第 2 章　Windows NT 安全原理

虽然本书以 Windows 2000 作为阐述 Windows 系统安全原理与技术的重点，但由于 Windows 2000 的大部分核心功能和面向对象的设计都来源于 Windows NT 4.0，大部分原有的基础安全结构仍然保持不变，因此本章将首先说明 Windows NT 4.0 系统安全的基本原理与技术。

2.1　Windows NT 系统安全体系

图 2-1 显示了 Windows NT 4.0 系统体系结构中的多个组成部分以及彼此之间的相关性。与其他大多数模型类似，这也是一种分层结构：计算机硬件位于底端，而高层的应用程序位于顶端。用户与最高层的部分进行交互，中间的所有层次都为上一层提供服务并与下一层进行交互。

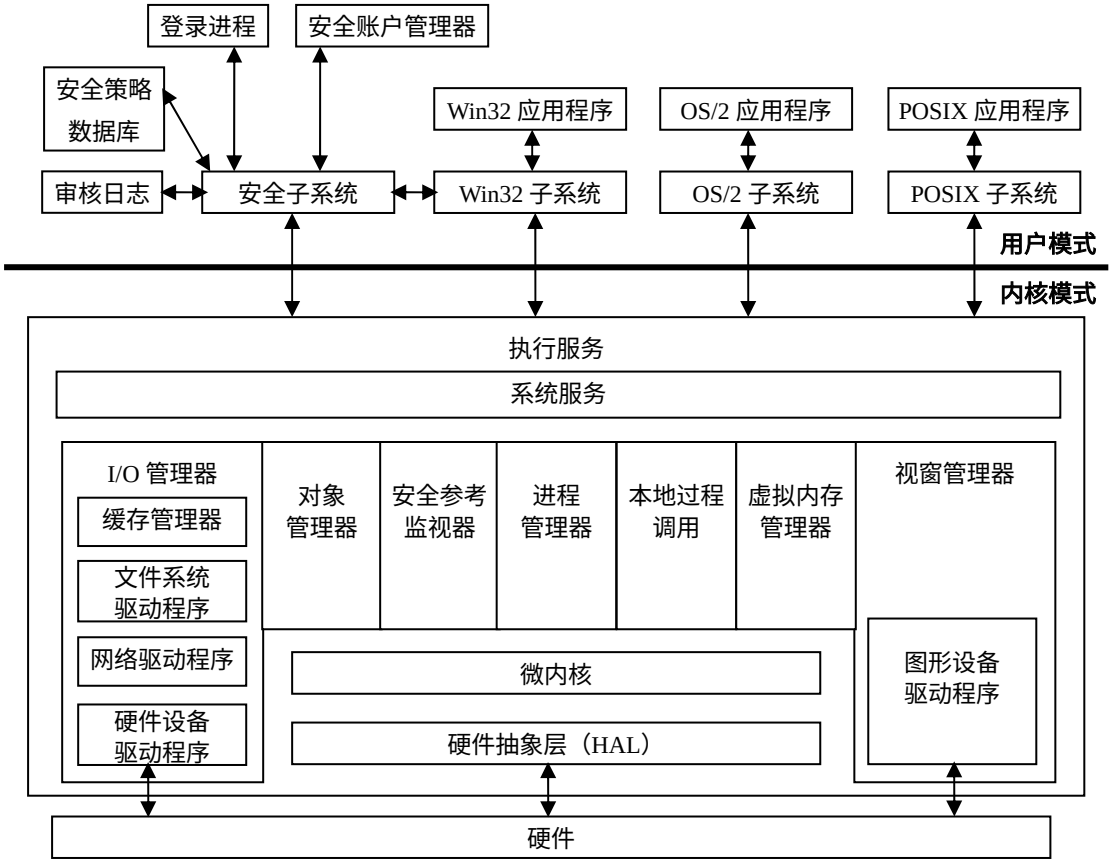

图 2-1　Windows NT4.0 安全体系结构

该体系结构分为两种模式：内核模式（Kernel Mode）和用户模式（User Mode）。

应用程序及其子系统运行在用户模式下。该模式拥有较低特权，不能对硬件直接进行访问。用户模式的应用程序被限定在由操作系统所分配的内存空间内，不能对其他内存地址空间直接进行访问。用户模式只能使用特殊的应用程序编程接口（API）来从内核模式组件中申请系统服务。

用户模式中包含有以下一些主要的子系统。

- Win32 子系统：这是主要的应用程序子系统，所有的 32 位 Windows 应用程序都运

行在这个子系统之下。

- 本地安全子系统：用来支持 Windows 的登录过程，包括对登录的身份验证和审核工作。安全子系统需要和 Win32 子系统进行通信。
- OS/2 子系统：被设计用来运行和 OS/2 1.x 相兼容的应用程序。
- POSIX 子系统：被设计用来运行和 POSIX 1.x 相兼容的应用程序。

而内核模式中的代码则具有极高的特权，可以直接对硬件进行操作以及直接访问所有的内存空间，并不像运行在用户模式下的程序那样被限定在自身特定的地址空间内。组成内核模式的整套服务被称为“执行服务”（有时也被称为 Windows NT Executive）。执行服务通过响应用户模式下应用程序发出的请求来提供内核模式服务。

