CA Access Control

selang 참조 안내서

r12.5

두 번째 버전

CA 제품 참조

이 문서는 다음 CA 제품을 참조합니다.

- CA Access Control Premium Edition
- CA Access Control
- CA Single Sign-On(CA SSO)
- CA Top Secret®
- CA ACF2™
- CA Audit
- CA Network and Systems Management(CA NSM, 이전 이름: Unicenter NSM 및 Unicenter TNG)
- CA Software Delivery(이전 이름: Unicenter Software Delivery)
- CA Enterprise Log Manager
- CA Identity Manager

설명서 규칙

CA Access Control 설명서는 다음과 같은 규칙을 따릅니다.

형식	의미
고정 폭 글꼴	코드 또는 프로그램 출력
기울임꼴	강조 또는 새 용어
굵게	표시된 대로 동일하게 입력해야 하는 텍스트
슬래시(/)	UNIX 및 Windows 경로를 기술하는 데 사용되는 플랫폼 독립적인 디렉터리 구분 기호

이 설명서는 또한 명령 구문과 사용자 입력(고정 폭 글꼴로 표시됨)을 설명할 때 다음과 같은 특별한 규칙을 사용합니다.

형식	의미
기울임꼴	반드시 입력해야 하는 정보
대괄호([ ]) 사이	선택적 피연산자
중괄호({ }) 사이	필수 피연산자 집합

형식	의미
파이프(|)로 구분된 선택 사항	대체 피연산자(하나 선택)를 구분합니다. 예를 들어, 다음은 사용자 이름 또는 그룹 이름 중 하나라는 의미입니다. `{username|groupname}`
...	앞의 항목 또는 항목 그룹이 반복될 수 있음을 나타냅니다.
밑줄	기본값
줄 마지막에 공백 다음의 백슬래시(\)	때때로 이 안내서에서 명령이 한 줄에 모두 표시되지 않는 경우가 있습니다. 이런 경우에는 줄 끝에 공백과 백슬래시(\)를 표시하여 명령이 다음 줄에서 계속됨을 나타냅니다. **참고:** 실제 명령을 입력할 때는 이러한 백슬래시를 포함하지 말고 줄바꿈 없이 명령을 한 줄에 입력하십시오. 백슬래시 및 줄바꿈은 실제 명령 구문에 포함되지 않습니다.

예제: 명령 표기 규칙

다음 코드는 이 안내서에서 명령 규칙이 사용되는 방식을 보여 줍니다.

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

설명:

- 표시되는 그대로 입력해야 하는 명령 이름(ruler)은 일반 고정 폭 글꼴로 표시됩니다.
- className 옵션은 클래스 이름(예: USER)이 들어갈 자리이므로 기울임꼴로 표시됩니다.
- 대괄호로 묶인 두 번째 부분은 선택적 피연산자를 의미하므로 이 부분 없이 명령을 실행할 수도 있습니다.
- 옵션 매개 변수(props)를 사용할 때 키워드 all 을 선택하거나 하나 이상의 속성 이름을 쉼표로 구분하여 지정할 수 있습니다.

기술 지원팀에 문의

온라인 기술 지원 및 지사 목록, 기본 서비스 시간, 전화 번호에 대해서는 http://www.ca.com/worldwide에서 기술 지원팀에 문의하십시오.

설명서 변경 사항

두 번째 버전

이 설명서의 두 번째 버전은 r12.5 GA 때 제공되었습니다.

이 버전에서 다음 항목이 추가 또는 업데이트되었습니다.

- editres config - 구성 설정 수정(페이지 161) - 업데이트된 항목에서 value, value+, value- 매개 변수의 용도에 대해 설명합니다.
- LOGINAPPL 클래스(페이지 337) - 업데이트된 항목에서 LOGINSEQUENCE 매개 변수의 사용 방법에 대해 설명합니다.

첫 번째 버전

설명서의 첫 번째 버전은 r12.5 에서 제공되었습니다. 이 버전에서 설명서의 r12.0 SP1 릴리스에 다음 사항이 업데이트되었습니다.

- selang 명령(페이지 39) - 업데이트된 장은 다음과 같은 새 명령 및 변경된 명령을 설명합니다.
  - get dbexport Command - 내보낸 데이터베이스 규칙 검색(페이지 117)
  - start dbexport 명령 - 데이터베이스 내보내기 시작(페이지 155)
  - editres config - 구성 설정 수정(페이지 161) - PMDB 필터 파일 기능에 대한 설명이 업데이트되었습니다.
  - showres config - 구성 정보 표시(페이지 165) - PMDB 필터 파일 기능에 대한 설명이 업데이트되었습니다.
- AC 환경의 클래스(페이지 224) - 다음과 같은 새 클래스 및 변경된 클래스에 대한 설명이 섹션에서 업데이트되었습니다.
  - ACVAR 클래스(페이지 224) - 새 클래스에 설명이 추가되었습니다.
  - DEPLOYMENT 클래스(페이지 254) - 새 TARGETYPE 속성으로 업데이트되었습니다.
  - GPOLICY 클래스(페이지 291) - 새 POLICY_TYPE 속성으로 업데이트되었습니다.
  - HNODE 클래스(페이지 312) - 새 NODE_TYPE, NODE_VERSION, NODE_INFO, UNAB_ID 속성으로 업데이트되었습니다.
  - LOGINAPPL 클래스(페이지 337) - LOGINFLAGS 속성의 pamlogin 값으로 업데이트되었습니다.

- POLICY 클래스(페이지 348) - 새 VARIABLES 및 POLICY_TYPE 속성으로 업데이트되었습니다.
- PROGRAM 클래스(페이지 358) - ACL, CALACL, NACL, PACL 속성의 값으로 업데이트되었습니다.
- RULESET 클래스(페이지 376) - 새 EXPANDED_COMMANDS 및 EXPANDED_UNDO_COMMANDS 속성으로 업데이트되었습니다.
- SEOS 클래스(페이지 385) - 새 SYSTEM_AAUDIT_MODE 속성으로 업데이트되었습니다.
- SPECIALPGM 속성(페이지 391) - SPECIALPGMTYPE 속성의 새 변경된 값으로 업데이트되었습니다.

목차

제 1 장: 소개

이 장은 아래의 주제를 포함하고 있습니다.

안내서 정보

이 안내서는 CA Access Control selang 명령, 데이터베이스 클래스와 속성 및 Windows 값에 대한 정보를 제공합니다. 이 안내서는 또한 엔터프라이즈 관리 및 보고 기능과 고급 정책 관리 기능을 제공하는 CA Access Control Premium Edition 과 함께 제공됩니다.

용어를 간단히 나타내기 위해 이 안내서에서는 제품을 CA Access Control 이라고 합니다.

안내서 사용자

이 안내서는 selang 명령을 실행하거나 CA Access Control 로 보호되는 환경을 유지 관리 및 구성하는 보안 및 시스템 관리자용으로 작성되었습니다.

제 2 장: selang 명령 언어

이 장은 아래의 주제를 포함하고 있습니다.

CA Access Control 명령줄 인터프리터

CA Access Control 은 CA Access Control 명령 언어인 selang 으로 알려진 명령 셸을 통해 관리됩니다. selang 명령 언어를 사용하면 CA Access Control 데이터베이스에서 정의를 만들 수 있습니다. selang 명령 언어는 명령 정의 언어입니다.

selang 유틸리티는 CA Access Control 설치 디렉터리의 bin 디렉터리에 있습니다. selang 환경에 들어가면 특수 selang 프롬프트가 나타납니다. 프롬프트의 정확한 형식은 작업 환경에 따라 다르며, 다음과 비슷합니다.

```
AC>
```

기본적으로 selang 명령 셸은 로컬 데이터베이스에서 작동합니다. 다른 스테이션의 CA Access Control 데이터베이스에서 작동하려면 selang 명령을 입력하기 전에 hosts 명령을 지정하십시오.

추가 정보:

selang 환경(페이지 34)
hosts 명령-원격 CA Access Control 터미널에 연결(페이지 122)

selang 유틸리티-CA Access Control 명령줄 실행

selang 유틸리티는 CA Access Control 데이터베이스 및 네이티브 환경에 대한 액세스를 제공하는 명령 셸을 호출합니다. 명령 셸 내에서 selang 명령을 작성함으로써 동적으로 데이터베이스가 업데이트됩니다.

참고: -o 옵션을 사용하지 않으면 명령 실행 결과는 표준 출력으로 전송됩니다.

UNIX 에서 이 명령의 형식은 다음과 같습니다.

```
selang [{-c command|-f file}] [{-d path|-p pmdb}] [-o file] [-r file] [-s] \
[-u user pass]
selang [-l] [-o file] [-r file] [-s] [-u user pass]
```

Windows 에서 이 명령의 형식은 다음과 같습니다.

```
selang [{-c command|-f file}] [{-d path|-p pmdb}] [-o file] [-r file] [-s] [-v]
selang [-l] [-o file] [-r file] [-s] [-v]
```

-c command

실행할 selang 명령을 지정합니다. selang 은 명령을 실행한 후 종료됩니다.

command 에 공백이 포함되어 있으면 전체 문자열 앞뒤에 따옴표를 입력하십시오. 예:

```
selang -c "showusr rosa"
```

-d path

selang 명령이 정의된 경로에서 데이터베이스를 업데이트하도록 지정합니다.

참고: 로컬 데이터베이스만 지정할 수 있습니다.

-f file

터미널의 표준 입력이 아닌 정의된 파일로부터 selang 명령을 읽도록 지정합니다.

selang 은 입력 파일의 명령을 실행하므로 실행 중인 명령의 행 수가 화면에 나타납니다. selang 프롬프트는 화면에 나타나지 않습니다. selang 은 file 에서 명령을 실행한 후 종료됩니다.

-h

이 유틸리티에 대한 도움말을 표시합니다.

-l

selang 이 기본 로컬 데이터베이스를 업데이트하도록 지정합니다. 이 데이터베이스는 대개 ACInstallDir/seosdb 이며, 여기서 ACInstallDir 은 CA Access Control 이 설치된 디렉터리입니다.

이 옵션을 -d 또는 -p 와 함께 지정할 필요는 없습니다.

참고: 이 옵션은 selang 을 대체합니다. 이 옵션은 seosd 가 실행되고 있지 않을 경우에만 유효하며 슈퍼 사용자만 실행할 수 있습니다.

-o file

selang 출력이 지정한 파일에 기록되도록 지정합니다. selang 을 호출할 때마다 새로운 빈 파일이 생성됩니다. 기존 파일의 이름을 지정하면 selang 은 파일에 있는 현재 정보를 덮어씁니다.

-p pmdb

selang 명령이 정의된 PMDB 의 데이터베이스를 업데이트하도록 지정합니다. 이 데이터베이스는 로컬 스테이션, 즉 PMDB 하위 디렉터리에 있습니다. 데이터베이스에 대한 변경 사항은 구독자에게 전파되지 않습니다.

참고: 이 옵션은 sepmdd 또는 seosd 가 지정된 PMDB 에서 실행되고 있으면 유효하지 않으며, hosts 명령을 사용하는 것과는 다릅니다.

중요! 이 모드에서 전파가 필요한 사항은 변경하지 마십시오. 업데이트 시 기본 모드를 사용하면 CA Access Control 은 기본 호스트 파일만 업데이트합니다(CA Access Control 구성 옵션에 정의되어 있는 대로).

-r file

selang 이 정의된 파일에서 명령을 읽도록 지정합니다. 파일은 세미콜론 또는 줄 바꾸기 문자로 구분된 일반 selang 구문의 명령으로 구성되어야 합니다. selang 은 file 의 명령을 실행한 후 사용자에게 입력하라는 메시지를 표시합니다.

