第1章　责任衡量和访问控制

本章涵盖的CISSP考试目标包括：

- √ 责任衡量
- √ 访问控制技术
- √ 访问控制管理
- √ 访问控制方法
- √ 标识和验证技术
- √ 访问控制方法论和实现

访问控制系统及其方法论属于CISSP认证考试的通用知识体系领域，它所要处理的课题和问题，涉及到准予或限制用户对资源进行访问时的监控、标识和授权。通常，访问控制是指所有硬件、软件、组织管理策略或程序，它们对访问进行授权或限制、监控和记录访问的企图、标识用户的访问企图，并且确定访问是否经过了授权。

这个领域将在本章和第2章“攻击和监控”中进行讨论。读者一定要对这两章的内容进行学习和研究，这样才能保证完全掌握CISSP认证考试中这个领域的本质内容。

访问控制综述

控制对资源的访问是安全性的中心话题。访问控制所涉及的内容要比简单地控制哪些用户可以访问哪些文件或服务多得多。访问控制是对主体和客体如何结合的管理。从客体到主体的信息传输叫做访问。主体（subjects）是活动的实体，它通过访问操作，寻找有关被动实体的信息，或者从被动实体中寻找数据。这个被动实体也可称为客体（objects）。主体可以是用户、程序、进程、文件、计算机和数据库等等。客体可以是文件、数据库、计算机、程序、进程、文件、打印机和存储介质等等。主体是接收有关客体的信息或来自客体的数据的实体。主体还是改变客体信息的实体，或改变存储在客体中数据的实体。客体始终是提供或控制信息或数据的实体。在主体和客体之间进行通信，执行一项任务时，两个实体的角色可以交换，如程序和数据库、处理过程和文件。

访问控制对于保护客体（其信息和数据）的保密性、完整性和可用性是很有必要的。术语访问控制可以用来描述广泛的控制，包括从强制用户提供有效的用户名和密码进行登录操作，到防止用户获得对资源超出其访问权限的操作。

CIA三元组

保密性（confidentiality）、完整性（integrity）和可用性（availability）的基本安全法则常常被称为CIA三元组（CIA Triad）。所有的安全性控制必须符合这个法则。这三个安全性法则常作为贯穿CISSP CBK的标准思路。每个领域都以自身独特的方式满足这些法则，

因此作为常规的术语并且在各自特定的领域中理解它们是很重要的：

- 对于那些没有向未授权主体公开的客体，保密性是最本质的法则。
- 对于那些保持其准确性，并且只由授权主体进行相关修改的客体，完整性是最本质的法则。
- 对于那些经授权的主体及时地获得准许，并对客体进行连续访问来说，可用性是最本质的法则。

不同的安全机制以不同的方式诠释这三个法则，并且对它们提供不同程度的支持或应用。客体必须进行适当地分类和优先级的区分，这样，正确的安全访问控制才可以进行。与CIA三元组相关的这些内容和其他一些问题都将在本书中得到讨论。

访问控制可以分为下列三类：

预防性的访问控制　进行预防性的访问控制是为了阻止不必要的或未授权的操作出现。举例来说，预防性访问控制包括防护、安全策略、安全感知训练和反病毒软件。

探查性访问控制　进行探查性访问控制是为了发现不必要的或未授权的操作出现。举例来说，探查性访问控制包括安全性防护、监督用户、时间调查和攻击监测系统。

纠正性访问控制　进行纠正性访问控制是为了在不必要的或未授权的操作发生后将系统恢复到正常的状态。举例来说，纠正性访问控制包括报警、圈套和安全性策略。

访问控制的实现可以按照行政、逻辑／技术或物理进行分类：

行政性访问控制　　行政性访问控制是依照机构的安全性策略定义的策略和执行过程，实现并加强全局的访问控制。举例来说，行政性访问控制包括策略、执行过程、雇用准则、背景调查、数据分类、安全培训、空缺记录、回顾、工作监督、人员管理和测试。

逻辑/技术性访问控制　逻辑性访问控制和技术性访问控制作为硬件或软件机制，可以用来管理对资源和系统的访问，并且提供对这些资源和系统的保护。举例来说，逻辑性或技术性访问控制包括加密、智能卡、密码、生物测定学、受限接口、访问控制列表、协议、防火墙、路由器、入侵检测系统和切割层。

物理性访问控制　物理性访问控制作为物理屏障，可以用来保护对系统的直接访问。举例来说，物理性访问控制包括防护装置、防护、移动探测器、闭锁的门、密封窗、灯光、线缆保护、笔记本电脑锁、刷卡、狗、摄像机、圈套和警报器。

访问控制控制着主体对客体的访问。这个过程的首要步骤是对客体进行标识。实际上，在对客体的实际访问之前还有几个步骤需要执行：标识、验证、授权和责任衡量。

标识（Identification）是一个主体表示其身份并进行责任衡量的过程。提供用户名、登录ID、个人身份号码（personal identification number，PIN）或智能卡的用户向我们描述了标识的过程。一旦主体完成了标识，那么这个身份标识就要对主体所进行的进一步操作负责了。信息技术（Information Technology，IT）系统通过身份标识来跟踪实际的操作，而不是通过主体自身进行跟踪。一台计算机并不认识我们，但是它却知道操作者的用户账号与其他人的用户账号是不同的。

验证（Authentication）是指对所宣称的身份标识的有效性进行校验和测试的过程。验证需要主体提供额外的信息，这些信息必须与身份标识所指示的内容完全相符。最常见的验证