在 Windows NT 4.0 中组成内核模式的各种执行服务组件包括：

- I/O 管理器　管理操作系统与外界的通信。它是一种软件模块，负责处理设备驱动程序，协助操作系统访问网卡、磁盘驱动器和缓存等物理设备。I/O 管理器的组件包括高速缓存管理器、各种文件驱动程序以及网络驱动程序，另外还有一个组件是用来完成硬件直接访问的硬件设备驱动程序。
- 对象管理器（Object Manager）　管理包括文件、文件夹、进程、线程和网络端口在内的对象，负责对象的命名、安全性维护、分配和处理等工作。
- 安全参考监视器（Security Reference Monitor, SRM）　用来验证访问权限，通过将进程（主体）的访问令牌与被访问对象（客体）的访问控制列表（ACL）相比较，确定是否应该授予进程所请求的权限。注意，对象管理器会调用安全参考监视器。
- 本地过程调用（Local Procedure Call, LPC）　在应用程序与操作系统的底层之间传递服务消息。
- 进程管理器　管理进程的创建和删除工作，为线程和进程提供 API。
- 虚拟内存管理器（Virtual Memory Manager, VMM）　用来在物理的磁盘空间中建立模拟的内存空间，定义和管理进程的地址空间，控制物理内存的分配。
- 视窗管理器　负责提供所有的图形用户界面。视窗管理器直接和图形设备驱动程序通信，而后者又和硬件直接通信。

2.2 Windows NT 和 C2 级安全

在美国国防部（DoD）的国家计算机安全中心（National Computer Security Center, NCSC）曾经颁布的《可信计算机系统评估准则》（TCSEC，即著名的“橘皮书”）中，将操作系统的安全等级分为 4 大类：A 类、B 类、C 类和 D 类。其中有些类还含有子类，如 C 类又分为 C1 级和 C2 级。在这些安全级别中，A 类的安全性最高，而 D 类则几乎不提供任何安全性保护。

B 类和 A 类只用在安全性要求极高的环境中，如政府和军事部门。而 D 类的典型例子就是 Microsoft 的 MS-DOS 操作系统。此外，C 类为大多数的商用操作系统所采用，C2 是 C 类中的最高安全级别。

C2 级别的安全性策略，简而言之，即自由控制的访问权限（Discretionary Access Control, DAC）。它的重要指标包括以下几项。

- 自由的访问控制（DAC）：资源的所有者必须能够控制对资源的访问。
- 对象的重用（Object Reuse）：操作系统必须能够保护对象在完成其使命后，不能够再被其他对象所利用。如释放的内存和删除的文件，不应为其他程序或进程再次访

问读取；或者当文件被删除时，用户决不能访问其数据，即使当其磁盘空间被另一个文件分配时也是如此。这种保护还必须扩展到磁盘、监视器、键盘、鼠标和任何其他设备。

- 强制的用户标识和认证：所有用户都必须以惟一的登录标识（ID）和密码来鉴别自身，从而跟踪用户的活动，而且只有授权的用户才能访问相应的系统资源。
- 可记账性和审核：系统管理员必须能够审核与安全性相关的事件，对其进行记录。事件的审核记录必须能够阻止非授权的访问。对这个审核数据的访问必须仅限于经授权的管理员。

此外，C2 级别的操作系统还必须能够阻止恶意的损害与篡改。

Windows NT 的安全模式设计是和 C2 级安全标准相兼容的，并于 1999 年 12 月在美国顺利地通过了 C2 级的安全测试（通过测试的包括 Windows NT 所有的 Server 版和 Workstation 版）。

2.3 Windows NT 的文件系统

Windows NT 支持两种文件系统：FAT（File Allocation Table）和 NTFS（NT File System），这两种文件系统有着不同的特征和性能。

2.3.1 FAT 文件系统

FAT 文件系统最初是为小磁盘和简单的目录结构设计的，但现在的设计也使它能对大磁盘和功能更强大的系统提供支持。FAT 在运行 MS-DOS 的系统中得到了非常广泛的应用。FAT 格式的文件卷是以簇（Clusters）为单位来分配的，默认簇的大小取决于卷的大小。在 FAT 文件格式的系统中，簇的数目必须在 16 位以内而且必须是 2 的倍数，这就导致了空间的浪费。当 FAT 卷大于 511MB 时，就非常容易造成磁盘空间的浪费，尤其是当碎小文件较多的时候。FAT 文件系统也无法用于大于 4GB 的卷。

FAT 文件系统比较适宜于较小的卷，如 400MB~500MB，但 FAT 文件系统不具有 NTFS 所提供的安全特性（如访问控制列表、加密、磁盘限额等）和磁盘自动修复功能。

2.3.2 NTFS 文件系统

事实上，Windows NT 4.0 的安全性能在很大程度上依赖于 NTFS 文件系统。所谓 NTFS，即 NT File System（NT 文件系统），它是从 Windows NT 所开始采用的独特文件系统结构。NTFS 建立在保护文件和目录数据基础上，同时以节省存储资源、减少磁盘占用率为设计目的。NTFS 较 FAT 文件系统而言，功能更强大，适合更大的磁盘和分区，支持安全性，是更为完善和灵活的文件系统。