이 옵션에서 파일이 정의되어 있지 않으면 selang 은 홈 디렉터리의 .selangrc 파일을 사용합니다.

-s

selang 이 저작권 메시지를 표시하지 않은 채 자동 모드에서 열리도록 지정합니다.

-u user pass

(UNIX 전용) selang 을 실행하기 위한 사용자 이름과 암호를 지정합니다.

이 옵션을 사용하려면 seos.ini 파일에서 check_password 토큰을 yes 로 설정해야 합니다. 그러면 selang -u 를 실행할 때 CA Access Control 에서 "암호를 입력하십시오."라는 메시지가 표시됩니다. 로그인은 세 번 시도할 수 있습니다.

seos.ini 파일의 [lang] 섹션에 있는 no_check_password_users 토큰에는 로그인 중 selang 에 대한 암호 검사를 바이패스하는 사용자 목록이 있습니다.

참고: check_password 토큰이 no 로 설정되어 있는 경우 selang 은 암호를 요구하지 않습니다.

-v

(Windows 전용) 출력할 명령줄을 작성합니다.

참고:

- -h 를 사용하면 모든 다른 옵션은 무시됩니다.
- -c 옵션은 -f 옵션과 함께 사용할 수 없습니다.
- -d 옵션은 -p 옵션과 함께 사용할 수 없습니다.
- -d 나 -p 를 지정하는 경우에는 -l 을 지정할 필요가 없습니다.

추가 정보:

hosts 명령-원격 CA Access Control 터미널에 연결(페이지 122)

selang 명령 셸의 기능

selang 명령 셸을 입력하고 나면 다음 프롬프트가 나타납니다.

```
AC>
```

프롬프트가 나타나면 selang 명령을 입력할 수 있습니다. 여러 명령을 입력할 때는 세미콜론(;)으로 구분합니다. 두 개 이상의 행에 명령을 입력하려면, 행의 끝에 백슬래시(\)를 입력하고 다음 행에서 계속 명령을 입력하십시오. 명령줄을 편집할 수 있습니다. 왼쪽 화살표 키와 오른쪽 화살표 키를 사용하여 행 사이를 이동합니다. 문자를 직접 입력하여 원하는 위치에 삽입할 수 있으며, 표준 백스페이스 및 Delete 키를 사용하여(UNIX 에서는 Ctrl+D 를 눌러) 문자를 삭제할 수 있습니다.

selang 은 tcsh 및 기타 스마트 셸에서 사용할 수 있는 많은 명령줄 입력 기능을 지원합니다. 여기에는 다음과 같은 기능이 포함됩니다.

- 특수 문자
- 바로 가기 키
- 명령 기록
- 특수 기능

참고: UNIX 에서는 사용자나 그룹의 추가 또는 업데이트 전후에 자동으로 실행되도록 지정(셸 스크립트 또는 실행 파일)할 수 있는 프로그램인 UNIX exit 를 사용할 수 있습니다. UNIX exit 에 대한 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

특수 문자

selang 은 다음의 특수 문자를 지원합니다.

문자	설명	의미
# 또는 *	파운드(해시) 또는 별표	행을 # 또는 *로 시작하면 행이 주석 처리되어 실행되지 않았음을 나타냅니다. 주석 행은 파일에서 selang 명령을 입력할 때 유용합니다.
!	느낌표	행의 시작 부분에서 행의 나머지가 셸 명령임을 나타냅니다. selang 은 실행을 위해 명령을 운영 체제 셸 프로그램으로 전송하며, CA Access Control 은 행을 실행하지 않습니다.
\	백슬래시	행의 마지막 문자로 명령이 다음 행에서 계속됨을 나타냅니다.
;	세미콜론	명령을 종료하고 동일한 행에 새 명령을 소개합니다.
|	파이프	이전 명령의 출력을 다음 명령의 입력으로 보냅니다(지정된 파이프).

바로 가기 키

selang 은 다음의 바로 가기 키를 지원합니다.

키	유효한 OS	의미
위쪽-화살표, 아래쪽-화살표, 또는 ^	모두	명령 기록에서 명령을 탐색 및 검색하기 위해 사용됩니다.

키	유효한 OS	의미
탭	UNIX	단어 완성에 사용됩니다.
Ctrl+D	UNIX	행의 끝에 커서를 위치시키면 명령줄의 단어 완성 문자열과 일치하는 단어의 목록을 표시합니다. 커서를 행의 원하는 위치에 둔 상태에서 커서의 오른쪽에 있는 문자를 삭제합니다.
Esc, Esc Ctrl+2	UNIX	명령줄에 명령 도움말 텍스트를 표시합니다. 명령줄의 텍스트는 모두 보존되므로, 입력을 멈춘 위치에서 명령을 계속 입력할 수 있습니다.
F1	Windows	문자 단위로 이전 명령을 삽입합니다.
F2	Windows	"Enter char to copy up to:"라는 지침이 있는 창을 표시합니다. 이전 명령의 문자를 입력하면 selang 은 명령을 문자의 첫 번째 인스턴스까지 입력합니다. 명령에 문자가 두 번 이상 나타나는 경우 F2 키를 다시 클릭하여 다음 인스턴스까지 삽입할 수 있습니다. 취소하려면 백스페이스 키를 사용합니다.
F3	Windows	이전 명령을 입력합니다(위쪽 화살표와 동일함).
F4	Windows	이전 명령을 편집합니다. "Enter char to delete up to:"라는 지침이 있는 창을 표시합니다. 취소하려면 백스페이스 키를 사용합니다.
F5	Windows	이전 명령을 입력합니다(위쪽 화살표와 동일함).
F6	Windows	명령행에 Ctrl Z(^Z)를 입력합니다. 이렇게 하면 Enter 키를 클릭한 후 다음 줄에 명령을 계속 입력할 수 있습니다.
F7	Windows	명령 기록을 나열하는 창을 표시합니다. 위쪽 및 아래쪽 화살표를 사용하여 이전 명령을 선택할 수 있습니다. 취소하려면 Esc 키를 사용합니다.
F8	Windows	위쪽 화살표를 사용할 때와 같이 이전 명령을 입력하지만 커서가 명령행의 끝이 아닌 명령행의 맨 앞에 위치합니다.
F9	Windows	"Enter command number:" 지침이 있는 창을 표시합니다. 번호를 입력하면 F7 목록의 해당 번호와 함께 명령이 삽입됩니다. 취소하려면 Esc 키를 사용합니다.

명령 기록

selang 은 실행된 명령을 기록 목록에 저장합니다. 위쪽 화살표와 아래쪽 화살표를 사용하여 기록 목록에서 명령행에 명령을 표시하십시오. 특수 문자 또는 문자열로 시작하는 명령만 보려면 위쪽 화살표와 아래쪽 화살표를 사용하기 전에 명령의 시작을 입력하십시오. Enter 키를 누르면 명령줄에 현재 표시된 텍스트가 실행됩니다.

이전에 실행한 명령을 보려면 history 명령을 입력하십시오.

selang 명령 셸은 기록 목록에 저장된 명령을 사용하는 다음 바로 가기를 지원합니다.

바로 가기	실행 명령
^^ [string]	이전 명령. string 을 지정하면 selang 은 원래 명령에 이 문자열을 추가합니다.
^n [string]	기록 목록에서 n 번째 명령을 실행합니다. 여기서 n 은 양의 정수입니다. string 을 지정하면 selang 은 원래 명령에 이 문자열을 추가합니다.
^-n [string]	목록의 끝에서 n 번째 명령으로, 여기서 n 은 양수입니다. string 을 지정하면 selang 은 원래 명령에 이 문자열을 추가합니다.
^mask [string]	mask 로 시작하는, 가장 최근에 실행한 명령입니다. 여기서 mask 는 텍스트 문자열입니다. string 을 지정하면 selang 은 원래 명령에 이 문자열을 추가합니다.

참고: Windows 에서 명령 기록을 보려면 F7 키를 사용할 수 있습니다.

특수 기능

여러 가지 추가 방법을 사용하여 selang 명령 셸에 키 입력을 저장할 수 있습니다.

참고: 레코드와 클래스 이름의 경우 UNIX 에서는 대/소문자를 구분하지만 Windows 에서는 구분하지 않습니다.

- 명령 인식

  사용 가능한 다른 명령과 구분될 만큼 충분한 문자를 입력하면 selang 은 실행할 명령을 즉시 인식합니다. 예를 들어, **ho** 를 입력하면 hosts 명령을 실행하는데, 이는 ho 로 시작되는 명령이 hosts 밖에 없기 때문입니다. **ho** 를 입력하자마자 selang 은 사용자가 의도하는 명령을 인식할 수 있습니다. 그러나 문자열 **new** 로 시작하는 명령은 여러 개가 있습니다. newusr, newgrp, newfile 및 newres 를 구분하기에 충분한 문자를 추가해야 합니다.

- 약어

  각 명령은 1 자에서 4 자까지의 약어와 연결되어 있습니다. 예를 들어, 문자열 new 로 시작하는 명령은 여러 개가 있으므로 **newusr** 명령의 약어인 **nu** 를 사용할 수도 있습니다. 이러한 약어의 설명은 각 명령에 대한 명령 구문의 일부로서 제공됩니다. 명령은 대/소문자를 구분하지 않고 입력할 수 있습니다.

- 단어 완성(UNIX 전용)

  단어 중간에 Tab 키를 누르면 단어가 완성됩니다. 단어 완성은 상황에 따라 달라집니다. 둘 이상의 단어가 지정된 문자열과 일치할 경우 selang 은 문자열과 일치하는 단어 일부 또는 가장 짧은 단어를 사용합니다. 예를 들어 n 문자를 입력할 경우 selang 은 ew 를 제공하여 단어 new 를 구성합니다. 필수 단어가 아닌 경우 한 두 개의 다른 문자를 입력한 후 Tab 키를 다시 눌러 단어를 완성합니다. 사용 가능한 모든 옵션을 보려면 Ctrl+D 를 누르십시오. 이 기능은 사용할 명령을 잘 모르는 경우 유용합니다. 위에 나온 예제를 사용하면, 단어 new 에 문자 u 를 추가하고 Tab 키를 누르면 selang 은 sr 을 추가하여 newusr 명령을 제공합니다.

  selang 명령에 해당되지 않는 단어는 나중에 같은 세션에서 단어 완성 기능에 사용될 수 있도록 메모리에 저장됩니다.예를 들어, newusr Mercedes 를 입력한 후 showusr Me 를 입력하고 Tab 키를 누르면 Me 는 다음과 같이 Mercedes 로 확장됩니다.

  ```
  showusr Mercedes
  ```

  단, "Me"로 시작하는 다른 사용자 이름을 입력하지 않았다고 가정합니다.

와일드카드 일치

selang 은 다음의 와일드카드 문자를 지원합니다.

*** (별표)**

0 개 이상의 임의 문자

? (물음표)

임의의 단일 문자(파일의 경로 구분자 제외).

하나의 문자를 임의의 다른 단일 문자와 일치하는 "모두 일치(do not care)" 문자로 만들려면 다음 예와 같이 물음표(?)를 사용하십시오.

지정 값	일치
mmc?	mmc3, mmcx, mmc5
mmc?.t	mmc1.t, mmc2.t
mmc04.?	mmc04.a, mmc04.1

0 개 이상 문자의 문자열을 일치시키려면 다음 예와 같이 별표(*)를 사용하십시오.