形式是密码。然而，至少有三种其他形式的信息可以用于验证：

类型1 类型1验证因素是指一些大家知道的内容，诸如密码、个人身份号码（PIN）、密码锁、母亲的娘家姓和喜欢的颜色等等。

类型2 类型2验证因素是指操作者所具有的内容，诸如智能卡、凭证设备和内存卡等等。它还可以包括操作者的物理位置，被称为“你在哪里”的因素。

类型3 类型3验证因素是指操作者所具有的一些如指纹、语音波纹、视网膜样本、虹膜样本、面容形状、掌纹和手型等等的内容。

除了这三个常见的因素之外，至少还有其他两个因素需要注意。第一个因素是“你的动作”，例如签名、键入密码短语（键盘动力学）或者如何说这个短语。“你的动作”常常包含在“你的身份”类型中。刚刚提到的“你的身份”就是另一个因素。例如特定的一台计算机终端，从一个有呼叫方ID标识的特定号码拨号，或者从一个有IP地址标识的特定国家地区拨号。你所在的地方通常包含在“你所拥有的”类别之中。当这两个因素都需要提供验证信息时，双因素验证就出现了。例如在商店中用支票付账时，你通常要提供你的驾驶执照（你所拥有的）和你的电话号码（你所知道的）。

一旦提供身份标识和验证因素的登录证书提交给系统，那么系统就会将它们与系统中的身份标识数据库进行核对。如果找到了身份标识，并且提供的验证因素也正确，那么主体通过验证。

然而，每当主体通过了验证，其访问还必须经过授权。授权（authorization）的过程确保了被请求的操作或目标访问可能获得了向经验证的身份标识（我们将它看做是这里的主体）赋予的权利和特权。大多数情况下，系统会估计一个访问控制矩阵，它会对主体、客体（目标）和预计的操作进行比较（我们在本章稍后的内容中对访问控制矩阵进行更详细的讨论）。如果指定的操作可以进行，那么主体就已经获得了授权。如果指定的操作没有被准许进行，那么主体就没有获得授权。

需要记住，主体只是经过标识和验证，并不同时意味着已经通过了授权。对于主体来说，登录到网络中（例如被标识和验证）却被阻止访问文件或从打印机打印是可能的（例如，没有经过授权来执行这项操作）。大多数网络用户只是被授权在指定的一组资源上执行一些有限的操作。身份标识和验证是访问控制的所有方面或者不是任何一个方面。对于环境中的每个单独的主体和客体，在全部或什么也不是之间，授权有着很大的区别。用户可能能够读取文件，却不能删除它。用户可能能够打印文档，但是不能修改打印队列。用户可能能够登录到系统中，但是无法访问任何资源。

理解身份标识、验证和授权之间的区别是很重要的。虽然它们很相似，并且是所有安全机制的本质内容，但它们是不同的，并且必须搞清楚，不能混淆。这些功能在本章稍后的内容中将进行非常详细的分析。

一个机构的安全策略只能在支持责任衡量的情况下，才可以正确地执行。换句话说，只有在主体对于它们的操作负有责任时，安全性才可以保持。有效的责任衡量依赖于检验主体的身份和跟踪其操作的能力。因此，责任衡量建立在身份标识、验证、授权、访问控制和审核的概念上。

身份标识和验证技术

身份标识是一个十分简明的概念。主体必须对系统提供身份标识，启动验证、授权和责任衡量过程。提供身份标识可以是输入用户名、刷卡、出示凭证设备、说一段话或将你的脸、手或指纹靠在照相机或扫描设备前。没有身份标识，系统是无法将验证内容与主体关联在一起的。主体的身份标识常被看做是典型的公共信息。

身份验证通过对比数据库中有效身份（比如用户的账户）的一个或多个因素对主体的身份进行证实。用来证实身份的验证因素常被认为是私有的信息。系统和主体维护身份验证的保密性的能力，会直接影响到系统的安全性级别。

标识和身份验证总是被放在一起成为单一的双步过程。第一步是提供标识符，第二步是提供身份验证因素。缺少这两步，主体是不能获得系统的访问能力的，只提供其中的任何一项都是没有用的。

一个主体可以提供几种类型的验证信息（例如所知道的、所拥有的等等）。每种验证技术或因素都具有其独特的优缺点。因此根据即将运行的环境对每种算法进行评估对于确定其生存能力是很重要的。

密码

身份验证技术最常见到的是密码，但是这些密码也被认为是最弱的保护形式。有很多原因使得密码成为不安全的算法，这其中包括：

- 用户常常选择他们很容易记忆的密码，因此这些密码很容易猜到或被解读。
- 随机生成的密码很难记忆，因此很多用户都会将它们写下来。
- 密码很容易共享、记录和忘记。
- 可以通过很多手段盗窃密码，包括观察、录音和回放，甚至对安全数据库的偷盗。
- 密码的传递常常会以明码的形式或易破解的协议进行。
- 密码库常常存储在可访问到的在线公共场所。
- 短密码可以通过暴力攻击很快攻破。

如果密码的选择很巧妙，并且管理得当，那么它们还是很有效的。密码有两种类型：静态的和动态的。静态的密码总是不变的。动态的密码在间隔一段时间或使用后会发生改变。一次性密码或专用密码是不同的动态密码，每次使用时，它们都会发生改变。在维护安全性方面的重要性一再增长的时候，更改密码的需要也变得更加频繁了。密码保持静态不变的时间越长，相同密码的使用越频繁，那么密码被泄漏或解读的可能性越大。

比较有效一点的密码是口令短语。口令短语通常是一串字符，其长度要比密码长。一旦输入口令短语，系统将为验证过程的使用将它转换为虚拟密码。口令短语常常是修改过的母语语句，这样可以简化记忆。举例来说，“She ell C shells ByE the c-shor”。

另一个有意思的密码算法是感知密码。感知密码通常是一系列问题，这些问题应该是只有主体才知道的事实或预定义的结果。举例来说，主体可能期望三到五个下面这样的问题：

- 生日是哪一天？
- 妈妈的娘家名字是什么？

- 部门的领导是谁？
- 最近的评估考试得多少分？
- 1984 年棒球联赛中你最喜欢的棒球手是谁？

如果所有的问题都答对了，那么主体则通过了身份验证。最有效的感知密码系统每次都会问一系列不同的问题。感知密码系统最大的局限在于，每个问题都必须在用户注册的时候（例如，用户账户建立时）回答，并且在登录期间重复回答，这增加了登录的时间。

很多系统都包括密码的策略，可以限制或规定密码的特性。通用的限制包括最小长度、最小期限、最大期限、需要三个或四个字符类型（例如大写字母、小写字母、数字和符号）和防止密码重复使用。在安全性需求增加时，这些限制应该进行加强。

