第 2 章　网络体系结构及协议基础

了解计算机网络知识是学习网络安全必不可少的基础。本章将对一些基本的、与网络安全联系紧密的网络知识作一个简单的介绍。通过本章的学习，应达到以下目标：

- 了解 OSI 模型及其安全体系
- 了解 TCP/IP 网络模型及其安全体系结构
- 掌握常用的网络协议和网络命令
- 掌握协议分析工具的使用方法

网络体系结构是计算机之间相互通信的层次以及各层中的协议和层次之间接口的集合。体系结构是一个抽象的概念，它精确定义了网络及其部件所应实现的功能，但这些功能究竟用何种硬件或软件方法来实现则是一个具体实施的问题。换言之，网络的体系结构相当于网络的类型，而具体的网络结构则相当于网络的一个实例。本章将从网络体系结构入手，讲述网络安全机制、安全服务以及协议和应用等一系列知识。

2.1　网络的体系结构

2.1.1　网络的层次结构

计算机网络系统是一个十分复杂的系统，将一个复杂系统分解为若干个容易处理的子系统，然后“分而治之”，逐个加以解决，是工程设计中常用的结构化设计方法。分层就是系统分解的最好方法之一。层次结构的好处在于使每一层实现一种相对独立的功能，每一层向上一层提供服务，同时接受下一层提供的服务。每一层不必知道下面一层是如何实现的，只要知道下层通过层间接口提供的服务是什么，以及本层向上层提供什么样的服务，就能独立地设计，这就是常说的网络层次结构，如图 2-1 所示。系统经分层后，每一层次的功能相对简单且易于实现和维护。此外，若某一层需要做改动或被替代时，只要不改变它和上、下层的接口服务关系，则其他层次都不会受其影响，因此具有很大的灵活性。分层结构还有利于交流、理解和标准化。

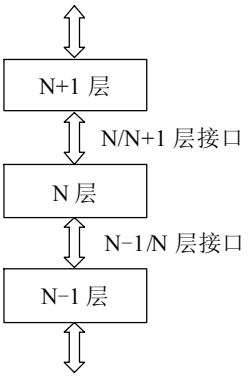

图 2-1　网络的层次结构

2.1.2　服务、接口和协议

每一层的活动元素称为实体，实体可以是软件实体（如进程），也可以是硬件实体。位于不同系统上的同一层中的实体称为对等实体，不同系统间进行通信实际上是各对等实体间的通信。在某层上进行通信所使用的规则、标准或约定的集合就称为协议（Protocol）。各层协议按层次顺序排列而成的协议序列称为协议栈。协议主要由以下 3 个要素组成：

（1）语义（Semantics）。涉及用于协调与差错处理的控制信息。

（2）语法（Syntax）。涉及数据及控制信息的格式、编码及信号电平等。

（3）定时（Timing）。涉及速度匹配和排序等。

除了在最底层的物理介质上进行的是实通信以外，其余各对等实体间进行的都是虚通信，即并没有数据流从一个系统的第 N 层直接流到另一个系统的第 N 层。每个实体只能和同一系统中上下相邻的层中的实体进行直接通信，不同系统中的对等实体是没有直接通信能力的，它们之间的通信必须通过其下各层的通信间接完成。第 N 层实体向第 N+1 层实体提供的在第 N 层上的通信能力称为第 N 层的服务。即第 N+1 层实体通过请求第 N 层的服务完成第 N+1 层上的通信，而第 N 层实体通过请求第 N−1 层的服务完成第 N 层上的通信，依此类推，直至到达最底层，最底层的对等实体通过连接在它们之间的物理介质进行直接的通信。

在接口处规定了下层向上层提供的服务，以及上下层实体请求或提供服务所使用的形式规范语句，这些形式规范语句称为服务原语。就是说，相邻层的实体通过发送或接收服务原语进行作用。下层向上层提供的服务分为两大类，分别为：面向连接的服务和面向无连接的服务。面向连接的服务是电话系统服务模式的抽象，每一次完整的数据传输都必须经过建立连接、使用连接和终止连接三个过程。面向连接的服务就像在两个实体间提供一个管道，发送者在一端输入数据，接收者从另一端取出数据。其特点是：收发数据不但顺序一致而且内容相同。无连接服务是邮政系统服务的抽象，其中每个数据分组都带有完整的信宿地址，各数据分组在系统中独立传送。无连接服务不能保证数据分组的先后顺序，由于先后发送的数据分组可能经不同的路由去往信宿，所以收到的顺序不确定。

2.2　OSI 模型及其安全体系

2.2.1　OSI/RM

1. OSI/RM 的层次结构

开放系统互连参考模型（Open System Interconnection/Reference Model，OSI/RM）是由国际标准化组织（ISO）制定的标准化开放式计算机网络层次结构模型，“开放”这个词表示能使任意两个遵守参考模型和有关标准的系统进行互连。

OSI 包括体系结构、服务定义和协议规范三级抽象。OSI 的体系结构定义了一个七层模型，用以进行进程间的通信，并作为一个框架来协调各层标准的制定；OSI 的服务定义描述了各层所提供的服务，以及层与层之间的抽象接口和交互用的服务原语；OSI 各层的协议规范，精确地定义了应当发送何种控制信息及用何种过程来解释该控制信息。直至今日，OSI/RM 仍是学习网络技术最好的模型，有助于对网络通信概念的理解。OSI 是一个概念上的框架，利用它可

以更好地理解不同网络设备间的交互。OSI 模型只定义需要完成的任务和提供的服务，实际工作由实际网络中相应的软件或硬件完成。

OSI/RM 采用分层的结构化技术，将整个通信网络划分为七层。每层按照一组协议来实现某些网络功能，同时，每一层为其上层提供服务。OSI/RM 的结构如图 2-2 所示。OSI 七层模型从下到上分别为物理层、数据链路层、网络层、传输层、会话层、表示层和应用层。整个开放系统环境由作为信源和信宿的端开放系统及若干中继开放系统通过物理媒体连接构成。这里的端开放系统和中继开放系统，都是国际标准 ISO 7498 中使用的术语。通俗地说，它们就相当于资源子网中的主机和通信子网中的结点机（IMP）。只有在主机中才可能需要包含所有七层的功能，而在通信子网中的 IMP，一般只需要最低三层甚至只要最低两层的功能就可以了。

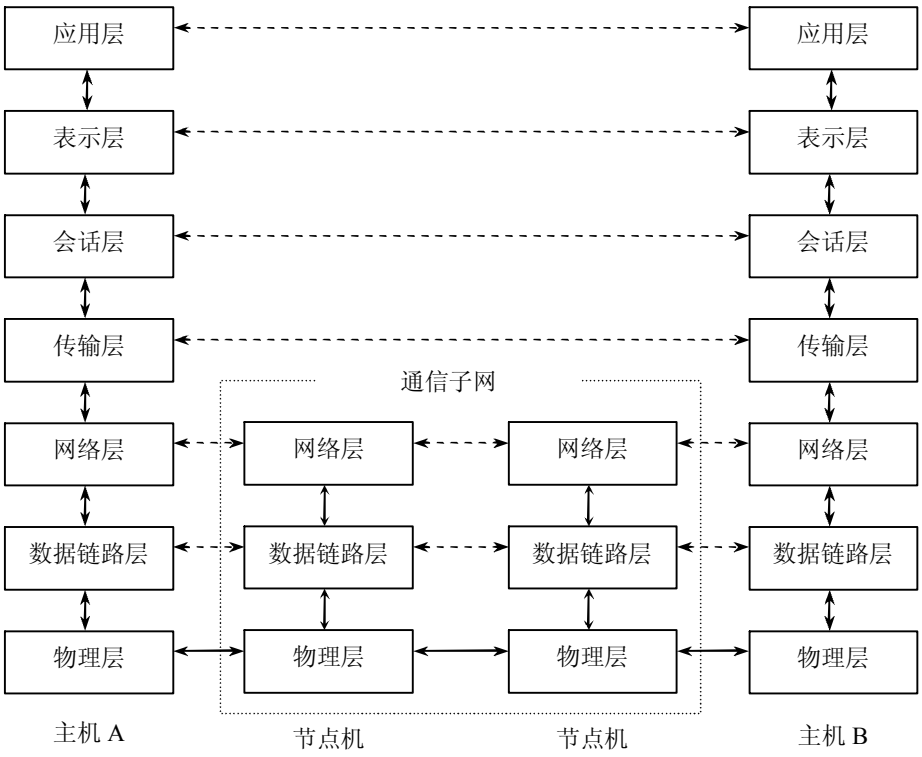

图 2-2　OSI 模型的分层示意图

（1）物理层（Physical Layer）。物理层是 OSI/RM 的最底层，它定义了通信介质的机械特性、电气特性、功能特性和过程特性，以建立、维持和拆除物理连接。物理层建立在物理通信介质之上，是系统和通信介质的接口。

（2）数据链路层（Data Link Layer）。数据链路层检测和校正物理层可能发生的差错，从而构成一条无差错的链路。数据链路层将从其上层接收的数据包封装成特定格式的数据单元，这种数据单元称为“帧”，在帧中除了数据部分外还附加了一些控制信息，如帧类型、流量控制、差错控制信息等，可以实现数据流控制、差错控制及发送顺序控制等功能。

（3）网络层（Network Layer）。网络层主要实现线路交换、路由选择和网络拥塞控制等功能，保证信息包在接收端以准确的顺序接收。

（4）传输层（Transport Layer）。传输层负责实现端到端的数据报文的传递。传输层提供了两端点之间可靠、透明的数据传输，执行端到端的差错控制、流量控制及管理多路复用。

（5）会话层（Session Layer）。会话层是网络会话控制器，它建立、维护和同步通信设备之间的交互操作，保证每次会话都正常关闭。会话层建立和验证用户之间的连接，控制数据交换，决定以何种顺序将对话单元传送到传输层，决定传输过程中哪一点需要接收端的确认。

（6）表示层（Presentation Layer）。表示层的目的是为了保证通信设备之间的互操作性。由于不同的计算机系统中数据的表示不同（如使用不同的编码方式），通过表示层的处理可以消除不同实体之间的语义差异。还可以代表应用进程协商数据表示，完成数据转换、格式化和文本压缩等。

（7）应用层（Application Layer）。应用层是用户与网络的接口，它直接为网络用户或应用程序提供各种网络服务。应用层提供的网络服务包括文件服务、事务管理服务、网络管理服务、数据库服务等。

OSI 协议有 3 个主要的概念，分别为：服务、接口和协议。服务定义某一层应该做什么；接口告诉处于上一层的进程如何访问该层；协议定义实体间数据通信的规则。

2. OSI/RM 的数据格式

当网络中的两个主机进行通信时，使用对等层通信协议，即在不同主机的同一层的数据具有相同的封装格式。从表面上看，好像数据是从一台主机的第 N 层直接到达另一台主机的第 N 层，而实际并非如此。假设数据从主机 A 发送到主机 B，数据从主机 A 的应用层依次向下一层传送，每经过一层都要加一个信息头，到达物理层后，数据通过传输介质传送到主机 B 的物理层，然后再依次向上一层传递，每经过一层去掉相应的信息头，这样不同的主机的同一层信息表示是相同的，系统可以根据对等层协议来理解和控制信息。图 2-3 表示了在主机 A、B 之间交换数据的格式和路径。实线表示数据实际的传输路径，虚线表示虚拟的对等层通信。

