CA Access Control

端点管理指南：用于 Windows

12.6.01

第三方通知

包含 AIX(TM)、Java(TM) 2 技术版、1.4 版模块的 IBM(R) 32 位运行时环境

示例脚本和示例 SDK 代码

CA Access Control 产品随附的示例脚本和示例 SDK 代码均“按原样”提供，仅供参考之用。 在特定环境下，可能需要对它们进行调整，而且在生产系统中部署它们之前，未经事先测试和验证不应将其用于生产。

CA Technologies 不向这些示例提供支持，而且对于这些脚本可能会引起的任何错误概不负责。

CA Technologies 产品引用

本文档引用以下 CA Technologies 产品：

- CA Access Control 企业版
- CA Access Control
- CA Single Sign-On (CA SSO)
- CA Top Secret®
- CA ACF2™
- CA Audit
- CA Network and Systems Management（CA NSM，以前为 Unicenter NSM 和 Unicenter TNG）
- CA Software Delivery（以前为 Unicenter Software Delivery）
- CA Service Desk（以前为 Unicenter Service Desk）
- User Activity Reporting（以前是 CA Enterprise Log Manager）
- CA Identity Manager

文档约定

CA Access Control 文档使用以下约定：

格式	含义
等宽字体	代码或程序输出
斜体	重点或新术语
粗体	必须完全按照显示内容键入的文本
正斜杠 (/)	用于描述 UNIX 和 Windows 路径的独立于平台的目录分隔符

文档在解释命令语法和用户输入（以等宽字体显示）时还会使用以下特殊约定：

格式	含义
斜体	您必须提供的信息
用方括号括起来 ([])	可选运算符

格式	含义
用大括号括起来 ({})	强制运算符集
用管道符 (|) 分隔的选项。	分隔可选运算符（选择一项）。 例如：下面的示例*既*可以表示用户名，*也*可以表示组名： `{username|groupname}`
…	指明前面的项或项组可以重复
下划线	默认值
前面带空格的行尾反斜杠 (\)	有时，在本指南中，一行无法容纳一个命令。 在这些情况下，行结尾处的空格加上反斜杠 (\) 就表示该命令延续到下一行。 **注意：**请勿复制反斜杠字符，并且请省略换行符。 这些不是实际命令语法的一部分。

示例：命令表示法约定

下面的代码说明了在本指南中使用命令约定的方式：

```
ruler className [props({all|{propertyName1[,propertyName2]...})]
```

在该示例中：

- 命令名称 (ruler) 以常规等宽字体显示，必须按照显示内容键入。
- *className* 选项以斜体显示，因为它是一个类名（例如 USER）的占位符。
- 即使不带有方括号中的第二部分，您也可以运行该命令，方括号表示该运算符是可选的。
- 使用可选参数 (props) 时，可以选择关键字 *all*，也可以指定一个或多个属性名（以逗号分隔）。

文件位置约定

CA Access Control 文档使用以下文件位置约定：

- *ACInstallDir*—默认 CA Access Control 安装目录。
  - Windows—C:\Program Files\CA\AccessControl\
  - UNIX—/opt/CA/AccessControl/
- *ACSharedDir*—CA Access Control for UNIX 使用的默认目录。
  - UNIX—/opt/CA/AccessControlShared

- *ACServerInstallDir*—默认 CA Access Control 企业管理 安装目录。
  - /opt/CA/AccessControlServer
- *DistServerInstallDir*—默认分发服务器安装目录。
  - /opt/CA/DistributionServer
- *JBoss_HOME*—默认 JBoss 安装目录。
  - /opt/jboss-4.2.3.GA

联系技术支持

要获取在线技术帮助以及办公地址、主要服务时间和电话号码的完整列表，请联系技术支持：http://www.ca.com/worldwide。

文档更改

从最新版本以来对该文档进行了以下更新：

- 管理资源－更新章节包括以下更改：
  - 内部文件规则 (Windows)

目录

第 1 章： 简介

此部分包含以下主题：

关于本指南

本指南将说明 CA Access Control for Windows 所使用的概念，CA Access Control for Windows 产品用于为开放式系统提供总体安全解决方案。 本指南还将说明 Windows 端点管理任务和概念。

本指南也随 CA Access Control 企业版 提供，CA Access Control 企业版 提供企业管理和报告功能，以及高级策略管理功能。

为了简化术语，在本指南中我们将此产品称为 CA Access Control。

使用本指南的用户

本指南是为负责实现和维护受 CA Access Control 保护的环境的安全管理员和系统管理员而编写的。

第 2 章： 管理端点

CA Access Control 是动态绑定到操作系统的软件产品，是开放系统的主动、全面的安全软件解决方案。 用户每次请求有关安全的操作（例如，打开文件、替换用户 ID 或获取网络服务）时，CA Access Control 会实时截获该事件并评估其有效性，然后才将控制权转交给标准的操作系统 (OS) 功能。

此部分包含以下主题：

什么是 CA Access Control？

CA Access Control 为您提供用来管理本地平台安全性的强大工具，从而能够实施可完全根据企业安全要求自定义的安全策略。 CA Access Control 可以为本地操作系统中可用用户、组和资源之外的用户、组和资源提供安全保护， 在整个组织范围内集中管理安全性，并将您的 Windows 和 UNIX 安全策略集成到一个异类环境中。

保护的对象是什么？

CA Access Control 保护下列实体：

- **文件**

  用户是否有权访问特定文件？

  CA Access Control 限制用户访问文件的能力。 您可以给予用户一种或多种访问权限，例如读取、写入、执行、删除和重命名。 访问权限的指定可以与单个文件相关，也可以与一组命名相似的文件相关。

- **终端**

  用户是否有权使用特定终端？

  该检查是在登录过程中完成的。 在 CA Access Control 数据库中，可以定义单个终端和终端组及其访问规则（即描述允许哪些用户或用户组使用终端或终端组）。 终端保护确保未经授权的终端或工作站不能用来登录到授予强大权限的用户帐户。

- **登录时间**

  用户是否有权在特定日期的特定时间登录？

  大多数用户只在工作日和工作时间使用其工作站；工作日和工作时间登录限制以及节假日限制，是为了防止黑客和其他未经授权的访问者的登录。

- **TCP/IP**

  另一个工作站是否有权从本地计算机接收 TCP/IP 服务？ 另一个工作站是否有权向本地计算机提供 TCP/IP 服务？ 是否允许另一个工作站从本地工作站的每个用户接收服务？

  开放系统（计算机和网络都开放的系统）的优点也是缺点。 一旦计算机连接到外面的世界，您就无法确保谁进入系统以及外来用户可以进行何种破坏（无论有意还是无意）。 CA Access Control 提供“防火墙”，可以防止本地工作站和服务器向未知工作站提供服务。

- **多个登录权限**

  是否允许用户从第二个终端登录？

  术语*并发登录*指的是用户从多个终端登录到系统的能力。 CA Access Control 可以阻止用户多次登录。 这可以防止入侵者登录到已经登录的用户的帐户。

- **用户定义的实体**

  可以定义和保护常规实体（例如 TCP/IP 服务和终端）和功能实体（也称为*抽象*对象；例如在数据库中执行事务和访问记录）。

- **管理员权限方面**

  CA Access Control 提供了向操作员指派超级用户权限而又同时限制超级用户帐户权限的方法。

- **注册表键**

  用户是否有权访问特定注册表键？

  CA Access Control 限制用户访问注册表键。 您可以授予用户一种或多种访问权限，例如读取 (READ)、写入 (WRITE) 和删除 (DELETE)。 可以指定与单个注册表键或一组命名相似的注册表键相关的访问权限。

- **程序**

  特定程序是否可受托？ 用户是否有权调用它？ 用户是否可以使用程序来访问特定资源？

  安全管理员可以对程序进行测试，确保它们不包含任何可用来获得未经授权的访问的安全漏洞。 通过测试并被视为安全的程序定义为受托程序。 CA Access Control 自我保护模块（也称为 **Watchdog**）知道哪个程序在特定时间处于控制之下，并检查该程序在被归为受托程序之后是否经过修改或移动。 如果受托程序经过修改或移动，则该程序不再被视为受托程序，CA Access Control 将不允许它运行。

另外，CA Access Control 可以防止各种故意的和偶然的威胁，包括：

- **终止尝试**

  CA Access Control 可以用来保护关键服务器和服务或后台程序，防止终止尝试。

- **密码攻击**

  CA Access Control 防止各种密码攻击、强制您的站点实施密码定义策略并检测入侵尝试。

- **密码缺点**

  CA Access Control 策略描述了强制用户创建和使用高质量密码的规则。 为了确保用户创建和使用可接受的密码，CA Access Control 可以设置密码的最长和最短使用期限、限制某些字词、禁用重复字符并强制遵守其他限制。 密码使用期限不能太长。

- **帐户管理**

  CA Access Control 策略确保正确处理睡眠帐户。

如何保护它？

CA Access Control 在操作系统完成初始化后立即启动。 CA Access Control 将 hook 放置在必须保护的系统服务中。 这样，便可以在执行服务之前将控制权传递给 CA Access Control。 CA Access Control 决定是否应将该服务授予用户。

例如，用户可以尝试访问受 CA Access Control 保护的资源。 该访问请求生成了对内核的系统调用，从而可以打开资源。 CA Access Control 截获系统调用，并决定是否授予访问权限。 如果授予权限，则 CA Access Control 将控制权转交给常规系统服务；如果 CA Access Control 拒绝权限，它会将标准权限拒绝错误代码返回到激活系统调用的程序，系统调用即结束。

授权决定基于数据库中定义的访问规则和策略。 数据库描述了两种对象：访问者和资源。 *访问者*是用户和组。 *资源*是要保护的对象，例如文件和服务。 数据库中的每个记录都描述了访问者或资源。

每个对象都属于一个类 - 同类对象的集合。 例如，TERMINAL 是包含 CA Access Control 所保护的终端（工作站）对象的类。

类激活

有关类状态的信息（即，该类是活动还是不活动）会保留在数据库中。 CA Access Control 检查数据库中的状态，并截获每次访问资源的尝试。 如果该类是非活动的，则允许访问，无需进一步检查权限。

CA Access Control 会在引擎启动以及用户更改类活动状态时发出活动类列表。 如果类是不活动的，则不会拦截对资源的访问，这将减少一定的开销。

访问者元素

每个用户由一个访问者元素 (ACEE) 代表（该元素是数据库中的用户记录在内存中的反映）。 CA Access Control 在登录过程中构建访问者元素。 访问者元素与用户的进程相关。 无论何时进程请求 CA Access Control 所保护的系统服务，或发出访问资源的暗示请求，CA Access Control 都会访问该资源的记录。 然后，它将确定以前创建的访问者元素中的信息（例如用户的安全级别、模式和组）是否允许用户访问该资源。

扩展本地安全性

下列 CA Access Control 功能可以扩展本地安全性。

超级用户帐户限制

管理操作系统的用户通常是在系统安装期间自动创建的预定义帐户的成员，例如 UNIX 系统中的 root 帐户及 Windows 系统中的管理员帐户。 每个预定义帐户的存在都是为了执行一组特定的系统功能。

担当 root 用户或管理员角色的用户可以执行多种任务，从创建、删除和修改用户到锁定、重新配置和关闭服务器。

这些操作系统中的主要安全风险之一是未经授权的用户可以获得对这些帐户的控制权。 如果发生这种情况，未经授权的用户可能会对系统造成巨大破坏。

通过 CA Access Control 可以限制授予这些帐户的权限，并限制作为用户组（这些帐户是该组成员）成员的用户的权限。 这将会减少操作系统的漏洞。

CA Access Control 管理员

安装 CA Access Control 时，系统会要求您命名一个或多个 CA Access Control 管理员。 CA Access Control 管理员有权修改整个或部分规则数据库。 您应该至少有一个具有完全权限的管理员。 该管理员可以自由修改或创建访问规则，并可以指定其他级别的管理员。

为系统定义用户后，可以通过向其他用户分配 ADMIN 属性来向他们分配管理权限。

注意：具有 ADMIN 属性的用户拥有强大的权限。 因此，应该严格限制 ADMIN 用户的数量。 在您设置一个或多个 CA Access Control 安全管理员后，将本地超级用户和 ADMIN 的角色分开，从超级用户删除 ADMIN 属性，这不失为一个好策略。

由于始终需要至少一个具有数据库管理权限的用户，因此 CA Access Control 不允许删除最后一个具有 ADMIN 属性的用户。

如果希望所有 CA Access Control 管理员都可以从该工作站管理其他主机，请确保该主机上的数据库中的规则向他们授予从该工作站读取和写入的访问权限。

子管理

CA Access Control 包含 *子层管理*功能。 通过该功能，管理员可授予特定权限，常规用户使用该权限可管理特定类。 这些用户则称为子管理员。

例如，可以允许特定用户只管理用户和组。

还可以通过不仅为特定类还为这些类中的特定对象授予访问权限，来指定更高级别的子层管理。

常规用户的管理权限

您可以使用 CA Access Control 向普通用户（即非管理员用户）授予必需的权限，以便这些用户不必成为管理员组的成员即可执行管理任务。 能够以这种精细方式通过授予管理权限指派任务是 CA Access Control 最重要的优点之一。

- SUDO 类中的记录存储了命令脚本，允许用户使用借来的权限运行该脚本。
- 数据属性值是命令脚本。 通过将可选脚本参数值添加到该值，可以对该值进行修改。
- SUDO 类中的每个记录都标识一个命令，一个用户可以借用另一个用户的权限来执行该命令。
- SUDO 类记录的关键字是 SUDO 记录的名称。 当用户执行 SUDO 记录中的命令时，会使用该名称来代替命令名称。

增强的文件保护

CA Access Control 支持逻辑文件名格式和绝对文件名格式。 例如，如果文件 foo.txt 位于逻辑驱动器 D 的目录 \tmp 下，且逻辑名“D:”分配给物理磁盘 1 的分区 0，则您可以使用逻辑文件名或绝对文件名将文件定义到 CA Access Control 数据库：

```
nr file D:\tmp\foo.txt
```

或

```
nr file \Device\HardDisk1\Partition1\tmp\foo.txt
```

注意：如果使用第二个格式，则即使更改了磁盘的逻辑名称，文件仍受保护。 对于 CA Access Control 常规文件保护，还支持绝对文件名格式。

CA Access Control 保护当前在支持的 Windows 操作系统中使用的所有文件系统。 两个最常用的文件系统是 Windows 文件系统 (NTFS) 和文件分配表 (FAT)。 CA Access Control 也支持 CDFS（专用于 CD 的文件系统）。

CA Access Control 提供了文件分配表 (FAT) 的总体安全解决方案，以及包括 NTFS 和 CDFS 的其他文件系统的额外安全层。

一般文件保护

CA Access Control 支持逻辑文件名和绝对文件名。 对于 CA Access Control 常规文件保护，还支持绝对文件名格式。

利用普通类别文件保护，可以保护所有符合指定的通配符模式（一般表达式）的文件。 名称与指定通配符模式匹配的任何资源都受指定的一般访问规则的保护。 通过 CA Access Control，您可以对文件进行常规保护。

如果一个资源与多个通用访问规则匹配，则 CA Access Control 将选择与该文件最匹配的规则。

使用一般文件保护，只须定义少数几个安全规则就可以保护需要保护的许多文件。

密码保护

本地 Windows 安全性可以通过许多方式保护密码和加强密码质量。 Windows 提供下列功能：

- 强制执行密码最长时限
- 强制执行最小密码长度
- 最多保存 24 代用户密码
- 重复登录失败后锁定帐户
- 强制用户登录到 Windows 后才能更改密码

CA Access Control 也会强制实施相同的规则，但它是通过其自身的独特机制来强制实施。 此外，CA Access Control 实施与大型机计算机的双向密码同步。

增强的密码保护

本地 Windows 安全对用户密码加以大量保护 (p. 23)。 不过，CA Access Control 大大扩展了密码保护，这样，黑客成功窃取密码的可能性大大减少。

使用 CA Access Control 时，可以创建其他规则来强制用户选择更加安全可靠的密码。 例如，可以要求用户至少选择一定数目的字母、数字、特殊、小写或大写字符。 也可以确保用户选择的新密码中不包含被替换的密码，并且被替换的密码中也不包含该新密码。

程序通路

*程序通路*是一种与文件关联的访问规则，该规则要求只能通过特定程序访问文件。 程序通路大大提高了敏感文件的安全性。 通过 CA Access Control 可以使用程序通路来为系统中的文件提供额外的保护。

B1 安全级别认证

CA Access Control 包括下列 B1“橙皮书”功能：安全级别、安全类别和安全标签。

- 可以向数据库中的访问者和资源分配一个*安全级别*。 安全级别是 1 至 255 之间的整数。 只有当访问者拥有的安全级别等于或大于分配给资源的安全级别时，访问者才可以访问资源。
- 数据库中的访问者和资源可以属于一个或多个*安全类别*。 只有当访问者属于分配给资源的所有安全类别时，访问者才能访问该资源。
- *安全标签*是将特定安全级别与一组（零个或更多）安全类别相关联的名称。 将用户分配给某一安全标签会授予该用户与该安全标签相关联的安全级别和所有安全类别。

注意：有关 B1 橙皮书功能的详细信息，请参阅*《实施指南》*。

设置审核过程

根据数据库中定义的审核规则，CA Access Control 可保留拒绝访问和授权访问事件的审核记录。 是否记录某个事件的决策是基于以下规则做出的：

- 每个访问者和资源都有 AUDIT 属性，可以设置该属性以表示访问成功还是失败或者是否应该记录成功和失败；另外，访问者的 AUDIT 属性可以表示登录成功还是失败或者是否应该记录成功和失败。
- 如果资源或访问者具有 AUDIT(ALL) 属性，则无论访问成功还是失败，均将记录与 CA Access Control 所保护的资源相关的所有事件。

- 如果对 CA Access Control 所保护资源的访问成功，且用户或资源具有 AUDIT(SUCCESS) 属性，则将记录该事件。
- 如果对 CA Access Control 所保护资源的访问失败，且用户或资源具有 AUDIT(FAIL) 属性，则将记录该事件。

只有系统审核者（向其分配了 AUDITOR 属性的用户）可以执行审核任务，例如，更改分配给用户和资源的审核属性。

如果资源处于警告模式，则任何违反资源访问规则的访问都将导致生成警告模式审核记录，该记录表明 CA Access Control 允许访问资源。

审核记录构成了称为审核日志 *(seos.audit)* 的文件。 注册表中指定了审核日志的位置，它与错误日志在相同的位置。

在以下注册表键中指定了审核日志（以及错误日志）：

```
HKEY_LOCAL_MACHINE\Software\ComputerAssociates\AccessControl\logmgr
```

审核日志是二进制文件，不能进行编辑或更改。 但是，可以使用 CA Access Control 端点管理 查看已记录的事件、按时间限制或事件类型筛选事件等等。 （还可以使用 seaudit 实用程序来完成这些相同的任务。）

如果考虑对旧的审核日志和错误日志进行存档（备份），则可以在以后扫描这些事件。

将审核事件发送到 Unicenter TNG

与 Unicenter TNG 的集成是在安装时设置的。

可以选择将审核数据发送到 Unicenter TNG，也可以选择允许从 Unicenter TNG 启动 CA Access Control，或选择两者。 这两个选项不相关。

选择第一个选项会在子键下设置注册表值：

```
HKEY_LOCAL_MACHINE\Software\ComputerAssociates\AccessControl\UCTNG
```

值 Integration 设置为 1（是），值 EvtManagerServer 将 Unicenter TNG 主机的名称接收为字符串值。

传递给 Unicenter TNG 的审核事件显示在 Unicenter Enterprise Management\Enterprise Managers\Windows NT\Event 窗口的“控制台”日志中。

审核事件	显示颜色	严重性
成功	蓝色	S
已拒绝	橙色	F
失败	橙色	F
警告	蓝色	W
CA Access Control 已停止（审核停止）	蓝色	I
CA Access Control 已启动（审核启动）	蓝色	I

使用第二个选项可以从 Unicenter WorldView 菜单启动 CA Access Control，方法是指向“托管对象”窗口中代表 TCP/IP 网络的图标，然后从右键菜单中选择 CA Access Control。

CA Access Control 还将发送以下有关事件的信息：

- 产品名（CA Access Control + 版本号）
- 用户名
- 终端名称
- 类名
- 资源名称
- 进程名称
- 事件时间
- CA Access Control 审核格式的完整审核消息

并不总是发送“用户名”、“终端名称”、“类名”、“资源名称”和“进程名称”字段的信息，这取决于事件类型。

组件

CA Access Control 包括一个数据库 (seosdb)、两个驱动程序（seosdrv 和 drveng）、多个服务（包括 Watchdog、代理、引擎 (seosd)、策略模型和任务指派）和一个图形用户界面。

数据库

数据库包含下列元素的定义：

- 组织中的用户和组
- 需要保护的系统资源
- 管理用户和组对系统资源进行访问的规则

驱动程序

驱动程序通过执行下列任务，对所有 CA Access Control 文件和注册表键进行保护：

- 拦截要打开文件或注册表键、终止进程和执行网络活动的每个请求
- 将这些请求传递给 CA Access Control 引擎，并从引擎接收是批准还是拒绝请求的决定。
- 将决定转发给操作系统的原始系统调用，操作系统然后根据从驱动程序收到的回答继续其处理。

服务

Watchdog

Watchdog 会不断检查其他 CA Access Control 服务是否正在运行。 偶尔，当监视程序发现另一个服务已经停止时，它会立即再次启动该服务。

代理

代理负责执行以下任务：

- 通过 TCP/IP 之上的专有应用程序协议与 CA Access Control 客户端进行通信
- 管理 CA Access Control 用户的安全

引擎

引擎负责执行下列任务：

- 管理数据库，包括控制所有数据库更新
- 决定是否批准从驱动程序和代理收到的访问请求
- 检查监视程序服务是否正在运行，如果发现监视程序已停止运行，则重新启动监视程序

引擎处理数据库访问请求*并*做出访问决定，从而创建有效服务。

策略模型

单独管理数十或数百个数据库并不现实。 因此，CA Access Control 提供了策略模型服务，这是可通过一台计算机管理多台计算机的组件。 使用策略模型服务是可选的，但是它将极大地简化大型站点上的管理。

借助策略模型服务，可以使用策略模型数据库 (PMDB)。 与其他 CA Access Control 数据库一样，PMDB 包含用户、组、受保护的资源和管理资源访问的规则。 此外，PMDB 还包含一个订户站列表。 订户站与 PMDB 链接，这样以来，对 PMDB 所做的任何更改都会自动发送到订户数据库。

您可以为组织创建基本安全策略，并在单一数据库（策略模型数据库）上实施所有必需的规则。 订户可以同时包括 Windows 和 UNIX 工作站，从而确保以最少的管理工作执行统一规则。

系统管理员或安全管理员将更新 PMDB。 然后，PMDB 会以批处理模式将所有更新从 PMDB 传播给其订户，从而将管理员解放出来执行其他工作。

PMDB 可以有两类订户：另一个 PMDB 或本地数据库。 该 PMDB 还包含它将数据库更新传播到的订户的列表。 您可以利用该功能生成 PMDB 的层级结构。 本地数据库可用于保护在工作站上定义的用户、组和资源。

selang

命令行语言 selang 执行 CA Access Control 的所有功能。 还可以在脚本中使用 selang。 也可以在脚本中使用 selang。

有关 selang 及其命令的详细信息，请参阅*《参考指南》*中的“selang 命令语言”一章。

端点管理

CA Access Control 提供了两种方式来管理企业中的资源并控制哪些用户可访问这些资源：

- **selang** - CA Access Control 命令语言。

  通过 selang 命令语言，您可以在 CA Access Control 数据库中进行定义。 selang 命令语言是命令定义语言。

  注意：有关使用 selang 的详细信息，请参阅 *《selang 参考指南》*。

- **CA Access Control 端点管理** - 端点管理界面。

  使用基于 Web 的界面可以通过中央管理服务器管理远程端点。

  注意：有关安装 CA Access Control 端点管理 的详细信息，请参阅*《实施指南》*。

第 3 章： 管理用户和组

此部分包含以下主题：

用户和组

在 CA Access Control 中，每个操作和访问尝试都是代表负责提交该请求的用户执行的。 因此，系统中的每个进程都与特定的用户名相关联。 用户名将用户标识到 CA Access Control。

*用户*是能够登录的人员，也可以是批处理或后台程序的所有者。 在 CA Access Control 中，每次访问尝试都由用户执行。 CA Access Control 可以使用 CA Access Control 数据库及企业用户存储中的用户信息。 CA Access Control 在其数据库的 USER 记录或 XUSER 记录中存储用户信息。

注意：*企业用户存储*是操作系统中存储用户或组的存储，例如 UNIX 系统的 /etc/passwd 和 /etc/groups，或 Windows 的 Active Directory。

*组*是用户的集合。 组为组中的用户定义通用访问规则。 组可以嵌套（属于其他组）。 CA Access Control 可以使用 CA Access Control 数据库及企业用户存储中的组信息。 通常，基于角色（例如，database_administrators）创建组并向其分配用户。

用户记录是关键的访问者记录。 在 CA Access Control 中使用组的主要目的是同时向组中的所有用户分配访问权限。 同时分配访问权限比分别向每个用户分配访问权限更轻松且更不容易出错。

关于访问者的信息的存储位置

CA Access Control 使用的关于用户和组的信息存储在 CA Access Control 数据库和主机操作系统中。 主机操作系统信息存储称为*企业用户存储*或简称为*企业存储*。 默认情况下，将配置 CA Access Control 以便其不使用企业存储。 然而，您也可以配置 CA Access Control，以便 CA Access Control 找不到在其数据库中定义的用户或组时，会查找在企业存储中定义的用户和组成员资格并使用其中的信息。

注意：CA Access Control 可以使用企业存储中的信息，但仅当您在本地环境中使用 selang 命令时才能向企业存储中写入信息。

检查授权时，CA Access Control 始终在检查企业存储之前先检查在其自己的数据库中定义的访问者：如果您的企业用户的名称与在 CA Access Control 数据库中定义的用户名称相同，则 CA Access Control 将忽略企业用户。

CA Access Control 如何查找用户记录

用户登录后，CA Access Control 按以下顺序执行搜索，直至其找到与该用户关联的记录：

1. CA Access Control 搜索在其数据库中定义的用户。
2. CA Access Control 为具有该名称的企业用户搜索缓存。

   网络出现故障后，操作系统 (OS) 允许用户使用 OS 缓存凭据登录。 CA Access Control 缓存的目的是使 CA Access Control 在此类情况下也能使用企业用户的记录。
3. CA Access Control 使用操作系统为具有该名称的用户搜索企业用户存储。
4. 如果 CA Access Control 没有在其数据库或企业存储中发现与该用户关联的记录，那么 CA Access Control 会将 _undefined USER 记录中的属性分配给用户。

与企业用户存储集成

通常，将 CA Access Control 配置为使用在企业用户存储中定义的组和用户。

如果您这样配置了 CA Access Control，在默认情况下，当创建与企业用户或组相关的访问规则后或当用户登录到操作系统后，CA Access Control 会在其数据库中为该用户或组创建记录（如果先前没有记录）。这些记录具有类 XUSER（对于企业用户）或 XGROUP（对于企业组），并且具有 CA Access Control 强制执行访问规则所需的属性。因为 CA Access Control 根据需要创建记录，所以无需对其进行管理。

CA Access Control 从企业用户存储提取的企业用户或组的属性仅是名称和组成员资格属性。

在企业存储中管理访问者的指南

如果您决定在企业用户存储中管理访问者，则应该仔细阅读以下部分中的指南。

必须在数据库中定义的用户和组

CA Access Control 需要在其数据库中定义某些用户和组，而不是在企业用户存储中定义。这些信息包括：

- 预定义用户 (p. 38)
- 预定义组 (p. 39)
- 一个 CA Access Control 管理员
- 配置文件组
- 逻辑用户

企业用户使用限制

CA Access Control 强制实行以下企业用户使用限制：

- 如果 CA Access Control 中的企业用户与在数据库中定义的用户名称相同，则不能创建或引用该企业用户。
- 不能使用 selang AC 环境创建、删除或修改企业用户。