然而，即使有强大的软件强制密码限制，仍然可能出现易猜到的或易破解的密码。组织机构的安全性策略必须清楚地定义不易破解的密码的需求，以及不易破解的密码是什么。需要对用户进行安全性的培训，这样他们才会重视机构的安全性策略，并且遵守它的规定。如果密码由最终用户自己制定，那么应该向他们提供建立不易破解的密码的建议。例如：

- 不要再使用名字、登录名、E-mail地址、雇员号码、社会保险号码、电话号码或其他标识身份的名字或代码。
- 不要使用字典中的词、俚语或行业缩写。
- 应使用非标准的大写和拼写方法。
- 应交换字母，并且利用数字来代替字母。

当怀有恶意的用户或攻击者寻求获得密码的时候，他们可以有几种办法。这些办法包括网络传输内容分析、密码文件访问、暴力攻击、字典程序攻击和社会工程学。网络传输内容分析是指当用户输入身份验证的密码时，对网络传输内容的截获（也被看做是探测）。一旦密码被破解，攻击者会试着向网络重新发送这个含有密码的包，以获得对网络的访问。如果攻击者可以获得对密码库文件的访问，那么他可以对这个文件进行拷贝，并且使用密码破解工具获得用户名和密码。暴力攻击和字典程序攻击属于密码攻击类型，这种类型可以对盗取来的密码库文件或系统的登录提示展开攻击。在字典程序攻击中，攻击者会使用由常用密码和字典中的词汇组成的脚本，企图破解用户账户的密码。在暴力攻击中，攻击者会使用所有字符可能的组合进行系统的测试，企图破解用户账户的密码。在社会工程学的攻击中，攻击者企图通过对用户的欺骗，通常是通过电话、对系统执行特定的操作（如改变不在现场的主管的密码，或者为一个不存在的雇员建立一个用户账户）对系统进行攻击。

增强密码的安全性存在几种方法。账户闭锁就是其中一种方法，它可以在失败的登录次数达到指定的数量后，关闭用户的账户。账户闭锁防止了暴力攻击和字典程序对系统登录提示的攻击。一旦达到了登录企图的限制，系统就会显示一则消息，报告最后一次成功或失败的登录企图的时间、日期和位置（例如，计算机名或IP地址）。怀疑自己的账户被攻击或已经被破解的用户可以向系统管理员报告这个信息。可以配置审核（Auditing），对登录的成功或失败进行跟踪。入侵检测系统可以轻易地识别登录提示攻击，并通知管理员。

还有一些其他的选择来增强安全性，它们通过密码的验证来实现：

- 对于密码的存储，可以使用单向加密的最强形式。
- 永远不要准许密码以明码的形式或者较弱的加密能力在网络中进行传递。
- 对自己的密码库文件使用密码验证工具和密码破解工具。要求所有易破解或解读的

密码的账户改变它们的密码。

关闭那些短时期内（例如一周或一个月）暂时不使用的用户账户。将那些再也不会使用的用户账户删除。

适当地对用户进行培训，告诉他们维护安全性和使用不易破解的密码的必要性。 对记录或共享密码的用户进行警告。提供一些技巧减轻维护安全的工作负担，或防止通过键盘记录截获密码。对如何建立不易破解的密码提供一些技巧和建议。

生物测定学

另一个常用的验证和身份标识技术是生物测定学。生物测定学关注于“你是什么”和“你有什么”的验证范畴。生物测定学的一个因素是主体独有的行为或生理上的特点。生物测定学有很多类型的因素，其中包括指纹、面容扫描、虹膜扫描、视网膜扫描、手掌扫描—也被认为是手掌外形或手掌特征、心跳或脉搏取样、语音取样、签字力度、按键取样等等。

生物测定学可以作为一项身份标识或验证的技术使用。使用一个生物测定学因素代替用户名或账户 ID作为身份标识，需要生物测定学取样对已存储的取样数据库中内容进行一对多的查找。作为一项身份标识技术，生物测定学被用做有形的（物理的）访问控制。使用生物测定学因素作为验证技术，需要在生物测定学取样和已存储的取样之间保持主体身份的一对一对应。作为一项验证技术，生物测定学被用在逻辑访问控制当中。

生物测定学的使用保证对星球上的每一个人提供绝对惟一的身份标识。不幸的是，生物测定学技术还没有做到这一点。由于生物测定学将被采用，因此它必须绝对灵敏。为了应用生物测定学作为身份标识的办法，生物测定学设备必须能够读取非常精确的信息，如人的视网膜中的血管变化，或者声音中音调和音质的变化。由于大多数人基本上都保持一致，因此对主体的验证所达到的细节要求经常导致负误判（ false negative ）和正误判（ false positive）验证。

生物测定学设备根据两个负验证条件评定它们的性能。大多数生物测定学设备都具有灵敏度校准，因此它们可以调节得更灵敏或不太灵敏。当生物测定学设备太灵敏时，就会出现 1类错误。 1类错误发生在合法主体没有经过验证的时候。对于合法验证的 1类错误发生率被称做误拒绝率(False Rejection Rate, FRR）。当生物学测定设备不够灵敏时，2类错误就会出现。 2类错误发生在合法主体经过验证的时候。对于合法验证的 2类错误被称为误接受率(False Acceptance Rate，FAR ）。FRR和FAR总是通过绘图来显示灵敏度调整的程度和FRR、FAR错误百分比的关系（如图 1.1所示）。FRR 和FAR等值的点被称为错误率交叉点（Crossover Error Rate, CER)。CER级别被用做测量生物学测定设备的标准评估点。在某些条件下，比如机场的金属探测器，设备的灵敏度高于CER是必要的。

除了生物测定学设备的灵敏度问题之外，还有其他一些因素可能会使它们变得不太有效，这些因素包括登记时间、处理能力和认可。对于生物学测定设备来说，主体必须登记或注册，它才能作为身份标识或验证机制使用。也就是说，主体的生物学测定必须被取样并且存储在设备的数据库中。扫描和存储生物测定所需要的时间很大程度上依赖于采用了什么样的检查或性能特性。利用生物测定学机制登记的时间越长，用户越不能接受这种麻烦。一般

