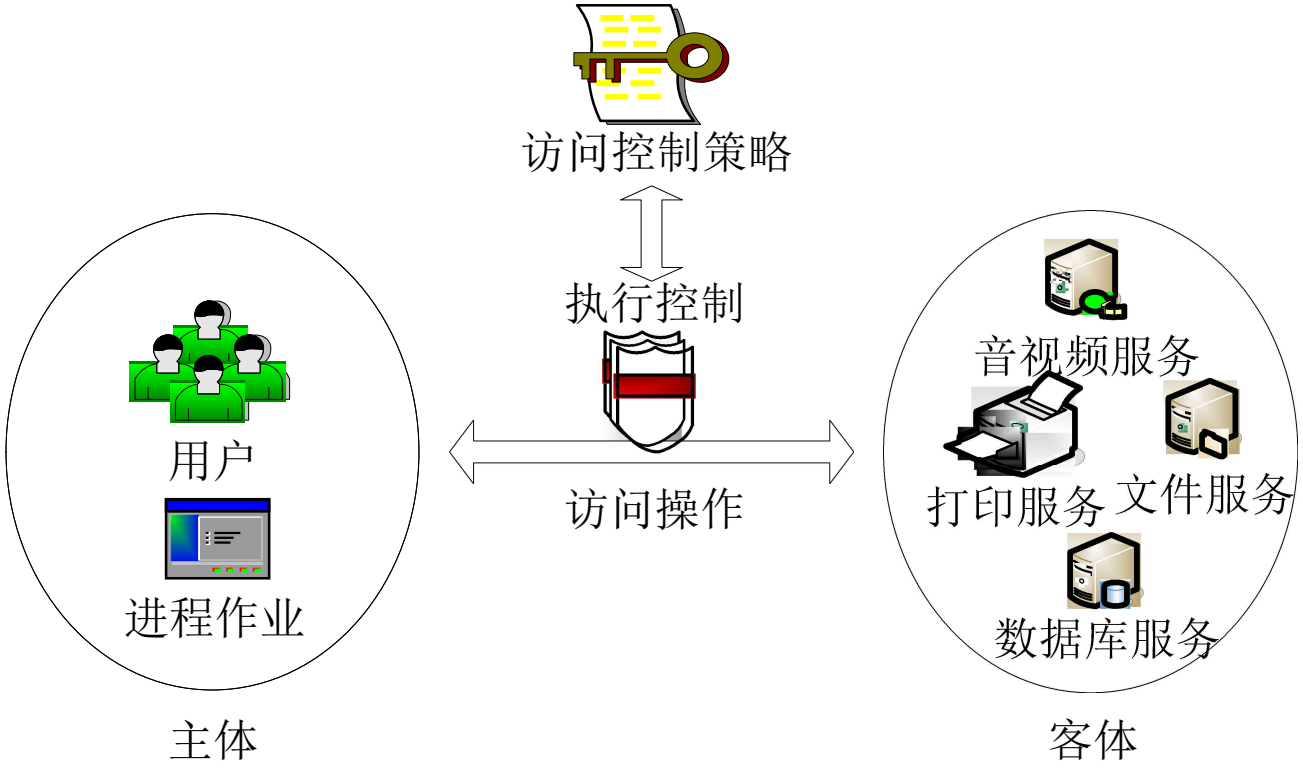
第5章 访问控制

罗文坚

内容

概述

- 身份认证：识别“用户是谁”的问题。
- 访问控制：管理用户对资源的访问。



- 主体对于客体的每一次访问，访问控制系统均要审核该次访问操作是否符合访问控制策略。影响访问控制系统实施效果的首要因素是访问控制策略。

访问控制的基本组成元素

- 主体(Subject)：是指提出访问请求的实体，是动作的发起者，但不一定是动作的执行者。主体可以是用户或其它代理用户行为的实体（如进程、作业和程序等）。

- 客体(Object)：是指可以接受主体访问的被动实体。客体的内涵很广泛，凡是可以被操作的信息、资源、对象都可以认为是客体。

- 访问控制策略（Access Control Policy）：是指主体对客体的操作行为和约束条件的关联集合。简单地讲，访问控制策略是主体对客体的访问规则集合。这个规则集合可以直接决定主体是否可以对客体实施的特定的操作。

内容

访问控制模型

- 访问控制模型是一种从访问控制的角度出发，描述安全系统以及安全机制的方法。
  - 访问控制模型，是对访问控制系统的控制策略、控制实施以及访问授权的形式化描述。

- **1985**年，美国军方提出可信计算机系统评估准则**TCSEC**，其中描述了两种著名的访问控制模型：
  - 自主访问控制**DAC (Discretionary Access Control)**
  - 强制访问控制**MAC (Mandatory Access Control)**

- **1992**年，美国国家标准与技术研究所**(NIST)**的**David Ferraiolo**和**Rick Kuhn**提出一个模型：
  - 基于角色的访问控制**RBAC(Role Based Access Control)**模型

自主访问控制DAC

- 自主访问控制DAC模型是根据自主访问控制策略建立的一种模型。
  - 允许合法用户以用户或用户组的身份来访问系统控制策略许可的客体，同时阻止非授权用户访问客体。
  - 某些用户还可以自主地把自己所拥有的客体的访问权限授予其它用户。