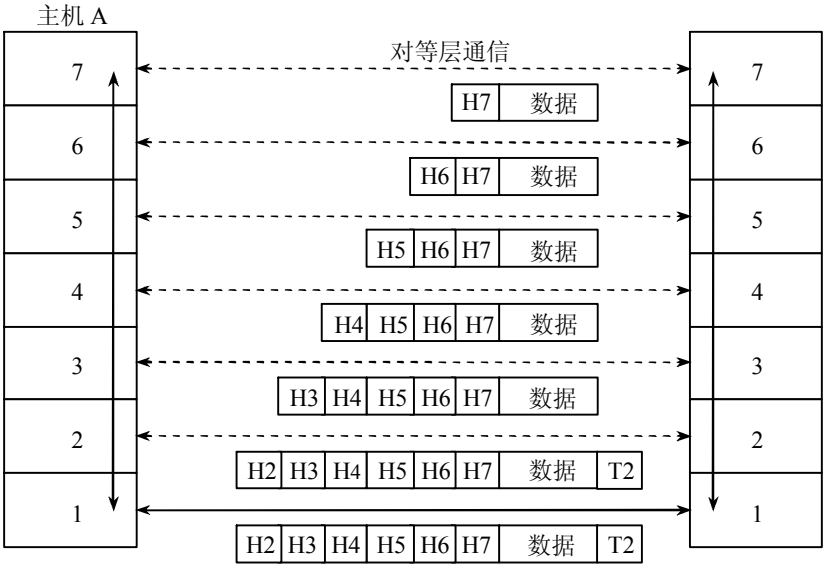

图 2-3　OSI-RM 中的数据交换路径和格式

2.2.2 OSI 模型的安全服务

安全服务是指计算机网络提供的安全防护措施。国际标准化组织（ISO）定义了以下几种基本的安全服务：认证服务、访问控制、数据机密性服务、数据完整性服务、抗否认服务。

1. 认证服务

认证服务用于确保某个实体的身份的真实性。可分为两种类型：

（1）对等实体认证。对等实体认证是指参与通信连接或会话的一方向另一方提供身份证明，接收方通过一定的方式来鉴别实体所提供的身份证明的真实性。对等实体鉴别是保障网络安全最基本的操作，实体的许多后继的活动都取决于鉴别的有效性。例如，实体身份一旦得到确认，就可以和访问控制列表中的权限关联起来，决定能否进行访问。

（2）数据源发认证。某个数据的发送者在发送数据时向接收方提交身份证明，这个身份证明同具体的某些数据相关联，用于确认接收到的数据的来源的真实性，这种鉴别称为数据源发认证。

2. 访问控制

访问控制的目标是防止对任何资源的非授权访问，确保只有经过授权的实体才能访问受保护的资源。所谓未授权访问包括未经授权的使用、泄露、修改、销毁以及发出指令等。访问控制对于保障系统的机密性、完整性、可用性及合法使用具有重要作用。

3. 数据机密性服务

数据机密性服务用于保护信息不泄露给那些没有授权的实体。具体分为以下几种：

（1）连接机密性服务。这种服务对某个连接上传输的全部数据进行加密。

（2）无连接机密性服务。这种服务对一个无连接数据单元的所有数据提供机密性保护。

（3）选择字段机密性服务。这种服务对某个数据单元中那些被选择的字段进行加密。

（4）通信业务流机密性服务。这种服务提供的保护，使攻击者无法通过观察通信业务流来推断出其中的机密信息。

4. 数据完整性服务

数据完整性服务对付主动威胁，保证数据在从起点到终点的传输过程中，如果因为机器故障或人为的原因而造成数据的丢失、被篡改等问题，接收端能够知道或恢复这些改变，从而保证接收到的数据的真实性。数据完整性服务可分为以下几种：

（1）带恢复的连接完整性服务。这种服务对某个连接上传输的所有数据进行完整性保护，并对检测到的数据的任何篡改、插入、删除进行补救或恢复。

（2）无恢复的连接完整性服务。与“带恢复的连接完整性”的服务相同，但不做补救或恢复工作。

（3）选择字段连接完整性服务。这种服务为在一次连接上传送的数据单元中的一些指定的字段进行完整性保护，以确定这些被选字段是否遭受了篡改、插入、删除或变得不可用。

（4）无连接完整性服务。这种服务为单个的无连接数据单元中的所有数据进行完整性保护。

（5）选择字段无连接完整性服务。这种服务为单个的无连接的数据单元中一些被选中的字段提供完整性保护。

5. 抗否认服务

否认是指参与通信的一方事后不承认曾发生过本次信息交换。抗否认服务就是用来针对

这种威胁的，可有以下两种形式：

（1）有数据源发证明的抗否认。在这种服务中，数据的接收者可以提供数据的源发证据，这将使发送者不承认发送过这些数据或否认其内容的企图不能得逞。

（2）有交付证明的抗否认。在这种服务中，数据的发送者可以提供数据已经交付的证据，这将使接收者事后不承认收到过这些数据或否认其内容的企图不能得逞。

2.2.3　OSI 模型的安全机制

安全服务是由各种安全机制来实现的，在本节中列出的 8 种安全机制可以设置在适当的某一层上，以提供 2.2.2 节中所述的某些安全服务。

1. 加密机制

加密就是对数据进行密码变换以产生密文。利用加密机制可以提供数据的安全保密，也可以提供通信的保密。加密可以根据不同的需求，在网络结构中的不同层次实现。比如，如果需要保证全部通信业务流的机密性，可以选取物理层加密。如果要求对不同应用提供不同的密钥或对协议中的某些字段进行保护，可以选取表示层加密。由于加密算法耗费大量的处理能力，所以选择字段保护是很重要的。如果希望实现所有端系统到端系统通信的简单块保护，可以选取网络层加密。如果要求带恢复的完整性，同时又具有细粒度保护，可以选取传输层加密。当关系到上述需求中的两项或多项时，可能需要在多个层上提供加密。

2. 数字签名机制

数字签名是对附加在数据单元上的一些数据，或是对数据单元所做的密码变换，这种变换可以使数据单元的接收者确认数据单元的来源和完整性，并使发送者有效地保护数据，防止被人伪造。数字签名机制包括两个主要的操作：对数据单元签名和验证数据单元的签名。签名使用公钥体制进行，签名过程是使用签名者的私钥对数据单元或由该数据单元生成的一个摘要进行加密；验证过程使用签名者的公钥来解密签名从而对其进行验证。签名机制的本质特征是：该签名只有使用签名者的私钥才能产生出来。因而，当该签名得到验证后，它能在事后的任何时候向第三方（例如法官或仲裁人）证明只有那个私有信息的唯一拥有者才能产生这个签名。

3. 访问控制机制

访问控制是依据实体所具有的权限，对实体提出的资源访问请求加以控制。访问控制机制依据该实体已鉴别的身份，或使用有关该实体的信息及该实体的权利进行。如果这个实体试图使用非授权的资源，或者以不正当的方式使用授权资源，那么访问控制功能将拒绝这一企图，另外还可能产生一个报警信号或把它作为安全审计跟踪的一部分进行记录。访问控制系统一般包括主体、客体和安全访问策略。简单地说，主体是指发出访问请求的实体；客体是指被访问的程序、数据等资源；安全访问策略就是一组用于确认主体是否对客体具有访问权限的规则。访问控制机制是系统安全防范应用的最普遍也是最重要的安全机制。

4. 数据完整性机制

实现消息的安全传输，仅用加密方法是不够的。攻击者虽无法破译加密消息，但如果攻击者篡改或破坏了消息，接收者仍无法收到正确的消息。因此，需要有一种机制来保证接收者能够辨别收到的消息是否是发送者发送的原始数据，这种机制称为数据完整性机制。决定数据的完整性涉及两个过程，一个在发送实体上，一个在接收实体上。发送实体给数据单元附加一个消息，这个消息为该数据的函数。接收实体根据接收到的数据能产生一个相应的消息，并通

过与接收到的附加消息的比较来确定接收到的数据是否在传送中被篡改过。

5. 鉴别交换机制

可用于鉴别交换的技术有：使用鉴别信息、密码技术、使用该实体的特征或占有物等。这种机制可设置在网络的第 N 层以提供对等实体鉴别。如果在鉴别实体时得到否定的结果，就会导致连接被拒绝或终止，或在安全审计跟踪中增加一个记录，或向安全管理中心报警。

6. 通信流量填充机制

通信流量填充机制用来防止对网络流量进行分析的攻击。有时攻击者通过对通信双方的数据流量的变化进行分析，根据流量的变化来推出一些有用的信息或线索。例如，监视某一项目两个研究小组之间的通信流量，如果流量突然减少，就说明某一项目的研究可能已结束或中止。因此，在此类机密通信中，可以通过生成一定的哑流量填充到业务流量中去，以保持网络流量的基本恒定，使攻击者无法捕获任何信息。

7. 路由选择控制机制

路由选择控制是指发送者可以指定数据通过网络的路径。这样就可以选择一条路径，这条路径上的结点都是可信任的，确保发送的信息不会因通过不安全的结点而受到攻击。

8. 公证机制

公证机制由通信各方都信任的第三方提供，由第三方来确保数据的完整性，数据源、时间及目的地的正确。例如，一个必须在截止期限前发送的消息必须带有由可信时间服务机构提供的时间戳，以证明自己的提交时间。该时间服务机构可以在消息上直接加入时间戳，必要时还可对消息进行数字签名。

2.2.4 OSI 安全服务与安全机制的关系

一种安全服务可由一种或多种安全机制来提供，一种安全机制可以用于不同的安全服务中。ISO 7498-2 标准对实现哪些安全服务应该采用哪种机制给出了一个说明性的描述，参见表 2-1。

表 2-1 OSI 安全服务与安全机制的关系

服务	机制							
	加密	数字签名	访问控制	数据完整性	鉴别交换	通信流量填充	路由控制	公证机制
对等实体鉴别	Y	Y			Y			
数据源发鉴别	Y	Y						
访问控制服务			Y					
连接机密性	Y						Y	
无连接机密性	Y						Y	
选择字段机密性	Y							
通信业务流机密性	Y					Y	Y	
带恢复的连接完整性	Y			Y				
不带恢复的连接完整性	Y			Y				

续表

服务	机制							
	加密	数字签名	访问控制	数据完整性	鉴别交换	通信流量填充	路由控制	公证机制
选择字段连接完整性	Y			Y				
无连接完整性	Y	Y		Y				
选择字段无连接完整性	Y	Y		Y				
有数据源发证明的抗否认		Y		Y				Y
有交付证明的抗否认		Y		Y				Y

注：Y 表示该机制适合提供该种服务，空格表示该机制不适合提供该种服务。

2.2.5　OSI 各层中的安全服务配置

OSI 安全体系结构总结了上述各项安全服务在 OSI 七层中的位置，参见表 2-2。

表 2-2　安全服务与层之间的关系

服务	协议层						
	1	2	3	4	5	6	7
对等实体鉴别			Y	Y			Y
数据源发鉴别			Y	Y			Y
访问控制服务			Y	Y			Y
连接机密性	Y	Y	Y	Y		Y	Y
无连接机密性		Y	Y	Y		Y	Y
选择字段机密性							Y
通信业务流机密性	Y					Y	Y
带恢复的连接完整性							Y
不带恢复的连接完整性			Y	Y			Y
选择字段连接完整性							Y
无连接完整性			Y	Y			Y
选择字段无连接完整性			Y	Y			Y
有数据源发证明的抗否认							Y
有交付证明的抗否认							Y

注：Y 表示该服务应该在相应的层中提供，空格表示不提供。对于第 7 层而言，应用程序本身必须提供这些安全服务。

安全服务的分层配置一般要遵循以下规则：

- 实现一种服务的不同方法越少越好。

- 在多个层上提供安全来建立安全系统是可取的。
- 为安全所需的附加功能不应该、也不必要重复 OSI 的现有功能。
- 避免破坏层的独立性。
- 可信功能度的总量应尽量少。

2.3 TCP/IP 模型及其安全体系

2.3.1 TCP/IP 参考模型

1. TCP/IP 参考模型的层次结构

TCP/IP 参考模型是因特网的前身 ARPANET 及因特网的参考模型。TCP/IP 参考模型共有 4 层，从上至下分别为应用层、传输层、网络层和网络接口层。TCP/IP 模型的结构及各层的数据封装格式如图 2-4 所示。TCP/IP 模型没有会话层和表示层，去掉了 OSI 模型中各层之间存在的一些重复的功能，在实现上比较简练高效。比如：并不是所有的服务都需要可靠的连接服务，如果在 IP 层进行可靠性控制会造成处理能力的浪费，因此 TCP/IP 模型把连接控制服务放到传输层进行，使 IP 层更加简洁。TCP/IP 注重实用的特性，使它在应用领域有着强大的生命力，而 OSI/RM 至今仍只是一种标准，没有推广到应用中去。TCP/IP 的体系结构与 OSI 七层模型的对应关系如图 2-5 所示。

