥构造的 MAC 并将其发送给 Bob。她知道只有 Bob 能够阅读这条信息，因为 Bob 与其共享解密时所需要的密钥。类似的，Bob 知道发送这条消息的只

有 Alice，因为只有 Alice 才具有创建消息上的 MAC 时所需要的密钥。这样，Bob 就可以知道是 Alice 发送的信息，而且还未被篡改。

那么，我们就有了所需要的一切，是吗？不。与每个人进行通信，仍然存在与其共享密钥的问题。周围有这么多密钥需要处理非常不方便。<设想一下如果有两个人那么需要交换 1 个密钥，如果有三个人侧需要交换 3 个密钥，如果有 n 个人，这时就需要 n（n-1）/2 个密钥>但更重要的是，这意味着为了进行密钥交换，你实际上必须要与每一个与之通信的人会面。这为通过因特网购买商品设置了障碍，除非你个人已经与供应商碰过面。这里面不便之处就是密钥管理的问题。

因为 Alice 与 Bob 要共享密钥，但是 A 和 B 没有碰过面，那么就要有一方要发送密钥给另一方，但是这个密钥是需要保密的，不能在网络上直接传送。所以就涉及到了密钥的管理问题。有人会说，直接传送能怎么样呢，这就会遭受密钥被攻击者截获（端认证没有被保证），消息被截获并且泄露。MAC 只能保证消息不被篡改，密钥用来保护消息不被泄露。

四 KDC、公用密钥加密和证书

4.1KDC（密钥分发中心）

针对密钥管理问题最流行的解决方案就是公用密钥加密（public key cryptography，PKC）。不过也存在一种只使用到目前为止所讨论的工具来解决密钥管理问题的措施。基本思想就是利用受信任的第三方，我们委托它对与我们通信的各方进行认证。这种第三方通常是由网络上某处一台安全的机器来实现的。这台机器被称做密钥分发中心（key distribution center，KDC）。每个需要保密通信安全的个人都与 KDC 共享一个密钥。当 Alice 想要与 Bob 进行通信时，她就给 KDC 发送消息，该消息以其与 KDC 共享的密钥加以保护，请求与 Bob 进

行通信。KDC 产生了一个新的用于 Alice 与 Bob 之间进行通信的加密密钥，再将其放在一条称做许可证（ticket）的消息中返回。

一条许可证消息由两条消息组成。第一条消息是给 Alice 的，其中带有新密钥。第二条消息是给 Bob 的，用 Bob 的密钥进行加密，其中也包含新密钥。Alice 将许可证中 Bob 的那一部分转交给 Bob，现在 Alice 与 Bob 共享密钥。这种协议的基础版本是由 Needham 和 chroeder[Needham 1978]发明的，但是部署最为广泛的变种为 Kerberos，在 MIT 及其他地方大量应用于认证与加密（请参见[Miller1987]）。

这种方案有两种主要缺点。首先，KDC 必须总是处于联机状态，因为如果它下线的话，就无法完成通信初始化。其次 KDC 能够读取任何两方之间传递的数据。它还能够伪造两方之间的通信。更糟的是，如果 KDC 被攻克的话，任何两个 KDC 用户之间的通信均将遭难。尽管如此，对于封闭系统来说，人们对这种类型的协议还颇有兴趣。

4.2 公用密钥加密

1976 年，Stanford 有几个非常聪明的人，想出了一种更好的解决密钥管理问题的方法。在一篇名为“密码术新动向”[Diffie1976]的论文中，Whitfield Diffie 和 Martin Hellman 提出了现在称之为公用密钥加密的方案。基本思想就是设计一种在加密和解密时使用不同密钥的函数。你公开自己的加密密钥（公用密钥），但解密密钥（私用密钥）要保密。（由于公用与私用密钥的不同，公用密钥加密有时被称做非对称加密，而共享密钥加密有时被称做对称加密）。这意味着无需碰面，任何人都能够给你发送保密信息。这在消除需要预先共享密钥的不方便之处的同时也解决了其中的保密性的问题。