지정 값	일치
i.c	main.c, list.c
st*.h	stdio.h, stdlib.h, string.h
*	지정한 클래스의 모든 레코드

selang 명령 구문

각 selang 명령은 CA Access Control 데이터베이스에서 특정 작업을 수행합니다. selang 명령의 구문은 다음과 같습니다.

```
commandname parameters
```

명령 이름은 실행할 명령을 CA Access Control 에 알려줍니다. 명령 뒤에는 일반적으로 명령을 실행하는 데 필요한 추가 정보를 CA Access Control 에 제공하는 하나 이상의 매개 변수를 사용합니다.

selang 매개 변수 구문은 다음과 같습니다.

```
parameterName[(arguments)]
```

매개 변수 이름은 CA Access Control 에 대한 매개 변수를 식별합니다. 매개 변수 중 다수는 매개 변수 처리에 필요한 정보를 CA Access Control 에 제공하는 인수를 요구합니다. 일부 매개 변수는 둘 이상의 인수를 사용합니다. 둘 이상의 인수를 지정할 때 쉼표 또는 공백으로 인수를 구분하십시오. 매개 변수의 인수 자체가 매개 변수가 될 수도 있습니다.

문자열이 인수를 정의할 때 레코드 속성을 제거하려면 간단히 빈 괄호**()**와 함께 속성을 입력하면 됩니다. 어떤 경우에는 별표(*****)를 인수로 사용하여 해당 인수에 대해 가능한 모든 값을 포함할 수 있습니다. 별표를 사용할 경우 별표는 동일한 인수에 특정 값을 제공하는 이전 또는 이후 명령보다 우선하지 않습니다. 또한 인수가 파일 이름일 경우 와일드카드를 파일 이름 패턴의 일부로 사용할 수 있습니다. 와일드카드는 *("0 개 이상의 문자")와 ? (하나의 문자)입니다.

UNIX 환경에서 사용자 제공 정보는 대소문자를 구분하며 소문자와 대문자를 모두 사용할 수 있습니다. 예를 들어, 사용자 ID 가 user53 인 사용자의 전체 이름을 Mike Jones 로 지정할 수 있습니다. Windows 에서는 대/소문자로 구분하여 정보를 인식하지 않지만 대/소문자를 구분하여 저장합니다. UNIX 워크스테이션에서 원격 Windows 호스트를 관리할 경우 UNIX 는 저장된 대로 사용자가 제공한 정보를 찾습니다. 예를 들어, 사용자가 Windows 환경에서 Mike Jones 로 식별되는 경우 로컬 CA Access Control 데이터베이스를 관리할 때 이 사용자의 이름을 mike jones 로 입력할 수 있습니다. 그러나 원격 UNIX 컴퓨터에서 데이터베이스를 관리하려면 이름을 Mike Jones 로 입력해야 합니다.

selang 명령 권한 부여

AC 또는 기본 운영 체제(네이티브 OS) 환경에서 레코드를 변경하는 selang 명령을 사용하려면 충분한 권한을 가지고 있어야 합니다. 대부분의 명령은 다음 조건 중 하나를 충족해야 합니다.

- 사용자가 리소스의 소유자입니다.
- 사용자가 ADMIN 특성을 가집니다.
- 리소스 레코드는 사용자가 GROUP-ADMIN 속성을 가지는 그룹의 범위 내에 있습니다.
- 사용자가 ADMIN 클래스의 레코드 ACL 에서 CREATE 또는 MODIFY 액세스 권한을 가집니다.

- (Windows) 설치에서 네이티브 Windows 환경의 관리만 허용하는 경우 사용자는 Windows 데이터베이스에서 CA Access Control 관리자 그룹의 구성원입니다.
- (UNIX) 설치에서 기본 UNIX 환경의 관리만 허용하는 경우 사용자는 로컬 UNIX 호스트의 보안 파일에서 CA Access Control 관리자 그룹의 구성원입니다.

참고: 이러한 일반 규칙에 대한 예외는 각 명령에 대한 설명에 나와 있습니다.

Access 제어 목록 지원

액세스 권한을 허용 또는 거부하려면 일곱 가지 유형의 액세스 제어 목록을 사용할 수 있습니다.

ACL

리소스에 액세스할 수 있는 사용자 이름 및 그룹 이름과 각 이름에 부여된 액세스 수준을 포함하는 표준 액세스 제어 목록입니다.

NACL

리소스에 액세스할 수 있는 권한이 없는 사용자 이름이나 그룹 이름이 들어 있는 네거티브 액세스 제어 목록입니다.

PACL

액세스 프로그램에 따라 달라지는 프로그램 액세스 제어 목록입니다. 각 PACL 에는 사용자 이름과 그룹 이름, 액세스 수준, 특정 리소스에 액세스하기 위해 사용자가 실행해야 하는 프로그램이나 셸 스크립트의 이름이 있습니다.

INET-ACL

인터넷 액세스 제어 목록입니다.

CACL

조건부 액세스 제어 목록입니다.

CALACL

달력 액세스 제어, 즉 Unicenter TNG 달력에 따라 결정되는 리소스 ACL 입니다.

AZNACL

권한 부여 ACL, 즉 리소스 설명에 따라 리소스에 대한 액세스를 허용하는 ACL 입니다.

CA Access Control 은 리소스에 액세스할 수 있는 사용자의 권한을 검사할 때 관련된 모든 목록을 사용합니다.

참고: 단일 권한 부여 명령으로는 임의의 단일 목록을 유지 관리할 수 있습니다. 둘 이상의 목록을 변경하려면 권한 부여 명령을 다시 실행해야 합니다. 하나의 권한 부여 규칙으로 여러 사용자 및 그룹에 대한 여러 액세스 권한을 정의할 수 없으므로, 규칙을 분리해야 합니다.

다음 표는 각 클래스와 함께 사용할 수 있는 액세스 컨트롤 목록을 보여줍니다. 표에 나타나지 않은 클래스는 액세스 제어 목록을 가지고 있지 않으므로 권한 부여 명령으로 제어할 수 없습니다.

클래스	ACL/ NACL	CALACL	PACL	INET-ACL	CACL	AZNACL
ADMIN	X	X	X			
APPL	X	X				X
AUTHHOST	X	X				X
CONNECT	X	X	X			
CONTAINER	X	X	X			
DOMAIN	X	X	X			
FILE	X	X	X			
GAPPL	X	X				X
GAUTHHOST	X	X				X
GFILE	X	X	X			
GHOST				X		
GSUDO	X	X				
GTERMINAL	X	X				
HOLIDAY	X	X				
HOST				X		
HOSTNET				X		
HOSTNP				X		
LOGINAPPL	X	X				
MFTERMINAL	X	X	X			
PROCESS	X	X	X			
PROGRAM	X	X				
REGKEY	X	X	X			

클래스	ACL/ NACL	CALACL	PACL	INET-ACL	CACL	AZNACL
REGVAL	X	X	X			
SUDO	X	X	X			
SURROGATE	X	X	X			
TCP	X	X	X		X	
TERMINAL	X	X	X			
UACC	X	X				
USER_DIR	X					X

클래스별 Access 권한

유효한 액세스 값은 리소스가 속한 클래스에 따라 달라집니다. 다음 표는 AC 환경의 유효한 액세스 값을 클래스별로 보여줍니다.

클래스	유효한 Access 값	Accessor 가 수행할 수 있는 작업
모든 클래스	all	클래스에 대해 모든 유효한 작업을 수행합니다.
	none	클래스에 대해 어떤 유효한 작업도 수행하지 않습니다.
ADMIN	create	이 클래스에서 레코드를 작성합니다.
	delete	이 클래스에서 레코드를 삭제합니다.
	join	USER 레코드에 그룹을 추가하고 사용자를 그룹에 연결합니다. **참고:** 접근자는 modify 액세스도 가지고 있어야 합니다.
	modify	기존 레코드를 수정합니다. **참고:** 사용자를 그룹에 연결하려면(사용자 이름을 GROUP 레코드에 추가) 접근자는 join 액세스도 가지고 있어야 합니다.
	password	다른 사용자의 암호를 변경합니다. **참고:** 이 액세스 유형은 USER 클래스에만 영향을 줍니다.
	read	이 클래스의 레코드를 나열합니다.
AUTHHOST	read	인증된 호스트로부터 로그인합니다.

클래스	유효한 Access 값	Accessor 가 수행할 수 있는 작업
CONNECT	read	원격 호스트에 연결합니다.
CONTAINER	inherited	**참고:** 이 클래스에 대한 유효한 액세스 값은 포함된 개체의 클래스에 대한 액세스 값입니다.
DOMAIN	chmod	한 도메인과 다른 도메인 간의 트러스트 관계를 작성 및 삭제합니다. **참고:** 두 도메인 모두 이 액세스 유형을 가지고 있어야 합니다.
	execute	도메인에서 구성원을 추가 또는 삭제합니다.
	read	도메인 구성원을 나열합니다.
FILE, GFILE	chdir	읽기 및 실행 권한과 동일한 권한으로 디렉터리에 액세스합니다.
	chmod	파일 시스템 모드를 변경합니다. **참고:** UNIX 호스트에서만 적용됩니다.
	chown	레코드의 소유자를 변경합니다.
	control	delete 및 rename 을 제외한 모든 유효한 작업을 수행합니다.
	create	이 클래스에서 레코드를 작성합니다.
	delete	이 클래스에서 레코드를 삭제합니다.
	execute	프로그램을 실행합니다. **참고:** 접근자는 read 액세스도 가지고 있어야 합니다.
	read	파일 또는 디렉터리를 변경하지 않고 사용합니다. **참고:** UNIX 에서 사용자가 파일에 대한 정보를 얻는 작업(예: ls -l)을 수행할 수 있는지 여부를 제어하기 위해 read 권한이 필요한 경우, STAT_intercept 구성 설정을 1 로 설정하십시오. 자세한 내용은 Reference Guide(참조 안내서)를 참조하십시오.
	rename	이 클래스에서 레코드의 이름을 바꿉니다.
	sec	이 클래스에서 레코드의 ACL 을 변경합니다.
	update	read, write 및 execute 작업을 결합하여 수행합니다.
	utime	파일의 수정 시간을 변경합니다.

클래스	유효한 Access 값	Accessor 가 수행할 수 있는 작업
		참고: UNIX 호스트에서만 적용됩니다.
	write	파일 또는 디렉터리를 변경합니다.
HNODE	read	클래스의 레코드를 나열합니다.
	write	레코드의 세부 정보를 편집합니다.
HOLIDAY	read	지정된 휴일 동안 로그인합니다.
KMODULE	load	커널 모듈을 로드합니다.
	unload	커널 모듈을 언로드합니다.
MFTERMINAL	read	메인프레임 터미널로부터 로그인합니다.
	write	메인프레임 터미널로부터 관리합니다.
POLICY	delete	정책을 삭제합니다.
	execute	정책을 배포합니다.
	read	정책 세부 정보를 봅니다.
	write	레코드의 세부 정보를 편집합니다.
	undeploy	delete 및 execute 작업을 결합하여 수행합니다.
PROCESS	read	프로세스를 중지합니다.
PROGRAM, SUDO, GSUDO	execute	프로그램을 실행합니다.
REGKEY	delete	Windows 레지스트리 키를 삭제합니다.
	read	Windows 레지스트리 키의 내용을 나열합니다.
	write	Windows 레지스트리 키를 변경합니다.
REGVAL	delete	Windows 레지스트리 값을 삭제합니다.
	read	Windows 레지스트리 값을 읽습니다.
	write	Windows 레지스트리 값을 변경합니다.
RULESET	read	레코드의 세부 정보를 봅니다.
	write	레코드의 세부 정보를 편집합니다.
SURROGATE	execute	사용자를 서로게이트합니다.
TCP	read	원격 호스트 또는 호스트 그룹에서 TCP 서비스에 액세스합니다.
TERMINAL, GTERMINAL	read	터미널에 로그인합니다.