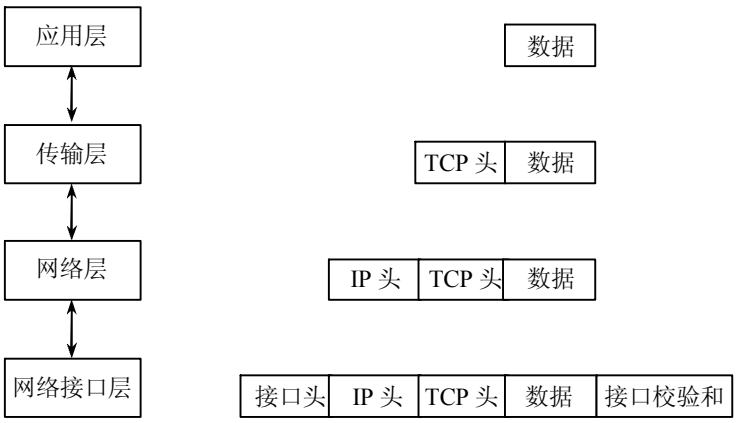

图 2-4 TCP/IP 网络的层次结构及信息格式

2. TCP/IP 模型各层的功能

（1）应用层：大致对应 OSI 的表示层、会话层和应用层，是 TCP/IP 模型的最上层，是用户访问网络的界面。包括一些向用户提供的常用应用程序，如电子邮件、Web 浏览器、文件传输、远程登录等，也包括用户在传输层之上建立的自己的应用程序。

（2）传输层：对应 OSI 的传输层。负责实现源主机和目的主机上的实体之间的通信。它提供了两种服务，一种是可靠的、面向连接的服务（TCP 协议）；一种是无连接的数据报服务（UDP 协议）。为了实现可靠传输，要在会话时建立连接，对数据包进行校验和收发确认，通信完成后再拆除连接。

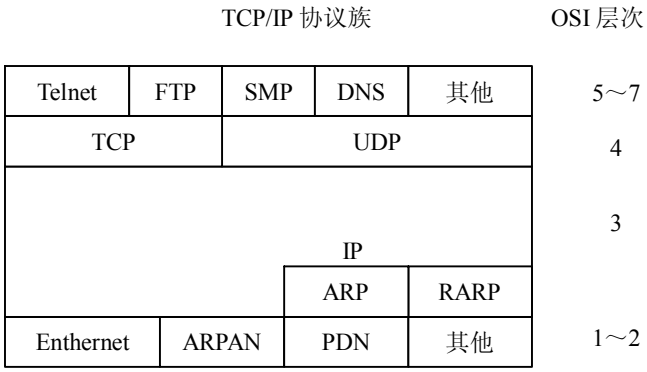

图 2-5　TCP/IP 模型各层包括的主要协议及其与 OSI 层次模型的对应关系

（3）网络层：对应 OSI 的网络层，负责数据包的路由选择，保证数据包能顺利到达指定的目的地。一个报文的不同分组可能通过不同的路径到达目的地，因此要对报文分组加一个顺序标识符，以使目标主机接收到所有分组后，可以按序号将分组装配起来，恢复原报文。

（4）网络接口层：大致对应 OSI 的数据链路层和物理层，是 TCP/IP 模型的最底层。它负责接收 IP 数据包并通过网络传输介质发送数据包。

2.3.2　TCP/IP 的安全体系

1. 链路层安全

只有在各个结点间安装或租用了专门的通信设施，才能对 TCP/IP 网络进行链路层保护。对网络的链路层保护一般可以达到点对点间较强的身份认证、保密性和连续的通道认证，在大多数情况下，也可保证数据流的安全。有些安全服务可以提供数据的完整性服务或至少具有防止欺骗的能力。

链路层保护不能提供真正的终端用户认证，也不能在合理成本下为被保护网络内的用户提供用户间的保密性。如果没有其他附加的安全机制，所有交换路由设备都是不安全的，包括无法限制设备的信息流。如果要求有防火墙之类的功能，应在链路层加密机制之前加入。

由于存在上述局限性，链路层保护可能并不十分有效，但有些保护机制在高层并不容易实现。第一种是通信安全机制。如果用户关心对迫近行为的指示和报警，就必须采用这种安全机制。第二种是高层不拥有的安全机制，如在限制隐通道数目方面的安全机制。隐通道的存在会像幽灵一样对系统构成威胁，数据包中任何字节的改变或传输参数的任何变化都是潜在的隐通道。链路层保护可以有效地去除诸如传输信息长度、时间以及地址的隐通道。其他隐通道起源于对未完全定义的传输杂项位进行利用并对其进行访问控制的能力。

另外，链路层系统设计较为简单，与其他层相比更容易达到预期目标。用户知道现在各种通信技术的限制，所以链路层解决方案将成为最容易被接受的解决方案。

2. 网络层安全

IP 包是一种面向无连接协议的包，通过对通信传输的控制，攻击者有可能修改网络的操作以达到他们的攻击目的，数据包有可能被路由器发往错误的地方，服务可能被局部或全部

拒绝。

动态路由机制确保了数据包在网络中的高效传输，这是任何 IP 网络的重要特征。路由信息和路由表的正确性是相当关键的，它能确保连接的路由不被拒绝，并有效使用网络资源。对网络的可用性来说，确保路由表免受攻击是相当关键的。

路由器间的更新信息必须使用完整性机制，以确保路由更新信息在网络上传送时不会被修改。路由器的内部也需要完整性机制。因此，路由表必须防止非授权用户的非法修改，以确保路由表信息的准确性。另外，还需要认证机制，以确保非授权资源不会将路由更新信息插入网络。

新一代的互联网协议 IPv6 在网络层提供了两种安全机制，即在报文头部包含两个独立的扩展报头：认证头（Authentication Header，AH）和封装安全负荷（Encapsulating Security Payload，EPS）。

认证头（AH）指一段消息认证代码（Message Authentication Code，MAC），在发送 IP 包之前，它已经被事先计算好。发送方用一个加密密钥算出 AH，接收方用同一或另一密钥对之进行验证。如果收发双方使用的是单钥体制，那它们就使用同一密钥；如果收发双方使用的是公钥体制，那它们就使用不同的密钥。在后一种情形下，AH 体制能额外地提供不可否认的服务。IP AH 可以提供认证和完整性控制的能力。

封装安全负荷（ESP）封装整个 IP 报文或上层协议（如 TCP、UDP、ICMP）数据并进行加密，然后给已加密的报文加上一个新的明文 IP 报头。这个明文报头用来对已加密的 IP 包在 Internet 上作路由选择。因而 ESP 提供了良好的保密能力。当认证和保密两者都需要时，AH 与 ESP 相结合，就可以获得所需的安全性。一般的做法是把 ESP 放在 AH 里，这允许接收者在解密之前对消息进行认证检查或者并行地执行认证和检查。

IP 安全性的主要优点是它的透明性，也就是说，安全服务的提供不需要应用程序，也不需要其他通信层次和网络部件做任何改动。它的最主要缺点是网络层一般对属于不同进程的包不作区别。对所有去往同一地址的包，它将按照同样的加密密钥和访问控制策略来处理，这会使性能下降。针对面向主机密钥分配的问题，RFC 1825 推荐使用面向用户的密钥分配，其中，不同的连接会得到不同的加密密钥。但是，面向用户的密钥分配需要对相对应的操作系统内核作比较大的改动。

网络层非常适合提供基于主机的安全服务。相应的安全协议可以用来在 Internet 上建立安全的 IP 通道和虚拟专网。例如，利用它对 IP 包的加密和解密功能，可以强化防火墙系统的防卫能力。

3. *传输层安全*

由于 TCP/IP 协议本身非常简单，没有加密、身份认证等安全特性，因此要向上层应用提供安全通信的机制就必须在 TCP 之上建立一个安全通信层次。传输层网关在两个通信结点之间代为传递 TCP 连接并进行控制，这个层次一般称作传输层安全。最常见的传输层安全技术有 SSL、SOCKS 和安全 RPC 等。

在 Internet 应用编程中，通常使用广义的进程间通信（IPC）机制来与不同层次的安全协议打交道。比较流行的两个 IPC 编程接口是 BSD Sockets 和传输层接口（TLI）。

在 Internet 中提供安全服务的首要想法是在它的 IPC 界面中加入安全支持，如 BSD Sockets 接口等，具体做法包括双向实体的认证、数据加密密钥的交换等。Netscape 公司遵循这个思路，

制定了建立在可靠的传输服务（如 TCP/IP 提供）基础上的安全套接层（SSL）协议。SSL 分为两层，如图 2-6 所示，上面是 SSL 握手层，双方通过协商约定协议版本、加密算法，进行身份验证、生成共享密钥等；下面是 SSL 记录层，它把上层的数据经分段、压缩后加密，由传输层传送出去。SSL 采用公钥方式进行身份认证，但是大量数据传输仍使用对称密钥方式。通过双方协商，SSL 可以支持多种身份认证、加密和检验算法（关于 SSL 更多的内容请参阅 9.5 节）。

图 2-6　SSL 结构图

网络层安全机制的透明性优点对于传输层来说是做不到的，这是传输层安全机制的主要缺点。原则上，任何 TCP/IP 应用，只要应用传输层安全协议（如 SSL），就必定要进行若干修改以增加相应的功能，并使用稍微不同的 IPC 界面。同时，公钥体系存在的不方便性 SSL 也同样存在，例如用户很难记住自己的公钥和私钥，必须依靠某些物理设备（如 IC 卡或者磁盘）来存储，这样对用户终端有一定要求。再有就是服务器方和客户方必须依赖 CA 来签发证书，双方都必须将 CA 的公钥存放在本地。为了保持在 Internet 上的通用性，目前一般的 SSL 协议只要求服务器方向客户方出示证书以证明自己的身份，而不要求用户方同样出示证书，在建立起 SSL 信道后再加密传输用户的口令实现客户方的身份验证。

同网络层安全机制相比，传输层安全机制的主要优点是它提供基于进程对进程（而不是主机对主机）的安全服务和加密传输信道，利用公钥体系进行身份认证，安全强度高，支持用户选择的加密算法。这一成就如果再加上应用级的安全服务，就可以提供更加安全可靠的安全性能。

4. *应用层安全*

网络层的安全协议能够为网络连接建立安全的通信信道，传输层安全协议允许为进程之间的数据通道增加安全性，但它们都无法根据所传送的不同内容的安全要求予以区别对待。如果确实想区分具体文件的不同的安全性要求，就必须在应用层采用安全机制。本质上，这意味着真正的数据通道还是建立在主机（或进程）之间，但却不可能区分在同一通道上传输的具体文件的安全性要求。例如，如果在一个主机与另一个主机之间建立起一条安全的 IP 通道，那么两个进程间传输的所有报文都要自动被加密。提供应用层的安全服务，实际上是最灵活的处理单个文件安全性的手段。例如，一个电子邮件系统可能需要对要发出的信件的个别段落实施数据签名。较低层的协议提供的安全功能一般不会知道要发出的信件的段落结构，从而不可能知道该对哪一部分进行签名。只有应用层是唯一能够提供这种安全服务的层次。

一般来说，在应用层提供安全服务有下面几种可能的做法。首先是对每个应用（及应用协议）分别进行修改和扩展，加入新的安全功能。一些重要的 TCP/IP 应用已经这样做了。例如，在 RFC 1421～RFC 1424 中，IETF 规定了私用强化邮件（PEM）来为基于 SMTP 的电子

邮件系统提供安全服务。

S-HTTP 是 Web 上使用的超文本传输协议（HTTP）的安全增强版本，提供了文本级的安全机制，因此每个文件都可以设置成保密/签字状态。用作加密及签名的算法可以由参与通信的收发双方协商。S-HTTP 提供了对多种单向散列（Hash）函数的支持，如 MD2、MD5 及 SHA；对多种私钥体制的支持，如 DES、三重 DES、RC2、RC4 以及 CDMF；对数字签名体制的支持，如 RSA 和 DSS。

S-HTTP 和 SSL 从不同角度提供 Web 的安全性，S-HTTP 对单个文件做“保密/签字”，而 SSL 则把参与通信的相应进程之间的数据通道按“保密”和“已鉴别”进行监管。