- 不能将企业用户用作逻辑用户。
- 默认情况下，不能在 CA Access Control 中创建企业用户，除非已经在企业用户存储中定义该用户。 但是，您可以在 UNIX 系统中启用或禁用此行为。

更多信息：

在 UNIX 中创建 XUSER 记录之前启用或禁用企业存储检查 (p. 36)

企业组使用限制

CA Access Control 强制实行以下企业组使用限制：

- 不能在 selang AC 环境中创建或删除企业组。
- 不能在 selang AC 环境中更改企业组的成员资格。
- 不能将企业组用作配置文件组 (p. 40)。

启用或禁用企业用户和组的使用

默认情况下，CA Access Control 不能使用在企业用户存储中定义的组和用户，但您可以让 CA Access Control 执行此操作。 我们建议您启用此功能，除非您需要与早期版本的 CA Access Control 兼容。

要让 CA Access Control 使用企业用户和组，请将配置设置 osuser_enabled 设置为 yes。 要禁用此行为，请将 osuser_enabled 的值设置为 no。

示例：在 Windows 中启用企业用户和组的使用

以下注册表设置可以在 Windows 中启用企业用户和组的使用：

- 注册表键：HKLM\SOFTWARE\ComputerAssociates\AccessControl\OS_user
- 名称：osuser_enabled
- 类型：REG_DWORD
- 值：yes

示例：在 UNIX 中启用企业用户和组的使用

以下命令可停止 CA Access Control、在 UNIX 中启用企业用户和组的使用以及重新启动 CA Access Control：

```
secons -s
seini -s OS_User.osuser_enabled yes
seload
```

在企业用户登录时启用或禁用 XUSER 记录的创建

如果启用 CA Access Control 对企业用户的使用，则默认情况下它会在用户登录时为该用户创建记录（在 XUSER 类中）。 有时您不需执行此操作，例如，如果每天成千上万的用户同时登录。

要防止 CA Access Control 在用户登录时创建 XUSER 记录，请将配置设置 create_user_in_db 的值更改为 0（零）。 要重新启用此行为，请将该值设置为 1（一）。

示例：在企业用户登录 Windows 时禁用 XUSER 记录的自动创建

以下注册表设置可以在 Windows 中禁用 CA Access Control 中企业用户记录的自动创建：

- 注册表键：
  HKLM\Software\ComputerAssociates\AccessControl\OS_user
- 名称：create_user_in_db
- 类型：REG_DWORD
- 值：0

示例：在企业用户登录 UNIX 时禁用 XUSER 记录的自动创建

以下命令可停止 CA Access Control、在 UNIX 中禁用 XUSER 记录的自动创建以及重新启动 CA Access Control：

```
secons -s
seini -s OS_User.create_user_in_db 0
seload
```

在 UNIX 中创建 XUSER 记录之前启用或禁用企业存储检查

有时，当未在企业用户存储中定义用户时您可能希望在 CA Access Control 中创建企业用户。 在 Windows 中，不能在 CA Access Control 中创建企业用户，除非该用户存在于 Windows 用户存储中。 在 UNIX 中，默认行为与 Windows 相反。 但是，在 UNIX 中，您可以启用或禁用此默认行为。

要禁用检查（以便在没有企业用户等同项时允许 CA Access Control 创建 XUSER 记录），请将配置设置 verify_osuser 的值更改为 0。 要强制执行检查，请将该值设置为 1。

示例：启用 XUSER 记录的创建而不检查企业用户存储

以下命令集可终止 CA Access Control、启用不具有企业存储等同项的 XUSER 记录的创建以及重新启动 CA Access Control：

```
secons -s
seini -s OS_User.verify_osuser 0
seload
```

Windows 中的循环企业存储帐户

*循环帐户*是已经删除然后又重新创建（使用相同名称）的企业存储用户或组。 从用户存储中删除一个用户（例如，当用户辞职时），然后为新用户创建一个与已删除的旧用户的名称相同的新帐户时，可能会产生循环帐户。

循环帐户存在安全问题，因为您不一定希望新访问者具有授予名称相同的旧帐户的那些访问权限。 要解决此问题，CA Access Control 授权要基于 SID。 这意味着当您创建新访问者，而该访问者名称与具有现有访问权限的已删除访问者的名称相同时，新访问者不自动接收旧访问者的旧权限。

重要说明！ 循环帐户访问者*不*继承旧的访问权限。 但是，依据涉及访问者名称（不是 SID）的数据库访问规则，可能看起来这些规则仍然适用。 使用 secons -checkSID 命令解决此问题。

在 Windows 中解析循环企业帐户

如果企业帐户（用户或组）具有相关联的数据库规则，然后被循环使用（删除然后使用相同名称创建），则可能看起来旧的数据库规则仍然适用于新帐户。 但是，由于 CA Access Control 授权基于 SID，这些规则不再适用，您需要为新组创建新规则。 可以创建新规则之前，您需要解析循环帐户。

要解析循环企业帐户，请打开命令提示符，然后运行以下命令：

```
secons -checkSID -users
secons -checkSID -groups
```

CA Access Control 处理它所具有的所有企业用户帐户（XUSER 记录），然后处理所有组帐户（XGROUP 记录），并标识 SID 与企业帐户的 SID 不同的帐户。 它使用以下命名约定在 CA Access Control 中重命名这些帐户：*SID* (*accountName*)

现在您可以为循环帐户创建新规则了。

注意：当用户登录或尝试访问资源时，将以此种方式解析循环用户帐户。 我们建议当您创建企业帐户时，将 secons -checkSID 命令作为排定任务运行。

示例：循环组帐户

公司 ABCD 在其企业存储中有个名为 *interns* 的组。 该组有九位成员，他们在从事 productA 的工作。 管理员使 CA Access Control 知道该组并为组成员分配访问文件所需的访问权限，如下所示：

```
nxg interns owner(msmith)
auth file c:\products\productA\materials\* xgid(interns) access(all)
auth file c:\HR\interns\* xgid(interns) access(read)
```

当 interns 在 ABCD 的使用期满后，企业存储管理员会删除该组。 三个月后会在企业存储中创建一个名称相同的新 interns 组，该组中有六位成员。 CA Access Control 数据库中的旧规则仍存在，因此看起来好像是新的 *interns* 组继承了旧组的权限。 但是，这些规则适用于旧的 interns 组，而 CA Access Control 管理员需要为新组创建新的规则。

要执行此操作，管理员需要识别并解析循环 interns 帐户，如下所示：

```
secons -checkSID -groups interns
```

此命令将 XGROUP 资源及参考该资源的任何访问规则重命名为“*SID* (*domain*\interns)”。 现在，管理员可以为从事 productB 工作的新 interns 组创建新的规则：

```
nxg interns owner(msmith)
auth file c:\products\productB\materials\* xgid(interns) access(all)
auth file c:\HR\interns\* xgid(interns) access(read)
```

注意：有关 secons 实用程序的详细信息，请参阅 *《参考指南》*。

数据库访问者

无论您决定如何管理您的用户，都必须在 CA Access Control 数据库中定义某些访问者，如以下部分中所述。

预定义用户

CA Access Control 预定义以下用户，您不能将其删除：

+devcalc

(Windows) CA Access Control 用于运行偏差计算进程 devcalc 的用户名。

_dms

_dms 安装在高级策略管理服务器组件的数据库（DMS、DH 读取程序和 DH 书写程序）中，policyfetcher 和 devcalc 使用 _dms 用户与 DH 和 DMS 进行通信。

nobody

nobody 用户是不能对应真实用户的用户记录。 使用此记录可以创建不授予任何用户相关联权限的规则。 例如，您可以将 *nobody* 设置为资源的所有者，这就是说任何用户都不能获得与拥有该记录相关联的权限。

+reportagent

CA Access Control 用于运行报告代理的用户名。

_seagent

_seagent 是 CA Access Control 用于运行某些内部进程的用户名，例如：

- PMDB 进程 sepmdd
- (UNIX) 偏差计算进程 devcalc
- 用户和组记录更新退出进程

_seagent 用户具有 SERVER 属性。

_sebuildla

(UNIX) _sebuildla 用户是 CA Access Control 运行 sebuildla 实用程序为 CA Access Control 后台进程 seosd 创建后备数据库的用户名。

_seoswd

(UNIX) _seoswd 是用户名，用于运行 seoswd watchdog 后台进程以监视程序（数据库中定义为受信程序）的文件信息和数字签名。

_undefined

_undefined 表示未在 CA Access Control 中定义的所有用户。可以使用 _undefined 在 ACL 中包括未定义的用户。

预定义组

CA Access Control 附带预定义组。 除了 _interactive 和 _network 组以外，您可以将用户以与添加到任何其他组相同的方式添加到这些组。

_abspath

如果用户在登录时位于 _abspath 组中，则该用户必须使用绝对路径名调用程序。

_interactive

属于 _interactive 组的用户仅可进行访问尝试。 如果用户登录到他们尝试访问的资源所在的主机，则这些用户是 _interactive 组的成员。 CA Access Control 动态并自动管理 _interactive 组的成员资格，您不能更改成员资格。

_network

这是 _interactive 的补充组。 属于 _network 组的用户仅可进行访问。 如果用户尝试访问的资源来自其他主机而不是资源所属的主机，则这些用户是 _network 组的成员。 CA Access Control 动态并自动管理 _network 组的成员资格，您不能更改成员资格。

_restricted

对于 _restricted 组中的用户，所有文件以及 Windows 注册表键都受 CA Access Control 的保护。如果文件或 Windows 注册表键没有显式定义访问规则，则将由该类（FILE 或 REGKEY）的 _default 记录控制访问权限。

注意：_restricted 组中的用户可能没有足够的授权开展工作。 如果您打算向 _restricted 组添加用户，请考虑在开始就使用警告模式。

_surrogate

如果用户使用 _surrogate 组的成员作为代理，则 CA Access Control 在代理操作审核跟踪中写入全部跟踪，并用原始用户名进行标记。

示例：使用 selang 将用户添加到 _restricted 组

以下 selang 命令将企业用户 john_smith 添加到 _restricted 组：

```
joinx john_smith group(_restricted)
```

配置文件组

*配置文件组*是在包含用户属性默认值的 CA Access Control 数据库中定义的组。 将用户分配到配置文件组后，配置文件组将向用户提供这些值，除非已经为用户设置这些值。

创建用户时可以为用户指定配置文件组，也可以稍后将用户分配到配置文件组。

通过配置文件组，管理员可以为分配给该组的任何新用户有效地创建带有特定权限的标准设置。 该设置可以指定以下内容作为该用户的主目录：审核属性、定义访问权限的 PMDB 以及影响配置文件组关联用户的各种密码规则。

CA Access Control 如何使用配置文件组确定用户属性

以下过程说明 CA Access Control 如何使用配置文件组确定用户属性。

1. CA Access Control 检查 USER 或 XUSER 类中的用户记录是否有针对属性的值。

   如果用户的记录有针对属性的值，CA Access Control 则使用该值。

2. CA Access Control 检查是否将用户分配到配置文件组。

   如果将用户分配到配置文件组，那么过程将继续。 如果未将该用户分配到配置文件组，那么 CA Access Control 将默认属性值分配给该用户。

3. CA Access Control 检查配置文件组是否有针对该属性的值。

   如果配置文件组有针对该属性的值，那么 CA Access Control 将该值分配给该用户。 如果配置文件组没有针对该属性的值，那么 CA Access Control 将默认值分配给该用户。

   注意：如果没有设置用户或配置文件组的审核属性，那么组的审核属性会影响用户的审核属性。

更多信息：

CA Access Control 如何为用户确定审核模式 (p. 107)

访问者管理

可以使用 CA Access Control 端点管理 或使用 selang 创建、修改和删除数据库或企业用户或组记录。

管理用户或组

如果要查看或修改特定访问者的属性，或者要删除访问者，必须先找到该访问者。

管理用户或组

1. 在 CA Access Control 端点管理 中，执行如下操作：
   a. 单击“用户”。
   b. 单击“用户”*或*“组”子选项卡。

   根据您的选择，将显示“用户”或“组”页面。
2. 在“搜索”区域中完成以下字段：

   用户/组名

   定义要查找的访问者的掩码。 您可以输入查找的访问者的全名，也可以使用掩码。 例如：使用 *admin* 列出名称包含“admin”的访问者。

   使用 *（星号）可列出所有访问者，使用 ?（问号）可代替单个字符。

   用户/组存储库

   指定要从中提取访问者列表的源。源可以是以下两者中的任意一个：

   - **内部帐户**－在 CA Access Control 数据库中定义的访问者。
   - **企业帐户**－在特定企业用户存储中定义的访问者。

仅显示 AC 帐户/配置文件

指定是否仅列出在 CA Access Control 数据库中有记录的帐户，如下所述：

- 如果选择了“内部帐户”，应用程序将仅列出存在于 CA Access Control 数据库中的帐户（无本地帐户）。
- 如果选择了“企业帐户”，应用程序将仅列出具有 CA Access Control 企业配置文件（XUSER 或 XGROUP 记录）的帐户。

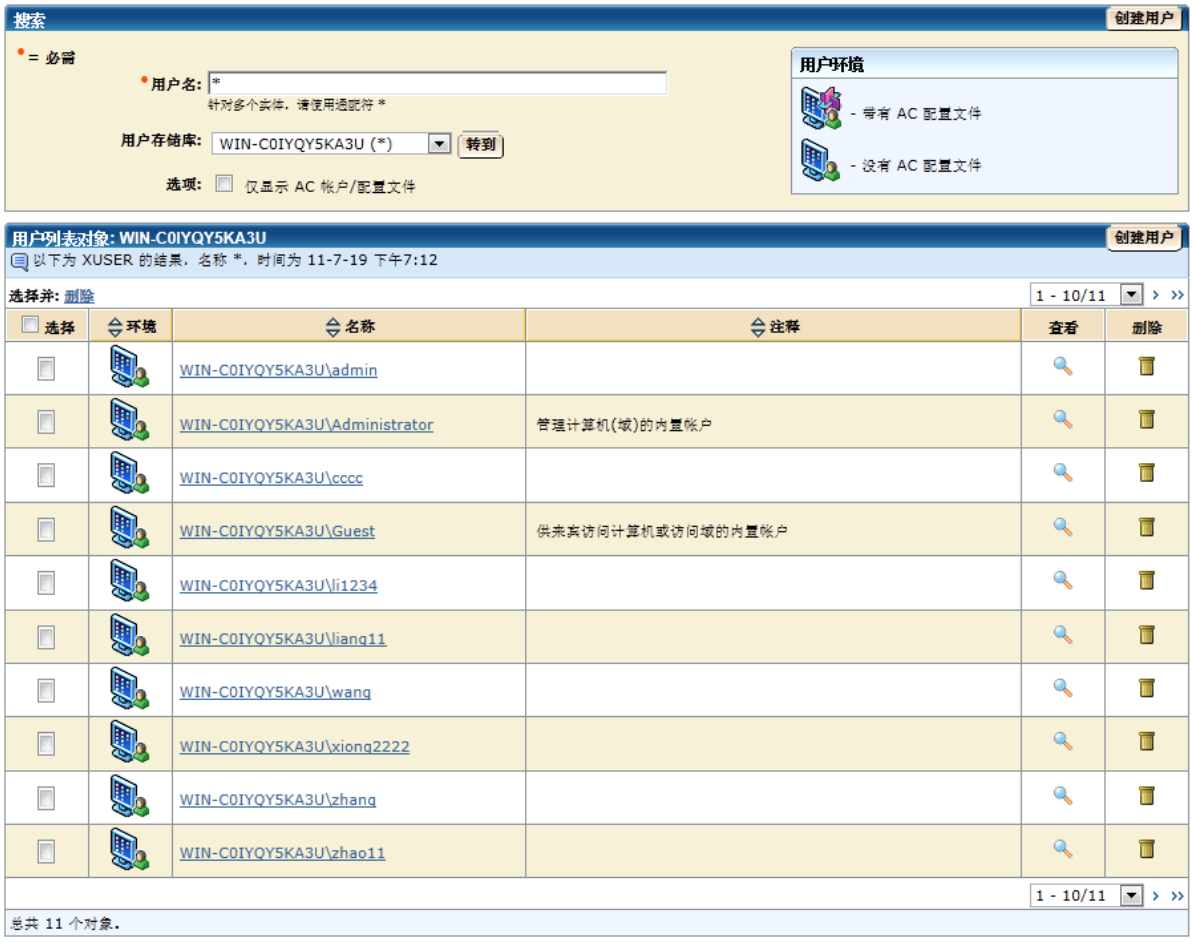
单击“执行”。

将显示您所选择的存储库中的访问者列表。

3. 请执行下列操作之一：
   - 单击“查看”列中的 以查看访问者的属性。
   - 单击“删除”列中的 以删除访问者。
   - 单击访问者的名称以修改访问者的属性。
   - 选择要删除的访问者，然后单击“删除”。
   - 单击“创建用户”或“创建组”以在 CA Access Control 数据库中创建用户或组记录。

示例：在存储库中搜索企业用户

以下图形向您显示了查找 ABC-DM1 企业用户存储中的所有用户的结果。



使用 selang 管理用户

将以下 selang 命令用于企业用户记录：

- **newxusr** 和 **editxusr** - 定义新的企业用户记录
- **chxusr** 和 **editxusr** - 更改企业用户的 CA Access Control 属性
- **find xuser** - 列出具有 CA Access Control 记录的企业用户
- **rmxusr** - 删除用户
- **show xuser** - 显示企业用户的 CA Access Control 属性

将以下 selang 命令用于 CA Access Control 数据库用户记录：

- **newusr** 和 **editusr** - 定义新的用户记录
- **chusr** 和 **editusr** - 更改用户的属性
- **rmusr** - 删除用户
- **find user** - 列出数据库用户
- **show user** - 显示用户的属性

示例：使用 selang 在数据库中定义用户

以下 selang 命令在 CA Access Control 数据库中定义了安全级别为 100 的新用户：

```
newusr internalUser level(100)
```

示例：使用 selang 更改企业用户的属性

以下 selang 命令向企业用户 Terry 授予了 AUDITOR 属性：

```
chxusr Terry auditor
```

使用 selang 管理组

除不能更改企业组的名称或成员资格外，您可以更改任意组的任意属性（从 CA Access Control 中）。

要更改组属性或分配与组相关联的访问权限，可以使用 CA Access Control 端点管理 或以下 selang 命令：

- **join[-]** 和 **joinx[-]**

  更改内部组的成员资格

  使用 join 将内部访问者添加到组。 使用 joinx 将企业组和用户添加到内部组。 使用命令的 -（减号）格式删除访问者。

- **editgrp**、**newgrp**、**chgrp**

  更改内部组的非成员资格属性

- **editxgrp**、**newxgrp**、**chxgrp**

  更改企业组的非成员资格属性

- **rmgrp**、**rmxgrp**

  删除用户组

示例：使用 selang 在数据库中定义组

以下 selang 命令在数据库中定义了新组“sales”。 该组的全名是“Sales Department”：

```
newgrp sales name('Sales Department')
```

示例：使用 selang 更改在数据库中定义的组的属性

以下 selang 命令使 CA Access Control 审核组 AC_admins 成员的所有事件：

```
chgrp AC_admins audit(all)
```

示例：使用 selang 将企业组添加到 ACL

以下 selang 命令将企业组 mygroup 添加到 myfile 的 ACL：

```
Authorize FILE (myfile) xgid(mygroup)
```

示例：使用 selang 将企业用户添加到在数据库中定义的组

以下 selang 命令将企业用户 mydomain\administrator 添加到在数据库中定义的组 AC_admins：

```
joinx mydomain\administrator group(AC_admins)
```

示例：使用 selang 将企业组添加到在数据库中定义的组

以下 selang 命令将企业组 Guests 添加到 _restricted 组：

```
joinx Guests group(_restricted)
```

第 4 章： 管理资源

此部分包含以下主题：

资源

*资源*是访问者可以访问且受访问规则保护的实体，或者是与该实体对应的 CA Access Control 数据库记录。 资源包括文件、程序、主机和终端等。

在 CA Access Control 中创建资源记录的主要目的是定义对与资源记录相对应的资源的访问权限。 访问资源所需的访问权限在资源记录的 Access Control 列表中指定。

资源组

*资源组*是包含其他资源列表的资源。 资源组是以下类之一的成员：CONTAINER、GFILE、GSUDO、GTERMINAL 或 GHOST。

由于资源组本身就是一种资源，因此它与其成员资源具有相同的属性。因此，使用资源组的优势就是简化了管理。 可以通过更改资源组的属性来更改所有成员资源的属性。

注意：在 Windows 上，检查用户对资源的授权时，CA Access Control 会考虑资源组所有权。 该操作在 12.0 中曾经介绍。 在先前版本中，授权进程只考虑资源的所有者。

例如，您使用没有所有者的默认访问权限来定义 FILE 资源。 而 FILE 资源是具有指定所有者的 GFILE 资源的成员。 在 CA Access Control r12.0 及更高版本中，命名的组所有者对该文件拥有完全访问权限。 在较早版本中，没有用户可以访问该文件。

类

在 CA Access Control 中，记录的*类*定义记录可以拥有的属性。 类中的所有记录具有相同的属性，尽管这些属性的值不同。

类包括：

- TERMINAL 类等。 该类包含终端的记录，例如 tty1、tty。
- FILE 类。 该类包含文件的记录。
- PROGRAM 类。 该类包含程序的记录。

每个记录均包含适用于记录类的属性的值。 例如，XUSER 类中的记录包括企业用户的位置和工作时间这样的属性，而 HOSTNET 类中的记录包括网络服务和 IP 地址数据这样的属性。

CA Access Control 包括预定义类。 还可以定义新类，称为用户定义的类。

类的默认记录

大多数类都可以包括指定该类资源访问类型的默认记录 (_default)，这些资源未在其自己的数据库记录中定义。

像其他资源记录一样，_default 记录可以包括 ACL 和 defaccess 字段。 您可以为除 USER、GROUP、CATEGORY、SECLABEL 和 SEOS 之外的所有类创建 _default 记录。

UACC 类（摒弃）

不再建议使用 UACC 类。 要为类中的记录指定默认值，请使用 _default 记录。

某些早期版本的 CA Access Control 将称为 UACC 的单独类用于与其他类的 _default 记录相似的记录。 现已不再建议使用 UACC 类，如果使用 _default 记录，则不检查 UACC 类中的同等记录。 在将来的版本中，可能不再支持 UACC 类。

例如，假设用户 Henderson 尝试终止进程 store_log。 CA Access Control 将按以下顺序检查授权。 主要问题是：是否在数据库中定义了进程 store_log？ CA Access Control 在数据库的 PROCESS 类中搜索名为 store_log 的记录。

- 如果未找到这种记录，则没有在 CA Access Control 中定义该进程。 在这种情况下，CA Access Control 会使用 PROCESS 类中的 _default 记录或 UACC 类中的 PROCESS 记录，以确定是否允许 Henderson 终止 store_log。
  - 如果用户 Henderson 出现在了 _default 记录的 ACL 中，则应用在其中指定的权限。
  - 如果 Henderson *未*出现在 _default 记录的 ACL 中，则应用在 _default 记录的 defaccess 属性中指定的权限。 该权限应用于 _default ACL 中未明确出现的所有用户。
- 如果数据库中定义了进程 store_log，则问题是用户 Henderson 是否出现在数据库进程 store_log 的 ACL 中。
  - 如果用户 Henderson 出现在进程 store_log 的 ACL 中，则应用在该处指定的权限。
  - 如果 Henderson *未*出现在 ACL 中，则 CA Access Control 应用在 store_log 资源的默认访问属性中指定的权限。 该权限称为资源的默认访问权限。

注意：如果 _default 的默认访问权限 (defaccess) 设置为 NONE，或者如果未指定 _default 且 UACC 类中相应资源的默认访问权限为 NONE，则拒绝任何尝试访问该类中未定义的资源的访问者访问资源。

如果 _default（或 UACC）的默认访问权限设置为最高权限（ALL，或在某些情况下为 READ 或 EXECUTE），则每个用户都可以访问未明确保护的任何资源。

预定义类

预定义类可以分为以下类型：

类类型	用途
访问者	定义访问资源的对象，例如用户和组
定义	定义对安全实体（例如安全标签和类别）进行定义的对象
安装	定义对 CA Access Control 的行为进行控制的对象
资源	定义访问规则所保护的对象

下表包含所有预定义类的列表。

类	类类型	说明
ADMIN	定义	使用该类可以将管理责任指派给不具有 ADMIN 属性的用户。 您可以为这些用户提供全局授权属性并限制他们的管理权限范围。
AGENT	资源	不适用于 CA Access Control
AGENT_TYPE	资源	不适用于 CA Access Control
APPL	资源	不适用于 CA Access Control
AUTHHOST	访问者	不适用于 CA Access Control
CALENDAR	资源	使用该类可以为实施时间限制的用户、组和资源定义 Unicenter TNG 日历对象。
CATEGORY	定义	使用该类可以定义安全类别。
CONNECT	资源	使用该类可以保护传出连接。 该类中的记录定义哪些用户可以访问哪些 Internet 主机。 激活 CONNECT 类之前，请确保数据流模块处于活动状态。
CONTAINER	资源	使用该类可以定义其他资源类中的一组对象，因此可以在某个规则适用于多个不同的对象类时简化定义访问规则的工作。
FILE	资源	使用该类可以保护文件、目录或文件名掩码。
GAPPL	资源	不适用于 CA Access Control
GAUTHHOST	定义	不适用于 CA Access Control

类	类类型	说明
GFILE	资源	该类中的每个记录定义一组文件或目录。 与将用户连接到组的方式一样，通过将文件或目录（FILE 类的资源）显式连接到 GFILE 资源，可完成分组。
GHOST	资源	该类中的每个记录定义一组主机。 与将用户连接到组的方式一样，通过将主机（HOST 类的资源）显式连接到 GHOST 资源，可完成分组。
GROUP	访问者	该类中的每个记录定义一个内部组。
GSUDO	资源	该类中的每个记录定义一个用户可以执行的一组命令（好像另一个用户正在执行一样）。 sesudo 命令使用该类。
GTERMINAL	资源	该类中的每个记录定义一组终端。
HNODE	定义	HNODE 类包含有关组织的 CA Access Control 主机的信息。 该类中的每个记录代表企业中的一个节点。
HOLIDAY	定义	该类中的每个记录定义用户需要额外权限才能登录的一个或多个时间段。
HOST	资源	该类中的每个记录定义一个主机。 主机由其名称或其 IP 地址标识。 对象包含确定本地主机是否可以从该主机接收服务的访问规则。 激活 HOST 类之前，请确保数据流模块处于活动状态。
HOSTNET	资源	该类中的每个记录均由 IP 地址掩码标识，且包含访问规则。
HOSTNP	资源	该类中的每个记录定义一组主机，其中，属于该组的主机都具有相同的名称模式。 每个 HOSTNP 对象的名称包含一个通配符。
LOGINAPPL	定义	LOGINAPPL 类中的每个记录定义一个登录应用程序，标识可以使用该程序进行登录的用户，并控制使用该登录程序的方式。
MFTERMINAL	定义	MFTERMINAL 类中的每个记录定义一个 CA Access Control 大型机管理计算机。
POLICY	资源	POLICY 类中的每个记录定义部署和删除策略所需的信息。 它包括指向 RULESET 对象的链接，而这些对象包含用于部署和删除策略的 selang 命令列表。
PROCESS	资源	该类中的每个记录定义一个可执行文件。
PROGRAM	资源	该类中的每个记录定义一个可与条件访问规则一起使用的受托程序。 受托程序是 Watchdog 进行监视以确保不被篡改的 setuid/setgid 程序。
PWPOLICY	定义	PWPOLICY 类中的每个记录定义一个密码策略。