클래스	유효한 Access 값	Accessor 가 수행할 수 있는 작업
	write	터미널을 관리합니다.
UACC	inherited	**참고:** 이 클래스에 대한 유효한 액세스 값은 정의 중인 클래스에 대한 액세스 값입니다.

참고: none 과 all 값은 모든 클래스에 적용됩니다. all 값은 특정 클래스에 대해 none 이 아닌 액세스 값의 전체 그룹을 나타냅니다. 액세스 권한에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

클래스별 Windows Access 권한

유효한 액세스 값은 리소스가 속한 클래스에 따라 달라집니다. 다음 표는 Windows(nt) 환경의 유효한 액세스 값을 클래스별로 보여줍니다.

클래스	유효한 Access 값	Accessor 가 수행할 수 있는 작업
모든 클래스	all	클래스에 대해 모든 유효한 작업을 수행합니다.
	none	클래스에 대해 어떤 유효한 작업도 수행하지 않습니다.
COM, DISK	change	delete, read 및 write 작업을 결합하여 수행합니다.
	changepermissions	리소스의 ACL 을 수정합니다.
	delete	리소스를 삭제합니다.
	read	리소스의 데이터를 변경하지 않고 리소스의 데이터에 액세스합니다.
	takeownership, chown, owner	지정한 리소스의 소유자를 변경합니다.
	write	지정한 리소스에 데이터를 기록합니다.
FILE		**참고:** FILE 리소스는 NTFS 파일에 대한 액세스 권한만 정의할 수 있으므로 FAT 파일은 액세스 권한을 가질 수 없습니다.
	change	delete, read 및 write 작업을 결합하여 수행합니다.
	changepermissions, sec	리소스의 ACL 을 수정합니다.
	chmod	delete 를 제외한 모든 작업을 수행합니다.
	chown	지정한 리소스의 소유자를 변경합니다.

클래스	유효한 **Access** 값	**Accessor** 가 수행할 수 있는 작업
	delete	리소스를 삭제합니다.
	execute	프로그램을 실행합니다. **참고:** 이 액세스를 사용하려면 접근자는 read 액세스도 가지고 있어야 합니다.
	read	리소스를 변경하지 않고 리소스에 액세스합니다.
	rename	리소스의 이름을 바꿉니다. **참고:** 파일의 이름을 바꾸려면 원본에 대한 delete 액세스 및 대상에 대한 rename 액세스를 가지고 있어야 합니다. 감사 로그는 이러한 이벤트 순서를 반영합니다.
	write	리소스를 수정합니다.
	update	read, write 및 execute 작업을 결합하여 수행합니다.
PRINTER	manage	프린터를 관리합니다. 예를 들면, 지정한 프린터에 대해 데이터를 설정하고, 인쇄를 일시 중지하고, 인쇄를 다시 시작하고, 모든 인쇄 작업을 지우고, ACL 을 업데이트하거나 프린터 속성을 변경합니다.
	print	프린터를 사용하여 인쇄합니다.
REGKEY	append, create, subkey	레지스트리 키의 하위 키를 작성 또는 수정합니다.
	takeownership, chown, owner	리소스 소유자를 변경합니다.
	changepermissions, sec, dac, writedac	리소스의 ACL 을 수정합니다.
	delete	리소스를 삭제합니다.
	enum	하위 키를 열거합니다.
	link	레지스트리 키에 대한 링크를 작성합니다.
	notify	레지스트리 키 또는 레지스트리 키의 하위 키에 대한 알림을 변경합니다.
	query	레지스트리 키의 값을 쿼리합니다.
	read	리소스를 변경하지 않고 리소스에 액세스합니다.
	readcontrol, manage	시스템(감사) ACL 의 정보를 제외한 채 레지스트리 키의 보안 설명자에서 정보를 읽습니다.

클래스	유효한 Access 값	Accessor 가 수행할 수 있는 작업
	set	레지스트리 키의 값을 작성 또는 설정합니다.
	write	레지스트리 키 및 하위 키를 변경합니다.
SHARE	change	리소스 속성을 변경하거나 리소스에서 공유를 제거합니다.
	read	리소스를 변경하지 않고 리소스에 액세스합니다.

참고: none 과 all 값은 모든 클래스에 적용됩니다. all 값은 특정 클래스에 대해 none 이 아닌 액세스 값의 전체 그룹을 나타냅니다. 액세스 권한에 대한 자세한 내용은 Windows 용 끝점 관리 안내서를 참조하십시오.

selang 환경

로컬 CA Access Control 데이터베이스에 대한 작업 외에도 selang 을 사용하여 기본(Windows 또는 UNIX) 데이터베이스, 로컬 PMDB(정책 모델 데이터베이스), CA Access Control 이 설치된 원격 호스트(Windows 또는 UNIX)의 데이터베이스 또는 CA Access Control 구성 설정의 데이터베이스를 수정할 수 있습니다. 환경을 전환하려면 모든 환경에서 사용할 수 있는 env(environment) 명령을 사용하십시오.

일부 명령은 서로 다른 환경에서 동일하지만 다른 매개 변수 및 인수를 포함할 수 있습니다. 따라서 새 환경에서 작업을 시작할 때는 구문을 주의 깊게 확인해야 합니다.

참고: env 를 사용하여 명령의 기본 속성을 입력하면 명령은 네이티브 환경과 현재 환경에 모두 입력됩니다.

지원되는 환경은 다음과 같습니다.

Environment	명령	프롬프트	설명
정책 모델	`env pmd`	`AC(pmd)>`	모든 selang 명령은 로컬 PMDB 에서 작동합니다.
기본 Windows	`env nt`	`AC(nt)>`	모든 selang 명령은 Windows 데이터베이스를 수정합니다.
AC	`env ac`	`AC>`	모든 selang 명령은 CA Access Control 데이터베이스에서 작동합니다. **참고:** 이것이 기본값입니다.

Environment	명령	프롬프트	설명
기본 UNIX	env unix	AC(unix)>	모든 selang 명령은 로컬 UNIX 호스트의 보안 파일에서 작동합니다.
기본	env native	AC(native)>	모든 selang 명령은 호스트의 네이티브 환경에서 작동합니다.
원격 구성	env config	AC(config)>	모든 selang 명령은 호스트에 대한 CA Access Control 구성 설정에서 작동합니다.

추가 정보:

environment 명령-보안 환경 설정(페이지 115)
AC 환경의 클래스(페이지 224)
UNIX 환경의 클래스(페이지 483)
Windows 환경의 클래스(페이지 452)

UNIX 에서의 selang 구성

UNIX 에서는 selang 의 작동 방식을 관리할 수 있습니다. 대부분의 옵션은 selang 이 UNIX 보안 시스템(selang UNIX 환경용)을 관리하는 방식과 관련이 있습니다.

selang 유틸리티는 구성 옵션에 대해 다음의 두 파일을 사용합니다.

seos.ini

CA Access Control 구성 옵션을 포함합니다. 이것이 CA Access Control 의 주요 구성 파일입니다.

lang.ini

selang 이 사용하는 구성 정보를 포함합니다.

selang 은 다음 위치 중 한 곳 또는 모두에 있는 lang.ini 파일을 사용합니다.

- seos.ini 파일이 있는 디렉터리
- 사용자 홈 디렉터리

이러한 lang.ini 파일 중 하나에서만 토큰을 지정할 경우 selang 은 해당 파일에 있는 값을 사용합니다. 두 lang.ini 파일에서 토큰을 다르게 지정하면 사용자 홈 디렉터리의 값이 다른 값보다 우선합니다.

서버의 seos.ini 파일에 있는 DefaultShell 및 DefaultHome 토큰 값이 lang.ini 파일에 있는 DefaultShell 및 HomeDirPrefix 토큰에 설정된 값보다 우선합니다.

참고: 샘플 lang.ini 파일은 ACInstallDir/samples/lang.init 디렉터리에 있습니다.

사용자 파일 변경

UNIX 사용자 업데이트를 위한 기본 파일은 /etc/passwd 이지만 이 기본 파일을 변경할 수 있습니다. 기본 파일을 변경하는 일은 NIS 에서 작업 중인 경우 NIS 서버에서 일반적으로 필요합니다.

사용자 파일을 변경하려면 seos.ini 파일의 passwd 섹션에서 사용자 파일의 전체 경로 이름을 가리키도록 YpServerPasswd 를 수정하십시오.

그룹 업데이트용 파일 변경

UNIX 그룹 업데이트를 위한 기본 파일은 /etc/group 이지만 이 기본 파일을 변경할 수 있습니다. 기본 파일을 변경하는 일은 NIS 에서 작업 중인 경우 NIS 서버에서 일반적으로 필요합니다.

그룹 업데이트용 파일을 변경하려면 **seos.ini** 파일의 **passwd** 섹션에서 사용자 파일의 전체 경로 이름을 가리키도록 **YpServerGroup** 를 수정하십시오.

UNIX 사용자 및 그룹 파일의 자동 백업

한 세션에서 UNIX 사용자를 처음으로 업데이트하기 전에 그리고 한 세션에서 UNIX 그룹을 처음으로 업데이트하기 전에, CA Access Control 은 /etc/passwd 파일이나 /etc/group 파일의 백업 복사본을 만듭니다. 백업 파일 이름은 각각 /etc/passwd.SeOS.bak 및 /etc/group.SeOS.bak 이 됩니다. UNIX 시스템 업데이트 시 오류가 발생할 경우 원래 정보를 복구할 수 있습니다. 백업은 selang 명령 셸 세션에서 UNIX 시스템으로 처음 변경하기 전에만 이루어집니다.

selang 도움말 보기

대화식 selang 명령 환경에서는 언제든지 도움말을 볼 수 있습니다.

selang 온라인 도움말을 보려면 다음 중 하나를 입력하십시오.

? 또는 help

현재의 환경용 selang 온라인 도움말 텍스트가 화면에 나타나고 목차가 표시됩니다.

help topic

topic

selang 명령 또는 selang 명령 셸과 관련된 기타 항목을 정의합니다.

항목을 설명하는 도움말 텍스트가 나타납니다.

help env

env

selang 환경을 정의합니다.

지정한 환경에 대한 도움말 텍스트가 화면에 나타나고 목차가 표시됩니다.

참고: UNIX 에서 명령줄의 텍스트를 삭제하지 않은 채 명령줄에 입력한 명령의 도움말 텍스트를 표시하려면 Ctrl+2 를 누르거나 Esc, Esc 를 누르십시오.

추가 정보:

help 명령-selang 도움말 이용(페이지 121)
selang 환경(페이지 34)
selang 명령 참조(페이지 39)

제 3 장: selang 명령

이 장은 아래의 주제를 포함하고 있습니다.

selang 명령 참조

다음 표는 모든 selang 명령을 알파벳순으로 보여줍니다.

참고: 모든 환경에서 동일한 방식으로 작동하는 명령은 AC 환경에 대해서만 설명되어 있습니다. 일부 명령은 여러 환경에서 사용할 수 있지만 각 환경에서 작동 방식이 다릅니다. 이러한 명령은 아래 표의 설명 열에서 별표(*)로 표시되었으며, 사용 가능한 각 환경에서 별도로 설명되어 있습니다.