除了电子邮件系统外，另一个重要的应用是电子商务，尤其是信用卡交易。为使 Internet 上的信用卡交易更安全，MasterCard 公司与 IBM、Netscape、GTE 和 CyberCash 等公司制定了安全电子付费协议（SEPP），Visa 国际公司与微软等公司制定了安全交易技术（STT）协议。同时，MasterCard、Visa 国际公司和微软公司已经同意联手推出 Internet 上的安全信用卡交易服务。他们发布了相应的安全电子交易（SET）协议，其中规定信用卡持有人用其信用卡通过 Internet 进行付费的方法。这套机制的后台有一个证书颁发的基础设施，提供对 X.509 证书的支持。SET 标准在 1997 年 5 月发布了第一版，它提供数据保密、数据完整性、对于持卡人和商户的身份认证以及其他安全系统的互操作性。

目前网络应用的模式正在从传统的客户机/服务器转向 BWD（Browser-Web-Database，浏览器－Web－数据库）方式，以浏览器作为通用客户端软件。由于 BWD 模式采用浏览器作为通用的客户方，原先的客户端软件工作很大部分变成了网页界面设计，各种数据库系统也提供了 Web 接口，可以在 CGI/Java 等网页创作工具中采用标准化的方式直接访问数据库，因此整个系统无论开发还是维护的工作量都大大减轻，特别是能够提供对内部网络应用和 Internet 统一的访问界面，使用十分方便。因此采用 BWD 模式改造现有的网络应用正在结合 Internet/Intranet 的建设迅速推行。在原先的客户机/服务器模式中，需要设计和实现各种专用的客户端软件以及客户端与服务器之间的安全措施，层次繁多复杂并且不具有通用性。转向 BWD 模式后，重点将放在浏览器到 Web 服务器之间以及 Web 服务器与数据库之间的安全上，因此应用层安全，特别是 WWW 的安全将成为至关重要的环节。

2.4 常用网络协议和服务

在本节中，只是简单介绍一下常用协议的基本原理及格式，以帮助读者对后续章节知识的理解。

2.4.1 常用网络协议

1. IP 协议

在网络协议中，IP 是面向非连接的协议，所以它是不可靠的数据报协议。IP 协议主要负责在主机之间寻址和选择路由。

IP 数据报的结构为：IP 头+数据，IP 头有一个 20 字节的固定长度部分和一个可选任意长度部分，格式如下：

<table>
<tr><td colspan="4">0　　4　　8　　16　　19　　31</td></tr>
<tr><td>版本</td><td>头长度</td><td>服务类型</td><td>封包总长度</td></tr>
<tr><td colspan="3">封包标识</td><td>标志　　分段偏移量</td></tr>
<tr><td colspan="2">生命期</td><td>协议</td><td>校验和</td></tr>
<tr><td colspan="4">源 IP 地址</td></tr>
<tr><td colspan="4">目的 IP 地址</td></tr>
<tr><td colspan="4">可选项（变长，可以是 0 或更多个字）
……</td></tr>
</table>

各个字段的含义解释如下：

- 版本：4bit，指明 IP 协议的版本号。每台机器上的版本可能不同，引入该字段可以在不同版本间传输数据。
- 头长度：4bit，因为 IP 头的长度是不固定的，所以用该字段指明头的长度。以 4 字节（即 32bit）为一个单位，该字段最小值为 5，表示没有可选部分，只有 20 字节（即 4 字节×5）的固定长度部分；该字段最大值为 15（即 4bit 全为 1），表示头部最大长度为 60 字节（4 字节×15），其中固定长度部分 20 字节，可选部分 40 字节。
- 服务类型：8bit，IP 优先级字段，主机可以通过该字段告诉网络它需要什么样的服务，是强调速度，还是强调准确性。
- 封包总长度：16bit，数据包中所有信息的长度，包括头部和数据。单位为字节，最大 65535（即 2^{16}–1）字节。
- 封包标识：16bit，用于让目的主机判断新来的分段属于哪个分组，所有属于同一分组的的分段包含同样的标识值。
- 标志：占 3bit，第 1bit 未用，只用到后面的两个 bit。

  DF（Don’t Fragment）用来标志是否允许路由器将数据报分段。

  　　DF=1，不允许分段。

  　　DF=0，允许分段。

  MF（More Fragment）用来标志是否所有的分组都已到达。

  　　MF=1，后面还有分段的数据包。

  　　MF=0，分段数据包的最后一个。
- 分段偏移量：13bit，用于说明分段在当前数据报中的相对位置。13bit 表示每个数据报最长可以有 8192（即 2^{13}）个分段，基本分段单位为 8 字节，所以最大的数据报长度为 65536（即 8192×8）字节，比前面提到的总长度字段的最大值 65535 还大 1。现在的数据报都没达到这么长，65535 的上限还是可以忍受的。
- 生命期：8bit，指分组的生命期，最长生命周期为 255，默认的单位是秒。当分组经过每个结点时，该字段的值递减，当值减到零时，该分组就要丢弃。
- 协议：8bit，该字段值表明 IP 数据包携带的是何种协议报文。

  1：ICMP

  6：TCP

  17：UDP

89：OSPF

- 检验和：16bit，指对 IP 协议头的校验和。
- 源 IP 地址：32bit，IP 报文的源地址。
- 目的 IP 地址：32bit，IP 报文的目的地址。
- 可选项：是为了允许后续版本的协议中引入新的信息、方便用户尝试新的想法以及避免让很少使用的信息占用头部位而提供的冗余字段。可选项是变长的。目前已定义了 5 个可选项，分别是：安全性、严格的源路由选择、松的源路由选择、记录路由、时间标记，但并不是所有的路由器都支持全部的 5 个可选项。

图 2-7 所示是对一个 IP 报头的解析：

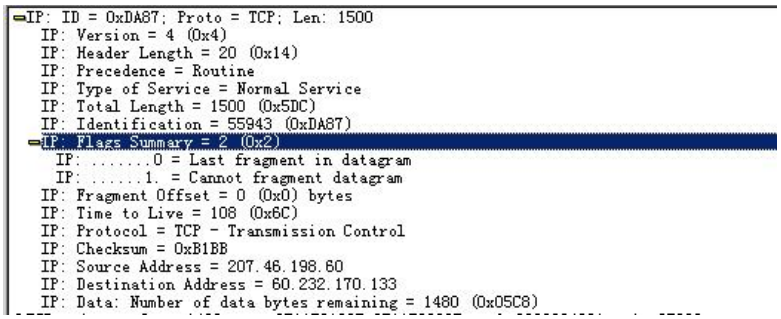

图 2-7　IP 报头解析

2. TCP 协议

TCP 是传输层协议，是专门设计用于在不可靠的因特网上提供可靠的、端到端的字节流通信的协议。

发送和接收方 TCP 实体以数据段（Segment）的形式交换数据。一个数据段包含一个固定的 20 字节的头（加上一个可选部分），后面跟着 0 字节或多字节的数据。TCP 协议的头的结构如下：

<table>
<tr><td colspan="4">0</td><td colspan="4">16</td><td colspan="8">31</td></tr>
<tr><td colspan="8">源端口</td><td colspan="8">目的端口</td></tr>
<tr><td colspan="16">顺序号</td></tr>
<tr><td colspan="16">确认号</td></tr>
<tr><td colspan="3">TCP
头长</td><td colspan="3"></td><td>U
R
G</td><td>A
C
K</td><td>P
S
H</td><td>R
S
T</td><td>S
Y
N</td><td>F
I
N</td><td colspan="8">窗口大小</td></tr>
<tr><td colspan="8">校验和</td><td colspan="8">紧急指针</td></tr>
<tr><td colspan="16">可选项（0 或更多的 32 位字）</td></tr>
</table>

各个字段的含义如下：

- 源端口：长度为 16bit 的源端口字段的值为初始化通信的端口号。
- 目的端口：长度为 16bit 的目的端口字段的值为传输的目的端口号。
- 顺序号：发送方向接收方发送的封包的顺序号，长度为 32bit。TCP 连接上的每个字

节均有它自己的 32bit 的顺序号，顺序号经过一段时间（如一个小时或更长）后会出现重复。

- 确认号：是指发送方希望接收的下一个封包的顺序号，长度为 32bit。
- 头长度：表明 TCP 头包含多少个 32 位字，长度为 4bit。

接下来的 6bit 未用；接着的 6 个标志位长度各 1bit，含义如下：

- URG：是否使用紧急指针。
  1：使用
  0：不使用
- ACK：是请求状态还是应答状态。
  1：应答，则确认号有效
  0：请求，则确认号被忽略
- PSH：PSH=1，表示接收方请求的数据一到便送往应用程序而不必等到缓冲区满才传送。
- RST：用于复位由于主机崩溃或其他原因而出现错误的连接。常可用于拒绝非法的数据或非法的连接请求。
- SYN：用于建立连接。在连接请求中，SYN=1，ACK=0，表示连接请求；SYN=1，ACK=1，表示连接被接受。
- FIN：用于释放连接。它表明发送方已没有数据发送了。
- 窗口大小：实现流量控制的字段，表示接收方想收到的每个 TCP 数据段的大小。若该字段值为 0 则表示希望发送方暂停发送数据。长度为 16bit。
- 校验和：是指对整个数据包的校验和，长度 16bit。
- 紧急指针：当 URG 为 1 时才有效，是发送紧急数据的一种方式，长度 16bit。
- 可选项：用于提供一种增加额外设置的方法，这种设置在常规的 TCP 包中是不包括的。

一个具体的 TCP 包头结构如图 2-8 所示。

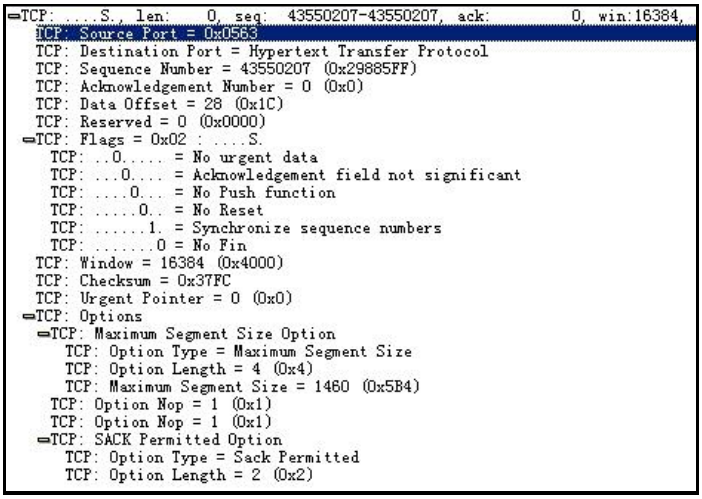

图 2-8　TCP 包头结构

3．UDP 协议

UDP 向应用程序提供了一种无连接的服务，通常用于每次传输量较小或有实时需要的程

序，在这种情况下，使用 UDP 开销较少，避免频繁建立和释放连接的麻烦。

一个 UDP 数据段包括一个 8 字节的头和数据部分。UDP 的协议头比较简单，如下所示：

0	16	31
源端口	目的端口	
封包长度	校验和	

UDP 头只包括 4 个字段，每个字段的长度为 16bit。

源端口、目的端口的作用与 TCP 中的相同。

封包长度：是指 UDP 头和数据的总长度。

校验和：与 TCP 头中的校验和一样，不仅对头数据进行检验，还对包的内容进行校验。

一个具体的 UDP 包头如图 2-9 所示。

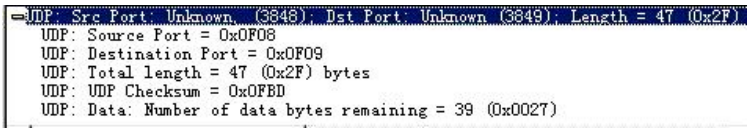

图 2-9　UDP 包头结构

4．ICMP 协议

ICMP 称为因特网控制消息协议。通过 ICMP 协议，主机和路由器可以报告错误并交换相关的状态信息。在下面几种情况中自动发送 ICMP 消息。

（1）IP 数据报无法访问目标。

（2）IP 路由器（网关）无法按当前的传输速率转发数据报。

（3）IP 路由器将发送主机重定向为使用更好的到达目标的路由。

一个具体的 ICMP 包头的解析如图 2-10 所示。

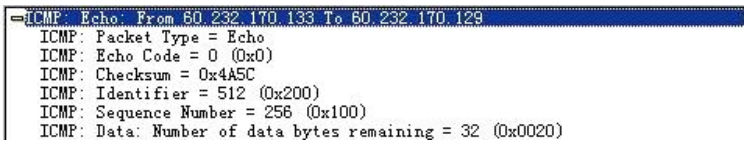

图 2-10　ICMP 包头的解析

2.4.2　常用网络服务

1．Telnet

Telnet 是一种因特网远程终端访问服务。它能够以字符方式模仿远程终端，登录远程服务器，访问服务器上的资源。Telnet 是最为简单的 Internet 工具之一，也是一种非常不安全的服务。Telnet 发送的信息都未加密，很容易被窃听。