来说，登记时间超过两分钟是不能接受的。如果生物测定特性随时间而变化，例如人的语调、头发或签字的方式，那么登记就必须定期重新进行。

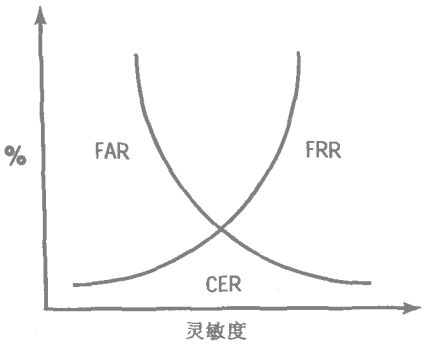

图 1.1 图解FRR和FAR错误，指示CER点

主体一旦被登记，系统扫描和处理主体所需要的时间就被看做是处理能力。生物测定特性越复杂、越详细，处理的时间也就越长。主体接受处理能力的典型时间是6秒钟或更短。

主体对于安全机制的接受程度依赖于很多主观感觉，包括隐私、侵害和心理或生理上的不舒服。主体还可能关注通过生物测定扫描设备带来的体液交叉和披露健康问题。

标 记

标记是密码生成设备，主体必须随身携带。标记设备是“你有什么”的一种形式。标记可以是静态的密码设备，如ATM卡。为了使用 ATM卡，你必须提供标记（ATM卡）和PIN（个人身份号码）。标记还可以是一次性或动态的密码设备，它看起来有些像小型的计算器。这个设备向你显示了向系统输入的一串字符（密码）。

有四种类型的标记设备：

- 静态标记
- 同步动态密码标记
- 异步动态密码标记
- 质询响应标记

静态标记可以是刷卡、智能卡、软盘、 USB RAM dongle或像开锁的钥匙一样简单的物品。静态标记提供物理的手段来提供身份。静态标记始终需要额外的因素提供验证，例如密码或生物测定。大多数设备的静态标记具有加密密钥（cryptographic key） 、数字签名或加密的登录证书。加密密钥可以当做身份标识或验证机制。由于加密密钥使用难破解的加密协议进行加密，因此它比密码更难破解，并且它只存在于标记中。静态的密钥最常被当做身份标识设备，而不是身份验证因素。

同步动态密码标记以固定时间间隔生成密码。时间间隔标记需要验证服务器上的时钟和标记设备上的时钟是同步的。生成的密码由主体协同PIN、通行口令或密码输入到系统中。生成的密码提供了身份标识， PIN/密码提供了身份验证。

异步动态密码标记基于出现的事件生成密码。事件标记需要主体在标记和验证服务器

上压制一个密钥。这个动作先于生成新的密码值。输入生成的密码和主体的PIN、通行口令或密码进行身份验证。

质询响应标记基于验证系统的指示生成密码或响应。身份验证系统显示质询，通常以代码或通行口令的形式显示。质询被输入到标记设备中。标记基于质询生成响应，并且随后响应被输入到系统中进行验证。

标记验证系统比起单独的密码验证来说，是更加难破解的安全机制。标记系统使用两个或更多因素建立身份标识，并且提供验证。除了知道用户名、密码、 PIN、代码等等内容外，主体必须在标记设备的物理掌控之中。

然而，标记系统仍然会失效。如果电池用尽或设备损坏，主体则不能获得访问。标记设备可能丢失或被盗。由于一旦标记系统受到损害，替代可能很困难，而且会很昂贵，因此应该巧妙地对标记进行存放和管理。

权证

权证（tickets）验证这种机制采用第三方实体证实身份，并且提供验证。最常用的也是最知名的权证系统是Kerberos。Kerberos 在麻省理工学院的Project Athena 下进行开发。其名字从希腊神话中来。一只三个头的狗名为Kerberos，它守护着通往阴间的大门。但是在神话中这只三个头的狗脸朝内，防止逃跑，而不是防止进入。

Kerberos 验证机制集中在一台（或多台）被信任服务器上，它负责提供的功能包括：密钥分布中心（Key Distribution Center，KDC） 、权证授权服务（Ticket Granting Service，TGS）和身份验证服务（Authentication Service，AS）。 Kerberos 对客户端和服务器使用对称密钥加密。所有的客户端和服务器都与KDC注册，这样KDC维护着所有网络成员的加密密钥。

单用户登录

单用户登录（Single Sign On，SSO）是准许主体在系统上只进行一次验证的机制。利用单用户登录，一旦主体被验证，那么它们可能在网络中自由漫游，并且不必再次对验证进行质询统可以访问资源和服务。 SSO的主要缺点是：一旦账户被破解，那么恶意主体具有无限制的访问。SSO通常准许使用较难破解的密码，这是因为主体必须记住惟一的密码。此外，SSO通过减少为主体进行账户定义的位置的数量来提供较容易的管理。 SSO 在加载时，可以通过验证系统或者提供登录证书的脚本自动启动。

脚本、Kerberos、SESAME和KryptoKnight都是SSO机制的例子。

在客户端、服务器和TGS 之间的复杂权证交换被用来证实身份，并且在客户端和服务器之间提供验证。在有十分的把握确保双方都是其所宣称的实体时，这个操作准许客户端从服务器请求资源。加密权证的交换还保证了登录证书、会话密钥或验证信息永远不会通过明码进行传递。

Kerberos权证具有指定的使用时限，并且使用一些参数进行操作。一旦权证过期，客户端必须请求更新或一个新的权证继续与服务器进行通信。

Kerberos作为一种通用的身份验证机制，可以用在本地局域网（local LAN）、本地登录、远程访问和客户端—服务器资源的请求中。然而，Kerberos具有单点故障——KDC。如果KDC被破解，那么网络中所有系统的安全密钥也都被破解。同样，如果KDC掉线，那么将不可能进行主体的验证。