类	类类型	说明
RESOURCE_DESC	定义	不适用于 CA Access Control
RESPONSE_TAB	定义	不适用于 CA Access Control
RULESET	资源	RULESET 类中的每个记录都定义一组用于定义策略的规则。
SECFILE	定义	该类中的每个记录定义一个不能更改的文件。
SECLABEL	定义	该类中的每个记录定义一个安全标签。
SEOS	安装	该类中的一个记录指定活动的类和密码规则。
SPECIALPGM	安装	SPECIALPGM 类中的每个记录对 Windows 中的备份、DCM、PBF 和 PBN 功能或 UNIX 中的 xdm、备份、邮件、DCM、PBF 和 PBN 程序进行注册，或将需要特殊授权保护的应用程序与逻辑用户 ID 关联。 这样便可根据所执行的操作而不是执行该操作的人员来设置访问权限。
SUDO	资源	sesudo 命令使用该类来定义一个用户（如常规用户）可以执行的命令（好像另一个用户（如 root 用户）正在执行一样）。
SURROGATE	资源	该类中的每个记录包含访问者的访问规则，此访问规则定义了谁可以将该访问者用作代理。
TCP	资源	该类中的每个记录定义一个 TCP/IP 服务，例如邮件或者 http 或 ftp。
TERMINAL	资源	该类中的每个记录定义一个终端（用户可以从其登录的设备）。
UACC	资源	定义每个资源类的默认访问规则。
USER	访问者	该类中的每个记录定义一个内部用户。
USER_ATTR	定义	不适用于 CA Access Control
USER_DIR	资源	不适用于 CA Access Control
XGROUP	资源	该类中的每个记录在 CA Access Control 中定义一个企业组。
XUSER	资源	该类中的每个记录在 CA Access Control 中定义一个企业用户。

注意：默认情况下，CA Access Control 数据库类 TCP 和 SURROGATE 处于非活动状态。

如果您从 TCP 类处于活动状态的早期版本升级，但是没有任何 TCP 记录且没有更改 _default TCP 资源，CA Access Control 则会在升级期间停用该类。 对于 SURROGATE 类同样如此。

如果您从 SURROGATE 类处于活动状态的早期版本升级，并且已经定义了 SURROGATE 记录或已经更改了 SURROGATE 记录的默认值，CA Access Control 则会在升级之后保留 SURROGATE 类配置。 该类仍然保持活动状态，而内核模式截获保持启用状态。

注意：有关 CA Access Control 类的详细信息，请参阅 *《selang 参考指南》*。

用户定义的类

您可以使用 CA Access Control 定义新类，以便可以通过为抽象对象创建适当的记录来保护抽象对象。

示例：用于数据库视图的用户定义的类

站点可以使用数据库存储和显示专有数据。

您可以定义用户定义的类 DATABASE_VIEWS，并将每个数据库视图定义为该类的资源成员。 为资源提供一个创建该数据库视图所需的定义访问权限的 ACL。 当用户尝试创建数据库视图时，CA Access Control 会检查该用户的访问权限，并基于 ACL 允许或禁止创建。

用户定义的类资源中的通配符

通过在用户定义的类中的资源名称中使用通配符，您可以创建对应于多个物理资源的资源记录：名称与通配符模式匹配的所有物理资源均受到与资源记录相关联的访问权限的保护。

可用通配符如下：

- * 表示任意数量的任意字符
- ? 表示任意单个字符

如果物理资源名称与多个资源记录名称相匹配，则最长的非通配符匹配项将用于该资源。

CA Access Control 不接受以下通配符模式作为资源名称：

- *
- /*
- /tmp/*
- /etc/*

用户定义的类 - 示例

假设您的系统为银行服务，并且您想要保护帐户间大笔金额的转帐，则可以根据以下概述来设置该安全性。

1. 定义包含描述转帐、调用（例如 TRANSFERS）的记录的类。
2. 对于您可能希望保护的每次资金转帐，请在 TRANSFERS 类中定义记录。

   例如，可以定义名为 Upto.$1K、Upto.$1M、Upto.$10M 和 Over.$10M 的记录。

   将您希望控制转帐的任何其他资源定义为 TRANSFERS 类的成员。
3. 要给予不同用户执行不同的最大金额转帐的权限，可以批准或拒绝他们访问 TRANSFERS 类中的各个记录。
4. 此外，要处理编程传输，在银行的资金转帐程序中插入对 CA Access Control API 的调用，从而 API 会在检查用户的权限后才允许转帐进行。

Windows 服务保护

通过 CA Access Control，您可以保护 Windows 服务。 *Windows 服务*是在 Windows 后台中运行的程序，是与 UNIX 中的后台程序等同的 Windows 后台程序。

CA Access Control Windows 服务保护将拦截源自以下其中一项的服务访问事件：

- 服务管理和信息事件。

  CA Access Control 将拦截每个服务访问的 services.exe 进程。 这包括启动或停止服务。 例如，net 启动*服务*、net 停止*服务*等等均受到保护。

  在这种情况下，已截获事件将通过受保护的服务名称进行审核。

- 服务数据库管理事件。

  CA Access Control 将拦截对服务控制管理数据库的注册表调用，以保护服务状态查询或更改。 这意味着 CA Access Control 将自动保护与受保护服务相关联的注册表区。 实际上，当您定义服务保护时，CA Access Control 将保护以下注册表键：

  HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services*service_name*
  HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services*service_name**

  在这种情况下，已截获事件将通过完整注册表路径进行审核。

您可以使用与保护其他资源相同的方法来保护 Windows 服务，即将资源分配到服务，并将访问者添加到资源的 Access Control 列表中。 Windows 服务的资源类是 WINSERVICE。 WINSERVICE 资源具有两个 Access Control 列表：ACL 和 NACL。 对于 WINSERVICE Access Control 列表中的条目，有效的访问类型为：

- 读取
- 修改
- 启动
- 停止
- 暂停
- 继续执行

启用和禁用 Windows 服务保护

您可以启用或禁用 Windows 服务的 CA Access Control 保护。

要启用 Windows 服务保护，请将 CA Access Control 注册表中 Instrumentation\PlugIns\WinServiceplg 部分的配置设置 OperationMode 设置为 1。 要禁用保护，请将 OperationMode 设置为 0。

默认情况下，CA Access Control 将启用 Windows 服务保护。

要使 CA Access Control 保护 Windows 服务，需要启用保护并激活 WINSERVICE 类。

保护 Windows 服务

您可以保护 Windows 服务，这样可以为 Windows 操作提供更多保护。

保护 Windows 服务

1. 确保您已启用 Windows 服务保护 (p. 56)。
2. 确保 WINSERVICE 类处于活动状态。 （默认情况下处于活动状态。）
3. 在 CA Access Control 中创建 WINSERVICE 记录，并使用您希望保护的 Windows 服务的名称命名该记录。

   注意：Windows 服务名称显示在“Windows 服务属性”对话框的“常规”选项卡中，但与该选项卡上的“显示名称”不同。
4. 将访问者及其访问权限分配给该服务。

   服务现已受到保护。

示例：限制对后台打印程序的访问

在 Windows 中，后台打印程序具有服务名 spooler。 下列 selang 命令确保 WINSERVICE 类处于活动状态并将后台处理程序的默认访问权限设置为 *读取*。

```
setoptions class+(WINSERVICE)
editres WINSERVICE(spooler) defacc(R)
```

非 IPv4 Telnet 连接在 Windows Server 2008 上不安全

在 Windows Server 2008 上，CA Access Control 无法保护不使用 IPv4 的 Telnet 连接。

要在 Windows Server 2008 上保护本地主机 telnet 连接（从本地主机到本地主机的 telnet），请修改/etc/HOSTS 文件如下：

```
127.0.0.1       localhost
#       ::1                 localhost
127.0.0.1       <your server name without domain suffix>
```

如果您的计算机在 IPv6 域上，请添加以下行：

```
127.0.0.1    <your server name with domain suffix>
```

查看对受保护的 Windows 服务的访问尝试

如果 CA Access Control 保护 Windows 服务，它将拦截与该服务相关联的访问尝试，并将其记录在审核日志中。 这些访问尝试可能是使用 services.exe 进程管理服务（启动、停止等等）所导致，也可能是对受保护服务的服务数据库管理区域的注册表访问所导致。 前一种访问是仅包含服务名的审核，后一种访问（注册表访问）包含完整的注册表路径。 要查看与 Windows 服务相关联的所有访问尝试，您需要使用通配符。

要查看对受保护的 Windows 服务的访问尝试，请创建用于筛选 WINSERVICE 类的审核记录和资源名称 **myService** 的审核筛选

CA Access Control 将显示您定义的 WINSERVICE 资源的所有审核记录（不论是通过注册表还是通过服务管理界面尝试访问）。

示例：查看对后台打印程序服务的所有访问尝试

该示例假定您将后台打印程序服务定义到 CA Access Control 而不进行任何访问，如下所示：

```
er winservice spooler defaccess(none) owner(nobody)
```

然后，您可以使用 seaudit 实用程序列出对后台打印程序服务的所有访问尝试，如下所示：

```
seaudit -resource WINSERVICE *spooler* *
```

该命令列出了 WINSERVICE 类的所有审核记录，这些记录是针对对后台打印程序服务的访问尝试所记录。 生成的输出结果如下所示：

```
seaudit - Audit log lister
03 Apr 2008 16:53:48 D WINSERVICE  bigHost1\Administrator Read     69  2 Spooler
   c:\WINDOWS\system32\services.exe bigHost1.comp.com
03 Apr 2008 16:53:48 D WINSERVICE  bigHost1\Administrator Read     69  2 Spooler
   c:\WINDOWS\system32\services.exe bigHost1.comp.com
03 Apr 2008 16:53:50 D WINSERVICE  bigHost1\Administrator Read     69  2 Spooler
   c:\WINDOWS\system32\services.exe bigHost1.comp.com
03 Apr 2008 16:53:50 D WINSERVICE  bigHost1\Administrator Read     69  2 Spooler
   c:\WINDOWS\system32\services.exe bigHost1.comp.com
03 Apr 2008 16:53:53 D WINSERVICE  bigHost1\Administrator Read     69  2 Spooler
   c:\WINDOWS\system32\services.exe bigHost1.comp.com
03 Apr 2008 16:53:53 D WINSERVICE  bigHost1\Administrator Read     69  2 Spooler
   c:\WINDOWS\system32\services.exe bigHost1.comp.com
03 Apr 2008 16:54:10 D WINSERVICE   bigHost1\Administrator Read       69  2
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Spooler
   C:\WINDOWS\regedit.exe bigHost1.comp.com
03 Apr 2008 16:54:10 D WINSERVICE   bigHost1\Administrator Read       69  2
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Spooler
   C:\WINDOWS\regedit.exe bigHost1.comp.com
03 Apr 2008 16:54:19 D WINSERVICE   bigHost1\Administrator Read       69  2
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Spooler
```

```
   C:\WINDOWS\regedit.exe bigHost1.comp.com
03 Apr 2008 16:54:26 D WINSERVICE   bigHost1\Administrator Read       69  2
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Spooler
   C:\WINDOWS\regedit.exe bigHost1.comp.com
03 Apr 2008 16:54:26 D WINSERVICE   bigHost1\Administrator Modify     69  2
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Spooler
   C:\WINDOWS\regedit.exe bigHost1.comp.com

Total records displayed 11
```

Windows 注册表保护

通过 CA Access Control，您可以保护 Windows 注册表项。

您可以通过将 REGKEY 类的资源分配给注册表键来对其进行保护。 然后可以与其他资源一样，指定对注册表键的访问权限。

指定对注册表键的访问权限不会影响对该注册表键的子键的访问，但枚举（列出）子键除外，该操作需要对注册表键的读取权限。

CA Access Control 仅在 Windows Server 2003 和更高版本的 Windows 系统上支持 AC 环境中的 REGVAL 资源。 在这些系统上，CA Access Control 将保护 REGVAL 类的注册表值，并且 REGKEY 访问权限不影响对注册表键值的访问。

在早期版本的系统上，CA Access Control 不支持 AC 环境中的 REGVAL 资源，并且 REGKEY 记录应用的访问权限会影响对注册表键值的访问。

REGKEY 和 REGVAL 记录具有相同的结构。 每个记录都包含以下 Access Control 列表：

- ACL
- CALACL
- NACL
- PACL

REGVAL 和 REGKEY 记录都允许相同的访问类型，如下所示：

- READ
- WRITE
- DELETE
- 无

注意：CA Access Control 注册表保护不会保护加载和卸载 Hive 的注册表操作。 在 Windows Server 2008 及更高版本的系统上，如果访问权为“NONE”的访问者尝试访问受保护的注册表值，CA Access Control 会返回值 REG_NONE。 REG_NONE 值将确认该值存在，但不会具体说明该值是什么。

保护 Windows 注册表项

您可以保护 Windows 注册表项，这样可以为 Windows 操作提供更多保护。

保护 Windows 注册表项

1. 如果您要使用 REGKEY 和 REGVAL 类记录，请确保这些类均处于活动状态。（它们在默认情况下处于活动状态。）
2. 创建 REGKEY 或 REGVAL 记录，并使用您要保护的注册表键或值的名称命名该记录。

   注意：使用完整的注册表路径名指定注册表键或值。 您可以使用通配符来指定嵌套在键下的所有子键或子键值。

   现在注册表项即通过 CA Access Control 为记录提供的默认访问权限受到保护。
3. （可选）将用户和组连同其访问权限分配到 REGKEY 或 REGVAL 记录中相应的 Access Control 列表。

示例：提供对注册表键的默认访问权限“无”

以下 selang 命令将提供对注册表键的默认访问权限“无”：

```
er REGKEY HKEY_LOCAL_MACHINE\SOFTWARE\Test\Key1 defacc(NONE) owner(nobody)
```

因此，对 key1 的默认访问权限如下所示：

操作	版本早于 Windows Server 2003 的系统	Windows Server 2003 及更高版本的系统	Windows Server 2008 及更高版本的系统
枚举子键	拒绝	拒绝	拒绝
查询、修改、重命名或删除键	拒绝	拒绝	拒绝
加载或卸载键上的 hive	拒绝	拒绝	拒绝
枚举值	拒绝	拒绝	允许
读取、创建、重命名或删除值	拒绝	允许	允许
枚举子键的子键	拒绝	允许	允许
创建子键	允许	允许	允许
查询、修改、重命名或删除子键	允许	允许	允许
加载或卸载子键上的 hive	允许	允许	允许

示例：提供对注册表键的默认访问权限“读取”

以下 selang 命令将提供对注册表键的默认访问权限“读取”：

```
er REGKEY HKEY_LOCAL_MACHINE\SOFTWARE\Test\Key1 defacc(READ) owner(nobody)
```

因此，对 Key 1 的默认访问权限如下所示：

操作	版本早于 Windows Server 2003 的系统	Windows Server 2003 及更高版本	Windows Server 2008 及更高版本
枚举子键	允许	允许	允许
读取键	允许	允许	允许
修改、重命名或删除键	拒绝	拒绝	拒绝
加载或卸载键上的 hive	拒绝	拒绝	拒绝
枚举值	允许	允许	允许
读取值	允许	允许	允许
创建、重命名或删除值	拒绝	允许	允许
枚举子键的子键	允许	允许	允许
创建子键	允许	允许	允许
查询、修改、重命名或删除子键	允许	允许	允许
加载或卸载子键上的 hive	允许	允许	允许
枚举子键值	允许	允许	允许
创建子键值	允许	允许	允许

示例：提供对注册表键通配符的默认访问权限“无”

以下 selang 命令将提供对注册表键中所有子键的默认访问权限“无”：

```
er REGKEY HKEY_LOCAL_MACHINE\SOFTWARE\Test\Key1\* defacc(NONE) owner(nobody)
```

通配符 (*) 不应用于 Key1，但应用于 Key1 的所有子键；这意味着对 Key1 所有子键的任何形式的访问都将被拒绝。 由于存在父保护规则，也会拒绝意在重命名或删除 Key1 的访问。

该命令允许对 Key1 的值进行访问。 对 Key1 的子键值（例如 Key1\subkey1\ 的值）的访问权限在不同的 Windows 系统之间有所差异：

- 在 Windows Server 2003 和更高版本的系统上，该命令拒绝可枚举 key1 的任何子键值的访问权限，但授予可创建、重命名、删除和读取这些值的访问权限。
- 在早于 Windows Server 2003 的系统上，该命令拒绝对 Key1 的子键值的所有访问权限。

示例：对注册表值提供默认访问权限“无”

以下 selang 命令在 Windows Server 2003 和更高版本的系统上将通过访问权限“无”来保护特定注册表值：

```
er REGVAL HKEY_LOCAL_MACHINE\SOFTWARE\TestKey\value1 defacc(NONE) owner(nobody)
```

注意：在 Windows Server 2008 及更高版本的系统上，如果访问权为“NONE”的访问者尝试访问受保护的注册表值，CA Access Control 会返回值 REG_NONE。 REG_NONE 值将确认该值存在，但不会具体说明该值是什么。

保护文件数据流

数据流是字节的序列。 文件数据流包含文件数据，并提供有关文件的其他信息。 例如，您可以创建包含关键字或元数据的数据流。

注意：文件数据流仅在 NTFS 文件系统上可用。 有关文件数据流的详细信息，请参阅 Microsoft Developer Network (MSDN) 库网站。

当您创建文件规则时，CA Access Control 将自动保护文件的默认数据流。 例如，保护文件 c:\foo.txt 的规则也管理对 c:\foo.txt::$DATA 的权限。 但是，CA Access Control 不会自动保护任何非默认的数据流，对于这些数据流，您需要创建其他文件保护规则。

要保护文件数据流，请执行以下操作之一：

- 要保护特定数据流，请按以下格式创建文件规则：

  drive:*path**filename.ext*:*stream*

- 要保护特定类型的特定数据流，请按以下格式创建文件规则：

  drive:*path**filename.ext*:*stream*:*type*

- 要保护所有数据流，请按以下格式创建通用文件规则：

  drive:*path**filename.ext*:*

示例：保护所有文件数据流

以下 selang 命令将创建可保护文件 c:\foo.txt 中所有数据流的通用文件规则：

```
er file c:\foo.txt:* owner(nobody) defaccess(none)
```

示例：保护特定数据流

以下 selang 命令将创建可保护文件 c:\foo.txt 中数据流 *mystream* 的文件规则：

```
er file c:\foo.txt:mystream owner(nobody) defaccess(none)
```

内部文件保护

在安装期间，CA Access Control 制定规则来保护两种类型的内部文件：

- 内部规则 — 保护配置文件、日志文件和数据库文件。

  您无法删除内部规则。

- 默认规则 — 保护敏感文件，例如您用来加密和验证通信的根证书和服务器证书。

  您可以在安装之后删除默认规则。

内部文件规则

内部文件规则保护配置文件、日志文件和数据库文件。 内部文件规则在 selang 中不可见且无法删除。 但是，您可以编写 FILE 规则来覆盖内部文件规则。 如果删除这些 FILE 规则，CA Access Control 会恢复到内部文件规则。

除了数据库文件之外，CA Access Control 使用内部文件规则保护的文件有以下访问权限：

- 对 CA Access Control 内部进程的完全访问
- 对所有其他访问者的读取和执行（有相关需要时）访问

CA Access Control 使用内部文件规则保护的数据库文件有以下访问权限：

- CA Access Control 内部进程对数据库有完全访问权限
- NT AUTHORITY\System 用户对数据库有读取权限
- 所有其他访问者对数据库都没有访问权限

  注意：所有其他访问者的默认访问权限在 r12.5 SP3 中更改。 在先前的版本中，默认情况下所有其他访问者对数据库文件都有读取访问权限。

CA Access Control 使用内部文件规则保护下列文件。 该表的第二列列出了指定文件位置的注册表子键和注册表项（如果适用）。 CA Access Control 在下面的注册表键下创建其注册表项：

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl
```

注意：一些文件位置是内部定义的，没有相应的注册表项。 您不能配置这些文件的位置。

文件	注册表子键和注册表项	默认文件位置
seosdrv.sys	-	%SystemRoot%\system32\drivers\seosdrv.sys
cainstrm.sys	-	%SystemRoot%\system32\drivers\cainstrm.sys
drveng.sys	-	%SystemRoot%\system32\drivers\drveng.sys
pwdchange.dll	-	%SystemRoot%\system32\pwdchange.dll
SUSRAUTH.dll	-	%SystemRoot%\system32\SUSRAUTH.dll
eACSubAuth.dll	-	%SystemRoot%\system32\eACSubAuth.dll
eACPasswordFltr.dll	-	%SystemRoot%\system32\eACPasswordFltr.dll

文件	注册表子键和注册表项	默认文件位置
所有数据库文件	SeOSD\dbdir	*ACInstallDir*\Data\seosdb
所有帮助文件	lang\help_path	*ACInstallDir*\Data\help
所有二进制文件	-	*ACInstallDir*\bin
seosd.trace	SeOSD\trace_file	*ACInstallDir*\log
seos.audit	logmgr\audit_log	*ACInstallDir*\log
seos.audit.bak	logmgr\audit_back	*ACInstallDir*\log
seos.error	logmgr\error_log	*ACInstallDir*\log
seos.error.bak	logmgr\error_back	*ACInstallDir*\log
seos.msg	message\filename	*ACInstallDir*\Data
stop.ini	STOP\STOPIniFileName	*ACInstallDir*\Data
stopsignature.dat	STOP\STOPSignatureFileName	*ACInstallDir*\Data
response.ini	SeOSD\ResponseFile	*ACInstallDir*\Data
audit.cfg	logmgr\AuditFiltersFile	*ACInstallDir*\Data

注意：有关配置设置的详细信息，请参阅*《参考指南》*。

默认文件规则

CA Access Control 在安装期间会创建默认文件规则来保护敏感文件。 默认文件规则在 selang 中可见且能够被删除。

下表列出 CA Access Control 使用默认文件规则保护的敏感文件以及这些文件的访问权限和允许的访问者。

在该表中，*PMDBDir* 是策略模型数据库 (PMDB) 所在的目录，而 *pmd_name* 是每个策略模型的名称。 默认情况下，*PMDBDir* 位于 *ACInstallDir*\Data。 *PMDBDir* 的位置在下列注册表项中有所定义：

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl\Pmd\_Pmd_directo
ry_
```

文件	默认访问权限	允许的访问者
ACInstallDir\data\crypto\crypto.dat	无	sechkey
ACInstallDir\data\crypto\def_root.pem*	无	sechkey
ACInstallDir\data\crypto\sub.key	无	sechkey

文件	默认访问权限	允许的访问者
ACInstallDir\data\crypto\sub.pem	无	sechkey
ACInstallDir\log\policyfetcher.log	读取	+policyfetcher
*PMDBDir**pmd_name*	读取，Chdir	-
*PMDBDir**pmd_name**	读取，执行	-

第 5 章： 管理授权

此部分包含以下主题：

访问权限

CA Access Control 的主要目的是分配和强制执行访问权限。

访问权限始终具有以下组件：

- 所访问的资源，例如，文件、主机或终端
- 访问的类型，例如读取、写入、删除、登录、运行
- 访问者，可以是用户也可以是组

如果以下一种或多种情况属实，则用户有权以某种方式访问资源：

- 用户具有由资源 ACL 授予的访问权限
- 用户是具有访问权限的组的成员。
- 用户运行具有访问权限的程序。 例如，用户具有在 SPECIALPGM 类中运行程序的权限，或者在 SUDO 类中运行命令的权限。

注意：有关按类的访问权限的详细信息，请参阅 *《selang 参考指南》*。

设置访问权限 - 示例

示例：向内部用户授予读取访问权限

以下 selang 命令将内部用户 internal_user 添加到终端 tty30 的 ACL，向终端授予读取访问权限：

```
authorize TERMINAL tty30 access(READ) uid(internal_user)
```

示例：向企业用户授予读取访问权限

以下 selang 命令将企业用户 Terry 添加到终端 tty30 的 ACL，向终端授予读取访问权限：

```
authorize TERMINAL tty30 access(READ) xuid(Terry)
```

示例：更改企业用户对资源的访问权限

以下 selang 命令将 Terry 对终端 tty30 的访问权限设置为 none，因此拒绝了 Terry 的访问：

```
authorize TERMINAL tty30 access(NONE) xuid(Terry)
```

示例：从资源中删除企业用户的访问权限

以下 selang 命令从终端 tty30 中的 ACL 中删除 Terry：

```
authorize- TERMINAL tty30 xuid(Terry) access-
```

Terry 现已具有对终端的默认访问权限。

示例：向企业用户授予子管理员访问权限

以下 selang 命令将企业用户 Terry 设置为具有管理用户和文件权限的子管理员：

```
authorize ADMIN USER xuid(Terry)
authorize ADMIN FILE xuid(Terry)
```

访问控制列表

对资源的访问权限在 Access Control 列表中指定。 每个资源记录至少具有两个 Access Control 列表：

ACL

指定被授予资源访问权限的访问者及其被授予的访问权限的类型。

NACL

指定被拒绝授权资源访问权限的访问者及其被拒绝的访问权限的类型。

访问权限还取决于访问权限所在的环境，例如用户是否是在本地登录。

条件访问控制列表

条件 Access Control 列表 (CACL) 提供了 ACL 的扩展。 当访问者尝试访问资源时，如果资源的 ACL 和 NACL 未为该用户定义访问权限，则 CA Access Control 将检查条件 Access Control 列表。

条件 Access Control 列表以一种特定方式指定对所访问资源的访问权限，例如通过使用指定的程序。

例如，您可以使用条件 Access Control 列表定义程序通路规则。

CA Access Control 允许使用以下条件 Access Control 列表：

- 程序 Access Control 列表 (PACL)
- TCP 类 Access Control 列表
- CALENDAR 类 Access Control 列表

要在条件 Access Control 列表条目中定义一个条目，可以使用 selang authorize 命令的 via 选项。

与其他 Access Control 列表一样，条件 Access Control 列表中的每个条目指定被授予访问资源权限的访问者及其被授予的访问权限的类型。 此外，条件 Access Control 列表中的条目还指定分配权限的条件。 对于 PACL，条件是程序的名称，访问者需要运行该程序才能具有访问权限。

示例：使用 PACL

要使企业用户 sysadm1 仅可通过运行程序 secured_su 成为超级用户，您可以使用以下 selang 命令指定相应的条件访问规则：

```
authorize SURROGATE user.root xuid(sysadm1) via(pgm(secured_su))
```

defaccess - 默认访问字段

资源的记录可以包括默认访问字段 defaccess。 defaccess 字段的值指定允许任何资源 Access Control 列表中均未包含的访问者使用的访问权限。

如何确定对资源的访问权限

当访问者尝试访问资源时，CA Access Control 通过按照预先确定的顺序运行一次或多次检查来检查访问权限，直至其获得结果。 如果在任何一次检查中生成了访问结果（拒绝或允许访问），则 CA Access Control 不会进一步执行检查而是返回结果。

CA Access Control 运行这些检查的顺序很重要。 默认情况下，对于每项资源，CA Access Control 均按以下顺序检查访问记录：

1. 基于资源时间的限制
2. 资源的所有权（允许所有者进行访问）
3. B1 检查
4. 资源的 NACL
5. 资源的 ACL
6. 资源的 PACL
7. 资源的 defaccess 字段

最后两项检查的顺序由 accpacl 选项的设置确定。 您可以通过使用 selang 命令设置选项 setpacl- 来禁用对资源 PACL 的使用。

一个 Access Control 列表可以包含多个影响用户的条目。 例如，可以包含与用户显式相关的条目，还可以包含用户所属的每个组的条目。 CA Access Control 在进入下个级别之前检查每个级别所有可能的条目。 有关 CA Access Control 如何解决每个级别的冲突规则的详细信息，请参阅《用户和组访问权限之间的互动》 (p. 73)。