NTFS 文件系统为 Windows NT 服务器或工作站提供了必需的安全保障，它的安全特性主要体现在以下几个方面。

- 在 NTFS 分区上支持随机访问控制和拥有权，对共享文件夹可以指定权限，以免受到本地访问或远程访问的影响。
- 对于在计算机上存储文件夹或单个文件，或者是通过连接到共享文件夹访问的用

户，都可以指定权限，以使每个用户只能按照系统赋予的权限进行操作，充分保护了系统和数据的安全。

- NTFS 使用事务日志自动记录所有文件夹和文件更新，当出现系统损坏和电源故障等问题而引起操作失败后，系统能利用日志文件重做或恢复未成功的操作。

正是由于具备了这些安全特性，Windows NT 下网络资源的本地安全性是通过 NTFS 权限许可来实现的。在一个格式化为 NTFS 的分区上，每个文件或者文件夹都可以被单独地分配一个许可，这个许可使得这些资源具备更高级别的安全性。用户无论是在本机还是通过远程网络访问这些设置有 NTFS 许可的资源，都必须具备访问这些资源的权限。

此外，NTFS 支持对单个文件或者整个文件夹进行压缩。这种压缩不同于 FAT 文件系统中对驱动器卷的压缩，其可控性和速度均要好得多。NTFS 文件系统还具备其他一些优点，例如：对于超过 4GB 以上的磁盘，使用 NTFS 分区可以减少磁盘碎片的数量，大幅度提升磁盘的利用率；NTFS 可支持的磁盘分区大小可达 64GB，远大于 FAT32 可支持的 4GB；支持长文件名等。

2.4 Windows NT 的用户和用户组

Windows NT 管理用户时使用本地用户账户、本地组账号和全局组账户。当第一次将系统安装成工作站或者独立服务器的时候，Windows NT 就会默认创建一批内置的本地用户和本地组账号，存放在本地计算机的 SAM 数据库中；而当安装成为域控制器的时候，Windows NT 则会创建一批域组账号。

2.4.1 用户账户

用户账户通过用户名和密码来进行标识。用户名是账户的文本标签，而密码则是账户的身份验证字符串。虽然 Windows NT 显示用户名来说明特权和权限，但账户的关键标识符却是 SID（安全标识符），我们将在下一节中讨论这个问题。

Windows NT 默认创建两个账户：Administrator 和 Guest。其中 Administrator 由管理所有配置的管理员使用，通过它可以管理安全性策略，创建、修改、删除用户和组，修改系统软件，创建管理共享目录，安装连接打印机和格式化硬盘等；Guest 用于临时登录的一次性用户，默认是禁止的，所有使用该账户登录的用户会获得相同的桌面设置。这两个默认账号均可以改名，但都不能删除。

2.4.2 用户组账户

除用户账户外，Windows NT 还提供组账户。在 Windows NT 系统中，具有相似工作或有相似资源要求的用户可以组成一个工作组（也称为用户组）。将对资源的存取权限许可分配给一工作组，就是同时分配给该组中的所有成员，从而可以简化管理维护工作。

Windows NT Server 有一些内置的组账户，每个组都被赋予了特殊的权限。

- Administrators：该组的成员可以控制整个系统，内置的 Administrator 账号即属于该组。
- Users：该组提供最终用户访问系统所必需的权限。除了内置的 Administrator、Guest

和初始用户之外，所有的系统用户都属于 Users 组。

- Backup Operators：该组允许用户备份和恢复计算机文件，其成员可以备份磁盘驱动器上的任何文件，而不管文件和目录是否被设置为禁止访问。
- Guests：该组提供对系统资源的有限访问。Guest 账户自动属于该组。
- Replicator：该组用于配置目录复制服务。
- Print Operators，Account Operators，Server Operators，Domain Administrators，Domain Users，Domain Guests 等组账户。

此外，Windows NT 系统中还有一些特殊组。特殊组无法创建用户，用户要么默认属于这些组，要么根据用户的网络活动成为这些组的成员。

- Network：包含正通过网络上其他计算机连到本地主机的用户。
- Interactive：本地登录的用户自动属于该组。
- Everyone：所有访问本机的用户都属于该组。
- Creator Owner：包括创建资源或获得资源所有权的用户账户。

删除一个组只是删除这个组，并不会删除组中的账户，但被删除的组是无法恢复的。使用账户管理工具创建的组是可以删除的，但 Windows NT 内置的组却不能删除。跟用户账户不同，用户组是无法改名或者被禁止的。

2.4.3 用户和组账户的管理工具

Windows NT 提供了下列几种工具来管理用户账户和组账户：

- 添加用户账户向导
- 用户管理器
- 域用户管理器
- 一组命令行工具

其中使用最多的管理工具是“用户管理器”和“域用户管理器”。“用户管理器（MUSRMGR.EXE）”是用来管理单个工作站资源的 Windows NT Workstation 工具，而“域用户管理器（USRMGR.EXE）”是通过 Windows NT 域管理账户的 Windows NT Server 工具。

从本质上说，“用户管理器”是“域用户管理器”的简化版。因为管理单个工作站时，Windows NT 域的许多选项都不适用，并不需要 Windows NT Server 工具所提供的额外功能。例如，Windows NT 允许通过“域用户管理器”创建的组类型包括本地组和全局组，这是“用户管理器”所不具备的。