명령	약어	환경	설명
alias		AC 및 Unix **참고:** UNIX 호스트 전용	selang 명령 및 속성의 별칭을 나열하거나 정의합니다.
authorize	auth	AC 및 nt	*특정 리소스에 액세스할 때 특정 접근자가 갖는 권한을 설정합니다.
authorize-	auth-	AC 및 nt	*특정 리소스에 액세스할 때 특정 접근자에게 이전에 부여된 권한을 제거합니다.
backuppmd		pmd	PMDB 데이터베이스의 데이터를 지정된 디렉터리에 백업합니다.
check		AC	사용자가 특정 리소스에 대해 액세스 권한을 가지고 있는지 검사합니다.
checklogin		AC	사용자의 로그인 권한, 암호 검사 필요 여부 및 터미널 액세스 검사 필요 여부를 결정합니다.

명령	약어	환경	설명
checkpwd		AC	암호를 변경하지 않고 사용자의 새 암호를 검사하여 암호 규칙을 준수하는지 검사합니다.
chfile	cf	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에서 파일 레코드의 정의를 변경합니다.
chgrp	cg	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에서 기존 내부 그룹 설정을 변경합니다.
chres	cr	AC 및 nt	*CA Access Control 또는 네이티브 OS 데이터베이스에서 기존 리소스 레코드를 변경합니다.
chusr	cu	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에서 기존 내부 사용자를 변경합니다.
chxgrp	cxg	AC	CA Access Control 데이터베이스에서 기존 엔터프라이즈 그룹 설정을 변경합니다.
chxusr	cxu	AC	CA Access Control 데이터베이스에서 기존 엔터프라이즈 사용자 설정을 변경합니다.
createpmd		pmd	원격 호스트에서 PMDB 를 만듭니다.
deletepmd		pmd	원격 호스트에서 PMDB 의 selang 보호 파일, PMDB 디렉터리의 내용 및 PMDB 디렉터리를 제거합니다.
deploy		AC	특정 POLICY 에 대한 RULESET 개체에 저장된 배포 selang 명령을 실행합니다.
deploy-		AC	특정 POLICY 에 대한 RULESET 개체에 저장된 정책 배포 취소 selang 명령을 실행합니다.
editfile	ef	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에서 파일 레코드의 정의를 추가 또는 변경합니다.
editgrp	eg	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에서 새 그룹을 추가하거나 기존 그룹 설정을 변경합니다.
editres	er	AC 및 nt	*CA Access Control 또는 네이티브 OS 데이터베이스에서 새 리소스 레코드를 추가하거나 기존 리소스 레코드를

명령	약어	환경	설명
			변경합니다.
editres config		config	지정한 원본의 구성 설정을 나열합니다.
editusr	eu	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에서 새 사용자를 추가하거나 기존 사용자를 변경합니다.
editxgrp	exg	AC	CA Access Control 데이터베이스에서 새 엔터프라이즈 그룹을 추가하거나 기존 엔터프라이즈 그룹 속성을 변경합니다.
editxusr	exu	AC	CA Access Control 데이터베이스에서 새 엔터프라이즈 사용자를 추가하거나 기존 엔터프라이즈 사용자 속성을 변경합니다.
end_transaction		AC	이중 제어 PMDB 프로세스에 대한 start_transaction 명령을 완료합니다.
환경	env	all	selang 이 작동하는 보안 환경을 설정합니다.
find	f	AC 및 기본	환경의 클래스 또는 클래스의 레코드를 나열합니다.
findpmd		pmd	컴퓨터의 모든 PMDB 를 나열합니다.
find config		config	이 호스트에서 관리할 수 있는 구성 설정(ini 파일 또는 레지스트리 항목)의 출처를 나열합니다.
find file		native	시스템 파일을 나열합니다.
find xgroup		nt	현재 도메인 또는 트러스트된 도메인에 있는 엔터프라이즈 그룹의 이름을 나열합니다.
find xuser		nt	현재 도메인 또는 트러스트된 도메인에 있는 엔터프라이즈 사용자의 이름을 나열합니다.
get dbexport		AC	CA Access Control 또는 PMD 데이터베이스에서 내보낸 규칙을 검색합니다.
get devcalc		AC	정책 위반 계산 결과를 검색합니다.
help		all	selang 도움말을 표시합니다.
history		all	세션에서 이전에 실행된 명령을 표시합니다.
hosts		all	selang 명령이 전달되는 호스트를 표시 또는 설정합니다.
join	j	AC 및 기본	*그룹에 사용자를 조인합니다.
join-	j-	AC 및 기본	*그룹에서 사용자를 제거합니다.

명령	약어	환경	설명
joinx	jx	AC	그룹에 엔터프라이즈 사용자를 조인합니다.
joinx-	jx-	AC	그룹에서 엔터프라이즈 사용자를 제거합니다.
list		AC 및 기본	find 명령의 별칭입니다.
listpmd		pmd	PMDB, 구독자, 업데이트 파일 및 오류 로그에 대한 정보를 나열합니다.
newfile	nf	AC	CA Access Control 데이터베이스에 파일 레코드의 정의를 추가합니다.
newgrp	ng	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에 새 그룹을 추가합니다.
newres	nr	AC 및 nt	*CA Access Control 또는 네이티브 OS 데이터베이스에 새 리소스 레코드를 추가합니다.
newusr	nu	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에 새 내부 사용자를 추가합니다.
newxgrp	nxg	AC	CA Access Control 데이터베이스에 새 엔터프라이즈 그룹을 추가합니다.
newxusr	nxu	AC	CA Access Control 데이터베이스에 새 엔터프라이즈 사용자를 추가합니다.
pmd		pmd	정책 모델 오류 로그를 정리하고, 구독자 목록을 업데이트하고, 구독자를 해제하고, 정책 모델 서비스를 시작 및 중지하고, 업데이트 파일을 자르고, 초기화 파일을 다시 로드합니다.
rename		AC	데이터베이스에 있는 개체의 이름을 변경합니다.
restorepmd		pmd	로컬 호스트에서 PMDB 를 복원합니다.
rmfile	rf	AC	CA Access Control 데이터베이스에서 파일 리소스 레코드를 제거합니다.
rmgrp	rg	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에서 그룹을 제거합니다.
rmres	rr	AC 및 nt	*CA Access Control 또는 기본 Windows 데이터베이스에서 리소스 레코드를 제거합니다.
rmusr	ru	AC 및 기본	*CA Access Control 또는 네이티브 OS

명령	약어	환경	설명
			데이터베이스에서 사용자를 제거합니다.
rmxgrp	rxg	AC	CA Access Control 데이터베이스에서 엔터프라이즈 그룹을 제거합니다.
rmxusr	rxu	AC	CA Access Control 에서 엔터프라이즈 사용자를 제거합니다.
ruler		AC 및 기본	show 명령을 실행할 때 표시되는 속성을 설정합니다.
search		AC 및 기본	find 명령의 별칭입니다.
setoptions	so	AC 및 nt	*데이터베이스 동작을 제어하는 전역 옵션을 설정 또는 표시합니다.
showfile	sf	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에 있는 파일 레코드의 속성을 나열합니다.
showgrp	sg	AC 및 기본	*CA Access Control 또는 네이티브 OS 데이터베이스에 있는 그룹 레코드의 속성을 나열합니다.
showres	sr	AC 및 nt	*CA Access Control 또는 기본 Windows 데이터베이스에 있는 레코드의 속성을 나열합니다.
showres config		config	지정한 원본의 구성 설정을 나열합니다.
showusr	su	AC 및 기본	*CA Access Control 데이터베이스 또는 네이티브 OS 데이터베이스에 있는 사용자 레코드의 속성을 나열합니다.
showxusr	sxu	AC	CA Access Control 에 있는 엔터프라이즈 사용자 레코드의 속성을 나열합니다.
source		all	지정한 파일에서 명령을 실행합니다.
start dbexport		AC	CA Access Control 또는 PMD 데이터베이스를 내보냅니다.
start devcalc		AC	정책 위반 계산을 트리거합니다.
start_transaction		AC	하나 이상의 명령을 사용하여 이중 제어 PMDB 프로세스에 대해 처리되지 않은 트랜잭션을 포함하는 파일의 레코딩을 시작합니다.
subs		pmd	부모 PMDB 에 구독자를 추가하거나 부모 PMDB 에 데이터베이스를 구독시킵니다.

명령	약어	환경	설명
subspmd		pmd	연결된 호스트에 있는 데이터베이스의 부모를 변경합니다.
unalias		AC 및 Unix	selang 명령 및 속성에 대한 별칭을 제거합니다.
undeploy		AC	deploy- 명령의 별칭입니다.
unsubs		pmd	PMDB 의 구독자 목록에서 구독자를 제거합니다.
xaudit		nt	감사 기준을 설정하고 액세스 이벤트 로깅을 시작합니다.
xaudit-		nt	감사 기준을 제거하고 액세스 이벤트 로깅을 중지합니다.

참고: 연결된 호스트의 운영 체제에 따라 네이티브 환경은 Windows(NT) 또는 UNIX 환경의 규칙을 따릅니다.

AC Windows 환경의 selang 명령

이 절에는 CA Access Control 데이터베이스에서 작동하는 모든 selang 명령(AC 환경의 명령)에 대한 참조가 알파벳순으로 포함되어 있습니다.

alias 명령 selang 별칭 정의

UNIX 호스트에 해당

selang 명령 및 속성에 대한 별칭을 나열 또는 정의하려면 alias 명령을 사용하십시오. 모든 사용자가 alias 명령을 사용할 수 있습니다.

참고: 별칭을 시작 파일에 정의하고 selang -r 명령을 사용하여 모든 selang 세션에서 사용할 일련의 별칭을 만들 수 있습니다.

이 명령의 형식은 다음과 같습니다.

```
alias [aliasName [aliasValue]]
```

aliasName

(선택 사항) 별칭으로 사용할 이름을 정의합니다.

이 옵션을 지정하지 않고 alias 명령을 사용하면 정의된 모든 별칭이 나열됩니다.

aliasValue

(선택 사항) selang 명령 셸과 aliasName 이 연결되어야 하는 의미를 정의합니다.

이 옵션을 지정하지 않고 alias 명령을 사용하면 지정된 별칭의 값이 나열됩니다.

또한 aliasValue 에 최대 10 개의 변수($0 ~ $9)를 포함할 수 있습니다. aliasValue 에 변수가 포함되어 있으면 별칭을 호출할 때 각 변수를 괄호로 묶은 적절한 값으로 바꿔야 합니다.

예: 변수를 사용하여 쉽게 새 관리자 작성

새 관리자를 데이터베이스에 더 쉽게 추가하는 별칭을 작성하려면 다음 명령을 입력합니다.

```
alias newadm newusr ($0) admin
```

이 별칭을 사용하려면 새 관리자의 이름을 괄호에 추가하기만 하면 됩니다. 예:

```
newadm(Terri)
```

이렇게 하면 Terri 라는 사용자가 데이터베이스에 추가됩니다. Terri 에게는 데이터베이스 관리에 필요한 ADMIN 특성이 부여됩니다. 이것은 다음 명령을 입력하는 것과 동일합니다.

```
newusr Terri admin
```

예: 속성 이름 단순화

속성 이름 access 를 줄임형 별칭 acc 로 바꾸는 별칭을 작성하려면 다음 명령을 입력합니다.

```
alias acc access
```

이제 이 별칭을 사용하려면 다음을 입력할 수 있습니다.

```
authorize file x uid(y) acc(z)
```

예: 상황에 따른 별칭 사용

별칭은 단순히 확장된 변수가 아닙니다. 별칭은 명령 이름 또는 속성 이름을 지정해야 하는 상황에서만 해석됩니다. 예를 들어 다음 별칭을 정의합니다.

```
alias newterm newres terminal
```

그 후 다음 명령을 실행합니다.

```
newterm newterm owner(nobody)
```

이 상황에서 문자열의 두 번째 인스턴스는 터미널 이름이 되어야 하므로, 첫 번째 newterm 문자열은 바뀌지만 두 번째는 바뀌지 않습니다. 이것은 다음 명령을 입력하는 것과 동일합니다.

```
newres terminal newterm owner(nobody)
```

추가 정보:

unalias 명령-selang 별칭 제거(페이지 160)
selang 유틸리티-CA Access Control 명령줄 실행(페이지 18)

권한 부여 명령-리소스에 대한 Access 권한 설정

AC 환경에 해당

접근자의 access authorities 를 리소스로 변경하려면 권한 부여 명령을 사용하십시오.