示例：对文件的结果权限

在下表中，假设名为 user1 的访问者尝试读取资源 file1。

在下表中，CA Access Control 按照 accpacl 选项的默认设置使用 PACL。

NACL 中用于 user1 的条目	ACL 中用于 user1 的条目	PACL 中用于 user1 的条目	defaccess 中的条目	结果权限
读取	*(Any)*	*(Any)*	*(Any)*	拒绝读取权限
(Not defined)	无	*(Any)*	*(Any)*	拒绝读取权限
(Not defined)	读取	*(Any)*	*(Any)*	授予读取权限

NACL 中用于 user1 的条目	ACL 中用于 user1 的条目	PACL 中用于 user1 的条目	defaccess 中的条目	结果权限
(Not defined)	*(Not defined)*	via pgm securereader	*(Any)*	允许通过 securereader 程序读取
(Not defined)	*(Not defined)*	*(Not defined)*	读取	授予读取权限

如果条目显示为 *(Not defined)*，则表示 Access Control 列表中不存在用于 user1 的条目。

如果条目显示为 *(Any)*，则表示该 Access Control 列表中的条目无关紧要，因为 CA Access Control 不对其进行检查。

CA Access Control 检查的顺序为从左到右。 请注意，对于所有行来说，具有定义的访问权限的单元格右侧的单元格均具有值 *(Any)*。 相反，包含定义的访问权限的单元格左侧的所有单元格均具有值 *(Not defined)*。

用户和组访问权限之间的互动

您可以向用户以及用户所属的组显式授予或拒绝访问权限。 有时这些权限可能会有冲突。 下例显示了如果在用户属于两个组（组 1 和组 2）时向同一资源分配有冲突的访问权限，会出现怎样的结果。

假设已设置“累积组权限” (p. 74)选项（默认设置）。

用户的访问权限	组 1 的访问权限	组 2 的访问权限	结果访问权限
访问被拒绝	*(Any)*	*(Any)*	访问被拒绝
访问已授权	*(Any)*	*(Any)*	访问已授权
(Not defined)	访问已授权	*(Not defined)*	访问已授权
(Not defined)	*(Not defined)*	访问已授权	访问已授权
(Not defined)	访问已授权	访问已授权	访问已授权
(Not defined)	访问被拒绝	*(Any)*	访问被拒绝
(Not defined)	*(Any)*	访问被拒绝	访问被拒绝

如果条目显示为 *(Not defined)*，则表示没有为用户或组定义任何条目。

如果条目显示为 *(Any)*，则表示访问权限无关紧要，因为 CA Access Control 不对其进行检查。

累积组权限 (ACCGRR)

*累积组权限*选项 (ACCGRR) 影响 CA Access Control 检查资源的 ACL 的方式。 如果启用 ACCGRR，则 CA Access Control 会检查 ACL 以获得用户所属的所有组授予的权限。 如果禁用 ACCGRR，则 CA Access Control 会检查 ACL 以查看是否有任何可应用的条目包含值 none。 如果有，则会拒绝访问。 否则 CA Access Control 将忽略所有组条目，Access Control 列表中的第一个可应用的条目除外。 默认情况下，该选项为启用。

要启用 ACCGRR 选项，可以使用以下 selang 命令：

```
setoptions accgrr
```

要禁用 ACCGRR 选项，可以使用以下 selang 命令：

```
setoptions accgrr-
```

安全级别、类别和标签

安全级别和安全类别提供了限制资源访问的其他方式，这是对 Access Control 列表用法的补充。

安全标签是一种将安全级别和类别捆绑在一起的方式，从而更轻松地对其进行管理。

安全级别

*安全级别*是可以分配给访问者和资源的 0 到 255 之间的整数。 如果访问者的安全级别小于分配给资源的安全级别，即使在资源的 Access Control 列表中向用户授予了访问权限，访问者也不能访问资源。 如果资源的安全级别为零，将不对该资源进行安全级别检查。

安全级别为零的访问者不能访问安全级别不为零的任何资源。

安全类别

*安全类别*是 CATEGORY 类中记录的名称。 可以向访问者和资源分配安全类别。 只有当将访问者分配给向资源分配的所有安全类别时，访问者才能访问该资源。

安全标签

*安全标签*是 SECLABEL 类中记录的名称。 安全标签将安全级别和一组安全类别捆绑在一起。 将安全标签分配给访问者或资源，会授予该访问者或资源与该安全标签相关联的组合的安全级别和安全类别。 安全标签会覆盖访问者或资源中任何特定的安全级别和类别分配。

示例：使用安全标签 High_Security

假设 High_Security 是包含安全级别 255 与安全类别 MANAGEMENT 和 CONFIDENTIAL 的安全标签。

如果将用户 user1 分配给安全标签 High_Security，则 user1 的安全级别为 255，另外还具有安全类别 MANAGEMENT 和 CONFIDENTIAL。

第 6 章： 保护帐户

此部分包含以下主题：

用户模拟保护

在 CA Access Control 中启用 SURROGATE 类时，同时启用了用户模拟保护。 通过用户模拟保护，您可以指定在特定规则允许更改的情况下，用户或组仅能将其 SID（安全标识符）更改为其他 SID。 这可以防止用户在无权操作的情况下模拟其他用户的身份。

注意：安全标识符是用于为操作系统标识用户或组的数字值。

例如，您定义防止任何用户模拟管理员的 CA Access Control 规则。 用户 Tom 试图以管理员身份运行程序以便执行某些任务。 而 CA Access Control 不允许程序执行，因为 Tom 没有权限模拟管理员。

您可以以两种模式运行用户模拟保护：

- 用户模式拦截
- 内核模式拦截

用户模式拦截

如果启用了用户模式拦截，CA Access Control 则仅拦截源自 Windows RunAs 实用工具的模拟请求。 用户模式拦截可在所有支持的 Windows 版本上使用。

注意：当启用用户模拟保护时（即启用 SURROGATE 类时），默认情况下会启用用户模式拦截。

用户模式拦截的优势包括以下内容：

- CA Access Control 识别提出最初模拟请求的用户。

  在许多 Windows 应用程序中（包括 RunAs 实用工具），NT AUTHORITY\SYSTEM 用户会模拟请求的用户并且提出模拟请求。 用户模式拦截会识别执行该实用工具的用户，而不是提出请求的 NT AUTHORITY\SYSTEM 用户。 例如，如果 Tom 执行 RunAs 来模拟管理员，而 NT AUTHORITY\SYSTEM 用户提出模拟请求，那么 CA Access Control 会将 Tom 识别为请求的用户。

- 仅当用户执行 RunAs 实用工具时，CA Access Control 才会拦截模拟请求。

  这就将对性能的影响最小化。

用户模式拦截的劣势是，CA Access Control 不拦截每个 Windows 进程的的每个模拟请求。

内核模式拦截

如果您启用内核模式拦截，CA Access Control 会拦截所有 Windows 进程的每个模拟请求。 内核模式拦截不可以在任何支持的 Windows 版本上使用。

注意：有关不可以使用内核模式拦截的 Windows 版本的更多信息，请参阅《*版本说明*》。

内核模式拦截的优势是，您可以保护在 Windows 计算机上提出的每个模拟请求。

内核模式拦截的劣势包括以下内容：

- 如果 NT AUTHORITY\SYSTEM 用户模拟请求的用户并且提出模拟请求，CA Access Control 不会识别提出最初模拟请求的用户。

  例如，RunAs、ftp 和 Telnet 请求都是由 NT AUTHORITY\SYSTEM 用户提出的。 如果 Tom 执行 RunAs 来模拟管理员，而 NT AUTHORITY\SYSTEM 用户提出模拟请求，那么 CA Access Control 会将 NT AUTHORITY\SYSTEM 识别为请求的用户。

- CA Access Control 会拦截操作系统将其作为正常操作的每个模拟请求，该操作可能有性能影响。

  虽然 CA Access Control 会缓存模拟请求，但是授权引擎仍必须授权许多模拟事件。

CA Access Control 对用户模拟请求的响应方式

SURROGATE 类中的每个记录都定义了保护用户免受模拟尝试伤害的限制。 CA Access Control 将模拟请求视为仅能由授权用户访问的抽象对象。 SURROGATE 类中的记录表示每个具有替换（模拟）保护的用户或组。

当用户或组提出模拟其他用户或组的请求时，CA Access Control 会执行以下操作：

1. 检查用户或组的 SURROGATE 记录的访问授权。 根据 SURROGATE 记录，会发生以下情况之一：
   - 用户或组的 SURROGATE 记录专门允许或拒绝模拟。

     CA Access Control 使用 SURROGATE 记录的访问授权来允许或拒绝模拟请求。
   - 用户或组没有 SURROGATE 记录。

     该过程转至步骤 2。
2. 检查用户或组的默认 SURROGATE 记录的访问授权，如下所示：
   - 如果请求人是用户，CA Access Control 会将 USER._default SURROGATE 记录中定义的访问类型授予用户。
   - 如果请求人是组，CA Access Control 会将 GROUP._default SURROGATE 记录中定义的访问类型授予用户。

   注意：USER._default、GROUP._default 和 _default SURROGATE 记录的默认访问授权是读取权限。 这意味着 CA Access Control 允许任何请求模拟用户或组，除非用户或组的 SURROGATE 记录禁止模拟请求。 要更改此行为，请更改 USER._default 和 GROUP._default 记录的访问授权。 您也能通过更改 _default SURROGATE 记录的访问授权来为用户和组设置相同的默认访问授权。

启用用户模拟保护

通过用户模拟保护可以设置规则来允许或拒绝模拟特定用户和组的请求。

启用用户模拟保护

1. （可选）启用内核模式拦截，如下所示：
   a. 停止 CA Access Control。
   b. 将下列注册表值更改为 1：

      ```
      HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl\
      SeOSD\SurrogateInterceptionMode
      ```

   c. 重新启动 CA Access Control。

   注意：默认情况下会启用用户模式拦截。
2. 打开 selang 命令提示符窗口。
3. 启用 SURROGATE 类：

   ```
   setoptions class+(SURROGATE)
   ```

4. 定义 SURROGATE 记录的 selang 规则以便进行 CA Access Control 实施。
5. （仅适用于内核模式拦截）定义规则，该规则允许 SYSTEM 用户模拟提出模拟请求的用户：

   ```
   auth SURROGATE USER.Administrator uid("NT AUTHORITY\SYSTEM") acc(R)
   ```

   Windows 将许多实用程序和服务（例如“运行身份”）识别为 "NT AUTHORITY\SYSTEM" 而不是运行实用工具的原始用户。 您必须为 SYSTEM 用户定义规则以便允许运行这些实用工具的用户模拟其他用户。

示例：允许任何模拟请求

除非数据库的记录明确阻止模拟，否则以下 selang 规则允许任何用户模拟其他用户：

```
editres SURROGATE _default defaccess(READ)
```

示例：阻止特定用户的模拟

除非数据库的记录明确允许用户模拟，否则以下 selang 规则阻止任何用户模拟管理员：

```
newres SURROGATE USER.Administrator defaccess(NONE)
```

示例：允许组模拟用户

下列规则允许管理员组的成员模拟管理员：

```
authorize SURROGATE USER.Administrator gid("Administrators")
```

设置 Surrogate DO 工具

操作员、生产人员和最终用户通常需要执行只有超级用户才能执行的任务。

传统的解决方案是向所有这些用户提供超级用户密码，这会危及到站点的安全。 安全的替代方法（即保持密码的保密性）会使系统管理员因处理用户要执行例行任务的合法请求而导致负担过重。

Surrogate DO (sesudo) 实用程序解决了这个难题。 它允许用户执行 SUDO 类（其中每条记录包含一个脚本）中定义的操作，指定哪些用户和组可以运行脚本，以及基于这一目的借给他们必要的权限。

例如，要定义启动“后台打印程序”服务的 SUDO 资源（就好像用户为系统一样），请输入以下 selang 命令：

```
newres SUDO StartSpooler data("net start spooler")
```

该 newres 命令将 StartSpooler 定义为受保护的操作，某些用户可能获得执行的系统权限。

重要说明！ 请在数据属性中使用完整的绝对路径名。 相对路径名可能会意外地执行未受保护的目录中植入的特洛伊木马程序。

此外，通过使用 authorize 命令，可以授权用户执行 StartSpooler 操作。例如，要允许用户 *operator1* 启动“后台打印程序”服务，请输入以下 selang 命令：

```
authorize SUDO StartSpooler uid(operator1)
```

还可以使用 authorize 命令明确防止用户执行受保护的操作。 例如，要防止用户 *operator2* 启动“后台打印程序”服务，请输入以下 selang 命令：

```
authorize SUDO StartSpooler uid(operator2) access(None)
```

运行 sesudo 实用程序将执行受保护的操作。 例如，输入以下命令后，用户 *operator1* 即可启动“后台打印程序”服务：

```
sesudo -do StartSpooler
```

sesudo 工具首先检查是否授权用户执行 SUDO 操作，然后，如果授权用户使用资源，则执行资源中定义的命令脚本。 在我们的示例中，sesudo 检查是否授权 operator1 执行 StartSpooler 操作，然后使用系统凭据调用命令“net start spooler”。

注意：有关 sesudo 实用程序的详细信息，请参阅 *《参考指南》*。

定义 SUDO 记录（任务指派）

SUDO 类中的记录存储命令脚本，这样，用户便可以通过借用的权限来运行该脚本。 借用权限的能力由 SUDO 记录以及执行这些脚本的 sesudo 命令严格控制。

注意：如果创建交互式 Windows 应用程序的 SUDO 记录，您必须设置 SUDO 记录的交互式标记。 如果不设置交互式标志，应用程序会在后台运行，而您无法与其进行交互。 有关详细信息，请参阅 *《疑难解答指南》*。

在 SUDO 记录中，注释属性用于特殊用途，通常也称为**数据属性。**

comment 属性的值是命令脚本，也可以在其中添加一个或多个要禁止或允许使用的脚本参数值。 整个 comment 属性值必须放在单引号中，并且应该使用可执行文件的完整路径名称引用可执行文件，以防止特洛伊木马获取它们的位置。

以下是 comment 属性的格式：

```
comment('cmd[;[prohibited-values][;permitted-values]]')
```

因为禁止和允许使用的值的列表是可选的，所以简单的 comment 属性值可以是：

```
newres SUDO NET comment('net use')
```

该命令中的简单值表示命令 sesudo NET 将执行命令“net use”。 不禁用特定脚本参数值；允许使用所有脚本参数值。

使用通配符和功能强大的变量，可以灵活地指定禁用的参数和允许使用的参数。 可以使用的通配符是标准的 Windows 通配符。 禁止使用的参数和允许使用的参数也可以包含下列变量：

变量	说明
$A	字母值
$G	现有 CA Access Control 组名
$H	（仅适用于 UNIX）以用户的主目录开头的参数
$N	数字值
$O	运行 sesudo 的 CA Access Control 用户的名称
$U	现有 CA Access Control 用户名
$e	空条目。 使用此变量为规则指定不带任何参数的 SUDO 命令。
$f	现有文件名
$g	现有 Windows 组名
$h	现有主机名
$r	具有 Windows 读访问权限的现有文件
$u	现有 Windows 用户名
$w	具有 Windows 写访问权限的现有文件
$x	具有 Windows 执行访问权限的现有文件

如果将*禁用*参数值的列表附加到脚本，请执行以下操作：

- 用分号分隔脚本和禁用的参数值，但将它们都保留在单引号内。 例如，如果要禁止用户使用 -start，但允许用户使用所有其他参数，请输入下面的命令：

  ```
  newres SUDO scriptname comment('cmd;-start')
  ```

  其中 *cmd* 代表您的脚本。

  另外，如果不允许使用任何参数值，但希望默认使用所有参数，请按以下方式定义 SUDO 记录：

  ```
  newres SUDO scriptname comment('cmd;*')
  ```

- 如果脚本参数有多个禁用的值，请使用空格字符作为分隔符。 例如，如果要禁止用户使用 -start 和 -stop，但却允许用户使用所有其他参数，请输入下面的命令：

  ```
  newres SUDO scriptname comment('cmd;-start -stop')
  ```

- 如果多个脚本参数有禁用值，请使用管道符 (|) 作为各组禁用值之间的分隔符。 例如，如果要禁止用户将 -start 和 -stop 用作脚本的第一个参数并禁止其将任何当前的 Windows 用户名用作第二个参数（请参阅前面的变量列表），请输入下面的命令：

  ```
  newres SUDO scriptname comment('cmd;-start -stop | $u')
  ```

  如果脚本的参数多于列表中的参数，则最后一组禁用参数将适用于所有其余参数。

如果将*允许*的参数值的列表附加到脚本：

- sesudo 实用程序检查参数值：
  - 请*勿*匹配任何相应的*禁用*值。
  - 匹配至少一个相应的*允许*值。

  这意味着如果某个参数值在禁用列表中，则即使在允许使用列表中指定了该参数，也不允许使用它。
- 用分号将*允许*值的列表与*禁用*值的列表分开，但要将它们都保留在单引号内。 即使您没有禁用值的列表，仍需要使用分号；否则，将禁用您要允许使用的值。 例如，如果仅允许值 NAME 作为脚本的参数值，请输入下面的命令：

  ```
  newres SUDO scriptname comment('cmd;;NAME')
  ```
- 正如在另一个列表中一样，
  - 如果脚本参数有多个允许值，请使用空格字符作为分隔符。
  - 如果多个脚本参数具有允许值，请使用管道符 (|) 作为各组允许值之间的分隔符。

  例如，如果您有两个参数，第一个参数必须是数值而不能是 Windows 用户名，第二个参数必须是字母而不能是 Windows 组名，则请输入下面的命令：

  ```
  newres SUDO scriptname comment('cmd;$u | $g ;$N | $A')
  ```

  如果脚本的参数多于列表中的参数，则最后一组允许使用的参数将适用于所有剩余参数。

因此，comment 属性的完整格式为：**首先是脚本，然后是逐个参数的禁用值，最后是逐个参数的允许值：**

```
comment('cmd; \
param1_prohib1 param1_prohib2 ... param1_prohibN | \
param2_prohib1 param2_prohib2 ... param2_prohibN | \
 ...
paramN_prohib1 paramN_prohib2 ... paramN_prohibN ; \
param1_permit1 param1_permit2 ... param1_permitN | \
param2_permit1 param2_permit2 ... param2_permitN |
 ...
paramN_permit1 paramN_permit2 ... paramN_permitN')
```

sesudo 实用程序通过以下方式检查用户输入的每个参数：

1. 测试参数 N 是否与允许的参数 N 匹配。（如果允许的参数 N 不存在，则使用上一个允许的参数。）
2. 测试参数 N 是否与禁止的参数 N 匹配。（如果禁止的参数 N 不存在，则使用上一个禁止的参数。）

如果所有参数都与允许的参数匹配，但任何参数与禁止的参数都不匹配，则 sesudo 执行该命令。

示例：设置允许用户运行 net send 的任务指派

以下过程说明了如何让用户 Takashi 执行 net send 命令，以及如何防止他执行 net start 命令：

1. 在 CA Access Control 端点管理 中，单击“用户”选项卡，然后单击“授权和指派”子选项卡。

   “授权和指派”菜单选项将显示在左侧。

2. 单击“任务指派”。

   将显示“任务指派”页面。

3. 单击“创建任务”。

   将显示“创建任务”页面。

4. 按如下所示填写对话框中的字段：

窗口项	值
名称	NET
数据	net;start;send *
所有者	nobody
默认访问权限	None（清除选项）
授权的访问者	USER: Takashi Allow: Execute

   单击“保存”。

   将创建新的任务指派 (SUDO) 记录。

5. 测试任务指派规则：
   a. 以 Takashi 身份登录。
   b. 打开命令提示，执行以下命令：

      ```
      sesudo -do NET start
      ```

      此时出现以下消息：

      ```
      sesudo：不允许使用 “start” 作为参数编号 1。
      ```

      注意： *net start* 将不执行，因为已将其定义为禁止的值。
   c. 执行以下值：

      ```
      sesudo -do NET send comp message
      ```

      应该执行该命令。

示例：授权用户使用交互式应用程序执行特权操作

用户可以使用任何管理单元 MSC 模块执行高权限操作，如下例所示：

1. 在 CA Access Control 端点管理 中，单击“用户”选项卡，然后单击“授权和指派”子选项卡。

   “授权和指派”菜单选项将显示在左侧。
2. 单击“任务指派”。

   将显示“任务指派”页面。
3. 单击“创建任务”。

   将显示“创建任务”页面。
4. 按如下所示填写对话框中的字段：

窗口项	值
名称	服务
数据	c:\winnt\system32\mmc.exe
所有者	nobody
选项	Interactive（选中选项）
默认访问权限	None（清除选项）
授权的访问者	USER: Tori Allow: Execute

单击“保存”。

将创建新的任务指派 (SUDO) 记录。 “交互式”选项提供了启动服务后登录者可以使用的桌面用户界面。 仅在服务作为 LocalSystem 帐户运行时，该选项才可用。

5. 测试任务指派规则：
   a. 以 Tori 身份登录。
   b. 打开命令提示，执行以下命令：

      ```
      sesudo -do services
      ```

   c. mmc.exe 将启动。

检查用户无操作状态

无操作状态功能防止通过其所有者离开的帐号或者组织不再采用的帐号未经授权地访问系统。 非活动日是指用户不登录的日子。 您可以指定在用户帐户被挂起和无法登录之前必须经过的不活动天数。 一旦挂起帐户，则必须手动重新激活它。

注意：在无操作检查方面，会将密码更改视为活动。 如果用户密码更改，则该用户不能因为无操作而被挂起。

可以使用 USER 类记录或 GROUP 类记录的无操作属性设置无操作天数。 后者只影响将该组作为配置文件组的用户。 您还可以使用 SEOS 类的 INACT 属性，为系统范围内的所有用户设置无操作。

在 selang 中，使用以下命令可以通过全局方式指定无操作状态：

```
setoptions inactive (numdays)
```

要为组设置天数（将覆盖该组的系统范围内的无操作设置），请使用以下命令：

```
editgrp groupName inactive (numdays)
```

要为用户设置天数（将覆盖该用户的组设置和系统范围设置），请使用以下命令：

```
editusr userName inactive (numdays)
```

要重新激活挂起的用户帐户，请使用以下命令：

```
editusr userName resume
```

要重新激活挂起的配置文件组，请使用以下命令：

```
editgrp userName resume
```

要在系统范围级别禁用无操作登录检查，请使用以下命令：

```
setoptions inactive-
```

要禁用对组的无操作登录检查，请使用以下命令：

```
editgrp groupName inactive-
```

要禁用对用户的无操作登录检查，请使用以下命令：

```
editusr userName inactive-
```

第 7 章： 管理用户密码

此部分包含以下主题：

管理密码和锁定策略

密码是最常用的身份验证方法，但密码保护方法却存在众所周知的问题：普通密码易于猜测；常年使用的密码和循环密码最终会被破坏；而通过网络明文发送的密码可能被侦听程序拦截。

Windows 有一套强制用户使用密码的密码规则和策略，可以避免大多数这些常见陷阱。 CA Access Control 具有确保用户选择更为安全的密码的更多规则。

您可以在 CA Access Control 中指定以下规则：

- 新密码不得与以前的密码匹配。 可在密码策略中指定 CA Access Control 存储的以前密码的个数。
- 新密码不得包含用户名。
- 新密码不得包含正在替换的密码。
- 新密码不得与正在替换的密码相匹配。 CA Access Control 不区分大小写。
- 新密码至少必须有在密码策略中指定的最低数量的字母数字字符、特殊字符、数字、小写字符和大写字符。
- 新密码中重复的字符数不得超过密码策略中指定的字符数。
- 新密码不得是 CA Access Control 的字典中受限制的单词之一。 该字典由注册表子键中的 Dictionary 值指定：

  ```
  HKEY_LOCAL_MACHINE\Software\ComputerAssociates\AccessControl\passwd
  ```

每个密码必须有最长使用期限（即，它到期必须失效），从而强制用户在某一时间间隔后选择新的密码。

- 每个密码必须有一个最短使用期限。 通过指定一个最短使用期限，您可以防止用户迅速频繁地更改密码。 使用频繁更改的密码，他们可以溢出密码历史堆栈，然后重复使用以前的密码。

配置密码质量检查

配置密码质量检查

1. 在 CA Access Control 端点管理 中，单击“配置”选项卡。

   在左侧将显示配置菜单选项。
2. 在“杂项”区域的选项中单击“类激活”。

   将显示“类激活”页面。
3. 在“用户身份控制”区域中选择“PASSWORD”，然后单击“保存”。

   此操作将激活密码质量检查。
4. 在“策略”区域的选项中单击“用户密码策略”。

   将显示“用户密码策略”页面。
5. 定义用于密码检查的规则，然后单击“保存”。

   您定义的密码检查规则将在密码被更改时立即强制执行。
6. （仅适用于 UNIX）通过使用 sepass 实用程序更新新密码。

   注意：有关 sepass 实用程序的详细信息，请参阅 *《参考指南》*。

示例：定义密码检查规则

以下 selang 命令将激活密码质量检查，并定义规定至少使用以下内容的密码规则：

- 六个字母数字字符
- 三个小写字母字符
- 两个数字字符

```
setoptions class+ (PASSWORD)
setoptions password(rules(alpha("6") lowercase("3") numeric("2")))
```

注意：有关 setoptions 命令格式的详细信息，请参阅 *《参考指南》*。

解析错误消息

如果您正在为 Windows NT 系统上的用户设置密码，则可能出现下列消息：

密码太短，不符合要求。

该错误表明密码不符合策略要求。 这可能是由以下任何一种原因造成的：

- 该密码短于或长于要求长度。
- 该密码最近已被使用，并存在于“Windows NT 更改历史记录”字段中。
- 该密码没有足够多的唯一字符。
- 该密码不符合其他密码策略要求（例如通过 CA Access Control 密码策略设置的要求）。

为了避免该错误，请确保设置符合所有相关要求的密码。

第 8 章： 监视和审核

此部分包含以下主题：

安全审核者

安全审核者和系统管理员的最重要的任务之一就是审核或监视系统活动，从而发现可疑或恶意的活动。 安全审核在安全环境中是一项必不可少的任务，CA Access Control 中的安全审核功能包括以下几个方面：

- 正确指出访问过系统的用户、已被访问的资源、访问资源的方式（例如读取文件）及访问资源的时间
- 在有人尝试进行安全破坏活动（即使尝试失败）时，通知相关的用户并发出警报
- 指明对安全规则做出的更改及更改者
- 提供在执行访问规则之前测试该规则效果的方式

CA Access Control 审核模仿真实审核：安全审核者的操作独立于系统和安全管理员，但如果某种其他模型更适合于您的环境，则可以更改实施方式，以便进行相应的更改。

安全审核者就是为其分配了 AUDITOR 属性的用户。 定义为安全审核者的用户可以执行审核任务，例如，更改为用户和资源分配的审核规则。此外，他们还可以使用 CA Access Control 审核工具，而无需拥有 ADMIN 属性。

事件截获

如果满足以下两个条件，CA Access Control 将截获事件：

- 相应的类处于活动状态。
- 参与该事件的规则在数据库中存在。

示例：可以使用以下通用规则审核对位于 c:\data\payroll 中所有文件的文件访问：

```
newres FILE c:\data\payroll\*
```