2.4.4 本地作用域和全局作用域

通过“用户管理器”和“域用户管理器”创建的用户和组账号可以有不同的作用域——本地作用域和全局作用域。

如果账户（包括用户账户和用户组账号）仅在创建它的工作站上有效，那么就称该账户具有本地作用域；而如果该账户在当前选定的整个域中都有效，那么就称该账户具有全局作用域。表 2-1 说明了使用“用户管理器”或者“域用户管理器”创建不同类型账户时的不同作用域类型。

表 2-1 账户管理工具和账户作用域对应关系表

工具	账户类型	作用域
用户管理器（Windows NT Workstation）	用户	本地

	组	本地
域用户管理器（Windows NT Server）	用户	全局
	本地组	本地
	全局组	全局

2.5 Windows NT 的工作组和域

在 Windows NT 中有两种类型的网络配置和信任关系：工作组和域，它们分别适宜于不同的网络和用户规模。

2.5.1 工作组

工作组（Workgroup）是一个不共享任何用户账户信息和组账户信息的小型 Windows NT 系统集合。工作组中每个系统之间是彼此独立的，在进行验证时都使用系统自身的 SAM 数据库。所以，这种配置仅适用于最小型的环境，否则将是难以管理的，并且根本无法集中进行控制。

2.5.2 域

不同于工作组的独立和松散，Windows NT 4.0 域是具有集中安全授权机构（如主域控制器，PDC）的一批计算机。它至少应该由一台 PDC 和若干台工作站和成员服务器所组成。在实际情况中，域中一般还存在备份域控制器（BDC），它用来提供整个域 SAM 数据库的多份完全复本，以提高有效性，并可以用来向多台服务器提供分布式的验证服务。

Windows NT 域为用户、组和计算机账户定义了安全边界的管理范围。由于集中安全授权机构的存在，一个域就允许在这个域中的所有用户共享普通的用户账户数据库和普通的安全策略，并不再需要每台计算机都各自提供自己的验证服务。一旦某用户通过了 PDC 或者 BDC 的域验证，那么该用户就可以在具有必要权限的域和主体内的任何地方访问资源了。

2.5.3 信任关系

在某个域中，所有的用户账户都集中在 PDC 上，可以进行集中的管理与控制。但是当用户进行跨域访问的时候，那么该如何进行安全控制呢？方法有两种：一种是给每个用户在每个域中分别开设账户，但这样就不利于集中管理，有悖于域的初衷；而另一种方法就是建立域与域之间的信任关系，将每个域的账户集中到一个域中来管理。所以这里的“ 信任关系 ”指的就是 Windows NT 域之间的关系。

信任关系就是两个域之间的通信连接。当连接建立后，其中一个域允许另外一个域的用户访问自己的资源而又不必在本域拥有这个用户的账户与口令，即将安全验证工作委托给另外一个域进行。显然，当网络中各域之间建立适当的信任关系之后，用户凭借一个账户就可以访问所有的域，网络中所有的计算机都可以识别这个用户账户。用户只须登录一次即可访问网上所有的计算机。在管理上，信任关系将两个独立的域连接为一个管理单元，账户管理就可以集中在一个域中进行而不必分散到各个域中。

两个域之间的信任关系有如下两种。

- 单向信任关系：信任只存在于一个方向上，即只是一个域信任另外一个域，反之则不然。信任域信任被信任域中的用户，允许被信任域中的用户访问信任域中的资源。
- 双向信任关系：信任建立在两个方向上，两个域之间彼此信任对方，每个域中的用户账户都可以授权访问另一个域中的资源。

信任关系只限于域和域之间，Windows NT 域与工作组之间不能建立信任关系。当域之间建立起信任关系之后，被信任域中的全局组可以成为信任域中的本地组成员，信任域可以在域中为该本地组指定资源访问许可和权限。在带信任关系的多域环境中，组策略一般也是把用户加入全局组，再将全局组加入本地组。只不过全局组在被信任域中定义，本地组在信任域中定义。信任关系之间也不具有传递性。如果我们假设域 A 信任域 B，域 B 信任域 C，那么将无法推出域 A 信任域 C 的结论。

我们可以利用信任这个基本概念创建多个域模型。每个域模型具有不同的管理效果，在特定的环境中都具有其特殊的优势。

（1） 单域模型

采用单域模型（Single Domain Model）的网络环境中只有一个域，不具有信任关系。单域模型的优势在于可以进行集中的账户管理和资源访问控制，适用于网络规模小，用户数不多的小型组织。

（2） 主域模型

采用主域模型（Master Domain Model）的网络环境中具有多个域。所有用户账户集中在某个域中（账户域）进行管理，我们称这个域为主域；而其他域则信任这个账户域，我们称之为资源域，如存放文件与打印服务器的域。用户放在主域的全局组内，并且这些组被加入到适当资源域的本地组中。账户域的域管理员负责管理用户账户，而每个资源域的域管理员负责管理资源服务器和对这些资源的访问权限。图 2-2 说明了这种模型。