Kerberos还有其他一些限制或问题：

- 字典程序攻击和暴力攻击对客户端初始TGS响应的攻击可能会显露主体的密码。
- 发布的权证被存储到客户端和服务器的内存中。
- 如果截获到的权证在使用时限内，那么恶意主体可能再次发送这些权证。

访问控制技术

一旦主体通过了识别和验证，并且建立了，那么它们必须被授权访问资源或进行操作。授权可以只在主体的身份通过验证得到证实后进行。系统通过访问控制提供授权。访问控制管理着主体对客体所拥有的访问类型和范围。目前有三种访问控制技术：任意的、强制的和不可任意支配的。

采用任意访问控制的系统允许客体的所有者或建立者控制和定义主体对客体的访问。换句话说，访问控制是基于拥有者的自由处理。举例来说，如果用户建立了一个新的电子表格文件，那么他们就是这个文件的拥有者。作为文件的拥有者，他们可以修改文件的许可权，对其他的主体授予或拒绝服务。任意的访问控制常常利用对客体的访问控制列表（access control lists，ACL）来执行。每个ACL定义了对个别或一组主体授予或限制的访问类型。由于拥有者可以修改对客体的ACL，因此任意的访问控制不提供集中的控制管理系统。这种访问要比强制的访问控制更加动态。

强制访问控制依赖于标签的使用。主体按照它们的分类标准或敏感度被贴上标签。例如军方使用绝密、机密、秘密、敏感但未分类和未分类的标签。在一个强制访问控制系统中，主体能够对具有相同或较低标签或分类的客体进行访问。这种访问控制方法的一种延伸被称为需要知道（need-to-know）。对于级别具有较大差别的主体，只有它们的工作任务需要进行这样的访问时，才被赋予权限访问具有更高机密级别的资源。如果不具备“需要知道（need-to-know）”，那么即使它们的级别差距满足要求，它们的访问仍然会被拒绝。

在强制访问控制中，安全标签的使用引出了一些有趣的问题。首先，强制访问控制系统若要运行，那么每个主体和客体都必须具有一个安全标签。依赖于运行的环境，安全标签可能涉及机密、分类、部门和项目等等。前面提到的军方的安全标签从最高机密到最低进行了分类：例如军方使用绝密、机密、秘密、敏感但未分类和未分类。普通的公司或商业安全标签级别包括秘密的、专利的、隐私的、敏感的和公开的。安全分类指出了敏感度的级别，但是每个级别都是不同的。实际上，这些级别间的区别在企图从一个级别向另一个级别移动客体时，会引发一些问题。已经出现了几个方案，它们将在本章后面的“访问控制模型”一节中进行讨论。

不可任意支配的访问控制也被称做是基于角色的访问控制。系统采用不可任意支配的访问控制，定义了主体通过主体的角色或任务访问客体的能力。如果主体处于管理位置上，那么他们将比那些处于临时位置的人具备更大的资源访问能力。由于访问基于工作的描述（如

角色或任务）而不是主体的身份，因此基于角色的访问控制在人员频繁变动的环境中很有用。

角色和组（group）服务于相同的目的，但是在使用和配置中它们是不同的。在作为容器将用户集中于可管理的单元方面，它们是类似的。然而，用户可以是多个组的成员。除了从每个组中获取权限和许可权之外，个人用户账户还可能具有直接分配给他的权限和许可权。当使用角色时，用户可以只有一个单一的角色。用户只具有分配给这个角色的权限和许可权，并且没有额外的个别分配的权限和许可权。

格状访问控制是不可任意支配的访问控制的一种变化形式。格状访问控制为主体和客体间的所有关系定义了访问的上限和下限。这个上下限可以是任意的，但是常常遵循军方或公司的安全标签级别。在图1.2中所示的具有格状许可权的主体可以访问到最高到私有的，最低到敏感的资源，但是他不可以访问秘密的、专利的或公共的资源。根据主体所分配的格子位置，在格状访问控制下的主体可以说具有了对标记客体最小的访问上限和最大的访问下限。

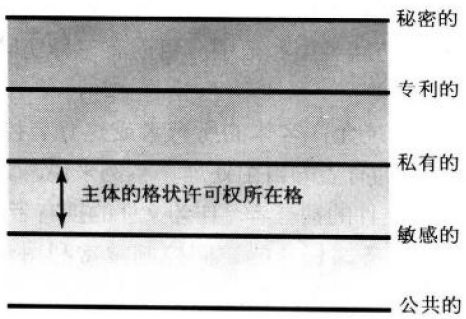

图1.2　格状基础上的访问控制所表现出的上下限

规则基础上的访问控制是强制访问控制的一种变化形式。规则基础上的系统使用一系列规则、限制或过滤器决定在系统上可以做什么，不可以做什么，如准许主体对客体进行访问或执行某个操作，或者访问某个资源。防火墙、代理和路由器是规则基础上的访问控制系统的常见例子。规则基础上的访问控制由系统管理员建立和维护，用户不能对其进行修改。

访问控制模型

访问控制模型是安全性策略的符号表示方式。在很多情况下，访问控制模型会生成一些复杂的安全性策略，这些策略可以通过由计算机必须进行访问控制的内容所定义的规则来理解。访问控制模型有很多种，其中包括：

- 状态机模型（State machine model）
  Bell-LaPadula
- Biba
- Clark-Wilson
  信息流模型（Information flow model）
  无干扰模型（Noninterference model）

- 获得授权模型（Take-Grant model）
- 访问控制矩阵（Access control matrix

状态机模型

状态机模型向我们描绘了一个无论处于何种状态下总是安全的系统。很多安全模型都是基于这个安全状态概念产生的。状态（state）是系统在某个特定事件中的即时快照。如果状态的所有方面都满足安全性策略的要求，那么这个状态就被认为是安全的。转换到下一步的操作就是接受输入或执行输出。转换总是会产生新的状态（也称做是状态转换）。所有的状态转换都必须进行评估。如果所有可能的状态转换导致了另外一个安全状态，那么系统可以标记为一个安全状态机。安全状态机模型系统总是会进入一个安全状态（在所有的转换中维护安全状态），并且准许主体只在安全行为适应安全性策略的情况下访问资源。安全状态机模型是其他许多安全模型的基础。