Telnet 远程登录的使用主要有两种情况。第一种是用户在远程主机上有自己的账号（Account），即用户拥有注册的用户名和口令；第二种是许多 Internet 主机为用户提供某种形式的公共 Telnet 信息资源，这种资源对于每一个 Telnet 用户都是开放的。

要建立一个到远程主机的对话，只需在命令提示符下输入命令：telnet　远程主机名。在

Windows 系统中，用户可以用具有图形界面的 Telnet 客户端程序与远程主机建立 Telnet 连接。Telnet 通过端口 23 工作。

2．FTP

文件传输协议 FTP 的主要作用是让用户连接上一个远程计算机（这些计算机上运行着 FTP 服务器程序），并察看远程计算机有哪些文件，然后把文件从远程计算机上下载到本地计算机，或把本地计算机的文件上传到远程计算机去。FTP 服务的端口为 21。

与大多数 Internet 服务一样，FTP 也是一个客户机/服务器系统。用户通过一个支持 FTP 协议的客户端程序，连接到在远程主机上的 FTP 服务器程序，并向服务器程序发出命令，服务器程序执行用户所发出的命令，并将执行的结果返回到客户机。例如，用户发出一条命令，要求服务器向用户传送某一个文件的一份拷贝，服务器会响应这条命令，将指定文件送至用户的机器上。客户端程序代表用户接收到这个文件，将其存放在用户目录中。

使用 FTP 时必须先登录，在远程主机上获得相应的权限以后，方可上传或下载文件。也就是说，要想向哪一台计算机传送文件，就必须具有哪一台计算机的适当授权。换言之，除非有用户 ID 和口令，否则便无法传送文件。这种情况违背了 Internet 的开放性，Internet 上的 FTP 主机何止千万，不可能要求每个用户在每一台主机上都拥有账号。匿名 FTP 就是为解决这个问题而产生的。用户可以使用 anonymous 作为用户 ID，E-mail 地址作为口令连接到提供了匿名 FTP 服务的远程主机上，并从那里下载文件，而无需成为其注册用户。

为了安全起见，不要在匿名 FTP 服务器上存放机密文件。

3．E-mail

电子邮件是最流行和最基本的网络服务之一。随着邮件病毒和垃圾邮件的泛滥，电子邮件的安全问题越来越突出。

电子邮件的工作方式遵循客户/服务器模式。使用电子邮件服务的每位用户必须在一个邮件服务器上申请一个电子邮箱。邮件服务器管理着众多的客户邮箱。

电子邮件系统由客户端软件和邮件服务端软件所组成。通常，客户端程序（如 Outlook、Foxmail 等）为用户提供友好的交互式界面，方便用户编辑、阅读、处理信件。服务器端程序，负责将信件从消息源传送到目的邮箱。

目前 E-mail 服务使用的两个最主要的协议是简单邮件传输协议 SMTP 和邮局协议 POP。SMTP 默认占用 25 端口，用于发送邮件；POP 占用 110 端口，用来接收邮件。

SMTP 协议支持的功能比较简单，并且有安全方面的缺陷。经过它传递的所有电子邮件都是以明码传输的文本形式，任何人都可以在中途截取并复制这些邮件，甚至对邮件内容进行篡改。邮件在传输过程中可能丢失。别有用心的人也很容易以冒名顶替的方式伪造邮件。为了克服上述缺陷，后来出现了 ESMTP（Extended SMTP，扩展的 SMTP）协议。

POP 协议是一种允许用户从邮件服务器收发邮件的协议。它有 2 种版本，即 POP2 和 POP3，都具有简单的电子邮件存储转发功能。POP2 与 POP3 本质上类似，都属于离线式工作协议，但是由于使用了不同的协议端口，两者并不兼容。与 SMTP 协议相结合，POP3 是目前最常用的电子邮件服务协议。

4．WWW

WWW 服务是目前最常用的服务，使用 HTTP 协议，默认端口为 80。在 Windows 下一般使用 IIS 配置 Web 服务器。

用户通过浏览器可以方便地访问 Web 上众多的网页，网页包含文本、图片、语音、视频等各种文件。大多数 Web 服务器比较安全，但也经常有一些网站被黑，还有一些恶意网站在网页中添加恶意代码，修改用户机器的注册表。

5. DNS

域名服务（Domain Name Service，DNS）用于映射网络地址，即寻找 Internet 域名并将它转化为 IP 地址。域名是有意义的、容易记忆的 Internet 地址。域名和 IP 地址是分布式存放的。DNS 请求首先到达地理上比较近的 DNS 服务器，如果寻找不到此域名，主机会将请求向远方的 DNS 服务器发送。例如，将域名 www.sina.com.cn 通过 DNS 解析为 211.95.77.3。

2.5 Windows 常用的网络命令

网络命令是在命令行方式下的网络操作工具集，利用这些网络命令，可以了解网络状态、进行网络配置、测试网络连通性、使用网络服务等。本节将详细介绍一些常用网络命令的格式及用法。

2.5.1 ping 命令

ping 命令用来检测当前主机与目的主机之间的连通情况，它通过从当前主机向目的主机发送 ICMP 包，并接收应答信息来确定两台计算机之间的网络是否连通，并可显示 ICMP 包到达对方的时间。当网络运行中出现故障时，利用这个实用程序来预测故障和确定故障源是非常有效的。

ping 命令的格式如图 2-11 所示（在命令行状态下输入 ping 即可显示其格式及参数的说明）。

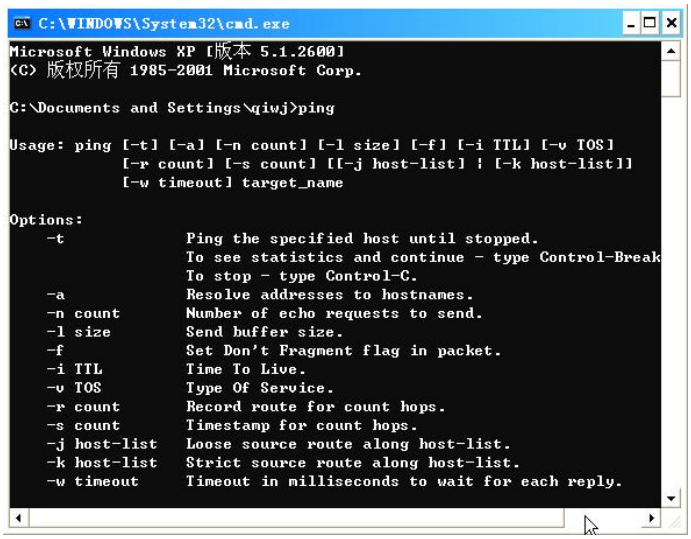

图 2-11 ping 命令参数

其中的常用参数说明如下：

- -t：使当前主机不断地向目的主机发送数据，直到按 Ctrl+C 键中断。
- -n count：指定要做多少次 ping，其中 count 为正整数值。

- -l size：发送的数据包的大小。
- -a：通过 IP 地址可以解析出对方的计算机。

一般使用的较多的参数为-t、-n、-l。

使用 ping 命令最简单的格式为：ping 主机域名，如图 2-12 所示。

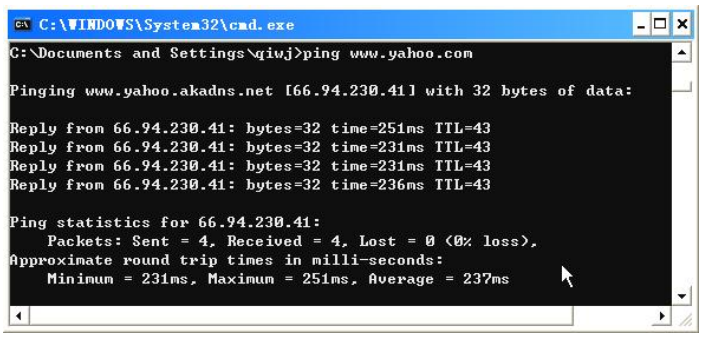

图 2-12　使用 ping 命令

如果 ping 某一网络地址，如 www.yahoo.com，出现 Reply from 66.94.230.41:bytes=32 time=251ms TTL=43，则表示本地主机与该网络地址之间的 IP 级连接是畅通的；如果出现 Request timed out，则表示此时发送的小数据包不能到达目的地，此时可能有以下两种情况，一种是网络不通，另一种是网络连通状况不佳。此时还可以使用带参数的 ping 命令来确定是哪一种情况，如使用 ping www.sina.com.cn -t -l 1500 不断地向目的主机发送数据，并且包大小设置为 1500 字节，此时如果都显示为 Request timed out，则表示网络之间确实不通，如果不是全部显示 Request times out 则表示此网站还是通的，只是响应时间长或通信状况不佳。

默认情况下，在显示 Request timed out 之前，ping 等待 1000 毫秒（1 秒）的时间让每个响应返回。如果通过 ping 探测的目标系统经由时间延迟较长的链路，如卫星链路，则响应可能会花更长的时间才能返回。此时可以使用-w（等待）选项指定更长时间的超时。

如果执行 ping 不成功，则可以预测故障出现在以下几个方面：网线不连通，网络适配器配置不正确，IP 地址不可用等；如果执行 ping 成功而网络仍无法使用，那么问题很可能出在网络系统的软件配置方面，ping 成功只能保证当前主机与目的主机间存在一条连通的物理路径。

另外，由于 ping 命令可以被攻击者用来收集主机信息和作为攻击的手段，因此，出于安全的考虑，许多主机的防火墙配置了“拒绝外部的 ICMP 包”这样的规则，这样的主机也是无法 ping 到的。例如，用命令 ping www.sohu.com，返回信息 Request timed out，看起来好像该主机不可达，而实际上，通过浏览器是可以正常访问该服务器的。

2.5.2　ipconfig 命令

ipconfig 用于在命令行方式下显示 TCP/IP 配置信息、刷新动态主机配置协议和域名系统设置。ipconfig 的命令及常用参数格式如下：

ipconfig [/? | /all | /release [adapter] | /renew [adapter]]

其中的参数的说明如下：

- /?：显示 ipconfig 的格式和参数的英文说明。

- /all：显示所有的配置信息。
- /release：为指定的适配器（或全部适配器）释放 IP 地址（只适用于 DHCP）。
- /renew：为指定的适配器（或全部适配器）更新 IP 地址（只适用于 DHCP）。

使用不带参数的 ipconfig 命令可以得到的信息如图 2-13 所示，包括 IP 地址、子网掩码、默认网关。

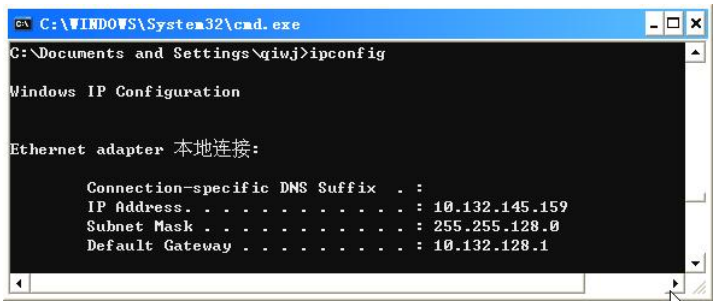

图 2-13　查看本机 IP 设置

使用 ipconfig /all 可以得到更多的信息，包括主机名、DNS 服务器、结点类型、网络适配器的物理地址、主机的 IP 地址（IP Address）、子网掩码（Subnet Mask）以及默认网关（Default Gateway）等。如图 2-14 所示为利用此命令显示所有的 IP 配置信息。