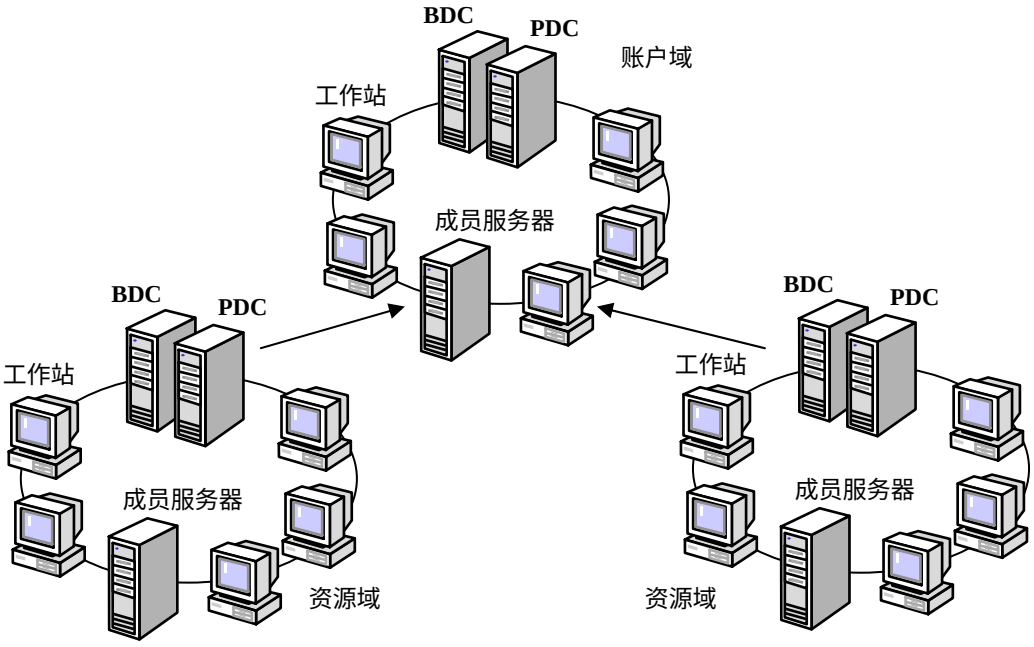

图 2-2 主域模型图

主域模型的优势在于对用户账户的集中管理，并对账户和口令有一个通用策略。管理用户账户的管理员并没有对资源的管理控制权，同时也不允许资源管理员创建有权访问本地信息之外任何信息的新用户账户，也不鼓励在资源域中创建用户账户。主域模型为 Windows NT 的用户提供了可用于资源域中所有 Windows NT 服务单一用户账户的单一登录（Single

Sign-on），适用于网络规模较大的组织。

（3） 多主域模型

多主域模型（Multiple Master Domain Model）则用多个主域控制账户，资源域信任所有的账户域，账户域彼此之间为双向信任。因此，用户账户就可以在任何一个主域中创建，并可以使用任何信任资源域中的资源。这一模型的其他方面与单主域模型相似，但却提供了更大的灵活性和可扩展性。图 2-3 说明了多主域模型。

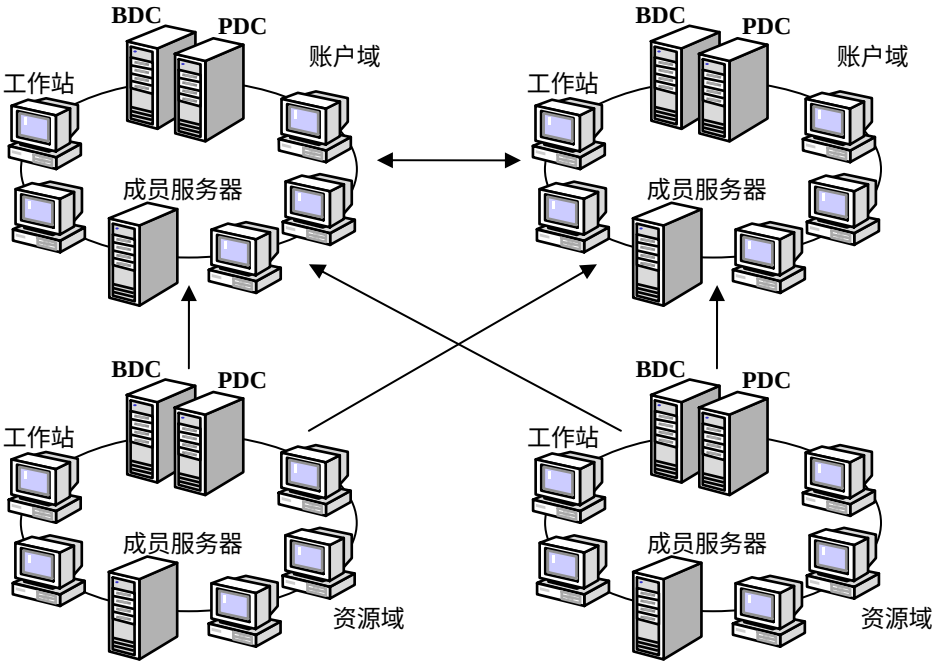

图 2-3 多主域模型图

多主域模型的优势在于，大型组织的不同分公司、分支单位（若地理位置相距很远）都可以独立管理自己的用户账户数据库，同时任何主账户域中的用户照样可以访问任何信任资源域中的共享资源。同时，多主域模型也适用于账户数量庞大，超过一个域所能承受极限的网络环境。