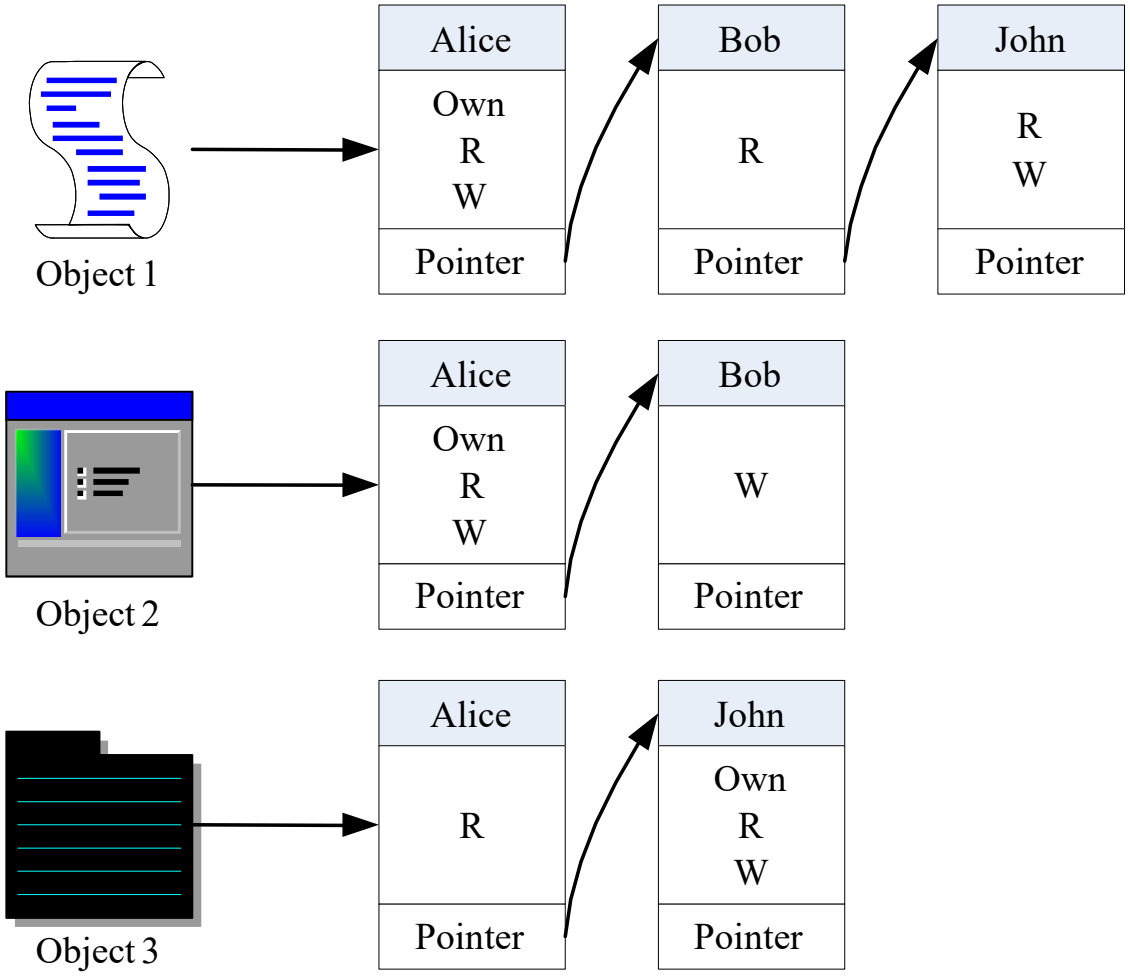
- **UNIX、LINUX**以及**Windows NT**等操作系统都提供自主访问控制的功能。

访问权限信息存储

- 从实现的角度来看，首先要对用户的身份进行鉴别，然后就可以按照访问控制列表所赋予用户的权限允许或限制用户访问客体资源。
- 主体控制权限的修改通常由特权用户或特权用户组实现。

- 特权用户为普通用户分配的访问权限信息的形式：
  1. 访问控制表**ACL**（**Access Control Lists**）
  2. 访问控制能力表**ACCL**（**Access Control Capability Lists**）
  3. 访问控制矩阵**ACM**（**Access Control Matrix**）
- 有关符号：
  - **Own**：管理操作；**R**：读操作；**W**：写操作。
  - 将管理操作与读/写操作分离，是因为管理员也许会对控制规则本身或是文件属性等做修改，即修改**ACL/ACCL/ACM**。

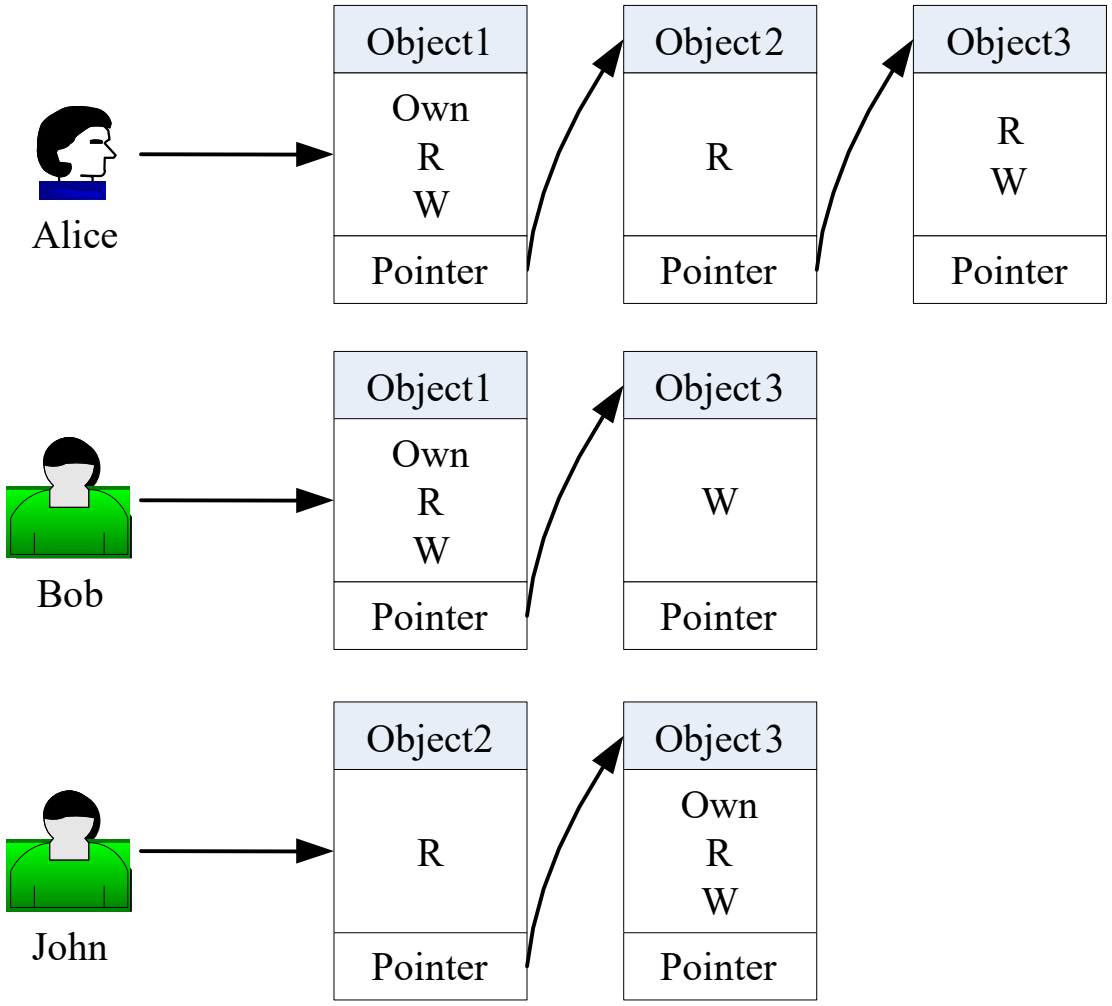
访问控制表ACL



- **ACL**是以客体为中心建立的访问权限表，其优点在于实现简单，系统为每个客体确定一个授权主体的列表。

- 目前大多数**PC**、服务器和主机都使用**ACL**作为访问控制的实现机制。

访问控制能力表ACCL



- **ACCL**是以主体为中心建立的访问权限表。
- 能力，可以解释为请求访问的发起者所拥有的一个授权标签。授权标签表明持有者可以按照某种访问方式访问特定的客体。