Bell-LaPadula模型

Bell-LaPadula模型是在美国国防部多级安全策略之外开发的模型。国防部的策略包括四个分类，从最敏感的到最不敏感的：绝密、机密、秘密和未分类。策略规定所有许可级别的主体可以访问其许可级别或之下的资源。然而对于秘密、机密和绝密的许可，访问只有在“需要知道”的基础上才能被准许。换句话说，访问一个特定的客体只有在特定的工作任务需要这样的访问时，才会得到准许。根据这些限制，Bell-LaPadula模型专注于保持客体的保密性。Bell-LaPadula没有说明客体的完整性或可用性方面的内容。

Bell-LaPadula 模型防止了分类信息泄漏或传输到安全许可级别较低的环境中。通过阻止较低类别的主体对较高类别的客体所进行的访问实现此模型的安全保护。

Bell-LaPadula模型以状态机模型为基础。它还采用强制访问控制和格子模型。格子等级是由组织机构的安全策略使用的分类标准（classification levels）。在这种模型中，安全状态由两种规则或特性所限制：

简单安全特性　简单安全特性（Simple Security Property，SS Property）规定在某个特定的分类标准下，主体不能够读取具有较高分类标准的数据。这时常被缩略为“不能向上读（no read up）”。

*安全特性　*安全特性规定在某个特定的分类标准下，主体不能向较低分类的标准进行数据写操作。这通常会被缩略为“不能向下写（no write down）”。

这两个规则定义了系统可能转换到的状态。其他的转换都是不准许的。所有可以通过这两个规则访问的状态都是安全状态。因此，Bell-LaPadula模型系统提供状态机模型安全性（参见图 1.3所示）。

说明：在Bell-LaPadula 模型中有一个例外，它规定了“受信任的主体”不受*特性的限制。受信任的主体被定义为“被保证即使可能也不必完成破坏安全信息的传输”。这意味着准许受信任的主体不必遵循＊特性，并且向较低分类的标准数据执行写操作。

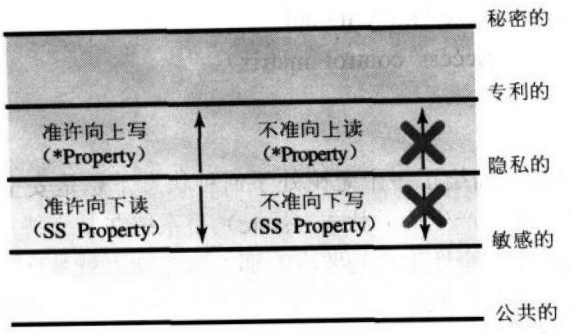

图1.3 Bell-LaPadula 模型

Bell-LaPadula有效地对机密性进行了管理，但是它却不能对其他众多的安全问题进行说明或管理：

- 没有说明完整性和可用性。
- 没有说明访问控制管理，也没有提供方法分配或改变客体或主体的分类标准。
- 没有提供暗道（covert channels）。在第12章“安全模型的原则”中将要讨论的暗道是指数据可以凭借它利用非正常、非期望或不可发觉的方法进行通信。
- 没有说明文件共享（网络系统中常见的特性）。

Biba

对于很多非军事组织来说，完整性比保密更重要。除了这个需要之外，许多专注于完整性的安全性方法也开发出来了，如由Biba和Clark-Wilson开发的方法。

Biba模型来自于对Bell-LaPadula的直接模拟。Biba还是以强制访问控制分格为基础的状态机模型。Biba用于：

- 防止未授权的主体对客体的修改。
- 防止授权的主体对客体进行未授权的修改。
- 保护内部和外部的客体一致性。

Biba有两个完整性规则：

简单完整性规则　简单完整性规则（Simple Integrity Axiom，SI Axiom ）规定在特定分类标准上的主体不能读取较低分类标准的数据。这通常会被缩略为“不能向下读”。

*完整性规则　*完整性规则（*Integrity Axiom，*Axiom）　规定在特定分类标准上的主体不能向较高分类标准进行数据写操作。这通常会被缩略为“不能向上写”。

这些Biba模型规则在图1.4中进行了图解。

Biba模型的评定提到了几个缺陷：

- 只说明了完整性，没有说明保密或可用性。
- 它专注于保护客体不受外部的威胁。它假设内部的威胁被有计划地进行控制。
- 它没有说明访问控制管理，也没有提供方法分配或改变主体的分类标准。
- 没有避免暗道（covert channels ，参见第12章）。

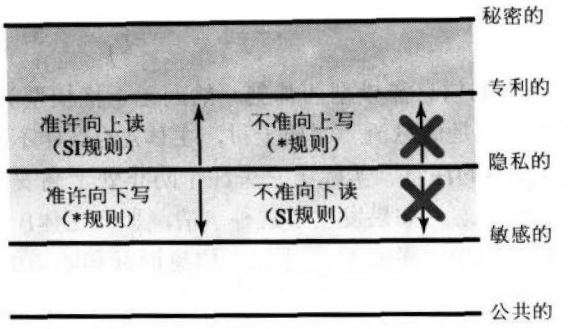

图1.4 Biba 模型

Clark-Wilson

Clark-Wilson模型还是完整性保护模型（integrity-protecting model）。Clark-Wilson 模型在Biba之后被开发，并且从一个不同的角度接近完整性保护。它使用称为三元组的主体／程序/客体的三部分关系，而没有采用格子的结构。主体并不对客体进行直接的访问。客体只能通过程序进行访问。通过两个原则的使用，即良好的处理（well-formed transactions）和责任分离（separation of duties）。Clark-Wilson 模型提供了保护完整性的有效的方法。

良好处理采用程序的形式。主体只能够通过程序访问客体。每个程序都对客体具有特定的可以或不可以的操作限制。这有效地限制了主体的能力。如果程序正确设计，那么三元组的关系提供了保护客体完整性的方法。