（4） 完全信任模型

完全信任模型（Complete Trust Model）中的所有域之间均是双向的信任关系。这个模型提供了分散式的用户账户、安全管理以及分散式的资源管理。每个部门维护自己的域，按照自己的需要管理用户账户和资源。

完全信任模型的优势在于，它建立在一个组织不同部门之间的松散结构关系之上，并允许来自任何部门的用户都可以访问任何有效的资源。但只有当负责各域的管理员之间能够完全合作时才能够达到足够的安全等级。

2.6 Windows NT 的访问控制

Windows NT 中的访问控制模型建立在用户安全访问令牌和访问控制列表（ACL）保护对象基础之上，为安全性提供了一个合理的基础。这一灵活、面向对象的安全模型的用户界面，提供了一种简单的管理对象安全属性和应用安全设置的方法，实现起来并不十分复杂和繁琐。

2.6.1 安全标识符

安全标识符，也就是我们经常所说的 SID（Security Identifier），是被 Windows NT 系统用来惟一标识安全主体的。安全主体即可以是系统内的用户，也可以是系统内的组，甚至是域。每当我们在系统中创建一个用户或一个用户组的时候，系统就会分配给该用户或组一个惟一的 SID。Windows NT 使用 SID 来独立于用户名之外跟踪账户。这样做的好处有：

- 更改用户名时，Windows NT 可以将其特定的 SID 重新映射到新的用户名上。这样就不会使原先设置在该用户身上的访问控制权限丢失。
- 删除账户时，即使以后创建具有相同用户名的账户，新账户也不会具有相同的访问控制权限，这是因为新账户将有一个新的 SID。

SID 能够保证它永远是惟一的，因为它的构成需要根据计算机名、当前时间、当前用户态线程的 CPU 耗费时间总和这几个参数来决定。

2.6.2 安全访问令牌

用户通过验证后，登录进程会给该用户一个访问令牌（Access token），该令牌相当于用户访问系统资源的凭证。访问令牌会伴随用户启动的每一个进程，当用户进程试图访问系统资源时，就会将访问令牌提交给 Windows NT，然后 Windows NT 检查用户试图访问对象上的访问控制列表，如果该用户被允许访问该对象，Windows NT 就会分配给其适当的访问权限。当用户注销时，与该用户相关联的所有进程都将终止，访问令牌也将被销毁。

访问令牌包含有下列信息：

- 用户的 SID
- 用户所在组的 SID 列表
- 用户或组的特权
- 所有者 SID
- 访问令牌的来源
- 当前模拟级别

2.6.3 访问控制项

访问控制项（Access Control Entries, ACE）包含了某用户或组的 SID 以及该用户或组针对某对象的访问权限。访问权限有两种：允许访问和拒绝访问，即允许用户访问某对象或者拒绝用户访问某对象。

ACE 所标识的访问权限是可以累加的。假设用户被授予对某对象读的权限，而其所在的组被授予对该对象写的权限，那么此用户就可以同时具有对该对象读和写的权限。虽然 ACE 可以累加，但拒绝访问的级别高于允许访问的级别。即假设用户已被明确拒绝访问某对象，那么这种拒绝就覆盖了其他任何允许访问该对象的权限。

2.6.4 访问控制列表

访问控制列表（Access Control List, ACL）是包含有访问控制项（ACE）的一个列表，它提供的安全描述用来保护 Windows NT 系统中的对象。

Windows NT 系统既支持表示允许访问的 ACE，也支持表示拒绝访问的 ACE。但是，在 Windows NT 设置安全权限的用户界面上却不能设置表示拒绝访问的 ACE（虽然可以在程序中用 Win32 安全 API 函数来定义）。

为了进一步支持并简化访问控制权限设置的传递，Windows NT 支持对象上的 ACE 继承，如子目录默认从上一级目录（父目录）继承所有的权限。对象能被定义成容器对象或者非容器对象。容器对象内还可以保存其他对象，比如目录就被定义成容器对象，在其中保存文件对象。

2.6.5 文件系统的访问控制

对文件系统的访问控制，仅适用于采用 NTFS 的磁盘分区。可以在设置文件或目录权限的对话框中对作用于文件或文件夹的访问控制列表（ACL）进行设置。ACL 中的每项 ACE 都会为用户或用户组分配一个或多个访问文件或目录的权限级别。

1. 文件访问权限

对文件而言，可以设置的权限级别为“ 无 ”(No Access)、“ 读取 ”(Read)、“ 更改 ”(Change) 和 “ 完全控制 ” (Full Control)。表 2-2 详细列出了这些权限。

表 2-2 标准文件访问权限

访问类型	*无*	*读取*	*修改*	*完全控制*
显示文件数据		√	√	√
显示文件属性		√	√	√
运行程序文件		√	√	√
显示文件所有者与权限		√	√	√
修改文件属性			√	√
修改文件中的数据			√	√
删除文件			√	√
修改文件所有者和权限				√