访问控制矩阵ACM

- **ACM**通过矩阵形式表示主体用户和客体资源之间的授权关系。
- 如果主体和客体很多，**ACM**会有大量的冗余空间。

主体 \ 客体	Object1	Object2	Object3
Alice	Own，R，W	R	R，W
Bob	R	Own，R，W	
John	R，W		Own，R，W

DAC小结

- DAC为用户提供了灵活的数据访问方式，授权主体（特权用户、特权用户组的成员以及对客体拥有Own权限的主体）均可以完成赋予和回收其他主体对客体资源的访问权限，使得DAC广泛应用在商业和工业环境中。

- DAC允许用户任意传递权限。
  - 例如，没有访问文件file1权限的用户A可能从有访问权限的用户B那里获得访问权限。

- 因此DAC模型提供的安全防护还是相对比较低的，不能为系统提供充分的数据保护。

强制访问控制MAC

- 强制访问控制**MAC**是一种多级访问控制策略。
  - 系统事先给访问主体和受控客体分配不同的安全级别属性。
  - 在实施访问控制时，系统先对访问主体和受控客体的安全级别属性进行比较，再决定访问主体能否访问该受控客体。

- **MAC**模型形式化描述：将访问控制系统中的实体对象分为主体集**S**和客体集**O**，然后定义安全类**SC(x) = < L ， C >**。
  - 其中，**x**为特定的主体或客体。**L**为有层次的安全级别**Level**；**C**为无层次的安全范畴**Category**。
  - 安全范畴**Category**用来划分实体对象的归属，而同属于一个安全范畴的不同实体对象由于具有不同层次的安全级别**L**，因而构成了一定的偏序关系。

强制访问控制MAC

- 访问的四种形式：
  - 向下读（RD，Read Down）：
    - 主体安全级别高于客体信息资源的安全级别时，即**SC(s)≥SC(o)**，允许读操作；
  - 向上读（RU，Read Up）：
    - 主体安全级别低于客体信息资源的安全级别时，即**SC(s)≤SC(o)**，允许读操作；
  - 向下写（WD，Write Down）：
    - **SC(s)≥SC(o)**时，允许写操作；
  - 向上写（WU，Write Up）：
    - **SC(s)≤SC(o)**时，允许写操作。

强制访问控制MAC

- **MAC**通过分级的安全标签实现了信息的单向流动，一直被军方采用。

- **Bell-LaPadula模型：** 只允许向下读、向上写。
  - 可以有效防止机密信息向下级泄露，保护机密性。

- **Biba模型：** 只允许向上读、向下写的特点。
  - 可以有效保护数据的完整性。

MAC信息流安全控制

主体 \ 客体	TS	C	S	U	
TS	R/W	R	R	R	High ↓
C	W	R/W	R	R	↓
S	W	W	R/W	R	↓
U	W	W	W	R/W	Low

- **TS-绝密（Top Secret），C-机密（Confidential），S-秘密（Secret），U-无秘（Unclassified）**

- **符合RD和WU，与Bell-LaPadula模型的信息流控制一致。**

基于角色的访问控制RABC

- MAC模型和DAC模型属于传统的访问控制模型。

- DAC虽然支持用户自主地把自己所拥有的客体的访问权限授予其他用户的这种做法，但当企业的组织结构或是系统的安全需求发生较大变化时，就需要大量繁琐的授权工作，系统管理员的工作势必非常繁重，更主要的是容易发生错误造成一些意想不到的安全漏洞。

- MAC虽然授权形式相对简单，工作量小，但其特点不适合访问控制规则比较复杂的系统。

- RBAC较好地综合了DAC和MAC的特点，基本解决了上述问题。

基于角色的访问控制RABC

- **Group的概念：**一般认为，**Group**是具有某些相同特质的用户集合。

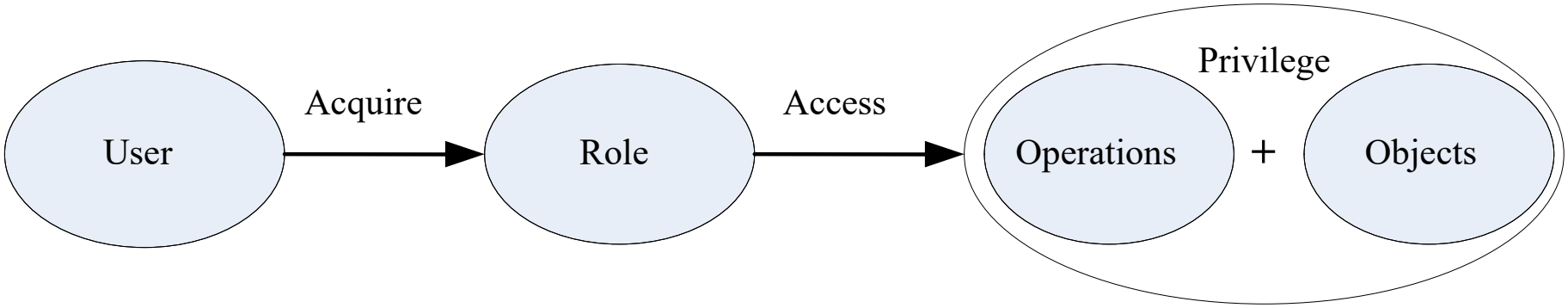
- 在**UNIX**操作系统中**Group**可以被看成是拥有相同访问权限的用户集合。
  - 定义用户组时，会为该组赋予相应的访问权限。
  - 如果一个用户加入了该组，则该用户即具有了该用户组的访问权限。