责任分离采用了将关键性功能分成两个或更多部分的形式。不同的主体必须完成每个部分。这保护了授权的主体不对客体进行未授权的修改。此外它还保护了客体的完整性。

除了这两个原则外，还需要审核。审核跟踪对客体的改变和访问，以及来自系统外部的输入。

Clark-Wilson 模型还可以称为约束接口模型（restricted interface model）。约束接口模型使用分类基础上的限制，只提供主体指定的授权信息和功能。在某个分类标准上的主体将可以看到一系列的数据，并且可以使用一系列的功能。反之，另一个分类标准上的另一个主体将可以看到不同系列的数据，并且可以使用不同系列的功能。

信息流模型

信息流模型专注于信息流。信息流模型以状态机模型为基础。Bell-LaPadula和Biba模型也都是信息流模型。Bell-LaPadula关注于防止信息从高安全级向低安全级流动。Biba关注于防止信息从低安全级向高安全级流动。信息流模型没有必要只对信息流的方向进行处理，它们还可以说明信息流的类型。

信息流模型被设计用来避免未授权的、不安全的或受限的信息流的出现。信息流可以出现于相同分类级别的主体和客体之间，也可以是不同分类级别的主体和客体之间。信息流模型准许所有的授权的信息流，无论是在相同的分类级别内还是处于不同的分类级别之间。信息流模型避免了所有未授权信息流的出现，无论是在相同的分类级别内，还是位于不同的分类级别之间。

无干扰模型

无干扰模型以松散的信息流模型为基础。然而无干扰模型关注的是主体影响系统状态或另一个主体的操作，不是信息流。本质上讲，主体A的操作不应该影响主体B的操作，或者甚至不应该引起主体B的注意。实际上它关注于防止处于高安全分类级的主体A的操作影响处于低分类级的系统状态。如果发生了这种事情，那么主体B可能会处于不安全的状态，或者可能会推导出较高级分类的信息。这属于信息泄漏和暗道的一种类型。

获得授权模型

获得授权模型采用定向的图标来指示权利如何从一个主体向另一个主体进行传递，或者从一个主体向一个客体传递。通过简单的移动，具有授权资格的主体可以向另一个主体或客体授予所拥有的其他任何权利。同样，具有获得权利的主体可以从另一个主体获得权利。

访问控制矩阵

访问控制矩阵是一个由主体和客体组成的表，它指出了每个主体可以对每个客体实施的操作或执行的功能。矩阵的每一列都是一个ACL。矩阵的每一行都是权利的列表（capability list）。ACL与客体相关，它列出了每个主体可以执行的有效的操作。权利列表与主体相关，它列出了可以在所有客体上进行的操作。

表1.1中所示的访问控制矩阵来自于一个任意访问控制系统。简单地利用分类或角色代替主体名，便可以得到强制或规则基础上的矩阵。系统使用访问控制矩阵快速地了解是否主体对客体请求的操作得到了授权。

表1.1 访问控制矩阵

主体	文档	打印机	网络文件夹共享
Bob	读	不能访问	不能访问
Mary	不能访问	不能访问	读
Amanda	读，写	打印	不能访问
Mark	读，写	打印	读，写
Kathryn	读，写	打印，管理打印队列	读，写，执行
Colin	读，写修改权限	打印，管理打印队列，修改权限	读，写，执行，修改权限

访问控制方法及其实施

有两种主要的访问控制方法：集中式和分散式（或分布式）。集中式访问控制（Centralized access control）暗示所有的授权验证都由系统中的单一实体执行。分散式访

问控制（Decentralized access control）或分布式访问控制（distributed access control）暗示授权验证由贯穿于系统中的不同实体执行。

这两种访问控制方法具有所有集中或分散式系统的优缺点。集中式访问控制准许小型团队或个人进行访问控制的管理。由于所有的更改都发生在单一位置，因此管理的负担较小。单个更改可以影响整个系统。然而，集中式访问控制也会成为单一故障点。如果系统组件不能访问集中式访问控制系统，那么主体和客体将不能相互联系。 RADIUS和TACACS 就是两个集中式访问控制的例子。

分散式访问控制常常需要几个团队或多个人对访问控制进行管理。由于更改必须在许多地方进行实施，因此管理的负担比较重。随着访问控制点的增加，系统的一致性维护工作变得越来越困难。单独的更改只会对与特定访问控制点相关的系统内容产生影响。分散式访问控制没有单点故障。如果一个访问控制点出现故障，那么其他的访问控制点可以均衡流量，直到控制点修复。加上那些与故障访问控制点无关的主体和客体，它们继续进行正常的通信。域和信任常常用在分散式访问控制系统中。

域（domain）是一个信任范围，或者说共享共同安全性策略的主体和客体的集合。当涉及多个域时，就会形成分散式访问控制。为了从一个域到另一个域进行资源的共享，必须建立信任关系。信任（trust）是一种建立在两个域之间的简单安全桥梁，它准许用户从一个域中访问另一个域中的资源。信任可以是单向的，也可以是双向的。

RADIUS 和TACACS

远程拨号用户授权服务（Remote Authentication Dial-In User Service，RADIUS） 被用于集中化远程拨号连接的授权。采用 RADIUS 服务器的网络经过配置，远程访问服务器可以将拨号用户的登录证书传到RADIUS服务器上进行验证。这个过程与域客户端向域控制器发送登录证书进行验证的过程是类似的。

终端访问控制器访问控制系统（Terminal Access Controller Access Control System，TACACS）是RADIUS的替代。TACACS 有三个可用的版本：TACACS 最初版、XTACACS（扩展的TACACS，Extended TACACS） 和TACACS+。TACACS 集成了验证和授权过程。XTACACS保持了验证、授权和记账过程的分离。 TACACS+通过增加两个方面的验证增强了XTACACS。TACACS 和RADIUS操作上类似，并且TACACS提供了与RADIUS相同的功能。然而RADIUS以因特网标准为基础，而TACACS却是更加私有化的方案（虽然得到了广泛的应用）。