이 명령은 리소스와 연결된 액세스 제어 목록을 수정합니다. 이 명령은 액세스 제어 목록의 항목을 한 번에 하나만 변경합니다.

접근자가 리소스에 액세스하려고 시도하면 CA Access Control 은 해당 액세스 제어 목록을 검사하여 액세스 권한을 확인합니다. 이러한 액세스 제어 목록은 리소스 레코드에 있는 목록을 포함하며, 리소스 그룹 레코드에 있는 액세스 제어 목록도 포함할 수 있습니다. 리소스가 속한 NACL 에서 접근자의 액세스 권한이 거부되면, 다른 ACL 에서 권한이 승인되더라도 해당 접근자의 권한은 거부됩니다.

리소스 소유자는 항상 리소스에 대해 모든 액세스 권한을 갖습니다. 소유자인 사용자의 액세스 권한을 변경하려면 새로운 소유자(예: 사용자 nobody)를 갖도록 리소스를 변경하십시오.

참고: 이 명령은 Windows 환경에도 있지만 작동 방식이 다릅니다.

권한 부여 명령을 사용하려면 충분한 권한이 있어야 합니다. 즉, 다음 중 하나 이상이 충족되어야 합니다.

- 사용자가 ADMIN 특성을 가집니다.
- 리소스가 구성원으로 속한 리소스 그룹에 대해 GROUP-ADMIN 특성을 가집니다.
- 사용자가 리소스의 소유자입니다.
- 리소스에 해당하는 ADMIN 클래스 레코드에서 사용자가 수정 액세스 권한을 가집니다.

권한 부여 명령의 형식은 클래스 집합에 따라 다릅니다. 이러한 집합은 다음과 같습니다.

- TCP
- HOST, GHOST, HOSTNET, HOSTNP
- 모든 다른 클래스

TCP 클래스에 대한 이 명령의 형식은 다음과 같습니다.

```
{authorize|auth} TCP tcpServiceName \
    [{access|deniedaccess}(accessType)] \
    [ghost(ghostName [,ghostName]...)] \
    [host(hostName [,hostName]...)] \
    [hostnet(hostNetName [,hostNetName]...)] \
    [hostnp(hostNamePattern [,hostNamePattern]...)] \
    {gid|uid|xgid|xuid}(accessor [,accessor]...]) ...
```

HOST, GHOST, HOSTNET 및 HOSTNP 클래스에 대한 이 명령의 형식은 다음과 같습니다.

```
{authorize|auth} {HOST|GHOST|HOSTNET|HOSTNP} stationName
    [{access|deniedaccess}(accessType)] \
    service({serviceName|serviceNumber|serviceNumberRange}) \
    { gid | uid | xgid | xuid}(accessor [,accessor...]) ...
```

다른 모든 클래스에 대한 이 명령의 형식은 다음과 같습니다.

```
{authorize|auth} className resourceName \
    [{access|deniedaccess}(accessType)] \
    [calendar(calendarName)] \
    [{unix|nt}]\
    [via (pgm ( program [,program]...))] \
    { gid | uid | xgid | xuid}(accessor [,accessor...]) ...
```

access (accessType)

리소스 ACL 액세스 제어 목록에서 액세스 권한 항목을 정의합니다. 이 ACL 은 접근자에게 허용할 액세스 권한을 지정합니다.

accessType

리소스 ACL 에서 액세스 유형(예: 읽기 또는 쓰기)을 정의합니다.

참고: 권한 부여 명령에서 액세스(accessType) 옵션과 거부된 액세스(accessType) 옵션을 모두 생략하면 CA Access Control 은 리소스 클래스의 UACC 클래스에서(예: 리소스가 파일인 경우 UACC 파일에서) 레코드의 암시적 액세스 속성에 의해 지정된 액세스를 할당합니다.

calendar(calendarName)

액세스 권한 결정에 사용할 달력을 지정합니다.

className

resourceName 이 속해 있는 클래스를 정의합니다.

deniedaccess(accessType)

NACL 리소스에서 액세스 권한을 변경합니다. NACL 은 접근자에 대해 거부되는 액세스 유형을 지정합니다.

accessType

거부할 액세스 유형(예: 읽기 또는 쓰기)을 지정합니다.

gid (accessor [,accessor...])

액세스 권한을 설정할 하나 이상의 내부 그룹을 정의합니다.

ghost(ghostName [,ghostName]...)

TCP/IP 서비스에 대한 액세스 권한을 설정할 하나 이상의 그룹 호스트를 정의합니다.

host(hostName [,hostName]...)

TCP/IP 서비스에 대한 액세스 권한을 설정할 하나 이상의 호스트를 정의합니다.

hostnet(hostNetName [,hostNetName]...)

TCP/IP 서비스에 대한 액세스 권한을 설정할 하나 이상의 HOSTNET 레코드를 정의합니다.

hostnp(hostNamePattern [,hostNamePattern]...)

TCP/IP 서비스에 대한 액세스 권한을 설정할 하나 이상의 HOSTNP 레코드를 정의합니다.

nt

Windows 에서 시스템 ACL 에 값을 추가할지 여부를 지정합니다.

FILE 클래스에만 해당.

resourceName

액세스 제어 목록을 수정하고 있는 리소스 레코드를 정의합니다.

service(serviceName|serviceNumber|serviceNumberRange)

로컬 호스트가 원격 호스트에 제공하도록 허용되는 서비스를 정의합니다.

serviceNumber |serviceNumberRange

서비스 번호 또는 범위를 정의합니다.

범위는 -(하이픈)으로 구분된 두 개의 정수(예: 1-99)로 지정하십시오.

제한: 정수의 범위는 0 ~ 65535 입니다.

stationName

표시된 클래스 내의 레코드 이름을 다음과 같이 지정합니다.

- **HOST-**단일 스테이션의 이름
- **GHOST-**ghost 명령으로 데이터베이스에 정의된 호스트의 그룹 이름
- **HOSTNET-**IP 주소의 mask 값과 match 값의 집합으로 정의된 호스트 그룹의 이름
- **HOSTNP-**이름 패턴으로 정의된 호스트 그룹의 이름

확인할 수 없는 호스트의 경우 IPv4 형식으로 IP 주소 범위를 지정하십시오.

tcpServiceName

액세스 권한을 설정 중인 CA Access Control TCP 서비스 레코드를 지정합니다.

uid (accessor [,accessor...])

액세스 권한을 설정할 하나 이상의 내부 사용자를 정의합니다.

모든 내부 사용자를 나타내려면 *를 사용할 수 있습니다.

unix

UNIX 에서 시스템 ACL 에 값을 추가할지 여부를 지정합니다.

ACL 을 지원하는 UNIX 환경 및 FILE 클래스의 레코드에만 해당됩니다.

via(pgm(programName [,programName]...))

조건부 프로그램 액세스를 위한 하나 이상의 프로그램을 정의합니다. via 매개 변수는 리소스의 PACL 에 항목을 지정합니다. programName 은 리소스에 액세스할 수 있는 프로그램을 지정합니다. programName 에는 와일드카드 문자를 포함할 수 있습니다. 하나의 프로그램이 PACL 의 여러 항목과 일치하면, 와일드카드 이외의 일치 중 가장 긴 일치가 있는 항목이 우선권을 갖습니다.

programName 이 PROGRAM 클래스에 정의되지 않은 프로그램이나 셸 스크립트를 지정하는 경우, CA Access Control 은 자동으로 PROGRAM 레코드를 만들어서 프로그램이나 셸 스크립트를 보호합니다.

xgid (accessor [,accessor...])

액세스 권한을 설정할 하나 이상의 엔터프라이즈 그룹을 정의합니다.

xuid (accessor [,accessor...])

액세스 권한을 설정할 하나 이상의 엔터프라이즈 사용자를 정의합니다.

예: Angela 에게 파일 읽기 권한 부여

다음의 selang 명령은 엔터프라이즈 사용자 Angela 에게 FILE 리소스 /projects/secrets 로 보호된 파일을 읽을 수 있는 권한을 부여합니다.

```
auth FILE /projects/secrets xuid(Angela) access(read)
```

예: Angela 에게만 파일 읽기 권한 부여

다음의 selang 명령은 다른 사용자가 아닌 오직 엔터프라이즈 사용자 Angela 에게만 FILE 리소스 /projects/secrets 로 보호된 파일을 읽을 수 있는 권한을 부여합니다.

```
auth FILE /projects/secrets xuid(Angela) access(read)
auth FILE /projects/secrets defaccess (none)
chres FILE /projects/secrets owner(nobody)
```

참고: UNIX 에서 사용자가 파일에 대한 정보를 얻는 작업(예: ls -l)을 수행할 수 있는지 여부를 제어하기 위해 read 권한이 필요한 경우, STAT_intercept 구성 설정을 1 로 설정하십시오. 자세한 내용은 Reference Guide(참조 안내서)를 참조하십시오.

예: 그룹의 모든 사용자에게 터미널이 로그인할 권한 부여

다음의 selang 명령은 엔터프라이즈 그룹 RESEARCH 의 모든 구성원에게 TERMINAL 리소스 tty10 으로 보호되는 터미널에 로그인할 권한을 부여합니다.

```
auth TERMINAL tty10 xgid(RESEARCH) access(read)
```

예: Joe 에게 파일 백업 권한 부여

다음의 selang 명령은 엔터프라이즈 사용자 Joe 에게 GFILE 리소스 secret_files 로 보호되는 파일의 백업 권한을 부여합니다.

```
auth GFILE secret_files xuid(Joe) \
via(pgm(/bin/backup)) access(read)
```

Windows 끝점에서 사용할 수 있는 이와 동일한 명령은 다음과 같습니다.

```
auth GFILE secret_files xuid(Joe) \
via(pgm(C:\WINDOWS\system32\ntbackup.exe)) access(read)
```

Joe 의 액세스 권한이 리소스의 ACL 또는 NACL 에 의해 결정되지 않은 경우에만 이러한 명령이 효과가 있습니다.

추가 정보:

authorize- 명령-리소스에서 액세스 권한 제거(페이지 51)
권한 부여 명령-Windows 리소스에 액세스하기 위한 접근자의 권한 설정(페이지 181)
chfile 명령-파일 레코드 수정(페이지 58)
ch[x]grp 명령-그룹 속성 변경(페이지 64)
chres 명령-리소스 레코드 수정(페이지 77)
ch[x]usr 명령-사용자 속성 변경(페이지 94)
authorize- 명령-Windows 리소스에 액세스하는 접근자의 권한 제거(페이지 183)

authorize- 명령-리소스에서 Access 권한 제거

AC 환경에 해당

리소스의 액세스 제어 목록(ACL)에서 접근자를 제거하려면 authorize- 명령을 사용하십시오.