角色Role的理解

- 角色Role的概念：一个角色是一个与特定工作活动相关联的行为与责任的集合。
- Role不是用户的集合，也就与组Group不同。
- 将一个角色与一个组绑定，则这个组就拥有了该角色拥有的特定工作的行为能力和责任。
- 组Group和用户User都可以看成是角色分配的单位和载体。而一个角色Role可以看成具有某种能力或某些属性的主体的一个抽象。

引入角色Role的目的

- 引入角色的概念，目的是为了隔离用户（Subject，动作主体）与Privilege（权限，指对客体Object的一个访问操作，即操作Operation+客体对象Object）。
  - Role作为一个用户与权限的代理层，所有的授权应该给予Role而不是直接给User或Group。



- RBAC模型的基本思想是将访问权限分配给一定的角色，用户通过饰演不同的角色获得角色所拥有的访问许可权。

例子

- 在一个公司里，用户角色可以定义为经理、会计、出纳员和审计员，具体的权限如下：
  - 经理：允许查询公司的经营状况和财务信息，但不允许修改具体财务信息，必要时可以根据财务凭证支付或收取现金，并编制银行账和现金帐；
  - 会计：允许根据实际情况编制各种财务凭证及账簿，但不包括银行账和现金帐；
  - 出纳员：允许根据财务凭证支付或收取现金，并编制银行账和现金帐；
  - 审计员：允许查询审查公司的经营状况和财务信息，但不允许修改任何账目。

RBAC小结

- RBAC的策略陈述易于被非技术的组织策略者理解，既具有基于身份策略的特征，也具有基于规则策略的特征。
- 在基于组或角色的访问控制中，一个用户可能不只是一个组或角色的成员，有时又可能有所限制。
- 例如，经理可以充当出纳员的角色，但不能负责会计工作，即各角色之间存在相容和相斥的关系。
- RBAC灵活、方便和安全，目前在大型数据库系统的权限管理中得到普遍应用。

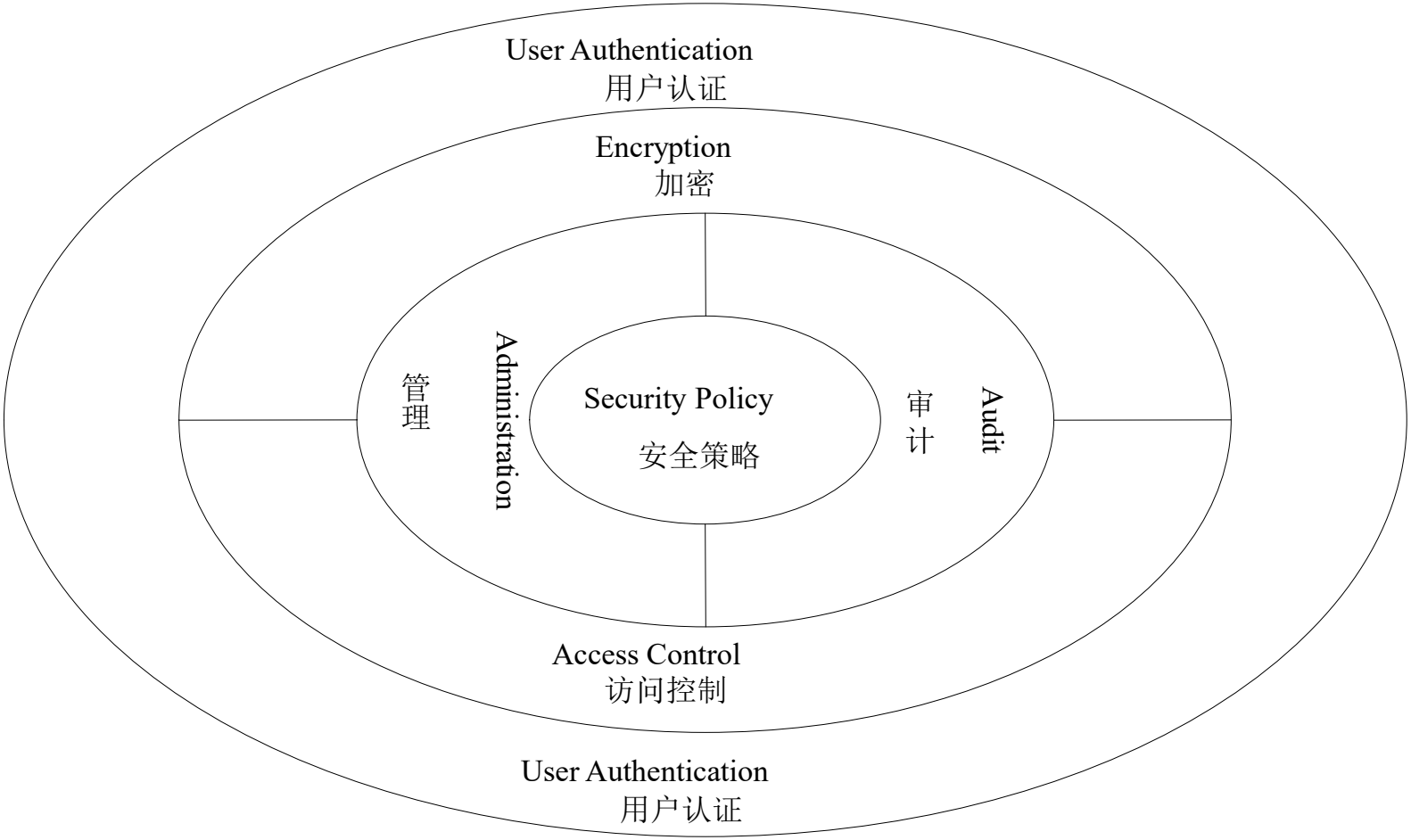
制定访问控制策略的三个基本原则

- 最小特权原则：
  - 是指主体执行操作时，按照主体所需权利的最小化原则分配给主体权力。
  - 最小特权原则的优点是最大限度地限制了主体实施授权行为，可以避免来自突发事件和错误操作带来的危险。
- 最小泄漏原则：
  - 是指主体执行任务时，按照主体所需要知道信息的最小化原则分配给主体访问权限。
- 多级安全策略：
  - 是指主体和客体间的数据流方向必须受到安全等级的约束。多级安全策略的优点是避免敏感信息的扩散。
  - 对于具有安全级别的信息资源，只有安全级别比它高的主体才能够对其访问。