您还需要确保 FILE 类处于活动状态（默认情况下）。

被截获事件的类型

CA Access Control 截获两种类型的事件：

- 截获事件

  来自截获事件的信息缓存为进程的一部分，以供审核事件将来使用。

- Audit 事件

截获模式

根据截获模式，CA Access Control 对授权进行截获、检查并记录访问请求事件的审核记录。 CA Access Control 具有以下截获模式：

- 完全强制模式
- 仅审核模式
- 无截获模式

注意：警告模式 (请参阅本页中的定义 97)不是截获模式，它仅在完全强制模式下工作且在实施过程中短期使用。

仅审核模式

*仅审核模式*将记录所有拦截事件，而不检查或强制执行访问规则。 使用这种模式可收集符合遵从要求或规定的数据。 在仅审核模式中，CA Access Control 将截获事件并把审核事件写入，但不处理对授权的请求，也不强制执行规则。 因此，CA Access Control 将允许它截获到的所有访问请求。 这意味着所有事件在审核日志中记录的授权结果为 *P*（允许）。

以下限制适用于仅审核模式：

- 没有审核记录会发送到 Unicenter。

  在仅审核模式中，允许所有事件 (*P*)。 允许的事件不会发送到 Unicenter。

- *不*考虑资源和用户的审核属性。

  仅审核模式记录*所有截获*事件，不考虑是特定于资源的设置还是特定于用户的设置。

设置仅审核模式

*仅审核模式*将记录所有拦截事件，而不检查或强制执行访问规则。 使用这种模式可收集符合遵从要求或规定的数据。

要设置仅审核模式，请将 SeOSD\GeneralInterceptionMode CA Access Control 注册表项设置为 1。

重要说明！ 如果使用仅审核模式，请确保有足够的磁盘空间来存储审核日志，并确保审核日志的大小限制足够大。 还要考虑到审核日志备份 (p. 125)的选项。

警告模式

*警告模式*是一个可应用到资源的属性、可应用到类的选项。 如果警告模式已应用到资源或类，且访问违反了访问规则，则 CA Access Control 会写入审核日志条目和返回代码 W，但允许访问资源。 如果类处于警告模式，则该类中的所有资源都处于警告模式。

仅在 CA Access Control 处于完全增强模式时警告模式才有效。

注意：完全强制模式是 CA Access Control for UNIX 所支持的唯一模式。 CA Access Control for Windows 还支持仅审核模式。

当您创建或修改访问策略时，可以使用警告模式。 如果使用警告模式，则在实施策略之前，您可以查看审核日志以预览该所需策略的结果。 您可以使用 seaudit 命令显示审核日志。

如果类具有属性 *warning*，则您可以将该类置于警告模式。 如果资源组或类处于警告模式，则当访问违反访问规则时，CA Access Control 将允许该访问，并在引用资源（非资源组或类）的审核日志中写入条目。

资源和类的警告模式设置是独立的：如果您将资源置于警告模式，即使它属于某个类并且您从该类中删除了警告模式，该资源仍将处于警告模式。

注意：如果资源或类具有属性 *warning*，则您可以将其置于警告模式；并非所有资源或类都有该属性。

更多信息：

仅审核模式 (p. 96)

将资源置于警告模式

将资源置于警告模式可监控访问规则的效果，而无需强制执行这些规则。

注意：除了将单独资源置于警告模式外，您还可以将类置于警告模式 (p. 99)。

将资源置于警告模式

1. 在 CA Access Control 端点管理 中编辑要置于警告模式的资源。

   将显示相应的“修改”页面。

2. 单击“审核”选项卡。

   将显示针对该资源的“审核模式”页面。

3. 选择“警告模式”，然后单击“保存”。

   您修改的资源现在处于警告模式。

注意：在警告模式中，当允许访问但该访问违反访问规则时，CA Access Control 始终将警告记录写入审核日志：您无需通过对资源设置审核属性来执行该操作。

使用 sereport 实用程序（报告编号 6）查看处于警告模式的所有资源。

示例：将文件置于警告模式

以下 selang 示例将文件 c:\myfile 置于警告模式：

```
chres FIlE c:\myfile warning
```

示例：清除文件的警告模式

以下 selang 示例将文件 c:\myfile 的警告模式清除：

```
chres FIlE c:\myfile warning-
```

myfile 现在不处于警告模式，因此 CA Access Control 将对 myfile 强制执行访问规则。

示例：将终端置于警告模式

以下 selang 示例将终端 myterminal 置于警告模式：

```
chres terminal myterminal warning
```

CA Access Control 允许任何授权用户从终端 myterminal 进行访问，但会对任何通常被拒绝从该终端访问的用户记录审核记录。

将类置于警告模式

您可以将类中的所有记录置于警告模式，而不是将记录逐个置于警告模式。 您可以使用警告模式来监控访问规则的结果，而无需强制执行这些规则。

将类置于警告模式

1. 在 CA Access Control 端点管理 中，执行如下操作：
   a. 单击“配置”。
   b. 单击“类激活”。
   将显示“类激活”页面。
2. 为要置于警告模式的类选中“警告”列中的复选框。
3. 单击“保存”。
   此时将显示一条确认消息，通知您已成功更新 CA Access Control 选项。

找出处于警告模式的资源

实施 CA Access Control 时，您应使用警告模式作为临时措施。 如果您确定用户具有访问其所需资源的相应权限，则应关闭警告模式，之后 CA Access Control 将开始强制执行相关联的规则。

要找出处于警告模式的资源，可以创建一个显示所有处于警告模式的资源的报告。

要创建报告，请输入以下命令：

```
sereport -f pathname.html -r 6
```

CA Access Control 将创建报告。

注意：有关 sereport 实用程序的详细信息，请参阅 *《参考指南》*。

找出处于警告模式的类

实施 CA Access Control 时，您应使用警告模式作为临时措施。 如果您确定用户具有访问其所需资源的相应权限，则应关闭警告模式，之后 CA Access Control 将开始强制执行相关联的规则。

要找出处于警告模式的类，您可以使 CA Access Control 显示此数据。

要显示此数据，请输入以下 selang 命令：

```
setoptions cwarnlist
```

CA Access Control 将显示一个表，其中显示处于警告模式的类。

注意：有关 setoptions 的详细信息，请参阅 *《selang 参考指南》*。

如何执行系统维护

您可能需要在特定时间执行系统维护来升级系统、安装新应用程序等等。 在系统维护期间，您应当在警告模式下设置 CA Access Control 规则。 一旦您确定维护不会影响用户访问其所需要的资源，则应关闭警告模式，之后 CA Access Control 将开始强制执行相关联的规则。

要在执行系统维护时使用警告模式，请执行以下操作：

1. 使用以下 selang 规则在开始维护之前将适当的类设置为警告模式：

   ```
   setoptions class(NAME) flags(W)
   ```

2. 执行维护。
3. 在执行维护后运行 seretrust 实用工具。

   seretrust 实用程序可生成重新托管在数据库中定义的程序和安全文件所需的 selang 命令。
4. 运行 selang 命令来重新信任在数据库中定义的程序。
5. 使用以下 selang 规则从类中删除警告模式以便启用策略实施：

   ```
   setoptions class(NAME) flags-(W)
   ```

6. 查看 CA Access Control 审核日志文件。

   审核日志包含受维护影响的资源的警告。

注意：有关 seretrust 实用程序的详细信息，请参阅*《参考指南》*。

监视访问控制活动

CA Access Control 跟踪是实时日志，可以显示 CA Access Control 执行的每项操作。 跟踪记录收集在 *ACInstallDir*\log\seosd.trace（其中 *ACInstallDir* 是您安装 CA Access Control 的目录）中。

或者汇集在指定为注册表子项中的 trace_file 值的任何文件中：

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl\SeOSD\
```

尽管您可以从跟踪文件中筛选记录，但是跟踪机制是为系统监视而设计的，而不是为安全审核而设计的。

默认情况下，CA Access Control 仅在初始化期间生成跟踪消息。 CA Access Control 初始化之后，它将停止跟踪机制，而且不会生成跟踪消息。

跟踪记录筛选

CA Access Control 生成两种类型的跟踪记录：

- 用户跟踪记录 — 记录用户完成的操作，例如，user1 访问文件 c:\tmp\tmp.exe。
- 一般跟踪记录 — 记录系统完成的操作，例如，Watchdog 将某程序设为非信任的。

跟踪记录被写入 seos.trace 文件，并且可以使用 trcfilter.ini 文件进行筛选。

如果将用户设置为可跟踪，则每次写入该用户的跟踪记录时，将向 seos.audit 文件写入匹配的审核记录。 审核记录根据 audit.cfg 文件进行筛选。

注意：由跟踪事件生成的审核记录不被缓存，并且总是经历完全的强制措施流。

下列 selang 命令可将用户设为可追踪的：

```
editusr userName audit(trace)
```

要查看跟踪记录或审核记录，请使用 seaudit 实用工具。

筛选跟踪记录

使用跟踪筛选文件，可以指定不应在跟踪文件中显示的某些活动类型。跟踪筛选文件是使用注册表键中的 *trace_filter* 值指定的：

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl\SeOSD
```

默认值为 *ACInstallDir*\log\trcfilter.ini（其中 *ACInstallDir* 是您安装 CA Access Control 的目录）。

重要说明！ CA Access Control 将在安装期间创建跟踪筛选文件，该文件中只有一行：*seosd.trace*。 请勿删除该记录。

跟踪筛选文件中的每一行都表示一种*不*应跟踪的访问或活动。 例如，要消除对用户访问 Microsoft Word 的跟踪，请在跟踪筛选文件中添加以下行：

```
*winword.exe*
```

CA Access Control 审核内容

为了安全审核，CA Access Control 根据在数据库中定义的审核规则以及操作中的强制模式，会保留截获事件的审核记录。 审核日志中的记录将根据这些审核规则进行汇集。

完全审核提供以下所有截获事件的审核记录：

- 文件访问（FILE 类）
- 程序执行（PROGRAM 类）
- 注册表访问（REGKEY 和 REGVAL 类）。
- 模仿控制（SURROGATE 类）
- 网络控制（CONNECT、TCP、HOST、GHOST、HOSTNET 和 HOSTNP 类）
- 登录（TERMINAL 类）

  注意：不会缓存截获的登录事件，它们始终遵守截获事件的审核进程。
- 服务保护（WINSERVICE 类）
- 密码验证失败（PASSWORD 类）
- 进程终止（PROCESS 类）

是否记录事件取决于 CA Access Control 截获模式。

登录拦截限制

Windows 上的登录拦截仅受 CA Access Control 子身份验证方法支持。

无法通过内核设置登录拦截。 因此，您应该考虑以下方面：

- 由于在 Windows 域环境中，子身份验证组件在域控制器 (DC) 级别上工作，且由操作系统来决定对用户登录事件（以及触发 CA Access Control 子身份验证模块）进行验证的 DC，因此，每个 DC 上均需安装 CA Access Control。
- 在 Windows 环境中运行时，CA Access Control 登录策略（TERMINAL 规则）需要位于 DC 上，而无需位于目标服务器上。

  例如，如果您希望保护或审核由文件服务器（该服务器属于 Windows 域的一部分，而不属于 DC）上的域用户执行的登录事件，则需要在 DC 上而非目标文件服务器上定义 CA Access Control 登录策略。 这是因为域用户访问共享文件目录时，将在 DC 而非文件服务器上发生登录身份验证。

- 存在多个 DC 时，可能会在其中一个 DC 上处理 CA Access Control 登录身份验证。 因此，我们建议您在所有 DC 之间同步 CA Access Control 登录策略。

  可以通过以下两种方法实施此同步：通过策略模型机制，在此机制中，所有 DC 均为 PMDB 的订户，或者通过将所有 DC 添加至主机组，然后使用高级策略管理部署常用策略。

- 登录事件相对应的某些用户属性，将在事件身份验证运行时更新。 这些属性可能并不同步，因为登录身份验证仅在其中一个 DC 上进行。 这些属性为 *Gracelogins*、*Last accessed* 和 *Last access time*。

  也就是说，例如，用户属性 *Last access time* 的值在 DC 之间可能会有所不同，因为 CA Access Control 子身份验证是在其中一个 DC 上触发的，而不是在所有 DC 上。

- 要强制执行本地用户（即不是域用户）登录事件，需要将 CA Access Control 安装在本地用户需具备访问权限的本地计算机上。 这是由于本地计算机被用作域计算机（域即为本地计算机）。

- 与先前 CA Access Control 版本设置相同，远程桌面协议 (RDP)/终端服务登录事件均强制在目标服务器上执行。 但是，对于 RDP 登录事件，应在目标服务器上定义 CA Access Control 登录策略。

完全强制模式中 CA Access Control 审核的内容

在完全强制模式（正常操作）中，CA Access Control 如下记录事件：

- 如果对截获资源*禁用*警告模式，则 CA Access Control 将强制执行规则并根据资源或用户的*审核*属性记录事件。

审核属性	记录的事件
ALL	*全部*
SUCCESS	允许访问
FAIL	访问被拒绝

- 如果对截获资源*启用*警告模式，则如果访问请求违反了访问规则，将向审核日志写入记录（如果强制执行规则，请求将会失败）。 审核记录会表明由于使用的是警告模式，允许此违反行为。

  该模式下不强制执行规则。

仅审核模式中 CA Access Control 审核的内容

在仅审核模式中，CA Access Control 不处理授权或强制执行规则的请求。 访问者的所有拦截登录事件以及 CA Access Control 保护的资源的所有拦截事件都将被记录，无论访问是否成功或失败。

如何更改 CA Access Control 写入审核日志中的内容

有两种方法可以更改 CA Access Control 写入审核日志的内容：

- 使用资源或访问者的 AUDIT 属性来定义 CA Access Control 写入审核日志的审核事件。

  注意：可以使用 GROUP 或 XGROUP 的 AUDIT 属性来设置组中所有成员的审核属性。 但是，您可以使用 AUDIT 属性来设置组成员的审核模式（如果用户的审核模式在 USER 记录、XUSER 记录或配置文件组中有所定义）。

- 使用审核配置文件 audit.cfg 来筛选 CA Access Control 发送到审核日志中的事件。 您不能使用 audit.cfg 文件将事件添加到审核日志。

要减少审核记录的数量，您也可以控制写入日志文件中的连续审核事件。 此自定义操作基于连续匹配审核事件（即使用相同的进程 ID、线程 ID、规则 ID、用户 ID 和访问掩码来访问资源）之间的时间间隔。 时间间隔以秒为单位，可以通过设置 AuditRefreshPeriod 注册表项的值来进行设置。 默认情况下，AuditRefreshPeriod 设置为零 (0)，这意味着所有事件均将写入日志文件。

设置审核规则

为了安全审核，CA Access Control 将根据在数据库中定义的审核规则，保留拒绝或授权访问事件的审核记录。

每个访问者和每种资源都具有 AUDIT 属性，可以设置为以下一个或多个值：

FAIL

记录访问者对资源的访问失败。

SUCCESS

记录访问者对资源的成功访问。

LOGINFAIL

记录访问者的每一次登录失败。 （该值不适用于资源。）

LOGINSUCCESS

记录访问者的每一次成功登录。（该值不适用于资源。）

ALL

记录与访问者的 FAIL、SUCCESS、LOGINFAIL 和 LOGINSUCCESS 或资源的 FAIL 和 SUCCESS 相同的信息。

无

不记录任何有关访问者或资源的信息。

无论您何时在数据库中创建或更新访问者或资源记录，都可以指定 AUDIT 属性。 此外，您还可以指定是否应该发送以及向何人发送记录事件的电子邮件通知。

审核日志中的记录将根据这些审核规则进行汇集。 是否记录某个事件的决策基于以下情况：

- 如果资源或访问者具有 AUDIT(ALL)，则将记录访问者的所有登录事件以及与 CA Access Control 所保护资源有关的所有事件，而无论访问失败还是成功。
- 如果对 CA Access Control 所保护资源的访问成功，且访问者或资源具有 AUDIT(SUCCESS)，则将记录该事件。
- 如果对 CA Access Control 所保护资源的访问失败，且访问者或资源具有 AUDIT(FAIL)，则将记录该事件。

此外，如果您将用户设置为可跟踪用户，该用户的每次跟踪记录被写入时，对应的审核记录也会被写入审核日志中。

定义 CA Access Control 写入审核日志的审核事件

CA Access Control 将成功和失败的访问记录写入到审核日志。 要定义 CA Access Control 将哪些访问事件写入到审核日志，请更改要审核的资源或访问者的 AUDIT 属性值。 您还可以使用该方法指定 CA Access Control 将每个跟踪事件记录到审核日志。

使用 AUDIT 属性来指定 CA Access Control 写入到审核日志的审核事件。 使用 selang 或 CA Access Control 端点管理 来为以下资源和访问者设置 AUDIT 属性：

AUDIT 值	CA Access Control 记录的内容	适用对象
FAIL	失败访问	用户和资源

AUDIT 值	CA Access Control 记录的内容	适用对象
SUCCESS	成功访问	用户和资源
LOGINFAIL	失败登录	用户
LOGINSUCCESS	成功登录	用户
ALL	等同于 FAIL、SUCCESS、LOGINFAIL、LOGINSUCCESS 和 INTERACTIVE	用户和资源
TRACE	等同于 ALL 和所有系统事件	用户
INTERACTIVE	UNIX 计算机上的用户会话	用户
无	无登录	用户和资源

注意：如果未设置用户的审核属性，组或配置文件组的 AUDIT 值则可以影响 CA Access Control 针对用户所使用的审核模式。

CA Access Control 如何为用户确定审核模式

用户的审核模式指定 CA Access Control 将哪些审核事件发送至该用户的审核日志中。 以下过程说明 CA Access Control 如何确定用户的审核模式：

1. CA Access Control 检查 USER 或 XUSER 类中的用户记录是否有针对 AUDIT 属性的值。

   如果用户的记录有针对 AUDIT 属性的值，CA Access Control 则使用此值作为该用户的审核模式。

2. CA Access Control 检查是否将用户分配到配置文件组。 如果将用户分配到配置文件组，CA Access Control 则检查 GROUP 类中配置文件组的记录是否有针对 AUDIT 属性的值。

   如果将用户分配到配置文件组且配置文件组的记录有针对 AUDIT 属性的值，CA Access Control 则使用此值作为该用户的审核模式。

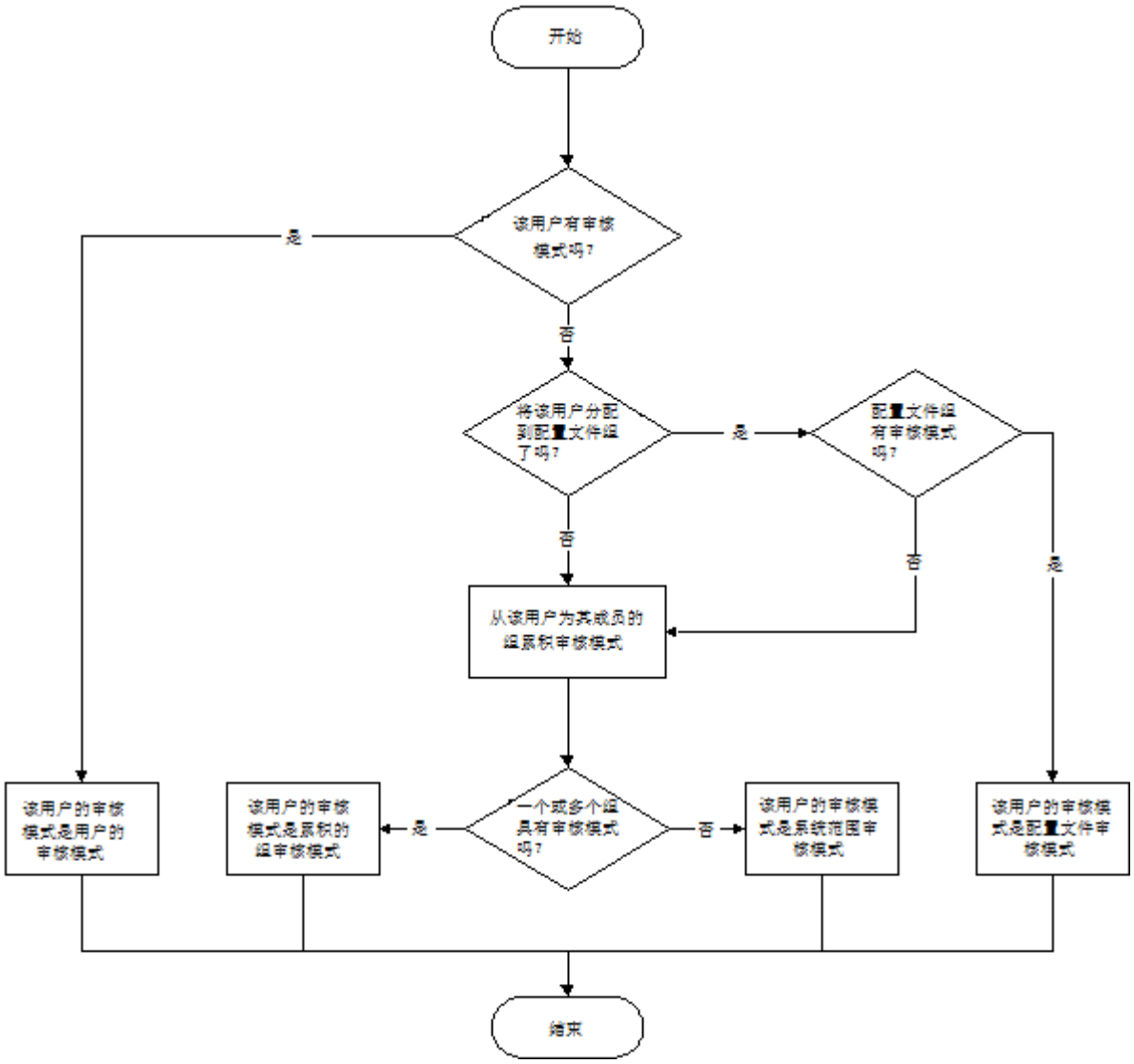
3. CA Access Control 检查用户是否为组中成员。 如果用户是组成员，CA Access Control 则检查 GROUP 或 XGROUP 类中组的记录是否有针对 AUDIT 属性的值。

   如果用户是组成员且组的记录有针对 AUDIT 属性的值，CA Access Control 则使用此值作为该用户的审核模式。 如果用户不是组成员或组的记录没有针对 AUDIT 属性的值，CA Access Control 则将系统范围审核模式分配给该用户。

   注意：如果用户是多个组中的成员且这些组有不同的审核模式，那么用户的审核模式会累积。 即该用户的审核模式是成员所在组的所有审核模式的总和。

注意：如果 CA Access Control 使用组的 AUDIT 属性值来确定用户的审核模式，而且在用户登录时更改了组的审核模式，那么已登录用户的审核模式也发生变化。 该用户不需注销使组审核模式中的更改生效。

以下图表显示 CA Access Control 如何确定用户的审核模式：



示例：按组审核

用户 Jan 同为组 A 和组 B 的成员。 组 A 有 FAIL 的审核模式，组 B 有 SUCCESS 的审核模式。 由于 Jan 是两个组的成员，因此 Jan 有 FAIL 和 SUCCESS 的累积审核模式。

更多信息：

CA Access Control 如何使用配置文件组确定用户属性 (p. 40)

用户和企业用户的默认审核模式

当您创建用户（USER 对象）时，CA Access Control 将默认 AUDIT_MODE 分配给该对象。 AUDIT_MODE 属性的默认值为 Failure、SuccessLogin、SuccessFailure。

当您创建企业用户（XUSER 对象）时，默认情况下，CA Access Control 不会将默认 AUDIT_MODE 值分配给该对象。

注意：(UNIX) 要针对 USER 对象更改 AUDIT_MODE 属性的默认值，请在 lang.ini 文件的 [newusr] 部分中编辑 DefaultAudit 值。

为某些用户更改为默认审核值

r12.0 SP1 CR1 之前，针对以下访问者默认审核模式为“无”：

- 在相应的 USER 类记录中没有定义的 AUDIT 值，没有与定义的 AUDIT 值的配置文件组相关联的用户。
- 没有在数据库（由 _undefined 用户记录表示）定义的任何用户。

  注意：如果使用企业用户，CA Access Control 则不会将任何用户看作未定义。 针对 _undefined 用户的属性在这种情况下不适用。

从 r12.0 SP1 CR1， 这些访问者的默认审核模式为 Failure、LoginSuccess 和 LoginFailure。 要获得早期行为，请将这些用户的 AUDIT 属性值设置为“无”。

对于 GROUP 记录，更改 AUDIT 属性的值

如果具有两个函数的 GROUP 记录：

- 为一组用户定义审核策略的配置文件
- 针对第二组用户的容器

r12.0 SP1 CR1 之前，GROUP 记录也为第二组用户定义审核策略。 为避免由于更改操作而可能引发的问题，请为第二组用户创建单独的 GROUP。

在 Windows 中设置审核策略

除了设置访问者和资源的访问规则以外，您还可以指定要写入审核日志中的 Windows 事件。 您可以按照组、配置文件组或按照用户，为整个组织指定此类审核策略。

示例：为配置文件组的所有成员设置审核策略

以下示例说明了如何为属于配置文件组一部分的所有用户设置审核策略：

1. 使用所需的审核模式创建新的配置文件组。 例如：

   ```
   newgrp profileGroup audit(failure) owner(nobody)
   ```

2. 创建新用户并将其附加到创建的配置文件组中。 例如：

   ```
   newusr user1 profile(profileGroup) owner(nobody)
   ```

3. 删除用户的审核设置。 例如：

   ```
   chusr user1 audit-
   ```

您现在可以检查该设置是否有效：

1. 以新用户的身份登录：

   ```
   runas /user:user1 cmd.exe
   ```

2. 在 user1 的命令提示窗口中，输入以下内容：

   ```
   secons -whoami
   ```

   该命令显示用于授权的信息，该信息保存在 user1 的 ACEE 中。

   ```
   ACEE audit mode is: Failure; Originated from Profile group definition
   ```

   该消息确认了审核策略是源于用户附加到的配置文件组所。

示例：为组成员设置审核策略

在此示例中，名为“Forward Inc”的虚构公司要使用 CA Access Control 来保护 /production 目录中的所有文件。 /production 目录在本地环境中具有完全访问权限。

Forward Inc 想要拒绝并审核所有试图访问 /production 目录的操作。 然而，Forward Inc 又允许开发人员对 /production 目录进行读取。 而且不审核该访问。 拒绝并审核开发人员试图写入 /production 目录的操作。

开发人员可以要求对 /production 目录的完全访问权限。 Forward Inc 审核具有完全访问权限的用户在 /production 目录中执行的所有活动。

以下过程说明 Forward Inc 实施以上方案要采取的步骤：

1. 在本地环境中创建名为“Developer”的组。 将所有开发人员添加到此组。
2. 在本地环境中创建名为“Dev_Access_All”的组。 不向此组添加任何用户。
3. 定义 /production 目录的一般访问规则，如下所示：

   ```
   authorize FILE /production/* access(none) uid(*)
   ```

   该规则将默认访问设置为“none”。
4. 定义 /production 目录的一般审核规则，如下所示：

   ```
   editres FILE /production/* audit(failure)
   ```

   该规则审核访问 /production 目录的所有失败操作。
5. 定义 Developer 组的访问规则，如下所示：

   ```
   authorize FILE /production/* access(read) xgid(Developers)
   ```

   该规则允许 Developer 组成员对 /production 目录具有读取权限。

   注意：在步骤 4 中设置的规则可以帮助确保 CA Access Control 审核所有用户（包括 Development 组成员）执行的所有失败访问操作。
6. 如下定义 Dev_Access_All 组的访问规则：

   ```
   authorize FILE /production/* access(all) xgid(Dev_Access_All)
   ```

   该规则允许 Dev_Access_All 组成员对 /production 目录具有完全访问权限。
7. 定义 Dev_Access_All 组的审核规则，如下所示：

   ```
   chxgrp Dev_Access_All audit(all)
   ```

   该规则审核 Dev_Access_All 组成员执行的每一个操作。
8. 当 Developer 组成员需要对 /production 目录的完全访问权限时，请将用户添加到本地环境中的 Dev_Access_All 组。