访问控制管理

访问控制管理是分配给管理员管理用户账户、访问和责任衡量的一组任务和责任。系统的安全性以有效的访问控制管理为基础。需要记住，访问控制依赖于四个原则：身份标识、验证、授权和责任衡量。在涉及到访问控制管理时，这些原则转换为三个主要的职责：

- 用户账户管理
- 操作跟踪
- 访问权利和许可权的管理

账户管理

用户账户管理涉及建立、维护和关闭用户账户。虽然这些操作可能看起来很普通，但是对于系统访问控制效力来说却是必要的。没有正确定义和维护的用户账户，系统就不能建立标识、实施验证、证实授权或跟踪责任。

新的用户账户的建立是一个简单的系统处理过程，但是它必须受到组织机构安全性策略的保护或保障。用户账户不应该凭管理员的一时兴起或任何人的请求而建立。相反，应该遵循人事部门的雇佣或职务提升手续执行严格的操作。

人事部门应该对新员工的用户账户做出正式的要求。要求应该包括分配给新员工用户账户的分类或安全性级别。新员工的部门经理和公司安全管理员应该检验安全分配。只有要求得到了验证，才能随后建立新的用户账户。不遵循已建立的安全性策略和手续建立用户账户，会带来漏洞和疏忽，被恶意主体利用。增加或减少现有用户账户安全级别也应该遵循类似的过程。

作为人员雇佣的手续，新员工应该接受公司安全性策略的手续的培训。在完成雇佣前，员工应该签署一份协议，承诺支持公司的安全标准。许多组织机构已经选择构思一份文档，规定违反安全性策略就会被解雇，以及依照联邦、州和地方法律进行起诉。当将用户的账户ID和临时密码交给新员工时，应当执行密码策略（password policy）的检查和可接受的使用约束。

新用户账户的最初建立常常被称为注册。注册过程生成新的身份，并建立起系统进行验证所需要的要素。注册过程完整并正确地完成是很关键的。通过组织机构采用的任何必要且充分的手段对注册个体的身份进行证实也是很关键的。在向安全系统注册这些人时，带照片的身份证（photo ID）、出生证（birth certificate）、背景调查（background check）、信用调查（credit check）、安全性级别证实（security clearance verification）、联邦调查局数据库检索（FBI database search），甚至职业证明（calling references）都是证实一个人身份的有效形式。

用户账户的整个生命周期内，持续的维护是必不可少的。那些有着相当稳定的组织结构和较少人员变动或升迁的组织，比起那些有着灵活的或动态的组织结构并且具有较高的人员变动和升迁的组织，将具有相当少的账户管理工作。大多数账户维护工作围绕着账户的权限和特权的更改而进行。应当建立起类似于新的账户建立时的那些过程，它们管理着在整个用户账户的生命周期内对访问的更改。未授权账户访问能力的增加或减少可以产生严重的安全影响。

当一个员工不再为公司工作时，其用户账户应该被关闭、删除或废除。无论何时，只要可能，这项工作就应该自动完成，并且应当与人事部门相配合。在大多数情况下，当薪水停止支付的时候，这名员工就应不再具有登录的能力。临时或短期的员工应当在他们的用户账户中拟订特定的截止日期。这样，账户生成时建立的控制级别得以维护，并且不会出现管理疏漏。

账户、日志和定期监控

操作审计、账户跟踪和系统监控也是访问控制管理中的重要内容。没有这些内容，把握主体的责任将是不可能的。在身份建立、验证和授权的过程中，跟踪主体的操作（包括他们访问了客体多少次）提供了直接且明确的责任。审核和监控，作为操作安全性和安全环境的必要组件，将在第14章“审核和监控”中进行讨论。

访问权限和许可权

为客体分配访问权限是实施组织机构安全性策略的重要部分。不是所有的主体都应当被赋予对所有客体的访问权限，也不是所有的主体都应具有客体所具有的相同的功能。一些特定的主体只应当访问一些客体。否则，某些功能只应当由一些特定的主体访问。

最少特权的原则（principle of least privilege）来自于当主体被赋予对客体进行访问的权限时所形成的复杂结构。这个原则规定主体只应当被授权访问那些完成其工作所需要的客体。这个原则还具有一个应当遵循的转换：应当阻止主体访问那些工作内容不需要的客体。

决定哪些主体对哪些客体具有访问权限，这是组织机构安全性策略、人员所在的组织层次和访问控制模型的实施的一项功能。因此，建立或定义访问权限的标准可以建立在身份、角色、规则、分类、位置、时间、接口和需要知道的等基础上。

讨论对客体的访问要用到三个主体标记：用户、所有者和管理人。用户就是任意的主体，他可以访问系统中的客体，执行一些操作或完成工作任务。所有者或信息所有者是对分类和标记客体、保护和存储数据负有最终法人责任的人。如果所有者在建立和执行安全性策略保护和维护敏感数据时没有做到适当勤奋（due diligence），那么可能要对疏漏承担责任。日常要对客体进行适当的存储和保护的工作，被分配或委派了这种工作的人称为管理员（custodian）.

用户就是系统中的任意最终用户。所有者常常就是CEO、董事长或部门领导。管理人员常常就是IT人员或系统安全管理员。

任务和责任的分离是通用的准则，它防止任意单一的主体回避或禁用安全机制。当核心管理或高级授权责任被分为几个主体时，没有一个主体具有足够的访问权限来执行有恶意的操作或绕过强迫执行的安全控制。任务的分离建立了一个检测和平衡系统，在这个系统中，多个主体可以相互校验操作，并且必须一起完成必要的工作任务。任务分离使得恶意的、欺诈的或其他未授权的操作变得更加困难，并且拓宽了检测和报告的范围。对于个人来说，如果他们认为可以侥幸成功，那么执行未授权的操作是简单的。一旦涉及两个或更多的人，那么未授权操作的承诺需要每个人同意保密才行。这通常会作为一种有效的威慑，而不是贿赂一组人的手段。

小结

CISSP CBK的第一个领域就是访问控制系统和方法论。访问控制是安全系统建立的核心。访问控制依赖于身份标识、验证、授权和责任衡量。访问控制是对授权或约束主体对客体进行访问的管理、执行和实施。