事实表明 PKC 针对问题中的认证部分也有一套解决方案。你的私用密钥可以用来创建某种称作数字签名的东西，它与 MAC 之间的关系如同公用密钥加密与秘密密钥加密之间的关系一样。你使用私用密钥对消息进行签名，而接收方使用你的公用密钥来验证你的签名。

<注意，数字签名具有一项 MAC 所不具备的重要属性：不可抵赖性（nonrepudiation）。发送方和接收方都可以产生 MAC，但是只有签名者才能够产生签名。这样，接收者就可以证明发送方对消息进行了签名而发送方无法抵赖。>

4.3 证书

尽管提供了我们解决问题所需要的工具，不幸的是，到目前为止我们所拥有的工具并不能完全解决密钥管理问题。问题处在来获取彼此的公用密钥的过程。如果这些密钥是以电子形式发表的，或是通信各方通过交换得到的话，那么攻击者就能够在这些密钥传递给接收者的过程中进行篡改。当两方打算进行通信时，攻击者截获他们的密钥，并代之再将自己的密钥发送给每一方。这样每一方都会按照攻击者的要求来进行加密，而攻击者根据真正的接收者重新进行加密，如图 1.2 所示，这被称作中间人攻击（man-in-the-middle attack）。然而，如果将密钥以物理方式印刷出来，则很不方便。解决方案（还是）就是通过称之为证书授予权（certificate authority，CA）的第三方。CA 发布以其私用密钥签名的目录。在实际应用中，CA 不是对目录进行签名，而是对包含密钥属主及其公用密钥的单一信息进行签名。这些信息一般被称作证书（Certificate），证书授予权因而得名。证书的主要标准为 X.509[ITU1988a]，它是在 RFC2459[Housley1999a]中为因特网编写的。