   该用户对 /production 目录有完全访问权限，且 CA Access Control 审核用户执行的每个操作。

   注意：该用户必须启动新的登录会话才能使组成员身份中的变化生效。
9. 用户完成了在 /production 目录中的任务后， 请从本地环境中的 Dev_Access_All 组中删除该用户。

   该用户现在对 /production 目录有读取权限。 CA Access Control 拒绝并审核 /production 目录中用户所执行的所有其他访问操作。

   注意：该用户必须启动新的登录会话才能使组成员身份中的变化生效。

审核进程

要配置 CA Access Control 以符合审核要求，您必须首先要了解审核的工作原理。 通过审核，您可以跟踪 CA Access Control 截获到的访问请求（事件）。 可以使用该数据满足遵从要求，分析并提取符合安全要求的访问规则或监控访问请求。

CA Access Control 所遵守的用于将审核事件记录到日志中的进程取决于截获的事件类型：

- 截获事件 (p. 114)

  注意：拦截登录事件（TERMINAL 类），由用户所生成的审核记录跟踪，不被贮藏；他们总是遵循拦截事件的审核过程。

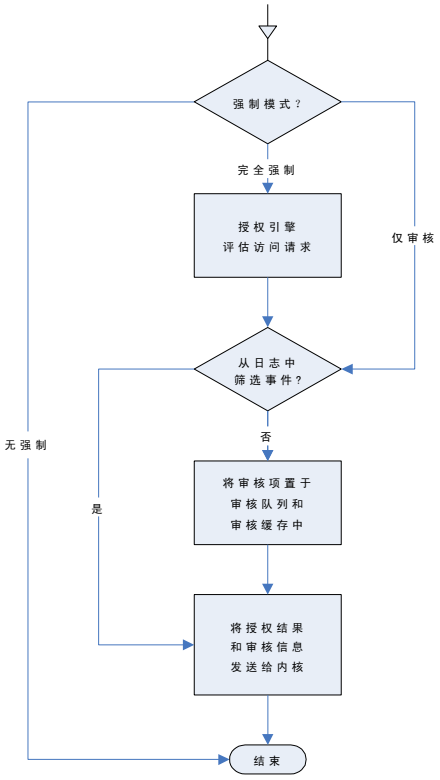
- 审核事件 (p. 115)

注意：CA Access Control 仅在相应的类处于活动状态且数据库包含参与该事件的规则时才截获事件。

截获事件的审核工作原理

*截获事件*是 CA Access Control 首次遇到的事件，其内核缓存中不存在授权信息或审核信息。

为记录审核记录，CA Access Control 将执行以下操作并对截获事件产生相应影响：



- 在非强制模式中，不截获或审核事件。
- 在完全强制模式中，CA Access Control 执行以下操作：
  1. 授权引擎根据授权结果将审核项目放置在审核队列和审核缓存中。

     CA Access Control 仅在将资源或访问者的审核属性设置为审核结果事件且没有将审核筛选文件设置为筛选该事件时，才写入审核项目。
  2. 授权引擎根据授权结果中返回信息答复，并将审核相关信息返回到内核。

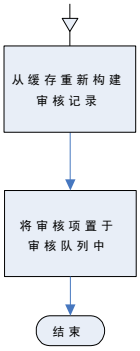
- 在仅审核模式中，CA Access Control 不处理授权请求。 无论资源和用户的审核属性是什么，将始终写入审核信息。

  仅在审核筛选文件没有设置为筛选该事件时，CA Access Control 才写入审核项目。 在该模式中的授权结果总为 *P*（允许）。

注意：不会缓存拦截到的登录事件（TERMINAL 类）和用户跟踪生成的审核记录；授权引擎总是写入这些事件的审核记录。

审核事件的审核工作原理

以下图表和步骤说明了审核事件的审核工作原理：



内核通知 CA Access Control 缓存的截获事件后，CA Access Control 将执行以下操作以记录该审核事件：

1. 使用审核缓存中内核发送的信息重新建立审核数据
2. 将审核项目放入审核队列中

内核和审核缓存

*内核缓存*包含有关先前截获事件的数据。 内核标识此类缓存的截获事件（审核事件），然后将其发送到 CA Access Control 进行处理。 本质上来讲，CA Access Control 使用内核缓存截获事件，该事件遵循与先前截获事件相同的模式。

*审核缓存*包含的数据可使 CA Access Control 重新建立再次发生的审核记录，并将这些记录发送到审核队列，而无需遵守授权进程。 这意味着如果截获事件在缓存（审核事件）中存在足够信息，截获事件就会被快速处理并添加到审核队列。 授权引擎提供存储在内核中的数据和来自截获的初始事件结果的审核缓存（截获事件）。

缓存重置

CA Access Control 在以下情况中清除内核和审核缓存：

- 数据库更改

  CA Access Control 在数据库信息更改时会清除所有缓存。 新的或修改的访问规则使现有缓存有可能不正确。

- 到时间检查点

  CA Access Control 在时间检查点影响任意事件的授权结果时会清除所有缓存。 DAYTIME 限制属性或 HOLIDAY 类记录更改时，授权结果可能也会更改且缓存可能不正确。

- PROGRAM 资源更改

  CA Access Control 在 Watchdog 标识 PROGRAM 资源已经更改且不受信任时会清除所有缓存。 不受信任的程序会影响有关该程序的授权请求结果。 这就会使缓存可能不正确。

- 审核缓存填充

  CA Access Control 在审核缓存被占满时会清除缓存项目（最近最少使用的项目）的 10%。

清除缓存后，来自新截获事件的信息需要重新填充缓存，让 CA Access Control 截获审核事件。

查看审核事件

CA Access Control 将审核事件发送到审核日志。 使用以下 CA Access Control 工具查看审核日志：

- CA Access Control 端点管理
- seaudit 实用程序

您可以配置 CA Access Control 将审核事件也发送到 Windows 事件日志。 事件日志存储来自单个收集中的各种应用程序的审核事件。 使用 Windows 事件查看器查看事件日志中的审核事件。

Windows 事件日志中的审核事件

Windows 事件日志存储来自单个收集中的各种源的审核事件。 如果您配置 CA Access Control 将审核事件传递到事件日志，那么每次 当 seosd 将审核事件写入 CA Access Control 审核日志时，都会将相应的事件发送给事件日志。

audit.cfg 文件筛选来自审核日志和事件日志的审核事件。 如果不将审核事件写入审核日志，则不会将其发送给事件日志。

Windows 2008 事件日志也将审核事件传递到名为“通道”的容器中，这取决于审核事件的数量、用户和原始应用程序。 CA Access Control 通道命名为 CA-AccessControl-AuthorizationEngine/Audit。

如果您已经在 Windows 2008 服务器上部署了 CA Access Control，您即可选择将审核事件发送到：

- 事件日志
- 通道
- 事件日志和通道
- 既不是事件日志也不是通道

将审核事件传递到 Windows 事件日志

如果您配置 CA Access Control 将审核事件传递到 Windows 事件日志，那么每次当 seosd 将审核事件写入 CA Access Control 审核日志时，都会将相应的事件发送给事件日志。 您也能配置 CA Access Control 将策略模型审核事件发送到事件日志。

将事件传递到到事件日志

1. 使用以下命令停止 CA Access Control：

   ```
   secons -s
   ```

   CA Access Control 停止。

2. 将 logmgr 部分中 SendAuditToNativeLog 配置设置的值设为 1。

   将审核事件发送给 Windows 事件日志。

3. （可选）将 Pmd 部分中 SendAuditToNativeLog 配置设置的值设为 1。

   将策略模型的审核事件发送给 Windows 事件日志。

4. 使用以下命令重新启动 CA Access Control：

   ```
   seosd -start
   ```

   CA Access Control 重新启动。

示例：将审核事件传递到事件日志

下列示例将审核事件传递到事件日志。 您必须是在远程配置环境 (env config) 中才能使用该命令：

```
er config ACROOT section(logmgr) token(SendAuditToNativeLog) value(1)
```

示例：将策略模型审核事件传递到事件日志

下列示例将策略模型审核事件传递到事件日志。 您必须是在远程配置环境 (env config) 中才能使用该命令：

```
er config ACROOT section(Pmd) token(SendAuditToNativeLog) value(1)
```

更多信息：

更改配置设置 (p. 173)

将审核事件传递到 Windows 事件日志通道

仅适用于 Windows Server 2008

如果您配置 CA Access Control 将审核事件传递到 Windows 事件日志，那么每次当 seosd 将审核事件写入 CA Access Control 审核日志时，都会将相应的事件发送给事件日志通道。 CA Access Control 事件日志通道命名为 CA-AccessControl-AuthorizationEngine/Audit。

您也能配置 CA Access Control 将策略模型审核事件发送到事件日志通道。 策略模型事件日志通道被命名为 CA-AccessControl-Policy Models/Audit。

将事件传递到到事件日志通道

1. 使用以下命令停止 CA Access Control：

   ```
   secons -s
   ```

   CA Access Control 停止。

2. 将 logmgr 注册表子键中的 SendAuditToNativeChannel 标记的值设为 1。

   将审核事件发送给 Windows 事件日志通道。

3. （可选）将 Pmd 注册表子键中的 SendAuditToNativeChannel 标记的值设为 1。

   将策略模型审核事件发送给 Windows 事件日志通道。

4. 使用以下命令重新启动 CA Access Control：

   ```
   seosd -start
   ```

   CA Access Control 重新启动。

示例：将审核事件传递到事件日志通道

下列示例将审核事件传递到事件日志通道。 您必须是在远程配置环境 (env config) 中才能使用该命令：

```
er config ACROOT section(logmgr) token(SendAuditToNativeChannel) value(1)
```

示例：将策略模型审核事件传递到事件日志通道

下列示例将策略模型审核事件传递到事件日志通道。 您必须是在远程配置环境 (env config) 中才能使用该命令：

```
er config ACROOT section(Pmd) token(SendAuditToNativeChannel) value(1)
```

审核日志

审核日志存储在文件中。 以下 Windows 注册表子键中的值 *audit_log* 指定了审核日志文件的位置：

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl\logmgr
```

该键的默认值为：

```
C:\Program Files\CA\AccessControl\\log\seos.audit
```

默认情况下，审核日志达到 1024 KB 时，CA Access Control 会自动对其进行备份。 您可以通过在子键中更改值 *audit_size* 值来更改此大小：

```
HKEY_LOCAL_MACHINE\Software\ComputerAssociates\AccessControl\logmgr
```

您还可以选择通过更改 Windows 注册表子键中的值 *BackUp_Date*，来定期（每天、每周或每月）备份审核日志：

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl\logmgr
```

注意：有关这些注册表子键的详细信息，请参阅*《参考指南》*。

使用审核日志

CA Access Control 提供两种内置工具，用于查看、筛选和搜索审核日志：

- CA Access Control 端点管理
- seaudit 实用程序

您可以显示审核日志中的每条记录，也可以使用筛选器从审核日志中选择特定记录。

本章的其余部分将说明在 CA Access Control 端点管理 中使用审核筛选时，如何查看审核日志中的记录。

审核记录筛选

audit.cfg 可通过定义不应发送至审核文件的记录来筛选主机上的审核记录。 文件中的每一行都代表一条用于筛选审核信息的规则（即，与行中条件匹配的审核记录将不出现在审核文件中）。 该筛选器可以只保留需要的记录，从而有助于限制 seos.audit 文件的大小。 您可以编辑 audit.cfg 文件来适合您的企业要求。

默认情况下，audit.cfg 文件位于 *ACInstallDir*/etc 目录 (UNIX) 或 *ACInstallDir*\data 目录 (Windows) 中。 您可以通过编辑 seos.ini 文件 (UNIX) 中的 [logmgr] AuditFiltersFile 标记或 logmgr 注册表键 (Windows) 中的 AuditFiltersFile 条目，更改 audit.cfg 文件的位置。

CA Access Control 引擎 seosd 在启动时会读取 audit.cfg 文件。 当向审核文件发送消息时，seosd 会检查该消息是否与 audit.cfg 文件中的下列规则之一匹配： 如果消息匹配规则，该消息不会被写入审核文件中。

注意：有关 audit.cfg 文件的详细信息，请参阅 *《参考指南》*。

审核显示筛选

审核日志中的记录数目可能会非常庞大。 要减少*显示*的记录数目，请使用筛选来选择要显示的记录类型。 您可以依据各种条件（包括时间或事件类型）来筛选事件。

注意：您也可以使用审核配置设置（audit.cfg 文件）筛选 CA Access Control *写入*审核文件的审核记录。

只需为筛选命名并至少选择一个开关参数，即可在 CA Access Control 端点管理 中创建筛选。 然后可以选择其他开关参数，并可以选择分配一个或多个选项。 还可以使用 seaudit 工具来筛选记录。

CA Access Control 端点管理 提供了几个预定义的筛选，您也可以创建自己的筛选。

筛选向导, 选择名称和开关参数页面

筛选向导的“选择名称和开关参数”页面用于定义要创建的审核显示筛选的名称，以及要对该筛选应用的开关参数。

此窗口包含下列窗口项：

筛选名

定义要创建的审核显示筛选的名称。

审核事件记录

指定是要通过该筛选显示所有审核记录还是仅显示选择的开关参数。

如果选择列出所有记录，则将不应用此页面上的开关参数。

列出主机和服务的 INET 审核记录

指定是否列出从指定服务的指定主机接收的 TCP 请求的 INET 审核记录。 主机 (host) 和服务 (service) 是标识搜索的主机和服务集的掩码。

显示终端上用户的 LOGIN

指定列出以下内容：

- 指定用户在指定终端上的 LOGIN 记录。 *user* 和 *terminal* 都是您定义的掩码。
- 多次输入无效密码时由授权引擎创建的记录。

列出用户资源的类的 RESOURCE 审核

指定是否列出资源记录。 可以在以后定义以下内容：

- *Class*—标识被访问资源所属类的掩码。
- *Resource*—标识被访问资源的名称的掩码。
- *User*—标识访问资源的用户名的掩码。

列出数据库更新

列出数据库更新审核记录。 可以定义：

- *Cmd*—确定要搜索的 selang 命令的掩码。
- *Class*—标识要搜索的类的掩码。
- *Object*—标识要搜索的记录的掩码。
- *User*—标识已执行这些命令的用户的掩码。

列出启动/关闭消息

指定是否列出来自 CA Access Control 服务的启动和关闭消息。

列出 WATCHDOG 审核记录

指定是否列出 Watchdog 审核记录。

仅显示跟踪记录

指定是否仅列出由跟踪工具发送到审核日志中的记录。

筛选向导，编辑选项页面

筛选向导的“编辑选项”页面用于定义要应用到审核显示筛选的选项。

此窗口包含下列窗口项：

列表开始日期今日

指定今日作为开始日期。 不列出今日之前记入日志的记录。

列表的开始日期

指定开始日期。 不列出该指定日期之前的记录。

列表的开始时间

指定开始时间。 不列出该指定时间之前的记录。

列表的结束日期。

指定结束日期。 不列出该指定日期之后的记录。

列表的结束时间

指定结束时间。 不列出该指定时间之后的记录。

显示 Internet 地址，而不是主机名

指定列出 Internet 地址，而不是 TCP/IP 记录中的主机名。

隐藏失败

指定不列出故障。

隐藏所有授权的访问

指定不列出成功（授权）的访问。

隐藏注销记录

指定不列出注销记录。

隐藏 NOTIFY 审核记录

指定不列出 NOTIFY 审核记录。

隐藏密码尝试和操作

指定不列出密码尝试记录。

隐藏警告记录

指定不列出警告记录。

显示端口号，而不是名称

指定列出端口号，而不是服务名。

只显示源于*主机*的记录

指定仅列出来源于指定主机的记录。该选项仅在连接至 UNIX 工作站时适用。

预定义筛选

CA Access Control 附带了以下预定义的筛选：

所有记录

显示审核日志中的每条记录。 不执行任何筛选。

今日记录

显示今天创建的每条记录。

最后 2 天记录

显示昨天和今天创建的每条记录。

最后 7 天记录

显示过去七天期间创建的每个记录。

与 CA Access Control 服务的连接

显示指明用户何时连接至 CA Access Control 服务（例如 CA Access Control 端点管理 或 selang）的记录。

注意：连接至 UNIX 工作站时，该筛选的名称将成为“登录记录”。该记录表示用户登录。

管理活动

显示更新 CA Access Control 或操作系统数据库的所有记录。数据库更新包括添加、删除和更改所有类型的记录。

创建用户定义的筛选

您可以根据需要创建任意数目的筛选器。 如果要仅查看一组特定的审核记录，请创建自定义筛选。

创建用户定义的筛选

1. 在 CA Access Control 端点管理 中，单击“审核事件”选项卡。

   审核记录查看器 -“筛选设置”区域显示已保存筛选的列表。

2. 在“已保存筛选”部分中，单击“创建筛选”。

   将显示“审核筛选向导”。

3. 完成向导中的各页面。

   选择名称和开关参数

   指定要在筛选中使用的开关参数 (p. 122)。

   编辑开关参数

   为选定的开关参数指定设置。实际上，它们是您可以为要筛选的审核事件而定义的掩码。

   编辑选项

   指定要为审核筛选设置的选项 (p. 123)。

   单击“完成”。

   将保存并加载您定义的新审核筛选。

审核日志备份

通过 CA Access Control，您可以自动备份审核日志文件以进行存档。

审核日志备份文件的名称在 CA Access Control 注册表项 logmgr\audit_back 中设置。

可以使用以下方法备份审核日志文件：

- 按大小触发的备份
- 按日期触发的备份

所选的备份审核日志文件的方法和设置取决于：

- 是否需要备份日志文件的副本
- 会在环境中生成多少审核数据
- 系统性能问题（如较大的审核日志文件会增加处理时间）

注意：默认情况下，如果您将设置配置为保留带有时间戳的备份，CA Access Control 将保护审核日志备份文件。 这种默认保护与按大小触发的审核备份文件所获得的保护相同。 要删除这些文件，需要在数据库中设置许可规则。

设置到达何种大小后自动备份审核日志

可以对审核日志文件的大小设置限制。 当文件到达定义的大小时，CA Access Control 将自动创建文件的备份副本并清除日志。 这就意味着将自动定期备份文件。

要设置在到达何种大小后自动备份审核日志，请在 CA Access Control 注册表项 logmgr\audit_size 中设置所需的最大大小 (KB)。

注意：可以通过设置 CA Access Control 注册表项 logmgr\audit_back 来定义备份文件的名称。

重要说明！ 如果 CA Access Control 注册表项 logmgr/BackUp_Date 设置为 yes（no 为默认值），则审核日志中的每个按大小触发的备份副本都会带有时间戳后缀。 在所有其他情况中，包括配置按日期触发的备份时，每个备份副本都会*覆盖*先前写入的备份副本。

示例：设置当文件到达 5 MB 时自动备份审核日志文件。

该示例说明如何设置当文件到达 5 MB (5120 KB) 时备份审核日志文件。 要执行该操作，请将 CA Access Control 注册表项 logmgr\audit_size 设置为 **5120**。

当审核日志文件到达 5 MB 时，CA Access Control 将创建文件备份副本（默认情况下名为 seos.audit.bak）并清除日志。

示例：设置当文件到达 1 MB 时使用自定义名称和时间戳自动备份审核日志文件

该示例说明如何设置当文件到达 1 MB (1024 KB) 时使用备份文件的自定义名称并向名称添加时间戳来备份审核日志文件

要执行该操作，请将以下 CA Access Control 注册表项设置为：

- logmgr\audit_size=1024
- logmgr\audit_back=log\ac_audit.old
- logmgr\BackUp_Date=yes

当审核日志文件到达 1 MB 时，CA Access Control 将创建文件备份副本并清除日志。 备份日志文件名称为 ac_audit.old.*timestamp*，其中 *timestamp* 是日期和时间，格式为 DD-Mon-YYYY.hhmmss。 例如：

```
ac_audit.old.06-Feb-2007.144330
```

设置自动备份审核日志的时间间隔

您可以定义时间间隔（每日、每周或每月），CA Access Control 将按照此间隔自动创建审核日志文件的备份副本并清除日志。

要设置自动备份审核日志的时间间隔，请在 CA Access Control 注册表项 logmgr\BackUp_Date 中设置时间间隔。 时间间隔可为以下选项之一：

每日

每天一次备份审核日志文件。

每周

每周一次备份审核日志文件。

每月

每月一次备份审核日志文件。

注意：可以通过设置 CA Access Control 注册表项 logmgr\audit_back 来定义备份文件的名称。

重要说明！ 如果到达备份时间间隔之前，审核日志到达了定义在 logmgr\audit_size CA Access Control 注册表键中的大小限制，那么 CA Access Control 会创建不带时间戳的文件备份副本。 每个此类备份可能会覆盖任何先前的副本。

示例：设置每日备份审核日志文件

该示例说明如何将审核日志文件设置为每日备份。 要执行该操作，请将 CA Access Control 注册表 logmgr\BackUp_Date 设置为 **daily**。

CA Access Control 将每天一次创建文件的备份副本并清除日志。 备份日志文件名称将带有 *.timestamp* 后缀，其中 *timestamp* 是日期和时间，格式为 DD-Mon-YYYY.hhmmss。 例如：

```
seos.audit.bak.06-Feb-2007.144330
```

第 9 章： 管理权限的范围

此部分包含以下主题：

全局权限属性

全局授权属性在用户记录中设置。 每个全局授权属性都允许用户执行某些类型的功能。 本节介绍每个全局授权属性的功能和限制。

ADMIN 属性

通过 ADMIN 属性，用户可以执行 CA Access Control 中几乎所有的命令。在数据库中定义的具有 ADMIN 属性的用户可以定义和更新数据库中的用户、组和资源。 这是 CA Access Control 中最强大的属性，但是它也有限制：

- 如果数据库中只有一个用户具有 ADMIN 属性，则无法删除该用户，而且也无法从记录中删除 ADMIN 属性。
- 具有 ADMIN 属性但没有 AUDITOR 属性的用户不能更改对用户、组或资源所做的审核的类型（审核模式）。 如果您有 ADMIN 属性且需要更改用户、组或资源的审核特性，请为自己分配 AUDITOR 属性。
- 具有 ADMIN 属性的用户无法删除超级用户（UNIX 上的 root 帐户或 Windows 上的 Administrator 帐户），但他们可以将 root 用户设置为非 ADMIN 用户。

AUDITOR 属性

具有 AUDITOR 属性的用户可以监视系统使用情况。 对于具有 AUDITOR 属性的用户，其显式权限包括以下内容：

- 用户可以显示数据库中的信息。

  审核员可以执行 selang 命令 *showusr*、*chgrp*、*chres* 和 *showfile*。

- 用户可以设置现有记录的审核模式。

  审核员可以执行 selang 命令 *chusr*、*chgrp*、*chres* 和 *chfile*。

OPERATOR 属性

具有 OPERATOR 属性的用户拥有对所有文件的 READ 访问权限。 使用该访问权限，他们可以列出数据库中的任何内容，且可以运行备份作业。 操作员可使用 showusr、showgrp、showres、showfile 和 find 命令列出数据库记录。 通过 OPERATOR 属性，用户还可以使用 secons 实用程序。

注意：有关 secons 实用程序的详细信息，请参阅 *《参考指南》*。

PWMANAGER 属性

PWMANAGER 属性为常规用户提供使用 chusr 或 sepass 命令更改其他用户的密码的权限。

注意：要通过 PWMANAGER 更改 ADMIN 用户的密码，请设置 setoptions 命令的 *cng_adminpwd* 选项。 有关详细信息，请参阅 *《参考指南》*。

PWMANAGER 属性不包括更改宽限登录次数、其他用户的密码间隔或常规密码规则的权限。

PWMANAGER 的权限还包括 showusr 和 find 命令的使用。

注意：如果用户将 nochngpass 属性设置为 yes，则 PWMANAGER 无法更改该用户的密码。

SERVER 属性

与许多其他安全模型一样，CA Access Control 不允许常规用户询问：“用户 A 是否可以访问资源 X？”，常规用户可以询问的唯一问题是：“我是否可以访问资源 X？”，不过，它应该允许向许多用户提供服务的进程（例如数据库服务器服务或内部应用程序）代表其他用户询问权限。。

SERVER 属性允许进程询问用户权限。 具有 SERVER 属性设置的用户可以发出 SEOSROUTE_VerifyCreate API。

注意：有关服务器属性和 CA Access Control API 的详细信息，请参阅*《SDK 指南》*。

IGN_HOL 属性

通过 IGN_HOL 属性，用户可以在假期记录中定义的任何时间段内登录。 HOLIDAY 类中的每个记录定义一个或多个时间段，在这些时间段内，用户需要额外权限才能登录。 通过 IGN_HOL 属性，用户可以随时登录，不受假期记录中定义的时间段的限制。

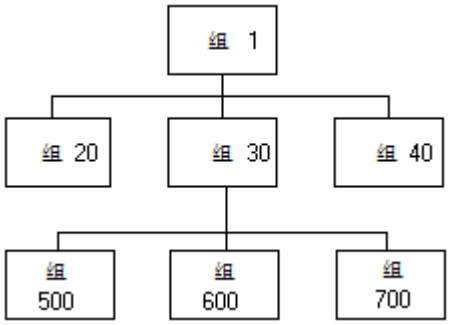
注意：有关 HOLIDAY 类的详细信息，请参阅*《参考指南》*。

组授权

在讨论组授权属性之前，有必要理解父子关系的概念。

父子关系

从属和超级组的概念（也称为父子关系），在讨论组管理权限时非常重要。 一个组可以是一个或多个组的父组（超级组）。 子组或从属组只能拥有一个父组。 为组分配一个父组是可选操作。 请考虑下图：



组 1 是 20、30 和 40 三个组的父组。 组 30 是 500、600 和 700 三个组的父组。 组 600 只有一个父组 30。 组 1 没有父组。

组授权限属性

包括像资源记录和访问者记录在内的所有记录都有所有者。 拥有一个记录意味着拥有查看、编辑和删除该记录的权限。

一个组可以拥有其自己的记录。 然而，在拥有记录的组中，只有某些特权用户才能管理记录。 这些特殊用户在自己的用户记录中设置了组授权属性。 组授权属性包括：

- GROUP-ADMIN
- GROUP-AUDITOR
- GROUP-OPERATOR
- GROUP-PWMANAGER

join 命令（只有正确授权的用户才可以发出）设置这些属性。 join 命令用于将用户置于组中，并指定用户的组授权属性（如果有）。

拥有权限的组成员可能或不可能被授权管理定义组成员的用户记录，这取决于谁拥有这些记录。

更多信息：

所有权 (p. 134)

GROUP-ADMIN 属性

拥有组管理授权属性的用户可以创建某组记录。 要创建记录，组管理员必须指定记录的所有者。

记录的所有者必须是该用户在其中拥有组授权属性的组。 如果该组是其他组的父组，则所有者还可以来自与其中的一个子组。 整组记录被称为组范围。 提供的授权示例说明了组范围的概念。

具有 GROUP-ADMIN 属性的用户对他们组范围中的记录拥有以下访问权限：

访问	说明	命令
读取	显示记录的属性。	showusr、showgrp、showres、showfile
创建	在数据库中创建新记录。 您必须指定所有者。	newusr、newgrp、newres、newfile
修改	更改记录的属性。	chusr、chgrp、 chres、chfile
删除	删除数据库中的记录。	rmusr、rmgrp、rmres、rmfile
连接	将用户加入组或者将用户与组分离。	join、join-