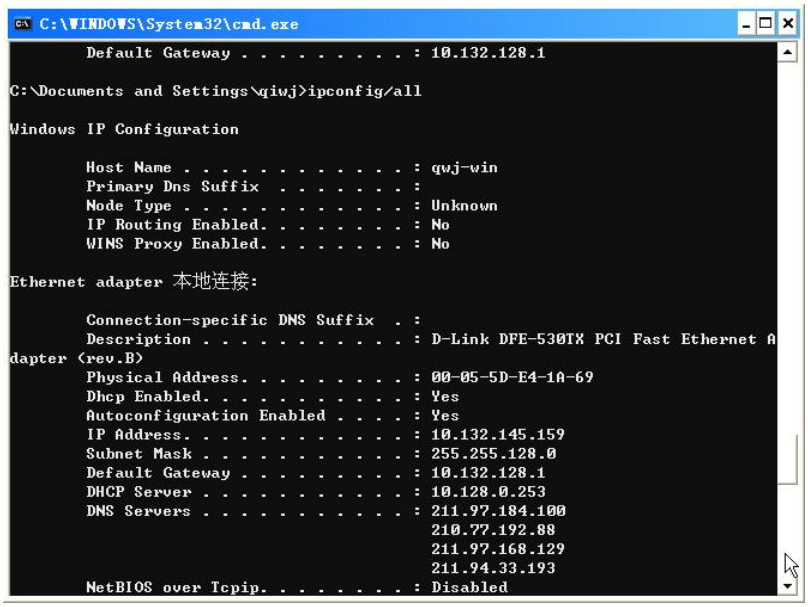

图 2-14　显示所有的 IP 配置信息

参数/renew 的功能是更新指定网络适配器（若未指定适配器，则指所有适配器）的 DHCP 配置。该参数仅在网卡自动获取 IP 的机器上可用。

2.5.3　netstat 命令

netstat 命令可用来显示当前的 TCP/IP 连接、Ethernet 统计信息、路由表等。Netstat 命令

的格式如下：

netstat [-a][-e][-n][-o][-s][-p proto][-r][interval]

netstat -a 命令将显示所有连接，而 netstat -r 命令用于显示路由表和活动连接。netstat -e 命令将显示 Ethernet 统计信息，而 netstat -s 用于显示每个协议的统计信息。如果使用 netstat -n，则不能将地址和端口号转换成名称。如图 2-15 所示是使用 netstat -a 命令时的输出示例。

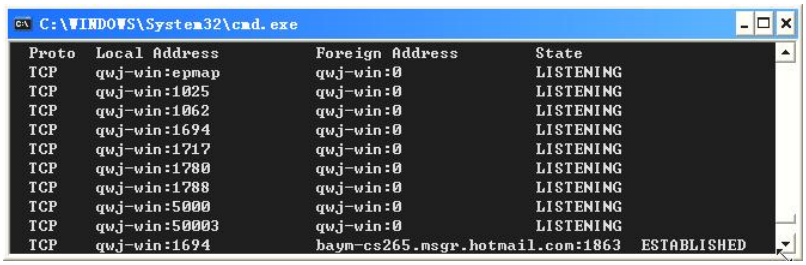

图 2-15　netstat -a 命令输出示例

从图 2-15 中可以看到，计算机打开许多端口，其中有些端口的状态为 LISTENING，表示该端口处于监听状态，没有和其他计算机建立连接；而有的端口状态为 ESTABLISHED，表明该端口正与某计算机进行通信。

2.5.4　tracert 命令

通过向目标发送不同 IP 生存时间（TTL）值的 ICMP 数据包，tracert 诊断程序确定到目标所采取的路由。路径上的每个路由器在转发数据包之前至少将数据包上的 TTL 递减 1，当数据包上的 TTL 减为 0 时，路由器将“ICMP 已超时”的消息发回源系统。tracert 先发送 TTL 为 1 的回应数据包，并在随后的每次发送过程将 TTL 递增 1，直到目标响应或 TTL 达到最大值，从而确定路由。通过检查中间路由器发回的“ICMP 已超时”的消息确定路由。tracert 命令按顺序打印出返回“ICMP 已超时”消息的路径中的近端路由器接口列表。

tracert 命令的用法示例如图 2-16 所示。

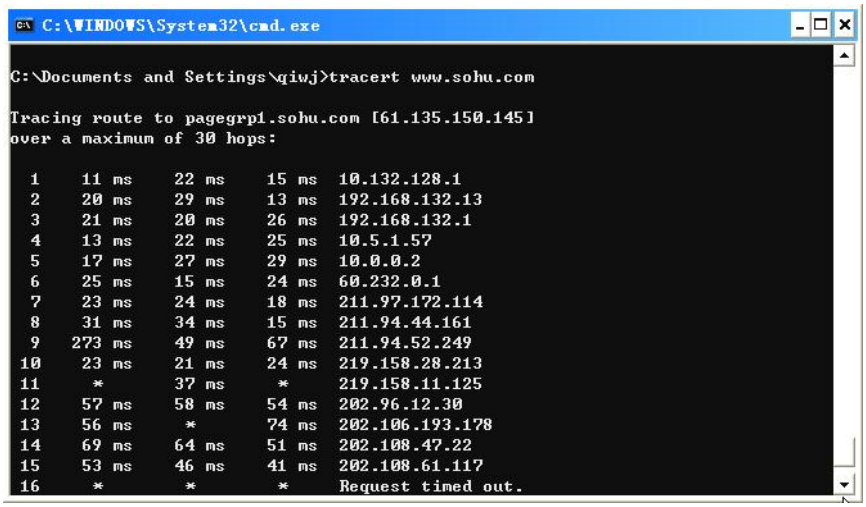

图 2-16　tracert 命令示例

2.5.5 net 命令

net 命令是网络命令中最重要的一个，必须透彻掌握它的每一个子命令的用法，因为它的功能非常强大，首先来看一下它都有哪些子命令，键入 net /?后回车，显示该命令的用法：

net [accounts | computer | config | continue | file | group | help | helpmsg | localgroup | name | pause | print | send | session | share | start | statistics | stop | time | use | user | view]

下面，重点介绍几个常用的子命令。

1．net start <service name>

使用 net start <service name>命令可以启动本地主机或远程主机上的服务。例如，用 net start telnet 就可以启动本地主机上的 telnet 服务，如图 2-17 所示。

图 2-17　启动 telnet 服务

2．net stop <service name>

使用 net stop <service name>命令来停止本地或远程主机上已开启的服务。如在 Windows 2000 的 cmd shell 下用命令 net stop server，就可以停止 Server 及与之关联的服务，如图 2-18 所示。

图 2-18　停止 Server 及与之关联的服务

3．net user

使用 net user 命令用来执行查看和账户有关的操作，包括新建账户、删除账户、查看特定账户、激活账户、账户禁用等。输入不带参数的 net user，可以查看所有用户，如图 2-19 所示。

（1）创建新账户。使用 net user peter 12345 /add 命令，新建一个用户名为 peter，密码为 12345 的账户，默认为 user 组成员，如图 2-20 所示。

（2）删除账户。使用 net user peter /del 命令，将用户名为 peter 的用户删除，如图 2-20 所示。

（3）禁用某个账户。设 123 为一个已存在的用户账户，则使用 net user 123 /active:no 命令，可将用户名为 123 的用户禁用，如图 2-21 所示。

图 2-19　显示所有用户

图 2-20　创建和删除用户

图 2-21　禁用和激活账户的操作

（4）激活某个账户。使用 net user 123 /active:yes，激活用户名为 123 的账户，如图 2-21 所示。

（5）查看用户信息。使用 net user 123 命令，查看用户名为 123 的用户的情况，包括用户账户的状态、密码有效期、所属组和上次登录时间等。

4. net localgroup

net localgroup 命令用于查看所有和用户组有关的信息和进行相关操作。输入不带参数的 net localgroup 即列出当前所有的用户组。可以用它来把某个账户提升为 Administrator 组账户，用法为：net localgroup groupname username /add。如把上面的用户账户 123 添加到管理员组中去，可以使用命令 net localgroup administrators 123 /add，123 就成为管理员组的成员，获得了管理员的权限。用命令 net localgroup administrators 123 /del，就可实现把 123 用户账户从管理员组删除。使用 net localgroup 命令如图 2-22 所示。

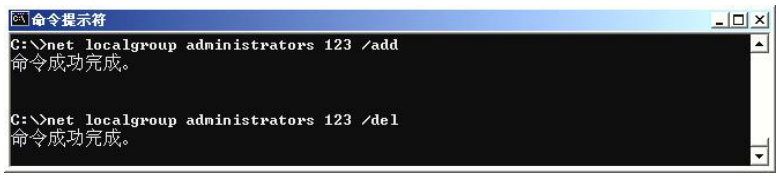

图 2-22　使用 net localgroup 命令

5. net view

net view 命令的格式如图 2-23 所示。

图 2-23　net view 命令的格式

使用 net view /domain:workgroupname 来查看名为 workgroupname 的域中的所有计算机。如执行 net view /domain:mshome，显示结果如图 2-24 所示。

图 2-24　查看域中的计算机

6. net share

不带参数的 net share 用于显示当前主机上的所有共享资源，如图 2-25 所示。

图 2-25　查看共享资源

关闭共享：net share 共享资源名 /del

如关闭 IPC$共享，使用的命令为：net share ipc$ /del

2.5.6　nbtstat 命令

nbtstat（TCP/IP 上的 NetBIOS 统计数据）实用程序用于提供关于 NetBIOS 的统计数据。运用 nbtstat，可以查看本地计算机或远程计算机上的 NetBIOS 名字列表。

常用选项：

- nbtstat -n：用于显示寄存在本地的名字和服务程序。
- nbtstat -c：用于显示 NetBIOS 名字高速缓存的内容。NetBIOS 名字高速缓存用于存放与本计算机最近进行通信的其他计算机的 NetBIOS 名字和 IP 地址对。
- nbtstat -r：用于清除和重新加载 NetBIOS 名字高速缓存。
- nbtstat -a IP：通过 IP 显示另一台计算机的物理地址和名字列表，显示的内容就像对方计算机自己运行 nbtstat -n 一样。执行结果如图 2-26 所示。

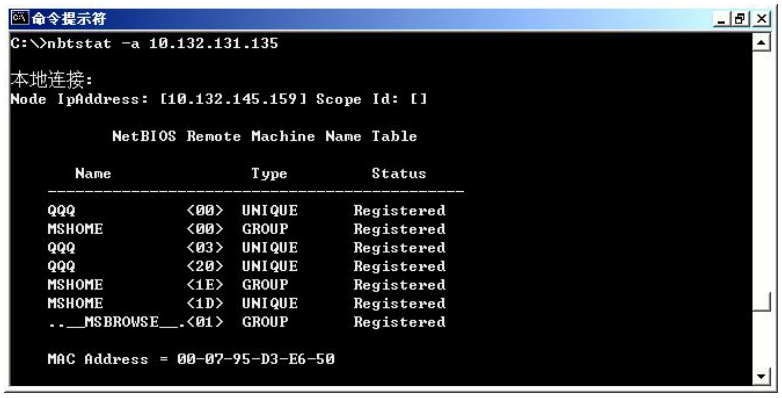

图 2-26　显示另一台计算机的物理地址和名字列表

2.5.7　ftp 命令

基本的 ftp 命令的使用方法为：首先在命令行键入 ftp 并回车，出现 ftp 的提示符，这时候可以键入 help 来查看帮助（任何 DOS 命令都可以使用此方法查看其帮助）。

首先是登录过程，直接在 ftp 的提示符下输入“open 主机 IP ftp 端口”回车即可，一般端口默认都是 21，可以不写，接着输入合法的用户名和密码进行登录，这里以匿名 ftp 为例介绍。在 user 后面，输入 anonymous，在 password 后面输入一个邮件地址作为口令，如图 2-27 所示。