内容

Windows系统安全体系结构

- **Windows系统采用层次性的安全架构。**



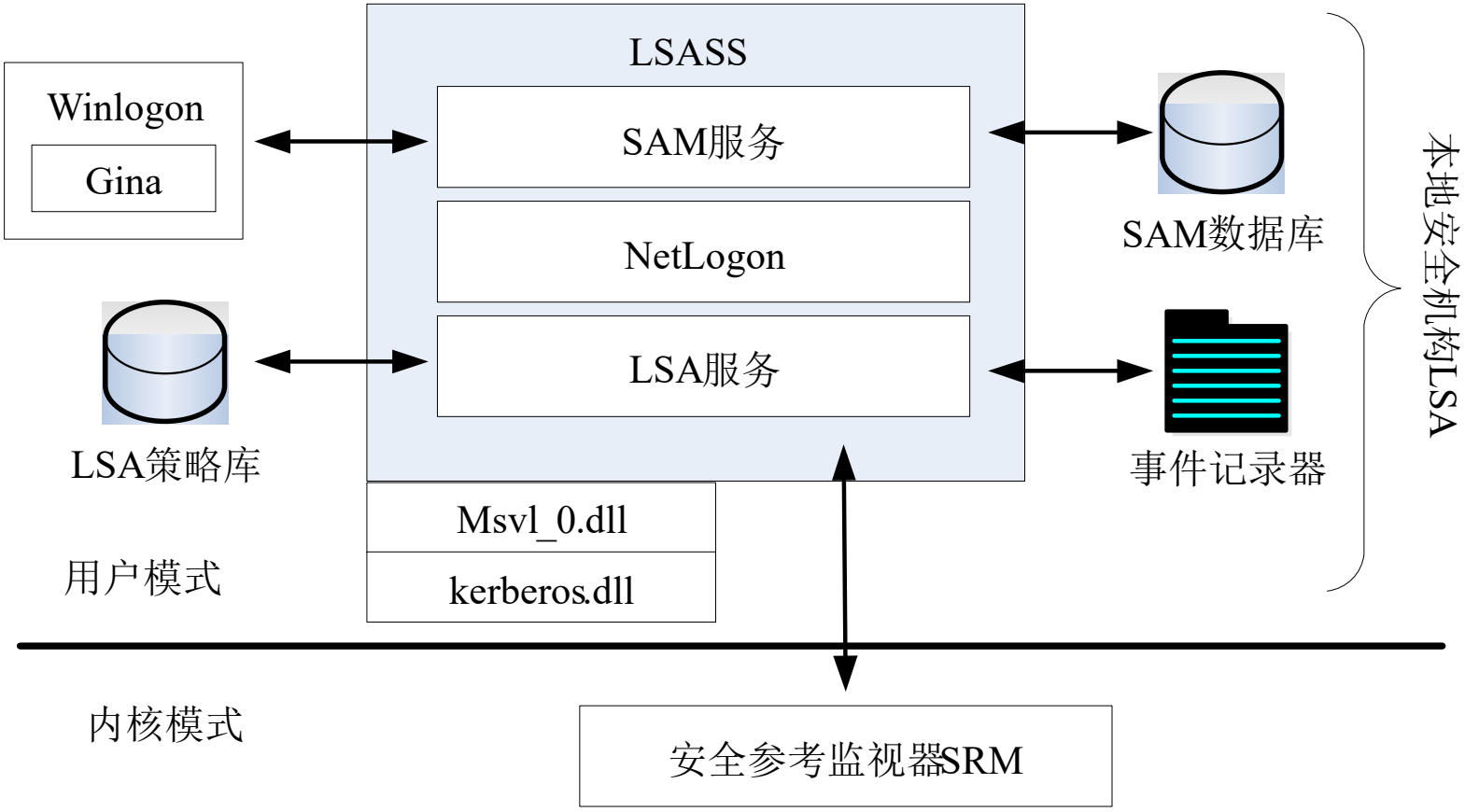
安全主体

- Windows系统的安全性主要围绕安全主体展开，保护其安全性。

- 安全主体主要包括用户、组、计算机以及域等。
  - 用户是Windows系统中操作计算机资源的主体，每个用户必须先行加入Windows系统，并被指定唯一的账户；
  - 组是用户账户集合的一种容器，同时组也被赋予了一定的访问权限，放到一个组中的所有账户都会继承这些权限；
  - 计算机是指一台独立计算机的全部主体和客体资源的集合，也是Windows系统管理的独立单元；
  - 域是使用域控制器(DC, Domain Controller)进行集中管理的网络。域控制器是共享的域信息的安全存储仓库，同时也作为域用户认证的中央控制机构。