참고: 이 명령은 네이티브 Windows 환경에도 있지만 작동 방식이 다릅니다.

authorize- 명령을 사용하려면 권한 부여 명령을 사용할 때와 동일한 액세스 권한이 필요합니다.

authorize- 명령의 형식은 클래스 집합에 따라 다릅니다. 이러한 집합은 다음과 같습니다.

- TCP
- HOST, GHOST, HOSTNET, HOSTNP
- 모든 다른 클래스

TCP 클래스에 대한 이 명령의 형식은 다음과 같습니다.

```
{authorize-|auth-} TCP tcpServiceName \
    {gid |uid |xgid |xuid } (accessorName [,accessorName]...)\
    [host(hostName [,hostName]...)] \
    [ghost(ghostName [,ghostname]...)] \
    [hostnet(hostNetName [,hostNetName]...)] \
    [hostnp(hostNamePattern [,hostNamePattern]...)]
```

HOST, GHOST, HOSTNET 및 HOSTNP 클래스에 대한 이 명령의 형식은 다음과 같습니다.

```
{authorize-|auth-} className stationName \
    service({serviceName | serviceNumber |serviceNumberRange})
```

모든 나머지 클래스에 대한 이 명령의 형식은 다음과 같습니다.

```
{authorize-|auth-} className resourceName \
    [{access-|deniedaccess-}]\
    [calendar(calendarName)] \
    {gid |uid |xgid |xuid } (accessorName [,accessorName]...)
```

access-

이 명령을 사용하여, 액세스 권한을 허용한 리소스 ACL(NACL 이 아님)에서 접근자를 제거해야 함을 지정합니다.

access-나 deniedaccess- 중 어느 것도 지정하지 않으면 이 명령은 두 ACL 에서 모두 접근자를 제거합니다.

calendar(calendarName)

액세스 권한 결정을 위해 지정한 달력을 제거합니다.

className

resourceName 이 속한 클래스의 이름을 지정합니다.

deniedaccess-

이 명령을 사용하여, 액세스 권한을 거부한 리소스 NACL(ACL 이 아님)에서 접근자를 제거해야 함을 지정합니다.

gid (accessor [,accessor]...)

항목을 제거해야 할 하나 이상의 내부 그룹을 정의합니다. 각 accessor 를 쉼표 또는 공백으로 구분하십시오.

ghost(ghostName)

GHOST 클래스에 있는 객체 이름을 지정합니다.

host(hostName)

HOST 클래스에 있는 객체 이름을 지정합니다.

hostnet(hostNetName)

HOSTNET 클래스에 있는 객체 이름을 지정합니다.

hostnp(hostNamePattern)

HOSTNP 클래스에 정의된 패턴을 지정합니다.

nt

Windows 에서 시스템 ACL 로부터 값을 제거할지 여부를 지정합니다.

FILE 클래스에만 해당.

resourceName

액세스 제어 목록을 수정할 리소스 레코드의 이름을 지정합니다. 리소스 레코드를 한 개만 지정합니다.

service(serviceName|serviceNumber|serviceNumberRange)

ACL 에서 제거할 서비스를 정의합니다.

stationName

표시된 클래스 내의 레코드 이름을 다음과 같이 지정합니다.

- **HOST-**단일 스테이션의 이름
- **GHOST-**ghost 명령으로 데이터베이스에 정의된 호스트의 그룹 이름
- **HOSTNET-**IP 주소의 mask 값과 match 값의 집합으로 정의된 호스트 그룹의 이름
- **HOSTNP-**이름 패턴으로 정의된 호스트 그룹의 이름

확인할 수 없는 호스트의 경우 IP 주소 범위를 지정하십시오.

serviceNumber |serviceNumberRange

서비스 번호 또는 범위를 정의합니다.

범위는 -(하이픈)으로 구분된 두 개의 정수(예: 1-99)로 지정하십시오.

제한: 정수의 범위는 0 ~ 65535 입니다.

uid (accessor [,accessor]...)

항목을 제거해야 할 하나 이상의 내부 사용자를 정의합니다. 각 accessor 를 쉼표 또는 공백으로 구분하십시오.

모든 내부 사용자를 지정하려면 uid(*)를 사용할 수 있습니다.

unix

UNIX 에서 시스템 ACL 로부터 값을 제거/추가할지 여부를 지정합니다.

ACL 을 지원하는 UNIX 환경 및 FILE 클래스의 레코드에만 해당됩니다.

xgid (accessor [,accessor]...)

항목을 제거해야 할 하나 이상의 엔터프라이즈 사용자를 정의합니다. 각 accessorName 을 쉼표 또는 공백으로 구분하십시오.

xuid (accessor [,accessor]...)

항목을 제거해야 할 하나 이상의 엔터프라이즈 그룹을 정의합니다. 각 accessor 를 쉼표 또는 공백으로 구분하십시오.

예: 파일에 Access 할 수 있는 그룹 권한 제거

다음 명령을 사용하면 리소스 /products/new 에 속한 파일의 ACL 및 NACL 모두에서 그룹 검색 권한이 제거됩니다.

```
auth- FILE /products/new xgid(research)
```

이제 검색 그룹은 파일에 대해 기본 액세스 권한만 갖게 됩니다.

추가 정보:

권한 부여 명령-리소스에 대한 액세스 권한 설정(페이지 46)
권한 부여 명령-Windows 리소스에 액세스하기 위한 접근자의 권한 설정(페이지 181)
authorize- 명령-Windows 리소스에 액세스하는 접근자의 권한 제거(페이지 183)
chfile 명령-파일 레코드 수정(페이지 58)
ch[x]grp 명령-그룹 속성 변경(페이지 64)
chres 명령-리소스 레코드 수정(페이지 77)
ch[x]usr 명령-사용자 속성 변경(페이지 94)

check 명령-사용자의 Access 권한 확인

AC 환경에 해당

사용자가 특성 리소스에 대해 액세스 권한을 가지는지 확인하려면 check 명령을 사용하십시오. 이 명령은 리소스의 ACL 과 기본 액세스 권한 속성에 따라 액세스 권한을 검사합니다. 하지만 이 명령은 PACL 을 지원하지 않습니다. 즉, 사용자가 특정 프로그램을 사용하여 리소스에 액세스할 수 있는지 여부를 알려주지 않습니다.

참고: seos 가 종료되면 이 명령을 사용할 수 없습니다. PACL 에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

이 명령을 사용하려면 리소스에 대한 충분한 권한을 가지고 있어야 합니다. 이러한 권한은 다음 조건 중 하나로 정의됩니다.

- 명령을 실행하는 프로세스가 SERVER 특성을 가집니다.
- 사용자가 ADMIN 특성을 가집니다.

이 명령의 형식은 다음과 같습니다.

```
check className resourceName uid(userName) access(authority)
```

access(authority)

uid 매개 변수로 식별된 접근자에 대해 검사할 액세스 권한을 정의합니다.

유효한 값은 검사하는 리소스에 따라 다릅니다.

className

resourceName 이 속한 클래스의 이름을 정의합니다.

resourceName

리소스 레코드의 이름을 정의합니다.

uid(userName)

resourceName 에 액세스하는 권한을 검증받아야 하는 CA Access Control 사용자의 이름을 정의합니다.

예: 사용자가 리소스에 대해 Access 권한을 가지고 있는지 확인

사용자 Alain 이 file 클래스의 testfile 리소스에 대해 쓰기 액세스 권한을 가지고 있는지 확인하려면 다음 명령을 입력합니다.

```
check FILE /testfile uid(Alain) access(w)
```

이 명령의 다음 샘플 출력은 사용자 Alain 이 정의된 파일에 대해 쓰기 액세스 권한을 가지고 있음을 나타냅니다. Alain 이 리소스의 소유자이기 때문입니다.

```
FILE /testfile 액세스 허가됨
단계: 리소스 OWNER 검사
```

checklogin 명령-로그인 정보 확인

AC 환경에 해당

사용자의 로그인 권한, 암호 검사 필요 여부 및 터미널 액세스 권한 검사 필요 여부를 확인하려면 checklogin 명령을 사용하십시오.

참고: seos 가 종료되면 이 명령을 사용할 수 없습니다.

이 명령을 사용하려면 리소스에 대한 충분한 권한을 가지고 있어야 합니다. 이러한 권한은 다음 조건 중 하나로 정의됩니다.

- 명령을 실행하는 프로세스가 SERVER 특성을 가집니다.
- 사용자가 ADMIN 특성을 가집니다.

이 명령의 형식은 다음과 같습니다.

```
checklogin userName [password(password)] [terminal(terminalName)]
```

password(password)

(선택 사항) 암호 검사가 활성화된 경우 CA Access Control 이 운영 체제 암호 및 데이터베이스에 대해 검사할 암호를 정의합니다.

userName

로그인 권한을 확인할 사용자의 이름을 정의합니다.

terminal(terminalName)

(선택 사항) 사용자가 로그인 권한을 가지고 있는지를 결정하기 위해 CA Access Control 이 확인할 터미널을 정의합니다.

예: 사용자가 로그인 권한을 가지고 있는지 확인

사용자 Frank 가 터미널 mutra 에서 localhost 에 대한 로그인 권한을 가지고 있는지 확인하려면 다음 명령을 입력합니다.

```
checklogin Frank terminal(mutra)
```

다음의 명령 출력은 사용자 Frank 가 터미널 mutra 에서 호스트 winsome(localhost)에 로그인할 수 있음을 나타냅니다.

```
frank 사용자의 winsome 호스트 로그인 허가됨
단계: 리소스 클래스 전역 범용 액세스
```

사용자 Frank 의 암호를 확인하려면 다음 명령을 입력합니다.

```
checklogin frank password(111) terminal(localhost)
```

CA Access Control 데이터베이스의 암호와 비교하여 사용자 Frank 의 암호를 확인하려면 다음 명령을 입력합니다.

```
so class+(PASSWORD) (localhost)
checklogin frank password(moonshine) terminal(tack)
```

위의 so 명령은 암호 검사를 활성화합니다.

checkpwd 명령-암호의 규칙 준수 여부 검사

AC 환경에 해당

사용자의 암호가 암호 규칙을 준수하는지 검사하려면 checkpwd 명령을 사용하십시오. 이 검사로 인해 암호가 변경되지는 않습니다.

이 명령을 사용하려면 ADMIN 특성을 가진 슈퍼 사용자여야 합니다.

새 암호는 CA Access Control 암호 규칙에 따라 승인되거나 거부됩니다.

- 새 암호가 승인되면 다음 성공 메시지가 표시됩니다.

  userName 의 암호를 변경할 수 있습니다.

- 새 암호가 거부되면 다음 실패 메시지가 표시됩니다.

  userName 의 암호를 변경할 수 없습니다.
  denied_reason

  여기서 denied_reason 은 통과하지 못한 실제 암호 규칙입니다.

  예:

  JDoe 의 암호를 변경할 수 없습니다.
  암호에 소문자가 너무 적습니다.

암호가 통과하지 못한 첫번째 규칙만 denied_reason 에 나타납니다. 예를 들어, 암호가 너무 짧고 암호에 대문자가 거의 없는 경우 암호가 너무 짧습니다.라는 메시지만 나타납니다.

참고: seos 가 종료되면 이 명령을 사용할 수 없습니다. 암호 규칙에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

이 명령의 형식은 다음과 같습니다.

```
checkpwd userName password(newPassword)
```

userName

새 암호를 검사할 CA Access Control 사용자의 이름을 지정합니다.

password(newPassword)

검사할 암호를 지정합니다.

chfile 명령-파일 레코드 수정

AC 환경에 해당

FILE 클래스의 레코드로 작업하려면 chfile, editfile 및 newfile 명령을 사용하십시오. 이러한 명령은 구조가 동일하며, 다음의 내용만 다릅니다.