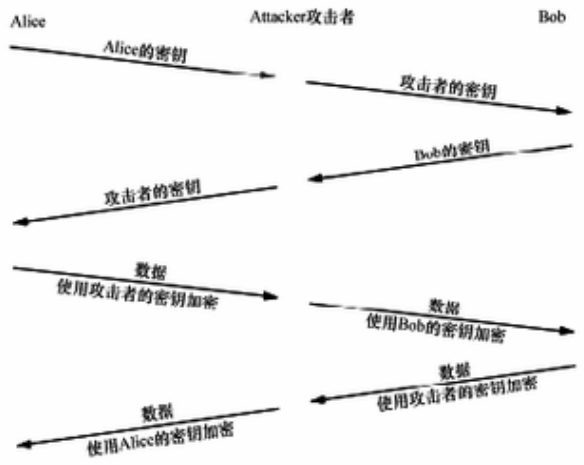

图 1.2　中间人攻击

CA 的公用密钥以某种物理形式发布，不过 CA 并不多而且不经常变换密钥，因此这在实际应用中不成问题。通常将 CA 密钥编译到需要使用它们的软件中，这样就能与软件一起发行。当软件以 CD-ROM 或软盘方式发布时就非常有意义。但有时软件只是下载得到的，在这种情况下就又回到了刚开始时遇到的问题——即人们做事常常并不理智。 图 1.3 描述了一份公用密钥证书展开后的视图。这里的细节并不重要，但请注意其中的基本结构：证书包含颁发者名称（issuer name）（证书签名者的名字，这里就是“SecurServer...”），主体名称（subject name）（证书所担保的密钥的持有者，在这里就是“www.amazon.com...”），主体公用密钥（subject public key）（即密钥本身），一组控制信息诸如有效期限、序列号以及对整个数据对象的签名。涉及证书的公用密钥解决方案仍然要包含受信的第三方（即 CA），但是它们的确修正了我们前面所描述的基于 KDC 系统的主要问题。由于同一个证书可以用来向任何人证明其公用密钥，所以 CA 没有必要为了让 Alice 与 Bob 进行通信，而始终处于联机状态。同时因为 CA 无法存取任何人的私用密钥，所以它也不能读取任何信息。

```
version:                              v1
serial number:                        2A 17 EF 73 97 07 74 7B E2 4B FB
      61 95 DB 4D 77
signature
      algorithm:                      md5WithRSAEncryption
issuer:
      C=US
      O=RSA Data Security, Inc.
      OU=Secure Server Certification Authority
validity
      not before:                      Sat Jan 28 02:21:56 1995
      not after:                       Thu Feb 15 02:21:55 1996
subject:
      C=US
      ST=Washington
      L=Seattle
      O=Amazon.com, Inc.
      OU=Software
      CN=www.amazon.com
subject public key
      algorithm:                       rsaEncryption
      modulus
            bit length:                1024
            value:
                  00 C8 1B 8B FA 40 C3 5B E3 46 3F 17 10 56 19 64 C4 F4 F9
                  CC AE CA F7 OB 02 1C C3 2D 27 60 91 16 CO A1 23 8B CA 90
                  77 31 25 CA D9 DE BO 87 F5 25 C9 12 7A 95 DF DC 6C E4 1C
                  C3 31 9F 77 BE 69 3E 9F BB 35 BF F3 3D BA 7A 72 DA 5D 0C
                  60 91 29 F8 89 67 50 5C 32 46 63 F2 FF 42 9D 24 F2 DC 6F
                  E5 CA D3 CD 3A AB 9D 5F A9 4D BO 82 91 E3 D3 EA AA EF 78
                  8A C1 06 B6 6D EA 56 B8 7E 68 5D AF 4D 85 AF
      public exponent:
            bit length:                2
            value:                     03
signature
      value:
            03 43 60 4B 5B 4B F1 78 56 BF B4 9B 81 E6 EE 0D 19 1B 4E 43 BD
            D9 C7 62 62 55 32 C7 15 A4 33 3A CA 0E 60 E5 FE D7 53 94 C6 AC
            17 D0 CE 7B 11 27 0C 3B 26 19 6D 35 55 4C D8 26 F4 5F FO 90 0D
            90 7F FC 39 47 FE EE B4 72 92 93 BF 93 7F 5C 56 38 10 F5 E5 58
            B5 6C 3E E0 B4 55 8D 74 BE 84 F1 53 67 49 5B 14 12 E6 A7 59 A9
            97  9E 6C E4  59 A6 8F 4E 7E B5 D9 2D 80 3F 38 3C 4C 11 A7 37
```

图1.3 公用密钥证书

五 主动攻击与被动攻击

5.1.被动攻击

被动攻击即窃听,是对系统的保密性进行攻击,如搭线窃听、对文件或程序的非法复制等,以获取他人的信息。被动攻击又分为两类:一类是获取消息的内容,很容易理解;另一类是进行业务流分析,假如通过某种手段,比如加密,使得敌手无法从截获的消息得到消息的真实内容,然而敌手却有可能获得消息的格式、确定通信双方的位置和身份以及通信的次数和消息的长度,这些信息对通信双方来说

可能是敏感的,例如公司间的合作关系可能是保密的、电子函件用户可能不想让他人知道自己正在和谁通信、电子现金的支付者可能不想让别人知道自己正在消费、Web 浏览器用户也可能不愿意让别人知道自己正在浏览哪一站点。

被动攻击因不对消息做任何修改,因而是难以检测的,所以抗击这种攻击的重点在于预防而非检测。

5.2 主动攻击

主动攻击包括对数据流的篡改或产生某些假的数据流。主动攻击又可分为以下 3 类:

①中断 是对系统的可用性进行攻击。如破坏计算机硬件、网络或文件管理系统。

②篡改 是对系统的完整性进行攻击。如修改数据文件中的数据、替换某一程序使其执行不同的功能、修改网络中传送的消息内容等。

③伪造 是对系统的真实性进行攻击。如在网络中插入伪造的消息或在文件中插入伪造的记录

绝对防止主动攻击是十分困难的,因为需要随时随地对通信设备和通信线路进行物理保护,因此抗击主动攻击的主要途径是检测,以及对此攻击造成的破坏进行恢复。

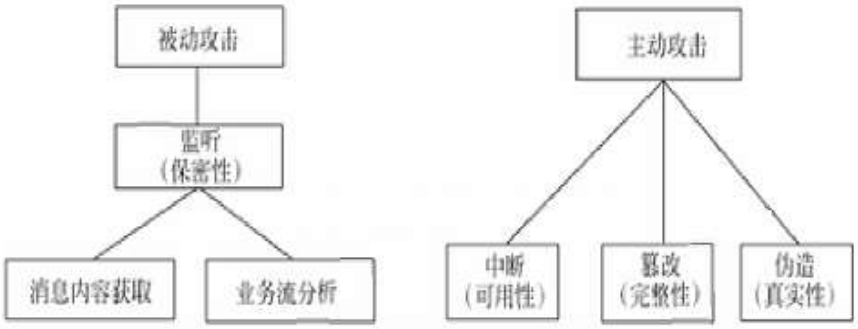

图　　攻击类型分类

补充：

这里就牵扯到了可靠性、可用性、保密性、完整性、不可抵赖性、可控性等的概念

1：可靠性

可靠性是网络信息系统能够在规定条件下和规定的时间内完成规定的功能的特性。可靠性是系统安全的最基于要求之一，是所有网络信息系统的建设和运行目标。

网络信息系统的可靠性测度主要有三种：抗毁性、生存性和有效性。

抗毁性是指系统在人为破坏下的可靠性。比如，部分线路或节点失效后，系统是否仍然能够提供一定程度的服务。增强抗毁性可以有效地避免因各种灾害（战争、地震等）造成的大面积瘫痪事件。

生存性是在随机破坏下系统的可靠性。生存性主要反映随机性破坏和网络拓扑结构对系统可靠性的影响。这里，随机性破坏是指系统部件因为自然老化等造成的自然失效。

有效性是一种基于业务性能的可靠性。有效性主要反映在网络信息系统的部件失效情况下，满足业务性能要求的程度。比如，网络部件失效虽然没有引起连

接性故障，但是却造成质量指标下降、平均延时增加、线路阻塞等现象。

可靠性主要表现在硬件可靠性、软件可靠性、人员可靠性、环境可靠性等方面。硬件可靠性最为直观和常见。软件可靠性是指在规定的时间内，程序成功运行的概率。人员可靠性是指人员成功地完成工作或任务的概率。人员可靠性在整个系统可靠性中扮演重要角色，因为系统失效的大部分原因是人为差错造成的。人的行为要受到生理和心理的影响，受到其技术熟练程度、责任心和品德等素质方面的影响。因此，人员的教育、培养、训练和管理以及合理的人机界面是提高可靠性的重要方面。环境可靠性是指在规定的环境内，保证网络成功运行的概率。这里的环境主要是指自然环境和电磁环境。

2：可用性

可用性是网络信息可被授权实体访问并按需求使用的特性。即网络信息服务在需要时，允许授权用户或实体使用的特性，或者是网络部分受损或需要降级使用时，仍能为授权用户提供有效服务的特性。可用性是网络信息系统面向用户的安全性能。网络信息系统最基本的功能是向用户提供服务，而用户的需求是随机的、多方面的、有时还有时间要求。可用性一般用系统正常使用时间和整个工作时间之比来度量。

可用性还应该满足以下要求：身份识别与确认、访问控制（对用户的权限进行控制，只能访问相应权限的资源，防止或限制经隐蔽通道的非法访问。包括自主访问控制和强制访问控制）、业务流控制（利用均分负荷方法，防止业务流量过度集中而引起网络阻塞）、路由选择控制（选择那些稳定可靠的子网，中继线或链路等）、审计跟踪（把网络信息系统中发生的所有安全事件情况存储在安全审计跟踪之中，以便分析原因，分清责任，及时采取相应的措施。审计跟踪的信

息主要包括：事件类型、被管客体等级、事件时间、事件信息、事件回答以及事件统计等方面的信息。）

3：保密性

保密性是网络信息不被泄露给非授权的用户、实体或过程，或供其利用的特性。即，防止信息泄漏给非授权个人或实体，信息只为授权用户使用的特性。保密性是在可靠性和可用性基础之上，保障网络信息安全的重要手段。