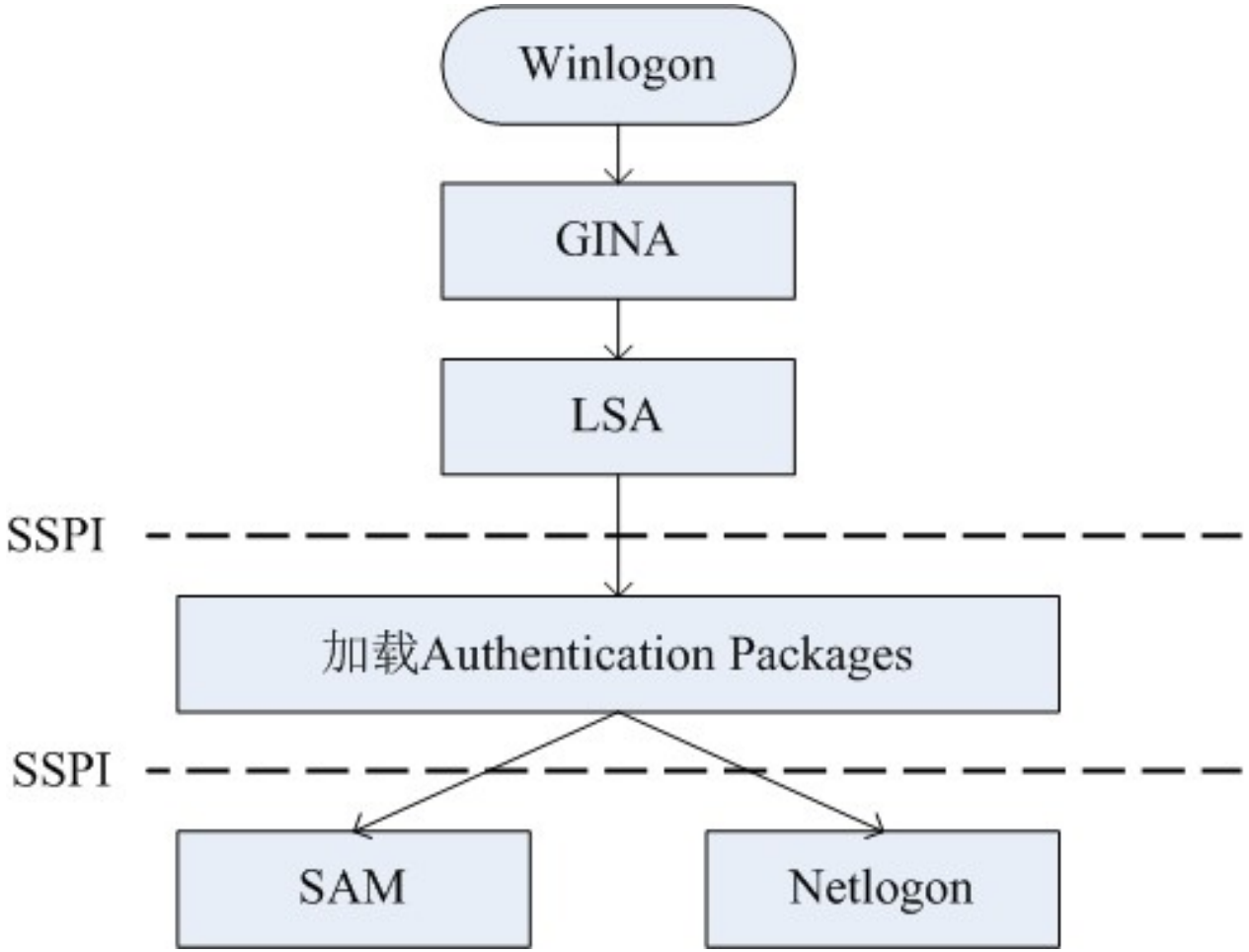
安全子系统

- 安全子系统既可以用于工作站，也可以用于服务器，区别在于服务器版的用户账户数据库可以用于整个域，而工作站版的数据库只能本地使用。



Windows登录认证流程

- **SSPI：Security Support Provider Interface。**



Windows系统的访问控制

- 访问控制模块的组成
  - 访问令牌（Access Token）和安全描述符（Security Descriptor），它们分别由访问者和被访问者持有。通过访问令牌和安全描述符的内容，Windows可以确定持有令牌的访问者能否访问持有安全描述符的对象。

- 访问控制的基本控制单元“账户”。
  - 账户是一种参考上下文(context)，是一个具有特定约束条件的容器，也可以理解为背景环境。
  - 操作系统在这个上下文描述符上运行该账户的大部分代码。
  - 那些在登录之前就运行的代码（例如服务）运行在一个账户（特殊的本地系统账户SYSTEM）的上下文中。

安全标识符SID

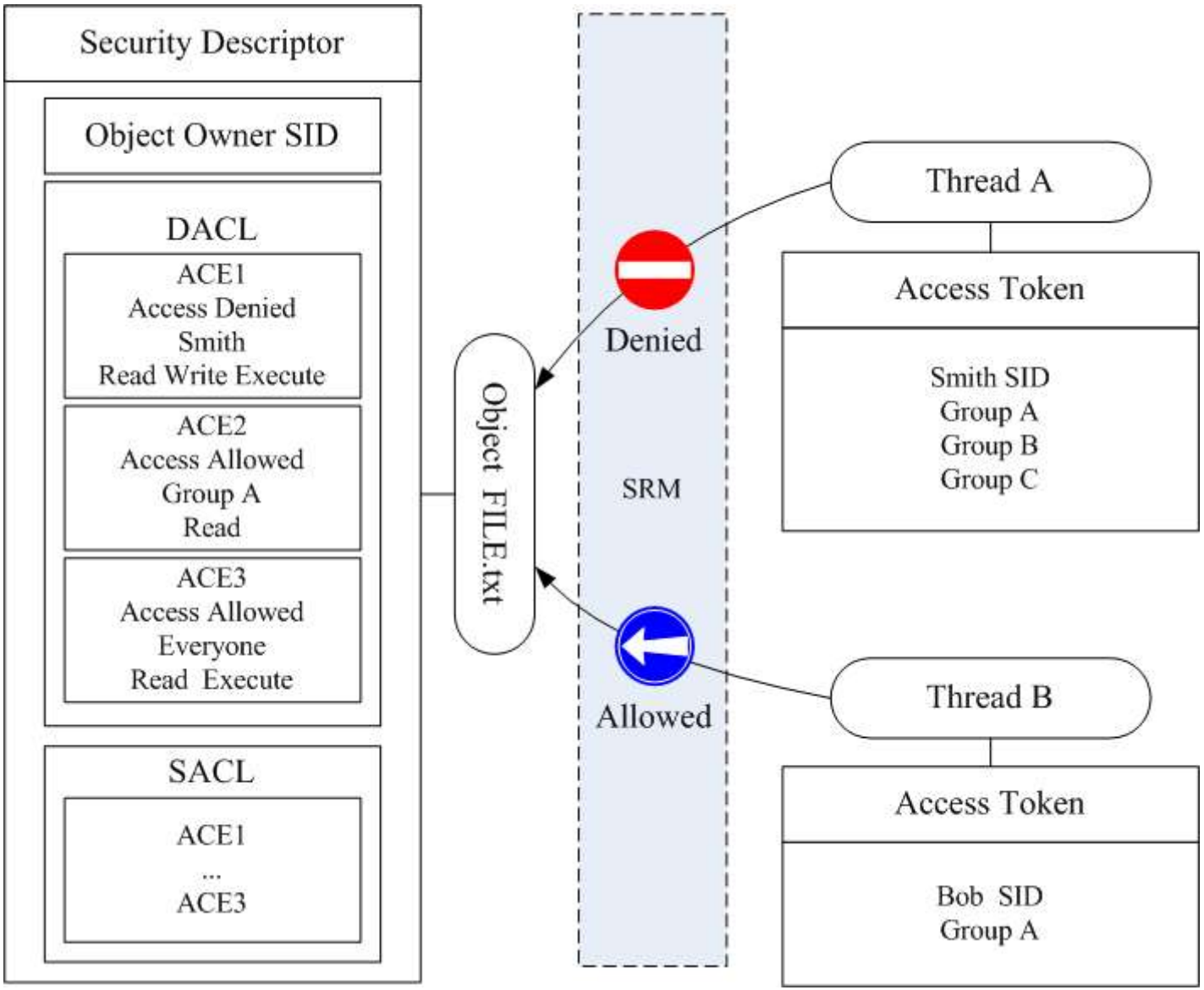
- Windows中的每个账户或账户组都有一个安全标识符SID（Security Identity）。
- Administrator、Users等账户或者账户组在Windows内部均使用SID来标识的。
- 每个SID在同一个系统中都是唯一的。
  - 例如S-1-5-21-1507001333-1204550764-1011284298-500就是一个完整的SID。
  - 第一个数字（本例中的1）是修订版本编号；
  - 第二个数字是标识符颁发机构代码（Windows 2000为5）；
  - 4个子颁发机构代码；
  - 相对标识符RID（Relative Identifier）。 RID 500代表Administrator账户， RID 501是Guest账户。从1000开始的RID代表用户账户。