GROUP-ADMIN 属性也有限制：

- GROUP-ADMIN 用户无法阻止自己对资源的访问，因此：
  - GROUP-ADMIN 用户无法分配高于自己安全级别的安全级别。
  - GROUP-ADMIN 用户无法分配他们没有的安全类别或安全标签。
- GROUP-ADMIN 用户无法从数据库中删除超级用户（UNIX 上的 root 帐户或 Windows 上的 Administrator 帐户）。
- 几种限制与本章中的“全局权限属性”所述的全局权限属性有关：
  - GROUP-ADMIN 用户无法删除数据库中唯一的 ADMIN 用户记录。
  - GROUP-ADMIN 用户无法删除数据库最后一个 ADMIN 用户记录中的 ADMIN 属性。
  - 没有 AUDITOR 属性的 GROUP-ADMIN 用户无法更新审核模式。只有具有 AUDITOR 属性的 GROUP-ADMIN 用户可以更新审核模式。
  - GROUP-ADMIN 用户无法为任何用户设置全局授权属性 ADMIN、AUDITOR、OPERATOR、PWMANAGER 和 SERVER。

GROUP-AUDITOR 属性

具有 GROUP-AUDITOR 属性的用户可以列出组范围中任何记录的属性。组审核者还可以为组范围中的任何记录设置审核模式。

GROUP-OPERATOR 属性

具有 GROUP-OPERATOR 属性的用户可以列出组范围中任何记录的属性。

GROUP-PWMANAGER 属性

具有 GROUP-PWMANAGER 属性的用户可以更改其记录位于组范围中的任何用户的密码。

所有权

数据库中的每个记录（包括访问者记录和资源记录）都有一个所有者。当您向数据库中添加记录时，您可以使用 owner 参数来显式指定其所有者，也可以通过 CA Access Control 将定义记录的用户指定为记录的所有者。

如果以下内容的*任何*一条为真，访问者即拥有记录：

- 他们被定义为记录的所有者。
- 他们是定义为记录所有者的组的成员*并且*已加入具有 GROUP-ADMIN 属性的组。
- 他们是资源所属的资源组记录的所有者。

如果您删除拥有数据库中记录的用户或组，则这些记录不再具有所有者。

拥有记录的用户对他们所拥有的记录享有以下访问权限：

访问	说明	命令
读取	显示记录的属性。	showusr、showgrp、showres、showfile
修改	更改记录的属性。	chusr、chgrp、 chres、chfile
删除	删除数据库中的记录。	rmusr、rmgrp、rmres、rmfile

访问	说明	命令
连接	将用户加入组或者将用户与组分离。	join、join-

如果您不想让某用户或组对特定记录具有所有权权限，则为该记录以及该记录所属的任何资源组记录分配所有者 *nobody*。

所有权权限的限制如下：

- 数据库中最后一个 ADMIN 用户的所有者无法删除该用户的记录。
- 不具有 ADMIN 属性的所有者无法更新审核模式。 只有具有 AUDITOR 属性的用户才可以更新审核模式。
- 超级用户（UNIX 上的 root 帐户或 Windows 上的 Administrator 帐户）的所有者不能从数据库中删除 root。
- 所有者无法为他们所拥有的用户设置全局权限属性 ADMIN、AUDITOR、OPERATOR 和 PWMANAGER。
- 所有者无法阻止自己对资源的访问，因此：
  - 所有者无法分配高于自己安全级别的安全级别。
  - 所有者无法分配他们没有的安全类别或安全标签。

文件所有权

通过 CA Access Control，文件所有者可以通过在 FILE 类中定义记录来保护文件。 文件的所有者对该文件的记录具有完全权限，所以所有者可以将具有所有参数的 newfile、chfile、showfile 和 authorize 命令用于保护该文件的记录。

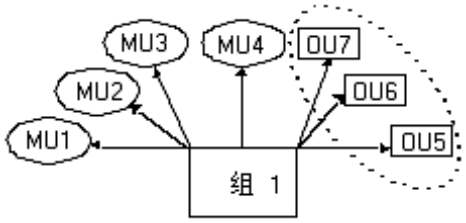
在 UNIX 上，当用户创建文件时，UNIX 会将该用户指定为该文件的所有者。 通过 CA Access Control，UNIX 文件所有者可以定义 FILE 记录，除非该功能被显式禁用。 如果您不需要文件所有者定义 FILE 记录，请确保将 seos.ini 文件 [seos] 部分中的 use_unix_file_owner 标记设置为 no。（这是默认设置）

授权示例

下图说明了组授权属性、父子关系、所有权、成员资格和组范围的概念。 这些图只包含用户和组，但所有权概念也适用于资源和文件记录。

单个组授权

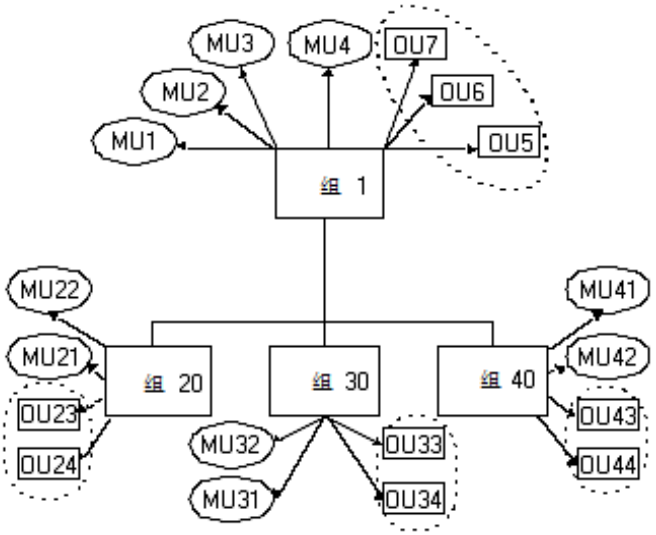
在下图中，四个用户 MU1、MU2、MU3 和 MU4 都是 Group 1 的成员。 Group 1 还拥有自己的三个用户 OU5、OU6 和 OU7。 成员 MU4 具有 GROUP-ADMIN 属性。÷



椭圆表明用户 MU4 执行的命令的组范围。 它包括组 1 拥有的所有用户，即 OU5、OU6 和 OU7。

父组和子组

在下图中，四个用户 MU1、MU2、MU3 和 MU4 都是 Group 1 的成员。 Group 1 还拥有自己的三个用户 OU5、OU6 和 OU7。 成员 MU4 在其记录中设置了 GROUP-ADMIN 属性。



组 1 是 20、30 和 40 三个组的父组。 其中的每个从属组都有两个属于该组成员的用户以及两个该组所拥有的用户。

四个椭圆表明用户 MU4 执行的命令的组范围。 它包括 Group 1 拥有的所有用户以及从属于 Group 1 的组所拥有的用户。 MU4 的组范围中的用户是 OU5、OU6、OU7、OU23、OU24、OU33、OU34、OU43 和 OU44。

如果有组从属于拥有用户、组或资源的组 20、30 或 40，则这些组所拥有的记录也属于用户 MU4 执行的命令的组范围。

子管理

安全管理员（具有 ADMIN 属性的用户）可以向常规用户授予特定管理权限。 这些常规用户则称为子管理员。 子管理员只有管理指定的 CA Access Control 类或对象的权限。 例如，可以授权子管理员只管理用户和组对象。 可以通过向子管理用户授予某类中特定对象的管理权限，来设置更高级别的子管理。

用户、组和资源的子管理员可以使用 selang 执行与这些资源相关的管理任务。

如何将特定管理权限授予常规用户

由于管理员（具有 ADMIN 属性的用户）几乎可以执行 CA Access Control 中的所有操作，因此您可能需要将特定管理任务指派给子管理员。 要执行此操作，您需要向这些用户授予对 CA Access Control 数据库中的类的权限，通过这些权限可以控制用户需要执行的特定管理任务，如下所示：

1. 识别控制您要指派的任务的一个或多个类。

   例如，CA Access Control 使用 USER 和 GROUP 类来创建访问者资源。如果您要指派访问者管理，那么您需要使用 ADMIN 类的 USER 记录和 GROUP 记录。

2. 将一个或多个子管理员授权到 ADMIN 类的适用资源。

   例如，要让一个子管理员查看和修改用户记录，请向该用户授予对 ADMIN 类的 USER 记录的*读取*和*修改*访问权限。

ADMIN 类

子管理员（类 ADMIN 中记录的 Access Control 列表 (ACL) 列出的用户）所拥有的权限与具有 ADMIN 属性的用户的权限相类似。 不过，ADMIN 类中记录的 ACL 中用户的权限仅限于记录代表的特定类。 例如，ADMIN 类中的 SURROGATE 记录确定哪些用户可以管理 SURROGATE 类的记录。

注意：有关 CA Access Control 类的详细信息，请参阅*《参考指南》*。

ADMIN 类中特定记录的 ACL 中的用户可以执行以下命令：

访问	说明	命令
读取	显示类中记录的属性。	showusr、showgrp、showres、showfile、find
创建	在类中创建新数据库记录。	newusr、newgrp、newres、newfile
修改	更改类中的属性。	chusr、chgrp、chres、chfile
删除	删除数据库中现有的类记录。	rmusr、rmgrp、rmres、rmfile
连接	向组中添加用户和删除组中的用户。该访问权限仅在 GROUP 记录的 ACL 中有效。	join、join-

访问	说明	命令
密码	控制数据库中所有用户的密码及其密码属性。 该访问权限授予的权限与具有 PWMANAGER 属性的用户的访问权限相同。 该访问权限仅在 USER 记录的 ACL 中有效。	chusr

拥有 ADMIN 类权限的用户有以下限制：

- 在 ADMIN 类的 USER 记录的 ACL 中定义的用户无法删除数据库中的最后一个 ADMIN 用户。
- ADMIN 类用户无法为他们所拥有的用户设置全局权限属性 ADMIN、AUDITOR、OPERATOR 和 PWMANAGER。
- ADMIN 类用户无法强制更新审核模式。 只有具有 AUDITOR 属性的 ADMIN 类才可以更新审核模式。
- ADMIN 类用户无法删除超级用户（UNIX 上的 root 帐户或 Windows 上的 Administrator 帐户），但他们可以将 root 设置为 NOADMIN。
- ADMIN 类用户无法阻止自己对资源的访问，因此：
  - ADMIN 类用户无法为资源分配高于自己安全级别的安全级别。
  - ADMIN 类用户无法分配他们没有的安全类别或安全标签。

这些限制是 B1 安全级别认证的一部分。

环境注意事项

您在环境中所占据的位置是决定您是否可以更新数据库中的信息的因素之一。

远程管理限制

您可以通过网络访问远程工作站，以及更新远程工作站上的数据库。要更新远程工作站上的数据库，您和您的终端都需要有权限。

- 您必须被显式定义为远程工作站数据库中的用户。 无论要执行何种命令，都必须在远程工作站数据库中您的用户记录中设置适当的属性。
- 您必须在规则中显式说明本地终端的需求，对本地终端授予访问远程工作站的写入访问权限；否则，您无法在远程工作站执行 CA Access Control 管理。

  借助默认访问字段 (_default) 或 UACC 类的写入访问权限，您可以在远程工作站输入 selang 命令 shell。 不过，您**无法执行任何 selang 命令或者以其他方式访问远程数据库。 使用读取访问权限，您可以登录远程工作站，但无法在该工作站执行 CA Access Control 管理。**

  以下示例说明了 WRITE 和 READ 权限之间的区别：

  1. 要指定默认访问权限为 READ 的新终端，即管理员可以从该终端登录，但却无法处理其中的数据库，请发出以下命令：

     ```
     newres TERMINAL tty13 defacc(read)
     ```

  2. 要授予用户 ADMIN1 权限，以便从新终端处理数据库（即授予 WRITE 权限和 READ 权限），请发出以下命令：

     ```
     authorize TERMINAL tty13 uid(ADMIN1) access(r,w)
     ```

UNIX 环境

对于管理 UNIX 中的用户和组，CA Access Control 中拥有全局或组授权属性的用户对 UNIX 的权限和限制与他们对 CA Access Control 的权限和限制相同。

如果您在*没*运行 seosd 后台进程时使用 selang（例如在安装时），则必须遵守以下规则：

- 必须在 selang 命令中包括 -l 选项。
- selang 的用户必须是 root 用户。 （这种独占的 root 权限符合常规的 UNIX 限制。）

Windows 环境

在本地 Windows 环境中有效

当 CA Access Control 正在运行时，如果您使用 selang 更改本机 Windows 环境中的资源，CA Access Control 代理则会在相应的 Windows 库中更改资源。 您不需要任何其他的 Windows 权限来更改资源。 这意味着，当 CA Access Control 中具有全局或组授权属性的用户在本机 Windows 环境中执行 selang 命令时，与其在 CA Access Control 中执行该操作具有相同的 Windows 权限和限制。

当 CA Access Control 未在运行时，如果您使用 selang 更改本机 Windows 环境中的资源，您必须遵守如下规则：

- 必须在 selang 命令中包括 -l 选项
- 必须具有 ADMIN 属性或子管理权限
- 必须具有足够的 Windows 权限来更改资源

  出现该限制的原因是由于 selang 进程（而不是 CA Access Control 代理）在 Windows 库中更改资源。

例如，用户 Emma 想使用 chfile selang 命令在本机 Windows 环境中更改文件 C:\tmp.txt 的所有者。 如果 CA Access Control 正在运行，Emma 则需要足够的 CA Access Control 权限来更改文件拥有者，但是不需要额外的 Windows 权限。 如果 CA Access Control 未在运行，Emma 则同时需要 CA Access Control 和 Windows 权限来更改文件拥有者。

访问数据库的默认权限

当 CA Access Control 正在运行时，它使用内部文件规则保护内部数据库、seosdb。 内部文件规则在 selang 中不可见且无法删除。 您可以编写 FILE 规则来覆盖内部文件规则。 如果删除这些 FILE 规则，CA Access Control 会恢复到内部文件规则。

当 CA Access Control 正在运行时，下列内部文件规则会保护数据库：

- CA Access Control 内部进程对数据库有完全访问权限
- NT AUTHORITY\System 用户对数据库有读取权限
- 所有其他访问者对数据库都没有访问权限

注意：所有其他访问者的默认访问权限在 r12.5 SP3 中更改。 在先前的版本中，默认情况下所有其他访问者对数据库文件都有读取访问权限。

默认情况下，在您安装 CA Access Control 或重新启动端点之后会自动运行 CA Access Control 服务。 因此，可以即时访问数据库的唯一用户是 NT AUTHORITY\System。 您在安装期间定义的 CA Access Control 管理员也能使用诸如 selang 等实用工具来更新数据库。

访问数据库的本机权限

当 CA Access Control 停止时，对数据库文件的访问权限由本机 Windows 权限决定。 权限从安装 CA Access Control 的父级目录继承。 由于此继承关系，当 CA Access Control 停止时，对数据库文件的默认访问权限是读取。

要在 CA Access Control 停止时对其进行保护，您可以改变对数据库文件的 Windows 权限以适合您的企业要求。 更改权限之前，请考虑以下事项：

- NT AUTHORITY\System 用户*必须*具有对数据库文件读写的 Windows 权限。

  CA Access Control 授权引擎从 NT AUTHORITY\System 用户那里继承权限。 如果该用户无法访问数据库，引擎就没有足够的本机权限更新数据库。

- 当 CA Access Control 停止时需要对其进行读写的用户必须具有对数据库文件读写的 Windows 权限。

  需要读写访问权限的用户包括备份、还原或升级 CA Access Control 的用户。

- 当 CA Access Control 停止时可以使用 selang（selang -l 选项）的用户必须具有以下权限：
  - ADMIN 属性或子管理权限
  - 对数据库文件读写的 Windows 权限
  - 更改本地存储库的 Windows 权限（如果需要）

    例如，要在 CA Access Control 停止时使用配置环境来更改 CA Access Control 注册表项，您必须有足够 Windows 权限更改注册表。

  当 CA Access Control 停止时，仅有 CA Access Control 管理员（具有 ADMIN 属性或子管理权限的用户）才可以使用 selang 来维护数据库。 当 CA Access Control 停止时，如果 CA Access Control 管理员无法访问数据库，那么没有用户可以执行脱机数据库维护，可能会发生死锁。

第 10 章： 管理策略模型

此部分包含以下主题：

策略模型数据库

单独管理数十或数百个数据库并不现实。 CA Access Control 提供策略模型服务，通过该服务组件，您可以通过一个中央数据库管理多个数据库。 对策略模型 (PMD) 服务的使用是可选的，但是它将大大简化大型站点的管理。

注意：在 Windows 任务管理器中，策略模型服务显示为 sepmdd.exe。

策略模型服务使用策略模型数据库 (PMDB)。 与其他 CA Access Control 数据库一样，PMDB 包含用户、组、受保护的资源和管理资源访问的规则。 此外，PMDB 还包含一个 *订户*数据库列表。 每个订户都是位于单独计算机上的 CA Access Control 数据库，或是位于同一或不同计算机上的另一 PMDB。 更新订户的 PMDB 是订户的*父*项。

在管理具有类似权限限制和访问规则的多个数据库方面，PMDB 是一个非常有用的工具。

策略模型名称在 Windows 上是区分大小写的，以便与 UNIX 兼容。 在命令中指定 PMDB 名称时，请确保您使用了正确的大小写形式。

注意：在 PMDB 和主机名中不能使用非英文字符。

虽然 PMDB 名称区分大小写，但是在同一计算机上不能具有两个仅字母大小写形式不同的 PMDB。 这是因为 CA Access Control 将 PMDB 名称用作文件路径的一部分，而 Windows 却又不区分大小写，因此不允许上述情况发生。 例如，myPMDB 和 MYpmdb 是两个不同的策略模型数据库，但是不能存在于同一系统中。

注意：有关管理 PMDB（sepmd 实用程序）的信息，请参阅*《参考指南》*。有关使用 selang 远程管理 PMDB 的信息，请参阅 *《selang 参考指南》*。

磁盘上的 PMDB 位置

计算机中的所有 PMDB 都驻留在一个公用目录中。 在下列 Windows 注册表子键中的 _pmd_directory_ 值指定该目录的名称：

```
HKEY_LOCAL_MACHINE\Software\ComputerAssociates\AccessControl\Pmd
```

_pmd_directory_ 在 NTFS 根目录中的默认值为：*ACInstallDir*\data，其中，*ACInstallDir* 是您安装 CA Access Control 的目录（默认为 C:\Program Files\CA\AccessControl\）。

每个 PMDB 都占用公共目录中的一个子目录。 子目录中的文件包含定义策略模型必需的所有数据。 策略模型配置设置存储在 CA Access Control 注册表设置的 Pmd 子键中。 该子键的名称即为策略模型的名称。

管理本地 PMDB

CA Access Control 提供用于管理 PMDB 的实用程序：

sepmd

您可以使用 PMDB 管理实用程序来执行以下任务：

- 管理订户
- 截短更新文件
- 管理双重控制
- 管理策略模型日志文件
- 执行其他管理任务

注意：有关 sepmd 的深入讨论，请参阅*《参考指南》*。

管理远程 PMDB

CA Access Control 还提供一系列可在 pmd 环境中使用的 selang 命令。 通过这些命令，您可以远程管理 PMDB：

backuppmd

备份 PMDB。

createpmd

创建 PMDB。

deletepmd

删除 PMDB。

findpmd

显示计算机上所有 PMDB 的名称。

listpmd

列出以下关于 PMDB 的信息：

- 订户及其状态
- PMDB 描述及其状态
- 更新文件中的命令及其偏移量
- 错误日志的内容

pmd

您可以使用 PMDB 管理命令来执行以下任务：

- 从不可用订户列表中删除订户
- 清除策略模型错误日志
- 启动和停止策略模型服务
- 锁定和解除锁定策略模型
- 截短更新文件

restorepmd

从其备份文件还原 PMDB。

subs

您可以使用 PMDB 订阅命令执行以下任务：

- 将现有的订户添加到父 PMDB 中
- 将新订户添加到父 PMDB 中
- 为数据库（CA Access Control 或另一 PMDB）指定父 PMDB

subspmd

为本地数据库指定父 PMDB。

unsubs

从 PMDB 中删除订户。

注意：有关可在 pmd 环境中使用的 selang 命令的深入讨论，请参阅 *《selang 参考指南》*。

体系结构相关性

部署 CA Access Control 时，您应考虑环境的层级结构。 在许多站点上，网络具有各种体系结构。 某些策略规则（例如受托程序列表）与体系结构相关。 另一方面，大多数规则都与系统体系结构无关。

您可以使用层级结构来包括这两种规则。 您可以为与体系结构无关的规则定义一个全局数据库，向它提供定义与体系结构相关的规则的订户 PMDB。

注意：根 PMBD 及其所有订户可以位于同一计算机或不同计算机上，这取决于您环境的实际需要。

示例：一个两层的部署层级结构

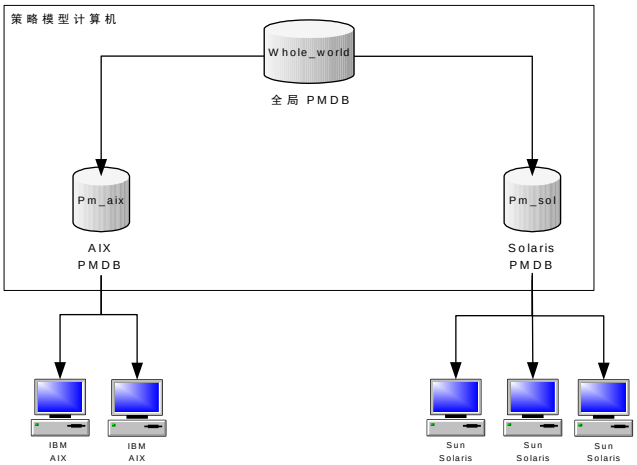
下面的 UNIX 示例也适用于 Windows 体系结构，不过需要做一些小修改。

在该示例中，站点包括 IBM AIX 和 Sun Solaris 系统。 由于 IBM AIX 上的受托程序列表与 Sun Solaris 上的受托程序列表不同，因此 PMDB 需要考虑体系结构的依赖性。

要设置多体系结构 PMDB，请按以下步骤设置 PMDB：

1. 定义名为 whole_world 的 PMDB，以包含用户、组和所有其他与体系结构无关的策略。
2. 定义名为 pm_aix 的 PMDB，以包含所有特定于 IBM AIX 的规则。
3. 定义名为 pm_sol 的 PMDB，以包含所有特定于 Sun Solaris 的规则。

   名为 pm_aix 和 pm_solaris 的 PMDB 是名为 whole_world 的 PMDB 的订户。 站点上的所有 IBM AIX 计算机都是 pm_aix 的订户。 站点上的所有 Sun Solaris 计算机都是 pm_sol 的订户。 下面以图表说明该概念。



4. 当您在 whole_world 中输入与平台无关的命令，例如添加用户或设置 SURROGATE 规则，则自动更新站点上的所有数据库。
5. 当您向 pm_aix 中添加受托程序时，只更新 IBM AIX 计算机，而不会影响 Sun Solaris 系统。

集中管理策略的方法

您可以使用 CA Access Control 采用以下方式从一台计算机管理多个数据库：

- **基于规则的自动策略更新** - 您在中央数据库 (PMDB) 中定义的常规规则会自动传播给已配置的层级结构中的数据库。

  注意：仅此方法提供双重控制，并且仅适用于 UNIX。 有关基于规则的自动策略更新的双重控制信息，可以在*《端点管理指南：用于 UNIX》*上找到。 有关基于规则的自动策略更新的信息，可以在*《端点管理指南：用于 Windows》*上找到。

- **高级策略管理** - 根据主机或主机组分配将您部署的策略（规则组）传播到所有数据库。 您还可以取消部署（删除）策略以及查看部署状态和部署偏差。 要使用此功能，您需要安装并配置额外的组件。

  注意：有关高级策略管理的信息，可以在*《企业管理指南》*中找到。

基于规则的自动策略更新

您在中央数据库中进行的单一规则策略更新（常规 selang 规则）将自动传播给订户数据库。 通过将若干计算机订阅到同一数据库，以及将一个数据库订阅到另一数据库，您可以创建层级结构。 安装后，您可以为基于规则的自动策略更新配置您的环境。

注意：这种管理策略的方法限于使您在整个层级结构中进行单一规则策略更新。 其他功能只能通过实施高级策略管理和报告来使用。

基于规则的自动策略更新原理

为基于规则的自动策略更新配置环境时，您在中心数据库中定义的每条规则自动通过以下方式传播给它的所有订户：

1. 必须为至少有一个订户的任何 PMDB 定义一条规则。
2. PMDB 向所有订户数据库发送命令。

3. 订户数据库应用传播的命令。
   a. 如果订户数据库没有响应，则 PMDB 按规定时间间隔（默认情况下为每隔 30 分钟）发送命令，直到更新了订户数据库为止。
   b. 如果订户数据库正在响应，但拒绝应用命令，则 PMDB 会将命令放在策略模型错误日志 (p. 156)中。
4. 如果订户数据库是其他订户的父项，则将命令发送给它的订户。

示例：从层级结构中的所有计算机上删除用户

如果使用 rmusr 命令从 PMDB 删除用户，则相同的 rmusr 命令将发送至所有订户数据库。 这样，一个 rmusr 命令即可从各台计算机上的多个数据库中删除用户。

您使用 PMDB 来传播配置设置的方式

当您编辑策略模型的配置时，新的配置值会被传播给策略模型的订户。

以下过程说明了配置更新被传播给策略模型的订户的方式：

1. 编辑一个或多个策略模型的配置值。
2. 策略模型将新的配置值写入虚拟配置文件。

   注意：虚拟配置文件不包含 audit.cfg 文件的值。 策略模型不会将您对该文件所做的任何更改写入到虚拟配置文件。
3. 策略模型将新的配置值发送给订户。
4. selang 命令使用新的配置值更新每位订户。

虚拟配置文件

每个策略模型都有虚拟配置文件，其中包含其订户的配置值。 虚拟配置文件位于 PMD 目录，并命名为 cfg_*configname*，其中 *configname* 是策略模型配置的名称。

虚拟配置文件不包含 audit.cfg 文件中所具有的配置值。

新订户的配置方式

策略模型使用现有的配置值配置每位新订户。 现有的配置值存储在虚拟配置文件中。

注意：虚拟配置文件不存储 audit.cfg 文件中的配置值。 您在创建新订户前对 audit.cfg 文件所做的任何更改都不会传播给新订户。

以下过程说明策略模型配置新订户的方式：

1. 创建一个新订户到策略模型。
2. 策略模型读取其虚拟配置文件中的值。
3. 策略模型将配置值从其虚拟配置文件添加到 updates.dat 文件。 updates.dat 文件还包含策略的访问规则。
4. 策略模型将 updates.dat 文件发送到新订户。
5. selang 命令使用 updates.dat 文件中的值配置新订户。

如何能够设置层级结构

CA Access Control 使用策略模型服务在已配置的层级结构中传播基于规则的策略更新。 通过为同一 PMDB 订阅多台 CA Access Control 计算机，以及通过为一个 PMDB 订阅另一 PMDB，您可以创建层级结构。

设置 PMDB 层级结构的最简单方式是在安装 CA Access Control 时进行设置，因此，在开始安装之前，应考虑要如何构建层级结构。 由于父 PMDB 及其订户必须能够相互通讯，因此必须确保 PMDB 层级结构中的所有主机都属于同一个网络。 也就是说，父 PMDB 必须能够按名称连接它的每个订户，而每个订户必须能够按名称连接到父 PMDB。

注意：有关安装 CA Access Control 的详细信息，请参阅 *《实施指南》*。

如果您想要更改在安装期间创建的配置，或者如果您在安装期间没有创建 PMDB 结构，您可以随时更改或创建 PMDB 配置。 您可以使用下列方式之一来执行该操作：

- 使用 CA Access Control 端点管理
- 利用 sepmd 实用程序

要创建 PMDB 层级结构并在安装后启用基于规则的自动策略更新，请执行以下操作：

1. 创建和配置主 PMDB。
2. （可选）创建和配置订户 PMDB。
3. 为订阅计算机（称为*端点*）定义父 PMDB。

更新订户

更新订户时，策略模型执行以下操作：

1. 当向策略模型中添加订户名称，或者从中删除订户名称时，策略模型将试图对其进行完全限定。
2. PMDB 服务即 sepmdd 尝试更新订户数据库。
3. 如果超出最长等待时间后，该服务仍无法成功更新某个订户，则它会跳过该订户，并尝试更新列表中的其余订户。
4. 完成订户列表的首次扫描后，sepmdd 会接着执行第二次扫描，在这次扫描期间，它会尝试更新在第一次扫描期间未成功更新的订户。