常用的保密技术包括：防侦收（使对手侦收不到有用的信息）、防辐射（防止有用信息以各种途径辐射出去）、信息加密（在密钥的控制下，用加密算法对信息进行加密处理。即使对手得到了加密后的信息也会因为没有密钥而无法读懂有效信息）、物理保密（利用各种物理方法，如限制、隔离、掩蔽、控制等措施，保护信息不被泄露）。

4：完整性

完整性是网络信息未经授权不能进行改变的特性。即网络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。完整性是一种面向信息的安全性，它要求保持信息的原样，即信息的正确生成和正确存储和传输。

完整性与保密性不同，保密性要求信息不被泄露给未授权的人，而完整性则要求信息不致受到各种原因的破坏。影响网络信息完整性的主要因素有：设备故障、误码（传输、处理和存储过程中产生的误码，定时的稳定度和精度降低造成的误码，各种干扰源造成的误码）、人为攻击、计算机病毒等。

保障网络信息完整性的主要方法有：

协议：通过各种安全协议可以有效地检测出被复制的信息、被删除的字段、

失效的字段和被修改的字段；

纠错编码方法：由此完成检错和纠错功能。最简单和常用的纠错编码方法是奇偶校验法；

密码校验和方法：它是抗窜改和传输失败的重要手段；

数字签名：保障信息的真实性；

公证：请求网络管理或中介机构证明信息的真实性。

5： 不可抵赖性

不可抵赖性也称作不可否认性，在网络信息系统的信息交互过程中，确信参与者的真实同一性。即，所有参与者都不可能否认或抵赖曾经完成的操作和承诺。利用信息源证据可以防止发信方不真实地否认已发送信息，利用递交接收证据可以防止收信方事后否认已经接收的信息。

6：可控性

可控性是对网络信息的传播及内容具有控制能力的特性。

概括地说，网络信息安全与保密的核心是通过计算机、网络、密码技术和安全技术，保护在公用网络信息系统中传输、交换和存储的消息的保密性、完整性、真实性、可靠性、可用性、不可抵赖性等。

针对以上个个特性，有以下安全业务进行解决。

1.保密业务

保护数据以防被动攻击。保护方式可根据保护范围的大小分为若干级,其中最高级保护可在一定时间范围内保护两个用户之间传输的所有数据,低级保护包括对单个消息的保护或对一个消息中某个特定域的保护。保密业务还包括对业务流实施的保密,防止敌手进行业务流分析以获得通信的信源、信宿、次数、消息

长度和其他信息。

2.认证业务

用于保证通信的真实性。在单向通信的情况下,认证业务的功能是使接收者相信消息确实是由它自己所声称的那个信源发出的。在双向通信的情况下,例如计算机终端和主机的连接,在连接开始时,认证服务则使通信双方都相信对方是真实的(即的确是它所声称的实体);其次,认证业务还保证通信双方的通信连接不能被第三方介入,以假冒其中的一方而进行非授权的传输或接收。

3.完整性业务

和保密业务一样,完整性业务也能应用于消息流、单个消息或一个消息的某一选定域。用于消息流的完整性业务目的在于保证所接收的消息未经复制、插入、篡改、重排或重放,即保证接收的消息和所发出的消息完全一样;这种服务还能对已毁坏的数据进行恢复,所以这种业务主要是针对对消息流的篡改和业务拒绝的。应用于单个消息或一个消息某一选定域的完整性业务仅用来防止对消息的篡改。

4.不可否认业务

用于防止通信双方中的某一方对所传输消息的否认,因此,一个消息发出后,接收者能够证明这一消息的确是由通信的另一方发出的。类似地,当一个消息被接收后,发出者能够证明这一消息的确已被通信的另一方接收了。

5.访问控制

访问控制的目标是防止对网络资源的非授权访问,控制的实现方式是认证,即检查欲访问某一资源的用户是否具有访问权。