访问令牌

- 每个访问令牌都与特定的**Windows**账户相关联，访问令牌包含该帐户的**SID**、所属组的**SID**以及帐户的特权信息。

```
Microsoft Windows XP [版本 5.1.2600]
(C) 版权所有 1985-2001 Microsoft Corp.
C:\>whoami /all
[User]   = "Smith\Administrator"  S-1-5-21-2000478354-842925246-1202660629-500
[Group 1] = " Smith \None"  S-1-5-21-2000478354-842925246-1202660629-513
[Group 2] = "Everyone"  S-1-1-0
[Group 3] = " Smith \Debugger Users"  S-1-5-21-2000478354-842925246-
1202660629-1004
[Group  4] = "BUILTIN\Administrators"  S-1-5-32-544
[Group  5] = "BUILTIN\Users"  S-1-5-32-545
[Group  6] = "NT AUTHORITY\INTERACTIVE"  S-1-5-4
[Group  7] = "NT AUTHORITY\Authenticated Users"  S-1-5-11
[Group  8] = "LOCAL"  S-1-2-0
……
```

Window 访问控制


Security Descriptor
Object Owner SID
DACL
ACE1
Access Denied
Smith
Read Write Execute
ACE2
Access Allowed
Group A
Read
ACE3
Access Allowed
Everyone
Read Execute
SACL
ACE1
...
ACE3
Object FILE.txt
Denied
SRM
Allowed
Thread A
Access Token
Smith SID
Group A
Group B
Group C
Thread B
Access Token
Bob SID
Group A

活动目录与组策略

- Windows的网络管理中有两个非常重要的管理技术，即活动目录AD （Active Directory）和组策略GP （Group Policy），它们的协调工作有效提升了Windows网络的安全性。