2. 特殊文件访问权限

标准文件权限有时并不能为所需保护级别提供足够的粒度。Windows NT 还允许用户利用 “ 特殊权限 ” 方式按照特殊的组合方式分配基本权限。表 2-3 展示了特殊访问权限的详细内容。

表 2-3 特殊文件访问权限

访问类型	*无*	*读取*	*写入*	*执行*	*删除*	*修改权限*	*获取所有权*	*完全控制*
显示文件数据		√						√
显示文件属性		√		√				√
运行程序文件				√				√
显示文件所有者与权限		√	√	√				√
修改文件属性			√					√
修改文件中的数据								
删除文件			√		√			√
修改文件所有者							√	√
修改文件权限						√		√

3. 目录访问权限

与文件权限设置类似，Windows NT 也支持全面的目录访问权限。对目录可以设置的权限级别为“无”（No Access）、“读取”（Read）、“写入”（Add）、“列出文件夹目录”（List）、“读取和写入”（Read & Add）、“修改”（Change）和“完全控制”（Full Control）。表 2-4 详细给出了这些权限。

表 2-4 目录访问权限

访问类型	*无*	*读取*	*写入*	*列出文件夹*	*读取和写入*	*修改*	*完全控制*
显示目录文件名		√		√	√	√	√
显示目录属性		√	√	√	√	√	√
改为子目录		√	√	√	√	√	√
修改目录属性			√		√	√	√
创建子目录与添加文件			√		√	√	√
显示目录所有者与权限		√	√	√	√	√	√
删除目录与子目录						√	√
修改目录权限							√
取消目录所有者身份							√

4. 文件与目录的继承访问权限

目录是容器对象，除了其自身的权限之外，还有目录内创建的子目录和文件（即容器对象）所继承的权限。其中，子目录总是从父目录继承所有的权限。而默认情况下，当用标准对话框设置目录权限时，文件权限按照表 2-5 所示完成设置。

表 2-5 文件继承访问权限

目录访问类型	*文件继承权限*
无	无
读取	读取、执行
写入	未定义
列出文件夹	未定义
读取和写入	读取、执行
修改	读取、写入、执行、删除
完全控制	全部

在设置目录权限之后，任何在该目录内创建的文件均具有继承来的文件访问权限。当然，如果需要的话，也可以直接在文件和目录上设置这些权限。当复制文件时，文件根据所要复制到的目标目录来继承权限，而不是源目录。如果文件的权限与其所在的目录权限不同的话，就会造成安全问题。Windows 2000 则提供了一种更健壮的 ACL 策略能力。它引入了新的继承模型，直接应用的 ACE 要优先于继承的 ACE。Windows 2000 通过把直接应用的 ACE 放置在继承的 ACE 之前来实现这种优先。

5. 共享权限

共享只适用于文件夹（目录）。如果文件夹不是共享的，那么在网络上就不会有用户看到它，也就更不能访问。网络上的绝大多数服务器主要用于存放可被网络用户访问的文件和目录，要使网络用户可以访问在 NT Server 服务器上的文件和目录，必须首先对它建立共享。共享权限（Share Permissions）用来决定用户从网络上访问系统资源对象的控制方式。如果把对象本身的权限比作房间钥匙的话，那么共享权限就是大楼的门卫。即便你有房间钥匙，

但如果门卫不让你进楼，那么你还是不能进入到你的房间里。表 2-6 列出从最大限制到最小限制的共享权限及相应级别允许的用户动作。

表 2-6 共享权限设置

共享权限级别	*允许的用户行为*
无（No Access）	禁止对目录和其中的文件及子目录进行访问
读取（Read）	允许查看文件名和子目录名，改变共享目录的子目录，还允许查看文件的数据和运行应用程序
修改（Change）	具有“读取”权限中允许的操作，此外还允许往目录中添加文件和子目录，更改文件数据，删除文件和子目录
完全控制（Full Control）	具有“修改”权限中允许的操作，此外还允许更改权限和获取所有权（这两项只适用于 NTFS 卷）

共享级访问权限并没有很好的控制粒度，仅限于“无”、“读取”、“修改”和“完全控制”。若要达到一个更好的控制级别，就还需要 NTFS 权限。通过联合使用共享级访问权限和目录文件访问权限就可以实现更大程度的控制能力。

2.6.6 注册表的访问控制

Windows NT 中也有对注册表的访问控制，它遵循与文件系统相类似的模型，表 2-7 详细列出了这些权限。标准的注册表键值权限设置对话框只提供了“读取”（Read）、“完全控制”（Full Control）和“特殊访问”（Special Access）这三个选项。同样地，在注册表键下面创建的子键也将会自动继承其权限，当上一级键的访问权限被修改时，其子键的权限也将被重置。

表 2-7 注册表访问权限

访问类型	说明
Query value	读取子键某值项的设置
Set Value	设置子键的值
Create Subkey	在所选键或子键内创建一个新的键或子键
Enumerate Subkey	确定某键或子键内的所有子键
Notify	接收子键产生的审核通知
Create Link	创建到子键的符号链接
Delete	删除所选键或子键
Write DAC	为所选键修改 DAC
Write Owner	去掉所选键或子键的所有者身份
Read Control	读取所选子键内的安全信息