注意：PMDB 将更新传播到订户时，无论何时出现错误，sepmdd 服务都会在策略模型错误日志文件 (p. 156)中创建一个条目。 该日志文件 ERROR_LOG 位于 PMDB 目录 (p. 146)中。

更新策略模型数据库

在 PMDB 所在计算机上的操作不会自动更新 PMDB 本身。 要更新 PMDB，需将其指定为目标数据库。

您可以使用 selang 或 CA Access Control 端点管理 来指定 PMDB。 要使用 selang 来指定目标数据库，请使用 selang 命令 shell 中的 hosts 命令：

```
hosts pmd_name@pmd_host
```

所有 selang 命令立即更新指定的策略模型数据库。 这些命令然后自动传播到此计算机和所有订户计算机上的活动数据库中。

示例：指定目标 PMDB

要将目标数据库设置为 myPMD_host 上的 policy1，请使用以下命令：

```
hosts policy1@myPMD_host
```

如果您现在输入 newusr 命令，则新用户将被添加到 policy1 数据库以及此计算机和所有订户计算机上的活动数据库中。

清理更新文件

sepmd 实用程序将自动写入它在 updates.dat 文件中接收到的每项更新。 为防止该文件变得过大，建议您定期删除文件中已处理的更新。

要清理更新文件，请使用以下命令：

```
sepmd -t pmdbName auto
```

sepmd 计算尚未传播的第一个更新条目的偏移量，并删除在它之前的所有更新条目。

注意：有关 sepmd 实用程序的详细信息，请参阅*《参考指南》*。

传播并同步密码

设置 PMDB 层级结构后，当使用 Windows 用户管理器或 CA Access Control 以外的软件更改用户密码时，您可以使用 PMDB 层级结构使用户密码在系统中保持同步。

注意：CA Access Control 也支持大型机密码同步。

传播并同步密码

1. 创建 PMDB 层级结构。
2. 在用户或管理员可以更改密码的每个工作站上，输入适当的父 PMDB 的名称作为注册表中的 passwd_pmd 项值。

   ```
   HKEY_LOCAL_MACHINE\Software\ComputerAssociates\AccessControl\AccessControl\passwd_pmd
   ```

   PMDB 然后将密码更改传播到它的所有订户。 如果 passwd_pmd 值为空，则 CA Access Control 将检查 secondary_pmd 值，并将新的和更新的密码发送到该值中列出的 PMDB（除非该值也为空）。

注意：如果 PMDB 将用户密码发送到没有定义用户的订阅者，则不会更改设置，并且仍未定义订阅者的用户。

删除订户

如果您不再希望将更新传播给某个订户，则应当将其删除。

删除订户

1. 将计算机从订阅列表中删除：

   ```
   sepmd -u PMDB_name computer_name
   ```

   从策略模型订阅列表中删除计算机。

2. 关闭在您从订阅列表中删除的计算机上的 seosd：

   ```
   secons -s
   ```

   seosd 服务被关闭。

3. 在您从订阅列表中删除的计算机上删除以下注册键中的 parent_pmd 注册表值：

   ```
   HKEY_LOCAL_MACHINE\Software\ComputerAssociates\AccessControl\AccessControl
   ```

   计算机将停止接受来自 PMDB 的更新。

4. 重新启动 seosd。

   在您从订阅列表中删除的计算机上的活动数据库不再是指定 PMDB 的订户。

注意：从 PMDB 取消订阅数据库后，PMDB 不会再发送命令。

筛选更新

如果希望 PMDB 更新不同订户数据库上的不同数据子集，您需要定义向订户数据库发送哪些记录。

筛选更新

1. 配置 PMDB 以便用作订户子集的父集。
2. 修改父 PMDB 的注册表键中的 *Filter* 注册表项，以指明要在同一计算机上设置的筛选文件。

   然后将对订户数据库的更新限于通过该筛选器的记录。

策略模型筛选器文件

筛选器文件由每行具有六个字段的行组成。 字段包含如下信息：

- 允许或禁用的形式。

  例如，EDIT 或 MODIFY

- 受影响的环境：

  例如，AC 或本地环境

- 记录的类。

  例如，USER 或 TERMINAL

- 规则涵盖的类中的对象。

  例如：User1、AuditGroup 或 COM2

- 记录授予或取消的属性。

  例如，筛选行中的 OWNER 和 FULL_NAME 意味着具有这些属性的任何命令都会被筛选。 必须按照*《参考指南》*所述准确输入每个属性。

- 这类记录是否应该转发到订户数据库：

  PASS 或 NOPASS

以下规则适用于筛选器文件中的每一行：

- 可以使用星号 (*) 表示任意字段中的所有可能值。
- 如果有多行含有相同的记录，则使用适用的*第一行*。
- 用空格分隔字段。
- 在具有多个值的字段中，使用分号分隔值。
- 以 # 开头的行被视为注释行。
- 不允许有空行。

示例：筛选文件

以下示例介绍筛选器文件中的行：

CREATE	AC	USER	*	FULL_NAME;OBJ_TYPE	NOPASS
访问形式	环境	类	记录名（* =全部）	属性	处理

在此示例中，如果我们将具有该行的文件命名为 Printer1_Filter.flt，并编辑 PMDB PM-1 的注册表，以使筛选为 C:\Program Files\CA\AccessControl\\Printer1_Filter.flt，则 PMDB PM-1 不会向其订户传播使用 FULL_NAME 和 OBJ_TYPE 属性创建新用户的任何记录。

策略模型错误日志文件

按时间先后顺序组织的策略模型错误日志看上去与以下内容类似：

错误文本	错误类别
20 Nov 03 11:56:07 (pmdb1): fargo nu u5 0 Retry ERROR: 登录过程失败 (10068) ERROR: 无法接受来自非父项 PMDB 的更新 (pmdb1@name.company.com) (10104)	配置错误
20 Nov 03 19:53:17 (pmdb1): fargo nu u5 0 Retry ERROR: 连接失败 (10071) 主机不可访问 (12296)	连接错误
20 Nov 03 11:57:06 (pmdb1): fargo nu u5 560 Cont ERROR: 创建 USER u5 失败 (10028) 已经存在 (-9) 20 Nov 03 11:57:06 (pmdb1): fargo nu u5 1120 Cont ERROR: 创建 USER u5 失败 (10028) 已经存在 (-9)	数据库更新错误

策略模型错误日志采用二进制格式，您只有通过输入以下命令才能查看它：

ACInstallDir/bin sepmd -e *pmdname*

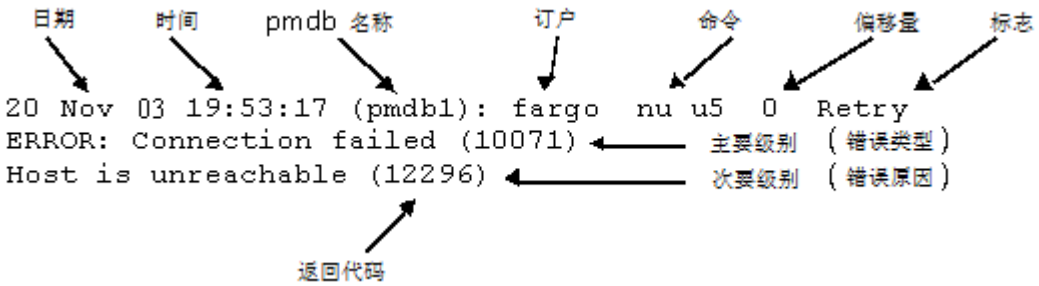
注意：不要手动删除错误日志（例如，使用 UNIX rm 命令）。只能使用以下命令删除日志：

ACInstallDir/bin sepmd -c *pmdname*

重要说明！ 在 CA Access Control r5.1 及更高版本中，错误日志的格式与早期版本的格式不兼容。sepmd 无法处理早期版本的错误日志。当您升级到具有该格式的版本时，旧的错误日志将复制到 ERROR_LOG.bak 中；在您启动 sepmdd 时将创建新的日志文件。

示例：PMDB 更新错误消息

以下示例显示典型的错误消息：



- 第一行总是由日期、时间和订户组成。 接下来显示产生错误的命令，然后是偏移量（十进制格式），指示更新文件中失败更新的位置。 最后，标志指示 PMDB 是自动重试更新，还是忽略该更新而继续。
- 第二行显示主要级别消息（发生的错误类型）的示例及消息的返回代码。
- 第三行显示次要级别消息（发生错误的原因）示例及消息的返回代码。

示例：错误消息

一个命令可能会产生并显示多个错误。 而且，一个错误可能包括主要级别消息、次要级别消息或同时包括这两种消息。

下列错误只有一个消息级别：

```
Fri Dec 29 10:30:43 2003 CIMV_PROD: 发布失败。 返回代码 = 9241
```

sepmd pull 尝试释放可用的订户时出现该消息。

本地策略模型存储库

您可以在 PMDB 中存储所有的本地环境用户和组对象类型。 通过在 PMDB 中存储该信息，您可以使用 show 命令（例如 show user 或 show group）接收有关对象的信息。 返回的对象是在 Windows 或 UNIX 订户中定义的实际对象的映像。

在连接到策略模型之后，用户可以选择下列环境：

- AC
- 本地
- NT
- UNIX
- 配置

注意：当您在 Windows 操作系统上工作时，本地环境完全与 Windows 一样运行，当您在 UNIX 操作系统上工作时，则本地环境完全与 UNIX 一样运行。

要使用本地环境存储库，请使用下列命令：

- 在 selang 提示符后输入下列命令：

  ```
  env NT; find
  ```

  您的结果将列出所有的本地环境对象类型。

  注意：有关这些对象类型的说明，请参阅*《参考指南》*中的 Windows 环境中的类和属性。

- 输入下列命令以接收 NT 和 Active Directory USER 属性列表：

  ```
  env NT; ruler user
  ```

- 输入下列命令以接收 NT 和 Active Directory GROUP 属性列表：

  ```
  env NT; ruler group
  ```

如果某个策略模型是另一个（父）策略模型的订户，它则通过传播接收父策略模型的数据，并在数据库中保存所有的用户和组属性，因此，您可以查看并更改这些属性。

注意：有关详细信息，请参阅*《参考指南》*中的 sepmd 实用程序。

策略模型备份

当备份 PMDB 时，您将策略模型数据库的数据复制到其他目录。 其中包括：

- 策略信息
- 策略模型的订户列表
- 配置设置
- 注册表项
- updates.dat 文件

您不能从使用其他平台、操作系统或 CA Access Control 版本的备份文件还原 PMDB。 确保将策略模型备份到运行相同的平台、操作系统和 CA Access Control 版本的主机上。

使用 sepmd 备份 PMDB

备份 PMDB 时，数据将从策略模型数据库复制到指定的目录。 您应将备份的 PMDB 文件存储至一个安全的位置，最好是受 CA Access Control 访问规则保护的位置。

您可以使用 sepmd 实用程序在本地主机上备份 PMDB。 您还可以使用 selang 命令在远程主机上备份 PMDB。

注意：您可以采用递归方式备份 PMDB。 递归备份可将一个层级结构中的所有 PMDB 备份到您指定的主机并修改 PMDB 订户，从而当备份移到该主机时订阅仍可进行。 当主 PMDB 和子 PMDB 部署在同一主机上时，您仅能使用递归备份。

使用 sepmd 备份 PMDB

1. 使用以下命令锁定 PMDB：

   `sepmd -bl pmdb_name`

   PMDB 将锁定，并且无法向其订户发送任何命令。

2. 请执行下列操作之一：

   - 使用以下命令备份 PMDB：

     `sepmd -bh pmdb_name [destination_directory]`

   - 使用以下命令采用递归方式备份 PMDB：

     `sepmd -bh pmdb_name [destination_directory] [backup_host_name]`

   注意：如果您不指定目标目录，备份将被存入以下目录：

   ACInstallDir\data\policies_backup\pmdb_name

3. 使用以下命令解锁 PMDB：

   `sepmd -ul pmdb_name`

   PMDB 将解锁，并且可向其订户发送命令。

使用 selang 备份 PMDB

备份 PMDB 时，数据将从策略模型数据库复制到指定的目录。 您应将备份的 PMDB 文件存储至一个安全的位置，最好是受 CA Access Control 访问规则保护的位置。

您可以使用 selang 命令在本地或远程主机上备份 PMDB。 您还可以使用 sepmd 实用程序在本地主机上备份 PMDB。

注意：您可以采用递归方式备份 PMDB。 递归备份可将一个层级结构中的所有 PMDB 备份到您指定的主机并修改 PMDB 订户，从而当备份移到该主机时订阅仍可进行。 当主 PMDB 和子 PMDB 部署在同一主机上时，您仅能使用递归备份。

使用 selang 备份 PMDB

1. （可选）如果要使用 selang 从远程主机连接 PMDB，请使用以下命令连接 PMDB 主机：

   ```
   host pmdb_host_name
   ```

2. 使用以下命令移至 PMD 环境：

   ```
   env pmd
   ```

3. 使用以下命令锁定 DMS：

   ```
   pmd pmdb_name lock
   ```

   PMDB 将锁定，并且无法向其订户发送任何命令。

4. 使用以下命令备份 DMS 数据库：

   ```
   backuppmd pmdb_name [destination(destination_directory)]
   [hir_host(host_name)]
   ```

   注意：如果您不指定目标目录，备份将被存入以下目录：

   ACInstallDir\data\policies_backup\pmdbName

5. 使用以下命令解锁 PMDB：

   ```
   pmd pmdb_name unlock
   ```

   PMDB 将解锁，并且可向其订户发送命令。

策略模型还原

当还原策略模型时，CA Access Control 会将 PMDB 备份文件复制到指定的目录。 原始 PMDB 文件中的所有内容都被复制到新的 PMDB 目录中，包括：

- 策略信息
- 策略模型的订户列表
- 配置设置
- 注册表项
- updates.dat 文件

如果目标目录中存在现有的 PMBD，CA Access Control 会在将还原文件复制到该目录之前删除现有文件。

您不能从使用其他平台、操作系统或 CA Access Control 版本的备份文件还原 PMDB。 确保将策略模型备份到运行相同的平台、操作系统和 CA Access Control 版本的主机上。

还原 PMDB

当您还原 PMDB 时，CA Access Control 将 PMDB 备份文件中的数据复制到您指定的目录里。 CA Access Control 必须在您执行还原的终端上运行。

注意：如果您在不同的终端上备份和还原 PMDB, PMDB 将不会在还原的 PMDB 数据库中自动更新终端资源。 您必须将新的终端资源添加到还原的 PMDB 中。 要添加新的终端资源，请停止还原的 PMDB，运行 *selang -p pmdb* 命令，然后再启动还原的 PMDB。

要还原 PMDB，请在想要还原 PMDB 的终端上运行以下内容之一：

- sepmd -restore 实用工具
- selang restore pmd 命令

注意：有关 sepmd 实用程序的详细信息，请参阅 *《参考指南》*。 有关 selang 命令的详细信息，请参阅 *《selang 参考指南》*。

将 PMDB 与 Unicenter 集成

将 PMDB 与 Unicenter TNG 集成使您可以使用 PMDB 来创建规则，以防止 Unicenter TNG 对象被各种 Unicenter TNG 组件操纵（例如，命令处理器、事件管理和工作量管理）。

您必须手动执行集成。

将 PMDB 与 Unicenter TNG 集成

1. 创建 PMDB。
2. 使用以下命令，将 Unicenter Security 选项迁移到 PMDB 中：

   `MigOpts` *`pmdb-name`*

   其中，*pmdb-name* 是 PMDB 的名称。

   注意：只有在您使用了 Unicenter 安全并在安装 CA Access Control 期间选择了“在 Unicenter 集成下的安全数据迁移”时，才需要执行该步骤。 如果您没有使用 Unicenter Security，而且从未建立任何安全选项，则没有内容要迁移到您的 PMDB 中。
3. 使用以下命令，为用户定义的任何 Unicenter TNG 资产类型创建类：

   `defclass.bat.` *`pmdb-name`*

   其中，*pmdb-name* 是 PMDB 的名称

   注意：仅当您使用了 Unicenter Security 并创建了用户定义的资产类型时才需要该步骤。 如果您在安装 CA Access Control 过程中选择了 Unicenter 集成，则在每个新的 PMDB 中将自动定义 Unicenter TNG 资产类型。

大型机密码同步

CA Access Control 支持运行 CA Top Secret、CA ACF2 或 RACF 安全产品（和 CA Common Services CAICCI 程序包）的大型机与运行 CA Access Control 的 Windows 或 UNIX 计算机之间的密码同步。 同步是使用标准 CA Access Control 密码策略模型方法完成的。

大型机用户执行的任何密码更改都将传播至该密码策略模型层级结构中的所有计算机。

大型机密码同步先决条件

要在安装了 TNG/TND/NSM 的服务器上使用“大型机密码同步”功能，CA Access Control 需要一个先决条件：TNG/TND/NSM 修正 T129430。 请与技术支持部门联系，以获取该修正。

第 11 章： 一般安全功能

此部分包含以下主题：

维护模式保护（无人值守模式）

CA Access Control 具有维护模式（也称为静默模式），用于在服务停止以进行维护期间提供保护。 在该模式下，CA Access Control 将在这些服务停止时拒绝事件。

CA Access Control 运行时将拦截安全敏感事件，并检查是否允许该事件。 在未激活维护模式的情况下，当 CA Access Control 服务关闭时，将允许所有事件。 在活动的维护模式下，CA Access Control 服务在停止时将拒绝事件，在系统维护时停止用户活动。

维护模式可进行调整，默认情况下该模式处于禁用状态。

CA Access Control 安全服务关闭时：

- 如果维护模式处于活动状态，将拒绝所有安全敏感事件（特殊情况和由维护用户执行的事件除外）。
- 如果已禁用维护模式，CA Access Control 将不做干预，执行将转到操作系统。

激活维护模式并且关闭安全机制时，阻止的事件不会记录到审核日志文件中。

要启用维护模式，请遵循下列步骤：

1. 确保 CA Access Control 服务已停止。
2. 使用注册表编辑器导航至注册表键

   \HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl\FsiDrv

   并更改下列值：

   - SilentModeEnabled = 1
   - SilentModeAdmins = *special_admins*

     变量 *special_admins* 用于定义 CA Access Control 服务停止时可以访问计算机的用户名列表。

     为每个用户使用新行。 无论是否指定，*SYSTEM* 均始终为维护模式用户。

     注意：在 Windows 2000 和 Windows NT 上，不能使用 regedit 编辑 SilentModeAdmins 注册表键，而应使用 Regedt32.exe。
3. 使用命令 shell 中的“seosd start”命令，或使用 Windows“开始”菜单中的选项，启动 CA Access Control 服务。

现在，如果 CA Access Control 服务已关闭，则只有 SilentModeAdmins 注册表键下列出的用户有权访问计算机，而所有其他用户的尝试活动都将收到拒绝的回答。

跳过驱动程序

要指定某些驱动程序可以不需提交操作即进行 CA Access Control 授权检查，请为这些驱动程序定义跳过。 例如，如果为防病毒程序驱动程序定义了跳过，那么它可以打开文件进行扫描不需 CA Access Control 授权检查。 如果没有跳过，驱动程序可能导致 CA Access Control 的死锁。

注意：对于当前版本的 Trend Micro™ PC-cillin Antivirus，跳过配置是预先配置好的。

跳过驱动程序

1. 为要定义跳过的驱动程序的数量设置 BypassDriversCount 注册表项值。

   您可以在 CA Access Control 注册表的 FsiDrv 键中找到该项。

   注意：必须停止 CA Access Control 之后才能更改 CA Access Control 注册表项。

2. 针对要跳过的每个驱动程序：

   a. 创建名为 DriverName_*drvNumber* 的类型 REG_SZ 的注册表项。

      第一项应为 DriverName_0，最后一项为 DriverName_*X*，其中 *X* 为 BypassDriversCount - 1。

   b. 编辑每个 DriverName_*drvNumber* 项，其值是要跳过的驱动程序的名称。

      值仅为驱动程序的名称（如 thisdrv.sys）。

3. 重新启动 CA Access Control。

   CA Access Control 重新加载并跳过在注册表中定义的驱动程序。

示例：跳过驱动程序解决兼容性问题

此示例通过定义要跳过的防病毒驱动程序（avDriverA.sys 和 avDriverB.sys），解决了防病毒产品与 CA Access Control 之间的兼容性问题。 在 FsiDrv 键下 CA Access Control 注册表树中设置驱动程序跳过的注册表项：

```
HKLM\SOFTWARE\ComputerAssociates\AccessControl\FsiDrv
```

如下设置注册表项：

名称	类型	数据
BypassDriversCount	REG_DWORD	2
DriverName_0	REG_SZ	avDriverA.sys
DriverName_1	REG_SZ	avDriverB.sys

BypassDriversCount 注册表键值 2 向 CA Access Control 说明要查找要跳过的两个驱动程序。 每个 DriverName_*drvNumber* 注册表项值定义要跳过的驱动程序。

切换驱动程序拦截

您可以激活或停用 CA Access Control 筛选器驱动程序的拦截。

注意：当拦截被停用时，仍会应用筛选器驱动程序不强制实施的 CA Access Control 保护。 该功能包括密码质量检查、登录事件、Windows 服务事件、STOP 等等。

要激活拦截，请将 UseFsiDrv 设置为 1；要停用拦截，将 UseFsiDrv 设置为 0。

您可以在 CA Access Control 注册表的 AccessControl 键中找到该配置设置。

更改该注册表值之后，请重新启动 CA Access Control 服务。

禁用 CA Access Control 内核拦截

您可以在内核级别禁用以下 CA Access Control 拦截：

- 网络拦截
- 进程拦截
- 注册表拦截
- 文件拦截

甚至在禁用网络、进程、注册表和文件类且不使用这些类拦截内核活动时，仍会在启动时初始化网络、进程、注册表和文件拦截处理代码并在运行时工作，影响性能。 要提高性能，可以在启动时从初始化中禁用一个或多个拦截。

要在内核级别禁用 CA Access Control 拦截

1. 创建一个或多个以下类型 REG_DWORD 的注册表项并将其值设置为 1。
   - DisableNetworkInterception - 禁用网络拦截
   - DisableProcessInterception - 禁用进程拦截
   - DisableRegistryInterception - 禁用注册表拦截
   - DisableFileInterception - 禁用文件拦截

   必须在以下注册表项下创建这些条目：

   `HKLM\SYSTEM\CurrentControlSet\Services\drveng\Parameters`
2. 重新启动计算机。

   CA Access Control 重新加载，无需初始化禁用的拦截类型。

堆栈溢出保护

堆栈溢出保护 (STOP) 是防止黑客创建和利用堆栈溢出进入系统的一种功能。 堆栈溢出使黑客可以在远程或本地系统上像管理员一样多次地执行任意命令。 他们利用操作系统或其他系统中的缺陷来进行该操作。 这些特殊类型的缺陷允许用户覆盖程序堆栈，更改要执行的下一个命令。

STOP 通过拦截对计算机上每个应用程序的重要操作系统调用来发挥作用。 如果调用可疑，则会对每个调用进行初始分析，然后发送调用以进一步分析。 进一步分析将通过 STOP 配置和签名文件中的数据来执行。

启用 STOP

STOP 可以防止黑客创建和利用堆栈溢出进入系统。 您可以在安装 CA Access Control 时启用 STOP。 也可以手动启用 STOP。

启用 STOP

1. 输入下面的命令：

   ```
   secons -s
   ```

   CA Access Control 将关闭。

2. 将 STOP *OperationMode* 注册表项设置为 1。

   可以在以下注册表键中找到注册表项：

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl\Instrumentation\PlugIns\StopPlg
   ```

   CA Access Control 启动后，将加载 STOP 模块，并在计算机上启用 STOP。

3. （可选）使用以下注册表键中的注册表项来调整 STOP 配置：

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl\Instrumentation\PlugIns\StopPlg
   ```

   ```
   HKEY_LOCAL_MACHINE\Software\ComputerAssociates\AccessControl\STOP
   ```

   注意：有关 STOP 注册表设置的详细信息，请参阅 *《参考指南》*。

4. 输入以下命令：

   ```
   seosd -start
   ```

   CA Access Control 将启动。

为接收签名文件更新配置 STOP

您可以确保环境中的所有计算机都具有防止堆栈溢出所需的最新 STOP 信息。 您可以通过在中央计算机上更新 STOP 签名文件，并将计算机设置为定期检索该文件来执行该操作。

为接收签名文件更新配置 STOP

1. 输入下面的命令：

   ```
   secons -s
   ```

   CA Access Control 将关闭。

2. 将 *STOPSignatureBrokerName* 注册表项设置为想要 CA Access Control 检索签名文件的计算机的主机名。

   可以在以下注册表键中找到注册表项：

   ```
   HKEY_LOCAL_MACHINE\Software\ComputerAssociates\AccessControl\STOP
   ```

   启动 CA Access Control（然后按照定义的时间间隔）后，CA Access Control 将从指定的计算机检索 STOP 签名文件。

3. 将 *STOPUpdateInterval* 注册表项设置为希望签名文件更新的时间间隔。

   CA Access Control 将按照指定的时间间隔从指定的计算机检索签名文件。

4. （可选）使用以下注册表键中的注册表项来调整 STOP 配置：

   ```
   HKEY_LOCAL_MACHINE\Software\ComputerAssociates\AccessControl\STOP
   ```

   注意：有关 STOP 注册表设置的详细信息，请参阅 *《参考指南》*。

5. 输入以下命令：

   ```
   seosd -start
   ```

   CA Access Control 将启动。

注意：您可以使用 eACSigUpdate 实用程序从任何主机检索签名文件。有关该实用程序的详细信息，请参阅 *《参考指南》*。

第 12 章： 配置设置

通过 CA Access Control，您可以远程管理 CA Access Control 端点配置设置。 您可以使用 CA Access Control 端点管理 或 selang 配置环境执行此操作。

此部分包含以下主题：

配置设置

CA Access Control 在以下位置存储其使用的端点和策略模型配置设置：

- Windows 计算机上的 Windows 注册表
- UNIX 计算机上的初始化文件 (.ini)

注意：关于您可进行的配置设置及其含义的信息，请参阅*《参考指南》*。

更改配置设置

要影响 CA Access Control 和任何策略模型的工作方式，需要更改配置设置。

更改配置设置

1. 在 CA Access Control 端点管理 中，执行如下操作：
   a. 单击“配置”。
   b. 单击“远程配置”。
   将显示“远程配置”页面。
2. 在左侧“远程配置区”窗格中，按要求展开配置树，以展示包含您要修改的配置设置的部分，然后单击该部分。
   将显示“区域: *sectionName* 系统标记”页面，其中显示了所有的配置设置。
3. 按要求找到并编辑配置设置，然后单击“保存标记”。
   将保存更改的配置设置。

更改审核配置设置

要影响 CA Access Control 生成和存储审核记录的方式，您需要更改审核配置文件中的设置。 使用 selang 命令来更改审核配置文件中的设置。

更改审核配置设置

1. （可选）如果使用 selang 连接到远程主机，请使用以下命令连接该主机：

   ```
   host host_name
   ```

2. 使用以下命令移至配置环境：

   ```
   env config
   ```

3. 使用 editres config 命令按照需要修改配置设置。

   审核配置设置被更改。

示例：修改审核配置文件

以下示例将行添加到审核配置文件中：

```
er CONFIG audit.cfg line+("FILE;*;Administrator;*;R;P")
```