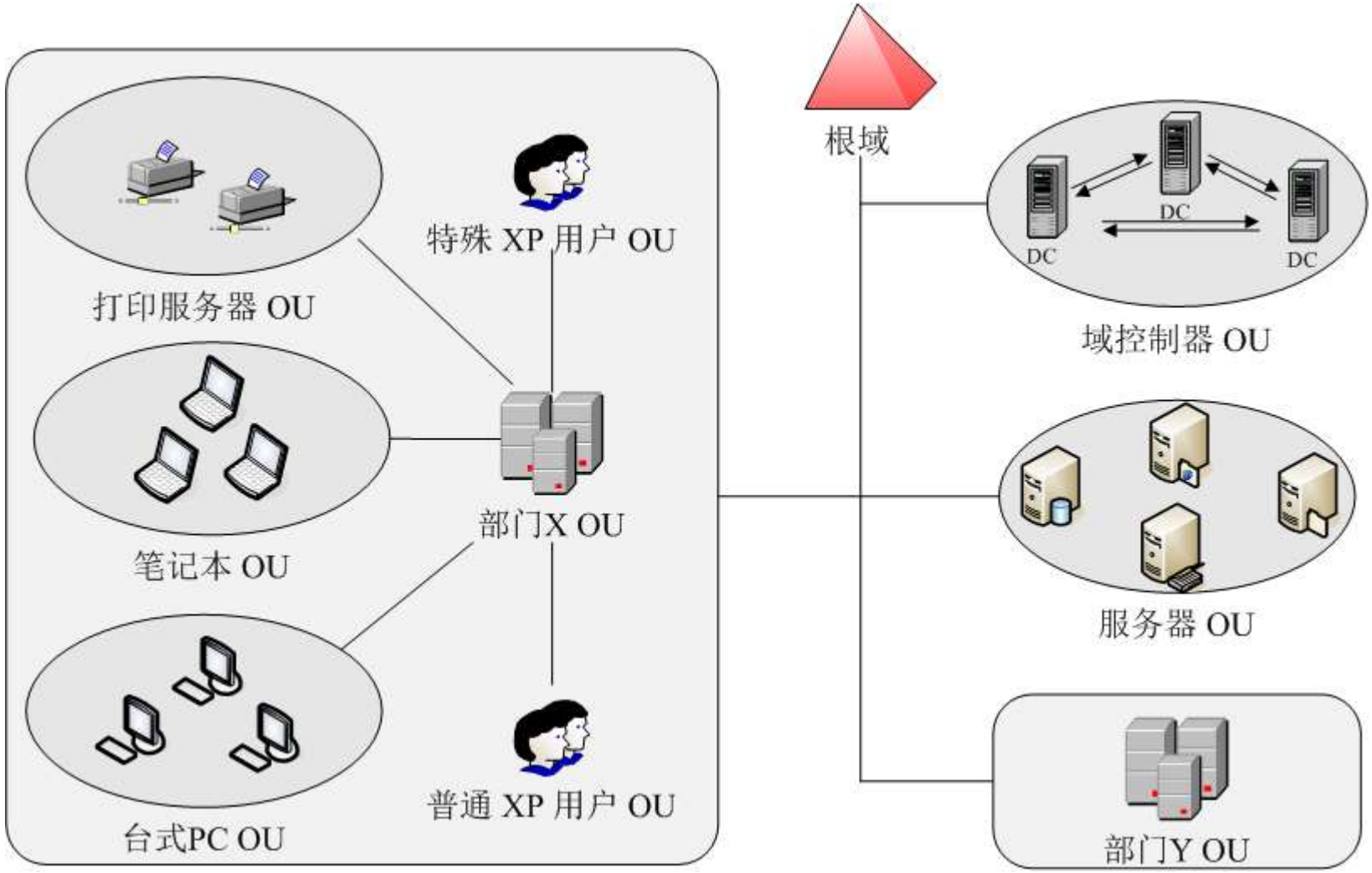
- 活动目录AD是一个面向网络对象管理的综合目录服务。

- 网络对象包括用户、用户组、计算机、打印机、应用服务器、域、组织单元（OU）以及安全策略等。

- AD 提供的是各种网络对象的索引集合，也可以看作是数据存储的视图，将分散的网络对象有效地组织起来，建立网络对象索引目录，并存储在活动目录的数据库内。

活动目录AD的管理划分

- AD把整个域作为一个完整的目录来进行管理。



组策略GP

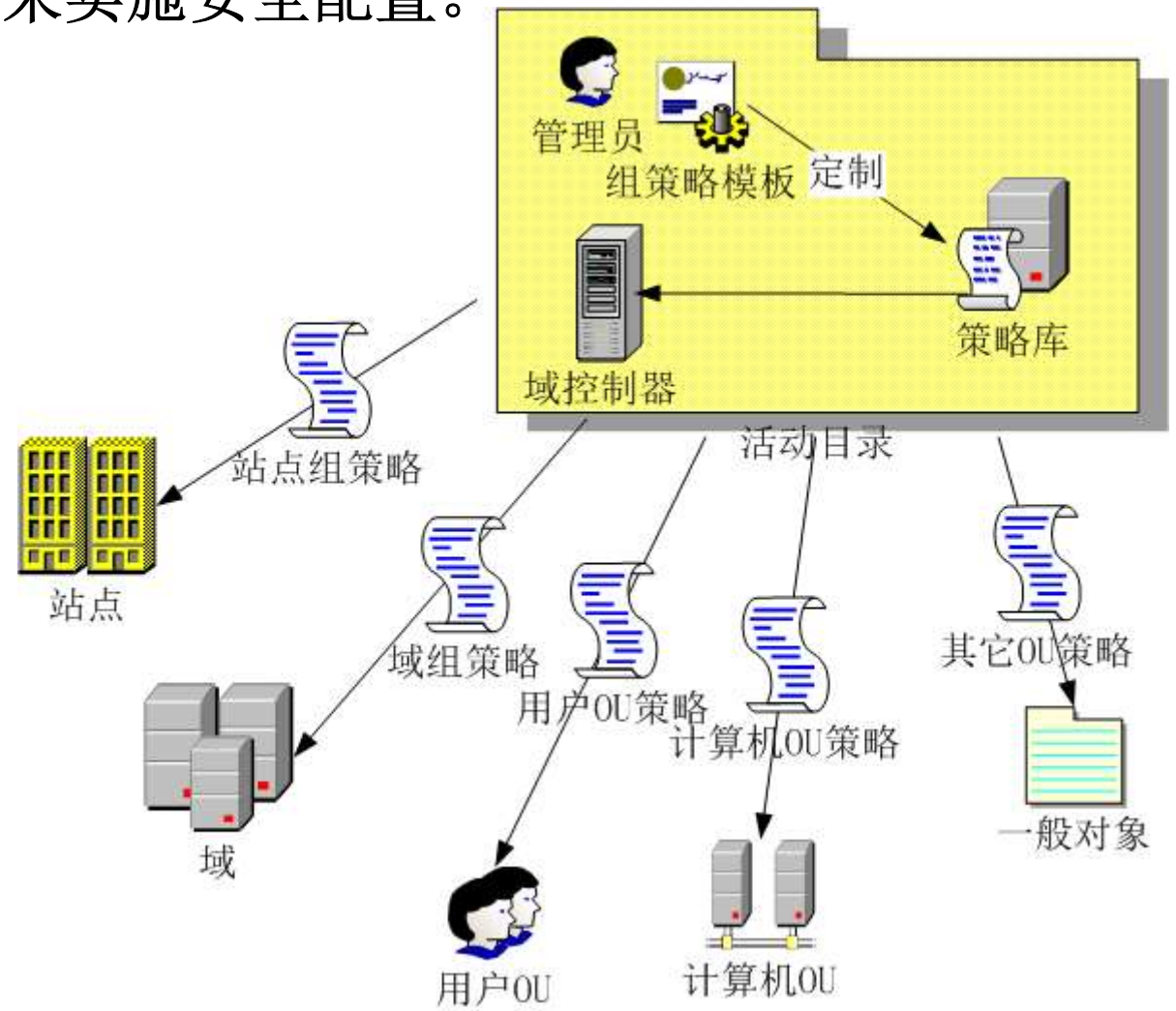
- 活动目录AD是Windows网络中重要的安全管理平台，组策略GP是其安全性的重要体现。

- 组策略可以理解为依据特定的用户或计算机的安全需求定制的安全配置规则。
  - 管理员针对每个组织单元OU定制不同的组策略，并将这些组策略存储在活动目录的相关数据库内，可以强制推送到客户端实施组策略。

- 活动目录AD可以使用组策略命令来通知和改变已经登录的用户的组策略，并执行相关安全配置。

组策略工作流程

- 用户完成网络登录后，就会受到**AD**直接控制管理，依据所在**OU**的**GP**来实施安全配置。



组策略的实施

- 注册表是Windows系统中保存系统应用软件配置的数据库。
- 很多配置都是可以自定义设置的，但这些配置发布在注册表的各个角落。如果是手工配置，可想是多么困难和繁琐。
- 组策略可以将系统中重要的配置功能汇集成一个配置集合，管理人员通过配置并实施组策略，达到直接管理计算机的目的。
- 简单点说，实施组策略就是修改注册表中的相关配置。

组策略和活动目录AD配合

- 组策略分为基于活动目录的和基于本地计算机的两种：
  - AD组策略存储在域控制器上活动目录AD的数据库中，它的定制实施由域管理员来执行；而本地组策略存放在本地计算机内，由本地管理员来定制实施。
  - AD组策略实施的对象是整个组织单元OU；本地组策略只负责本地计算机。

- 组策略和活动目录AD配合：
  - 组策略可以部署在OU、站点或域的范围内，也可以部署在本地计算机上。
  - 部署在本地计算机时，组策略不能发挥其全部功能，只有和AD配合，组策略才可以发挥出全部潜力。

组策略的主要工作

① 部署软件

② 设置用户权力

③ 软件限制策略

- 管理员可以通过配置组策略，限制某个用户只能运行特定的程序或执行特定的任务。

④ 控制系统设置：

- 允许管理员统一部署网络用户的**Windows**服务。

⑤ 设置登录、注销、关机、开机脚本。

⑥ 通用桌面控制

⑦ 安全策略

⑧ 重定向文件夹

⑨ 基于注册表的策略设置

作业

- 课后阅读**5.3**节。

1. 习题**2**（**1**）：**DAC**与**MAC**有什么不同？
2. 习题**2**（**3**）：角色与组的区别是什么？
3. 系统**2**（**5**）：**Windows**系统的安全体系结构包括哪些内容？