图 2-27　使用 ftp 服务

接下来介绍具体命令的使用方法。

- dir：跟 DOS 命令一样，用于查看服务器的文件，直接键入 dir 并回车，就可以看到此 ftp 服务器上的文件，如图 2-28 所示。

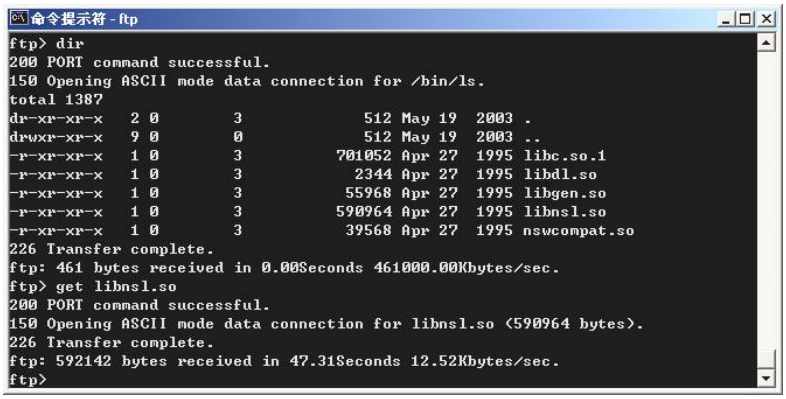

图 2-28　ftp 服务器上的文件列表

- cd：用于进入某个文件夹。
- get：用于下载文件到本地机器。
- put：用于上传文件到远程服务器。这就要看远程 ftp 服务器是否给了可写的权限。
- delete：用于删除远程 ftp 服务器上的文件。这也必须保证有可写的权限。
- disconnect：用于断开当前连接。
- bye：用于退出 ftp 服务。
- quit：同上。

2.5.8 telnet 命令

telnet 是功能强大的远程登录命令。它操作简单，如同使用自己的机器一样，使用如下方法建立 telnet 连接：

方法一：telnet 主机名（IP）。

方法二：首先输入 telnet，按回车键；然后在提示符下输入 open IP 并回车，显示登录界面，输入合法的用户名和密码，输入任何密码时都是不显示的。

当输入的用户名和密码都正确后就成功建立了 telnet 连接，这时用户就在远程主机上具有了相应的权限。

2.6 网络协议分析工具——Wireshark

网络分析程序（Network Packet Analyzer）可以帮助网络管理员捕获网络中传输的数据包和分析数据包信息。比较常用的网络分析工具有 Wireshark、Microsoft Network Monitor、Capsa Packet Sniffer、NetworkMiner 等。网络分析工具的用途广泛，网络管理员使用它来检测网络问题，网络安全工程师使用它来检查信息安全的相关问题，开发者可以使用它来为新的通信协议除错，普通使用者可以使用它来学习网络协议的相关知识。当然，也会有少数“居心叵测”的人用它来寻找一些敏感信息。

网络包分析工具 Wireshark，其前身是 Ethereal，是目前最流行的开源网络分析工具之一，它具有的主要功能和特性如下：

- 支持 UNIX 和 Windows 平台。
- 在网络接口实时捕捉包。
- 能显示包的详细协议信息。
- 可以打开/保存捕捉的包。
- 可以导入/导出其他捕捉程序支持的包数据格式。
- 可以通过多种方式过滤包。
- 可以通过种方式查找包。
- 通过过滤以多种色彩显示包。
- 创建多种统计分析。

下面简单介绍如何安装和使用网络分析工具 Wireshark。

2.6.1　Wireshark 的安装

1．下载

可以从 Wireshark 官方网站下载最新版本的软件，网址为 http://www.wireshark.org/download.html。Wireshark 通常在 4～8 周内发布一次新版本，目前的最新版本为 1.8.4。还可以订阅 Wireshark-announce 邮件列表获得 Wireshark 发布的消息。

2．安装环境

Wireshark 可以运行在 Windows 和 UNIX 的多个系统平台上，其中 Windows 操作系统包括 Windows 2000/XP/2003/Vista 等；UNIX/Linux 操作系统平台包括 Apple Mac OSX、Debian GNU/Linux、FreeBSD、Solaris、OpenPKG 等。Wireshark 的最低运行环境要求为：

- 任何 32 位 x86 或 64 位 AMD64（x86-64）处理器；
- 128MB 可用系统内存；
- 75MB 可用磁盘空间（如果想保存捕捉文件，需要更多空间）；
- 800×600（建议 1280×1024 或更高）分辨率，最少 65536（16bit）色。

支持的网卡包括：以太网、无线局域网网卡。由于 Wireshark 底层使用 Libpcap/Winpcap，因此与它们具有相同的局限性，对于 Wireshark 在不同平台上能捕获哪些接口上的数据，这里不再详细介绍，参见http://wiki.wireshark.org/CaptureSetup/NetworkMedia。

3．在 Windows 下安装 Wireshark

本节将探讨在 Windows 下安装 Wireshark 二进制包。

下载的 Wireshark 二进制安装包的名称类似 Wireshark-setup-x.y.z.exe，如 Wireshark-win32-1.8.4.exe。Wireshark 安装包已经包含 Winpcap，所以不需要单独下载安装它，只需要下载 Wireshark 安装包并执行它即可。在安装的过程中，如果不了解设置的作用，尽量保持默认的设置。安装中还可以选择想要的组件，如图 2-29 所示。

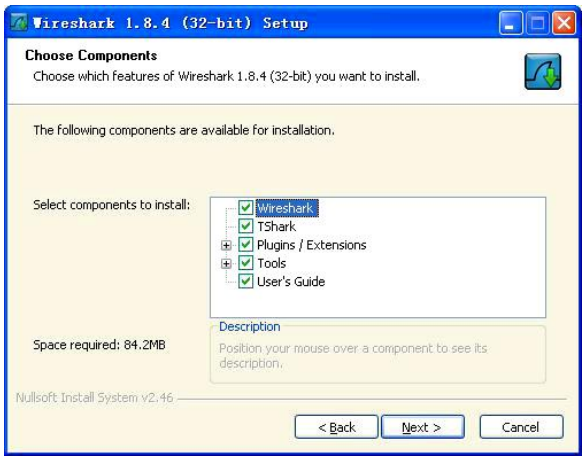

图 2-29　选择组件

这些组件包括：

（1）Wireshark：Wireshark GTK 是一个基于图形用户界面的协议分析器。

（2）TShark：一个基于命令行的网络分析工具。

（3）插件/扩展（Wireshark，TShark 分析引擎）：

- Dissector Plugins：分析插件，带有扩展分析的插件。
- Tree Statistics Plugins：树状统计插件，统计工具扩展。
- Mate - Meta Analysis and Tracing Engine（experimental）：可配置的显示过滤引擎，参考http://wiki.wireshark.org/Mate。
- SNMP MIBs：SNMP、MIBS 的详细分析。

（4）Tools/工具（处理捕捉文件的附加命令行工具）：

- Editcap：用来读一个捕获文件并把其中的部分或全部包写入另一个文件的程序。
- Text2Pcap：用来将 ASCII 形式的十六进制转存成 libpcap 格式的捕获文件。
- Mergecap：用于将多个捕获文件合并成一个单独的输出文件的程序。
- Capinfos：提供捕获文件的信息。
- Rawshark：是一个原始包过滤器。

（5）User's Guide（用户手册）：本地安装的用户手册。如果不安装用户手册，单击“帮助”菜单的大部分按钮的结果可能是访问 Internet。

2.6.2 Wireshark 主窗口

1. *启动* Wireshark

Wireshark 软件安装成功后，可以通过“开始”菜单启动 Wireshark 程序。在主界面或 Capture 菜单项中可以选择要捕获的网络接口并启动捕获过程，主窗口界面如图 2-30 所示。

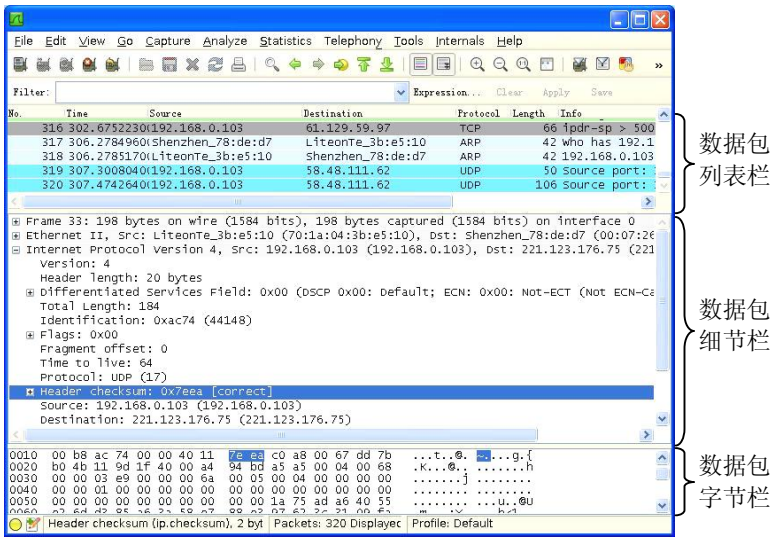

图 2-30 Wireshark 主窗口

2. *主窗口界面及功能*

和大多数图形界面程序一样，Wireshark 主窗口由如下部分组成：

- 菜单：用于开始各种功能。
- 主工具栏：提供快速访问菜单中经常用到的项目的功能。
- 过滤工具栏：提供处理当前显示内容所使用的过滤方法。

- 数据包列表栏：如图 2-30 中数据包列表栏中显示打开文件或当前捕获的每个包的摘要。单击其中的一条项目，该数据包的详细情况将显示在另外两栏（数据包细节栏及数据包字节栏）中。
- 数据包细节栏：显示在数据包列表栏中选中的数据包的更多详情。
- 数据包字节栏：显示在数据包列表栏中被选中的数据包的数据，并且当在数据包细节栏中选择某一字段的内容时，会在此栏中高亮显示对应的字节的内容。

其他菜单项的功能不再一一描述，仅介绍最常用的菜单项 Capture、Analyze，它们包括的子菜单项分别如图 2-31、图 2-32 所示。其中几个常用子菜单项的功能描述如表 2-3、表 2-4 所示。

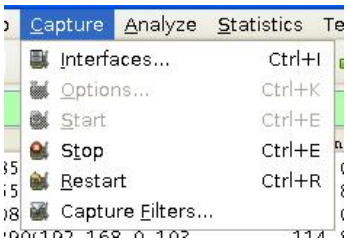

图 2-31　Capture 子菜单项

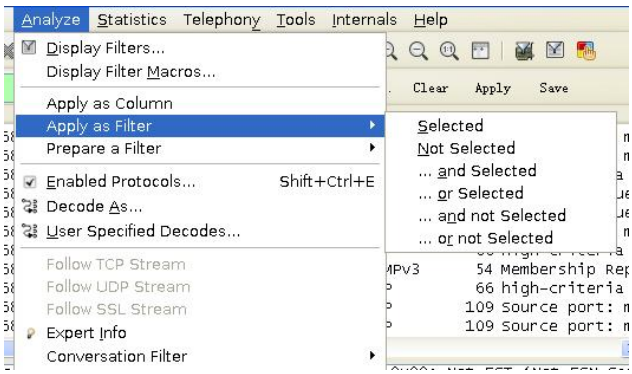

图 2-32　Analyze 子菜单项

表 2-3　Capture 子菜单项及功能说明

菜单项	说明
Interfaces...	在弹出的对话框中选择要进行捕捉的网络接口
Options...	打开设置捕捉选项的对话框
Start	立即开始捕捉，默认参照最后一次设置
Stop	停止正在进行的捕捉
Restart	正在进行捕捉时停止捕捉，并按同样的设置重新开始捕捉，仅在认为有必要时使用
Capture Filters...	打开对话框，编辑捕捉过滤设置，可以命名过滤器，保存为其他捕捉时使用

表 2-4　Analyze 主要子菜单项及功能说明

菜单项	说明
Display Filters...	打开过滤器对话框编辑过滤设置，可以命名过滤设置，保存为其他场合使用
Apply as Filter	更改当前过滤显示并立即应用。根据选择的项，当前显示字段会被替换成在 Detail 面板选择的协议字段
Prepare a Filter	更改当前过滤显示，但不会立即应用。同样根据当前选择项，过滤字符会被替换成在 Detail 面板选择的协议字段
Enabled Protocols...	是否允许协议分析

2.6.3 数据包捕获

1. 选择接口

在 Capture 菜单中选择子菜单项 Interfaces…，打开 Capture Interfaces（捕获接口）对话框，如图 2-33 所示。选择相应的端口，按 Start 按钮开始捕获数据包。若选择 Options 按钮，则可弹出 Capture Options（捕捉选项）对话框，如图 2-34 所示，可以对一些捕捉选项进行设置。