2.6.7 打印机的访问控制

从某种程度上来说，打印机也是系统中重要的对象。与文件系统、注册表一样，打印机也受到访问控制列表的保护。打印机的访问权限是通过打印机属性对话框进行设置的，其中权限设置对话框显示了用户和用户组访问特定打印机的设置。有以下四种权限可供设置：“无”（No Access）、“打印”（Print）、“管理文档”（Manage Documents）和“完全控制”（Full Control）。表 2-8 显示了每一种权限可以进行的操作。

表 2-8 打印机访问权限

操作	*无*	*打印*	*管理文档*	*完全控制*

打印文档	√		√
文档设置控制		√	√
暂停、继续、重启和删除文档		√	√
改变文档打印顺序			√
暂停、继续、清洁打印机			√
修改打印机属性			√
删除打印机			√
修改打印机权限			√

一台打印机可以在网络上共享（如用作打印服务器），由其 ACL 对本地与远程网络访问进行控制，而没有单独的打印机共享权限控制列表。

2.7 Windows NT 的安全审核

Windows NT 已经开始具有广泛的审核功能了。但是审计子系统默认是关闭的，审计管理员可以在服务器的域用户管理或工作站的用户管理中打开审计并设置审计事件类。审核允许管理员有选择性地跟踪用户和系统的活动，只要建立了安全审核策略，就可以对操作系统内的绝大部分与安全相关的操作进行审核。安全审核策略用来确定 Windows NT 执行的安全性记录的类型和数量，当发生安全审核事件时，就会在系统的安全日志中添加一项条目进行记录以备事后分析（安全日志可在“事件查看器”中查看）。

2.7.1 安全审核的内容

安全审核策略可选择是否对下列类型的安全事件进行审核：

表 2-9 安全审核事件类型

事件	选择	描述
登录和注销	成功	用户成功登录或注销工作站或者用户从网络连接到计算机
	失败	用户企图（但不允许）登录或注销工作站或者用户企图生成网络连接但失败
文件或对象访问	成功	用户成功地访问审核的目录、打印机或文件
	失败	用户企图但未能访问审核的目录、打印机或文件
用户权限使用	成功	用户成功使用用户权限（不包括登录和注销相关的权限）
	失败	用户企图但未能使用用户权限
用户和组管理	成功	用户成功地创建、改变或删除用户账户或成功地更改口令
	失败	用户企图创建、改变或删除用户账户但失败或企图更改口令失败
安全策略改变	成功	用户权限和审核策略成功改变
	失败	对用户权限和审核策略的改变企图失败
重启动、关闭和系统安全性	成功	成功地重启动或关闭计算机，或者发生影响系统安全的事件
	失败	重启动或关闭计算机的企图失败
过程跟踪	成功	详细地跟踪一些事件信息如成功的程序启动、处理复制的一些格式、间接对象访问或过程退出

	失败	详细跟踪一些事件信息，如失败的程序启动、处理复制的一些格式、间接对象访问和过程退出

2.7.2 安全审核数据的存储

系统运行中产生三类日志：系统日志、应用程序日志和安全日志。可使用事件查看器浏览和按条件过滤显示。前两类日志任何人都能查看，它们是系统和应用程序生成的错误警告和其他信息；安全日志则对应审计数据，它只能由审计管理员查看和管理，前提是它必须存于 NTFS 文件系统中，以使 Windows NT 的访问控制生效。审计数据以二进制结构文件形式存储于物理磁盘，每条记录包括事件发生时间、事件源、事件号和所属类别、机器名、用户名和事件本身的详细描述。

2.7.3 客体访问和审计

用户登录到系统时，WinLogon 进程为用户创建访问令牌，包含用户及所属组的安全标识符（SID），作为用户的身份标识。文件等客体则含有自主访问控制列表（DACL），表明谁有权访问；还含有系统访问控制列表(SACL)，标明哪些主体的访问需要被记录。用户进程访问客体对象时，通过 Win32 子系统向核心请求访问服务，核心的安全参考监视器（SRM）将访问令牌与客体的 DACL 进行比较，决定是否拥有访问权限，同时检查客体的 SACL，确定本次访问是否落在既定的审计范围内，是则送至审计子系统。整个过程如图 2-4 所示。

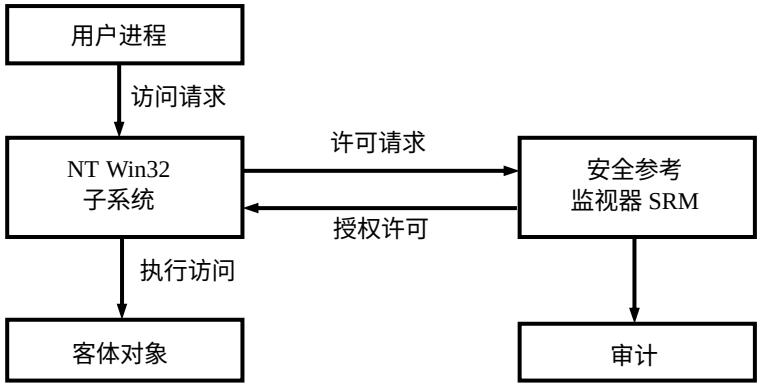

图 2-4 Windows NT 的客体访问和审计示意图