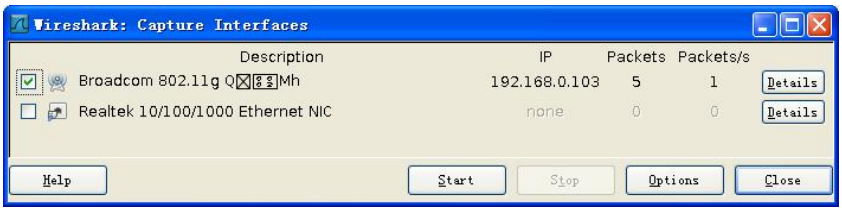

图 2-33 Capture Interfaces 对话框

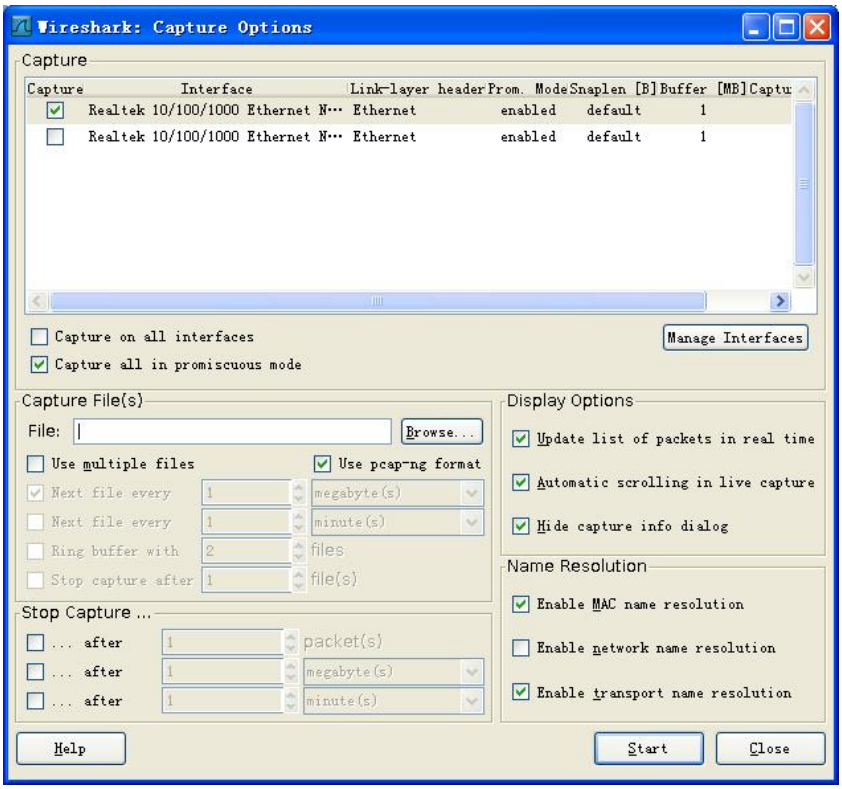

图 2-34 Capture Options 对话框

首先，在图 2-34 中双击 Capture 栏中的某个接口，弹出图 2-35 所示的 Edit Interface Settings（接口设置）对话框，显示以下内容，可对其中一些选项进行设置：

- Interface：该字段显示用于进行捕捉的接口。
- IP address：显示接口的 IP 地址。如果系统未指定 IP 地址，将会显示为 unknown。
- Link-layer header type：链路层包头类型，除非有些特殊应用，尽量保持默认选项。

- Buffer size：输入用于捕捉的缓冲区的大小。该选项用于设置写入数据到磁盘前保留在核心缓存中捕捉数据的大小，如果发现丢包，尝试增大该值。
- Capture packets in promiscuous mode：指定 Wireshark 捕捉包时，设置接口为混杂模式。如果未指定该选项，Wireshark 将只能捕捉进出计算机的数据包（不能捕捉整个局域网段的包）。
- Limit each packet to n bytes：指定捕捉过程中，每个包的最大字节数。如果禁止该选项，默认值为 65535，这个长度适用于大多数协议。
- Capture Filter：指定捕捉过滤器，默认情况下是空的。

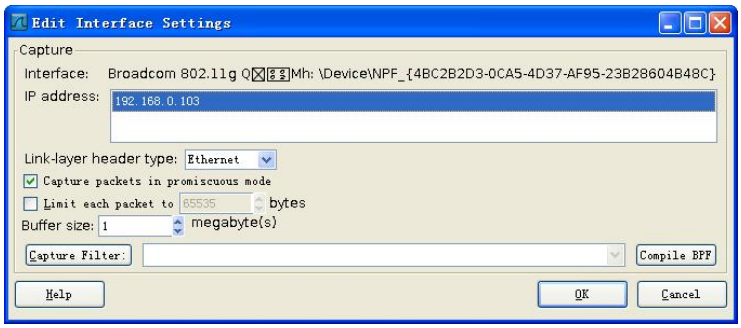

图 2-35　Edit Interface Settings 对话框

在图 2-34 中，还可以对 Capture File(s)、Stop Capture、Display Options、Name Resolution 等选项进行设置，如：

- File：指定将用于捕捉的文件名。该字段默认是空白。如果保持空白，捕捉数据将会存储在临时文件夹。可以单击文本框右侧的按钮打开浏览窗口，设置文件存储位置。
- Use multiple files：如果指定条件达到临界值，Wireshark 将会自动生成一个新文件，而不是使用单独文件。如 Next file every n megabyte(s)是指如果捕捉文件容量达到指定值，将会切换到新文件。

其他的关于 Stop Capture、Display Options、Name Resolustion 的各选项本文不再详细介绍，具体内容参考软件的使用手册。

2. 过滤器定义

Wireshark 基于 Winpcap 过滤器对捕获数据进行过滤。过滤器是一个包含过滤表达式的 ASCII 字符串，如果没有给定的过滤表达式，过滤引擎将会接收所有的数据包；否则，只有带入表达式之后其值为 true 的包才会被接收。过滤表达式可以包括关系运算和标准二进制操作，如“ether[0] &1!= 0”表示捕获所有的多播流量。更复杂的过滤表达式还可以综合运用逻辑运算符 and（&&）、or（||）和 not（!）。如使用过滤器“ip.dst == 122.97.252.78 && http”，将过滤目的 ip 地址 122.97.252.78 的 http 数据包，如图 2-36 所示。

3. 数据包捕获解析

图 2-36 所示是对捕获的报文进行解码的显示界面，目前大部分网络分析工具都采用这种三层的显示结构。要求解码分析人员对协议比较熟悉，这样才能看懂解析出来的报文，如图 2-37 和图 2-38 所示分别是对 IP 协议、TCP 协议的 HTTP 协议的解码分析。关于协议方面的细节，这里不再详细讨论，请参阅有关协议方面的资料来理解这些解码的含义。

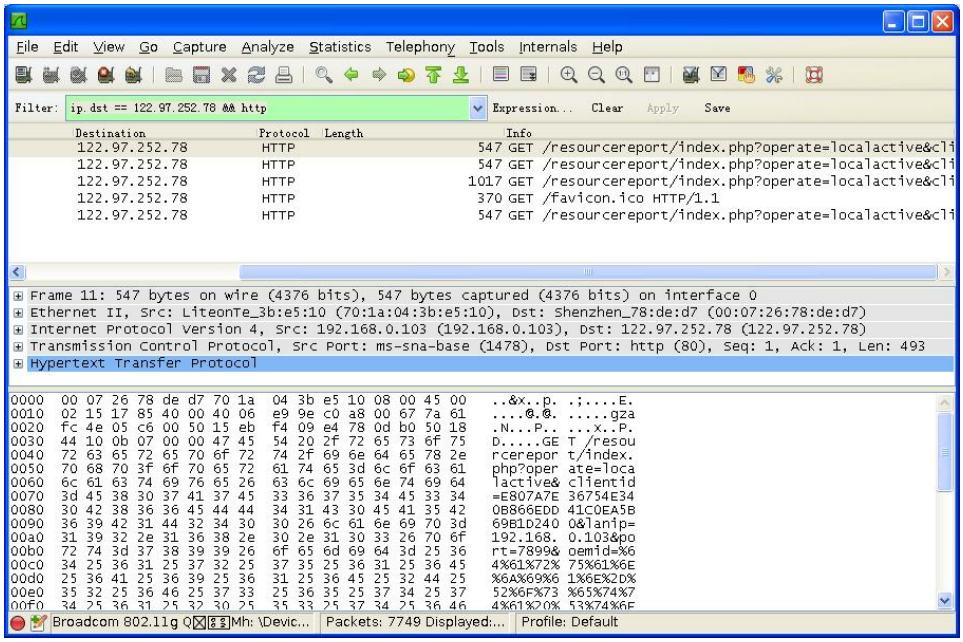

图 2-36　捕获符合条件的数据包

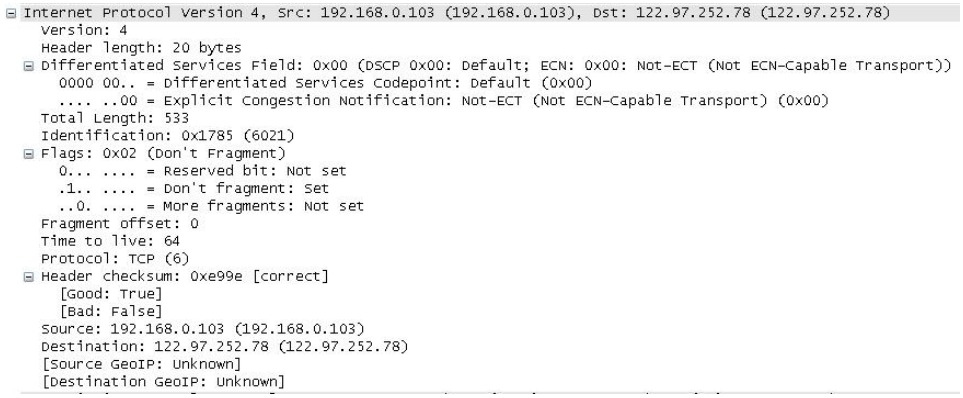

图 2-37　对 IP 协议的解码分析

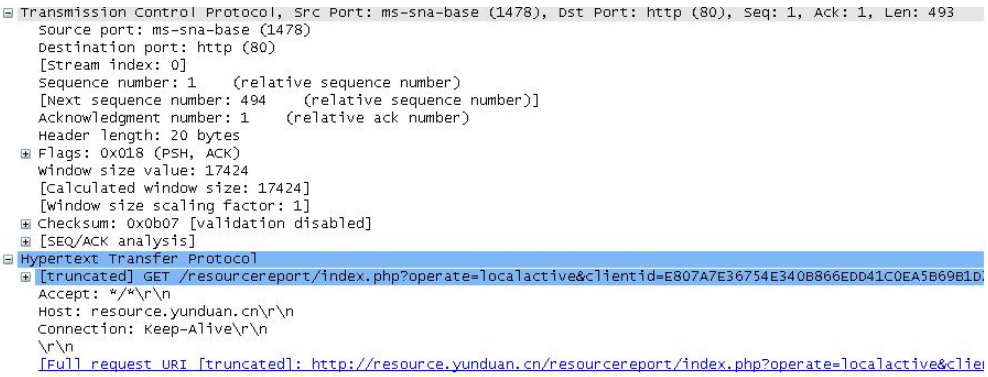

图 2-38　对 TCP 和 HTTP 协议的解码分析

一、思考题

1．说明 OSI 安全体系结构中，定义了哪些安全服务和安全机制。

2．说明 TCP/IP 的网络层安全和应用层安全是如何实现的。

3．分析 TCP 和 UDP 协议的异同。

4．分析 ping 命令可能造成的安全威胁。

5．根据 ICMP 协议的定义来分析 ICMP 攻击的机制。

6．net 命令可以实现哪些网络管理功能？

7．tracert 命令实现路径跟踪的原理是什么？

8．通过 netstat 命令可以了解什么信息？

9．使用命令 ping www.sohu.com -l 6000，返回的信息为 Reply from 199.188.36.12: bytes=6000 time=127ms TTL=128，请解释命令及返回信息中每个方框中部分表示的含义。

二、实践题

1．如何使用 net 来添加用户？如何把添加的用户提升到管理员组？如何显示共享和关闭共享？

2．使用 Sniffer 进行数据包的捕获及数据包结构的分析，理解协议对数据的封装。

3．如何登录 ftp 服务器并下载文件？

4．利用 tracert 命令跟踪到达www.amazon.com的路由。

5．查阅相关资料，研究 IPC$共享的用途及可能存在的威胁，并讨论如何降低 IPC$共享带来的风险。