- chfile 명령은 FILE 클래스에 있는 하나 이상의 레코드를 수정합니다.
- editfile 명령은 FILE 클래스에 있는 하나 이상의 레코드를 작성 또는 수정합니다.
- newfile 명령은 FILE 클래스에 있는 하나 이상의 레코드를 작성합니다.

참고: 이 명령은 네이티브 환경에도 있지만 작동 방식이 다릅니다.

FILE 클래스에 속한 파일에 대해 레코드를 추가 또는 변경하려면 파일에 대해 충분한 권한을 가지고 있어야 합니다. 다음의 조건 중 하나가 충족될 때까지 CA Access Control 은 다음 검사를 수행합니다.

1. 사용자가 ADMIN 특성을 가집니다.
2. 리소스 레코드는 사용자가 GROUP-ADMIN 속성을 가지는 그룹의 범위 내에 있습니다.
3. 레코드를 변경할 때 해당 레코드의 소유자입니다.
4. ADMIN 클래스에 있는 FILE 레코드의 ACL 에 CREATE(newfile 또는 editfile 의 경우) 또는 MODIFY(chfile 의 경우) 액세스 권한을 가집니다.
5. seos.ini 파일의 토큰 use_unix_file_owner 가 yes 로 설정되어 있으면 해당 사용자는 파일의 소유자입니다(네이티브 OS 에 존재하는 CA Access Control 에 대해 파일을 정의할 때).

```
{{chfile|cf}|{editfile|ef}|{newfile|nf}} filename... \
    [audit{none|all|success|failure}] \
    [category[-](categoryName)] \
    [comment(string)|comment-] \
    [defaccess(accessAuthority)] \
    [label(labelName)|label-] \
    [level(number)|level-] \
    [notify(mailAddress)|notify-] \
    [gowner(groupName)] \
    [owner({userName|groupName})] \
    [restrictions( \
        [days({anyday|weekdays|{[mon] [tue] [wed] \
                [thu] [fri] [sat] [sun]}})] \
        [time({anytime|startTime:endTime}) \
    |restrictions-] \
    [warning|warning-]
```

audit{none|all|success|failure}

로그에 기록되는 액세스 이벤트를 지정합니다. 액세스 유형은 다음과 같습니다.

- **all**-CA Access Control 은 권한 있는 액세스 및 검색된 권한 없는 액세스 시도를 모두 기록합니다.
- **failure**-CA Access Control 은 검색된 권한 없는 액세스 시도를 기록합니다. 이것이 기본값입니다.
- **none**-CA Access Control 은 레코드를 로그 파일에 기록하지 않습니다.
- **success**-CA Access Control 은 리소스에 대한 권한 있는 액세스를 기록합니다.

참고: audit 매개 변수를 사용하려면 AUDITOR 특성을 가지고 있어야 합니다.

category(categoryName)

파일에 할당할 보안 범주 레코드(CATEGORY 클래스에 정의됨)의 목록을 공백 또는 쉼표로 구분하여 정의합니다.

CATEGORY 클래스가 활성화되지 않았을 때 category 매개 변수를 지정하면 CA Access Control 은 데이터베이스의 파일 정의를 업데이트 합니다. 그러나 업데이트된 범주 할당은 CATEGORY 클래스가 다시 활성화되기 전까지 적용되지 않습니다.

참고: 보안 범주 검사에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

category-(categoryName)

리소스 레코드에서 하나 이상의 보안 범주를 삭제합니다. 둘 이상의 보안 범주를 제거하려면 보안 범주 이름을 공백 또는 쉼표로 구분하십시오.

지정된 보안 범주가 CATEGORY 클래스의 활성 여부에 상관 없이 리소스 레코드에서 삭제됩니다.

참고: 이 매개 변수는 레코드를 수정할 경우에만 유효합니다.

comment(string)

영숫자 문자열을 최대 255 자까지 파일 레코드에 추가합니다. 문자열에 공백이 포함되어 있으면 작은따옴표로 문자열을 둘러쌉니다. 이 문자열은 이전에 정의한 기존의 모든 주석을 대체합니다.

comment-

파일 레코드에서 주석 문자열을 삭제합니다.

참고: 이 매개 변수는 레코드를 수정할 경우에만 유효합니다.

defaccess(accessAuthority)

파일의 기본 액세스 권한을 지정합니다. 기본 액세스 권한은 파일에 대한 액세스를 요청하는 모든 접근자에 부여되지만 파일의 액세스 제어 목록에는 존재하지 않는 권한입니다. 기본 액세스는 데이터베이스에 정의되지 않은 사용자에게도 적용됩니다.

fileName

파일 레코드의 이름을 정의합니다. 파일 이름을 최소한 하나 이상 지정해야 합니다.

일반 파일 이름을 사용하여 FILE 클래스에서 레코드를 추가 또는 변경하는 경우 selang 에서 허용되는 와일드카드 표현식을 사용하십시오. 하나 이상의 레코드를 정의하거나 변경할 때 파일 이름 목록을 괄호 안에 넣고 파일 이름은 공백이나 쉼표로 구분하십시오.

참고: 둘 이상의 파일 이름을 지정할 경우 CA Access Control 은 지정된 매개 변수에 따라 각 파일 레코드를 독립적으로 처리합니다. 파일을 처리할 때 오류가 발생하면 CA Access Control 은 메시지를 작성하고 목록의 다음 파일을 계속 처리합니다.

gowner(groupName)

CA Access Control 그룹을 파일 레코드의 그룹 소유자로 할당합니다. 소유자의 보안 수준, 보안 레이블 및 보안 범주 권한이 파일에 액세스하기에 충분하다면 파일 레코드의 소유자는 무제한의 파일 액세스 권한을 가집니다. 파일의 그룹 소유자는 항상 파일 레코드를 업데이트하고 삭제할 수 있습니다.

label(labelName)

보안 레이블(SECLABEL) 클래스에 정의된 보안 레이블을 파일에 할당합니다. 특정 보안 수준과 0 개 이상의 보안 범주 사이의 관계를 나타내는 보안 레이블입니다. 리소스 레코드에 현재 보안 레이블이 있는 경우, 여기에 지정된 보안 레이블은 현재 보안 레이블을 대체합니다.

참고: 보안 레이블 검사에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

label-

파일 레코드에 정의된 보안 레이블을 삭제합니다.

참고: 이 매개 변수는 레코드를 수정할 경우에만 유효합니다.

level(number)

리소스 레코드에 보안 수준을 할당합니다. 1-255 사이의 양의 정수를 입력합니다. 이전에 보안 수준이 리소스 레코드에 할당된 경우 새 값은 기존 값을 대체합니다.

참고: 보안 수준 검사에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

level-

CA Access Control 이 리소스에 대해 보안 수준을 검사하지 않도록 합니다.

참고: 이 매개 변수는 레코드를 수정할 경우에만 유효합니다.

notify(mailAddress)

리소스 레코드로 표시되는 파일에 성공적으로 액세스할 때마다 CA Access Control 에게 알림 메시지를 보내도록 지시합니다. 사용자 이름이나 전자 메일 주소를 입력하거나 별칭이 지정된 경우 메일 그룹의 전자 메일 주소를 입력하십시오.

통지는 로그 라우팅 시스템이 활성 상태인 경우에만 발생합니다. 통지 메시지는 로그 라우팅 시스템의 설정에 따라 화면 또는 사용자의 사서함으로 전송됩니다.

통지 메시지가 전송될 때마다 감사 레코드가 감사 로그에 기록됩니다.

통지 메시지의 수신인은 자주 로그인하여 각 메시지에서 설명하고 있는 무단 액세스 시도에 응답해야 합니다.

제한: 30 자.

참고: 감사 레코드의 필터링 및 보기에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

notify-

CA Access Control 이 레코드로 표시되는 파일에 대한 액세스를 허용할 때 아무에게도 알리지 않도록 지정합니다.

참고: 이 매개 변수는 레코드를 수정할 경우에만 유효합니다.

owner(Name)

CA Access Control 사용자 또는 그룹을 파일 레코드의 소유자로 할당합니다. 소유자의 보안 수준, 보안 레이블 및 보안 범주 권한이 파일에 액세스하기에 충분하다면 파일 레코드의 소유자는 무제한의 파일 액세스 권한을 가집니다. 파일의 소유자는 항상 파일 레코드를 업데이트하고 삭제할 수 있습니다.

restrictions(days(dayData) time(timeData))

사용자가 파일에 액세스할 수 있는 요일 및 하루 중의 시간을 지정합니다.

days 인수를 생략하고 time 인수를 지정할 경우, 시간 제한은 레코드에 이미 표시된 요일 제한에 적용됩니다. time 을 생략하고 days 를 지정할 경우 요일 제한은 레코드에 이미 표시된 시간 제한에 적용됩니다. days 와 time 을 모두 지정할 경우 사용자는 지정된 요일의 지정된 시간 동안만 시스템에 액세스할 수 있습니다.

days(**dayData**)

사용자가 파일에 액세스할 수 있는 요일을 지정합니다. days 인수는 다음 하위 인수를 가집니다.

- **anyday**-모든 요일에 파일 액세스 권한을 제공합니다.
- **weekdays**-주중(월요일부터 금요일까지)에만 리소스 액세스 권한을 제공합니다.
- **mon tue wed thu fri sat sun**-지정된 요일에만 리소스 액세스 권한을 제공합니다. 순서에 상관 없이 요일을 지정할 수 있습니다. 둘 이상의 요일을 지정하려면 공백 또는 쉼표로 날짜를 구분하십시오.

time(timeData)

사용자가 파일에 액세스할 수 있는 기간을 지정합니다. time 인수는 다음 하위 인수를 가집니다.

- **anytime**-시간에 상관없이 언제든지 리소스 액세스 권한을 제공합니다.
- startTime**:**endTime-사용자가 지정된 기간 동안에만 리소스에 액세스할 수 있습니다. startTime 및 endTime 의 형식은 hhmm 입니다. startTime 및 endTime 형식은 모두 hhmm 이며, 여기서 hh 는 24 시간 표기(00 - 23) 시간이고 mm 은 분(00 - 59)입니다. 2400 은 올바른 시간 값이 아닙니다. startTime 은 endTime 보다 작아야 하며, 두 시간 모두 같은 날에 속해야 합니다. 터미널의 시간대가 프로세서와 다른 경우, 터미널의 시작 시간과 종료 시간을 프로세서에 대해 동등한 로컬 시간으로 변환하여 시간 값을 조정합니다. 예를 들어, 프로세서가 뉴욕에 있고 터미널이 LA 에 있는 경우 LA 에서 오전 8 시에서 오후 5 시까지 터미널에 대한 액세스를 허용하려면 시간(1100:2000)을 지정하십시오.

restrictions-

파일 액세스 권한을 제한하는 모든 제한 규칙을 삭제합니다.

참고: 이 매개 변수는 레코드를 수정할 경우에만 유효합니다.

warning

파일을 경고 모드에 둡니다.

warning-

파일을 경고 모드에서 해제합니다.

예: 슈퍼 사용자를 제외한 모든 사용자에 대해 파일에 대한 Access 제한

슈퍼 사용자를 제외한 모든 사용자에 대해 /etc/passwd 파일에 대한 READ 액세스를 제한하려면 다음 명령을 입력합니다.

```
chfile /etc/passwd defaccess(read) owner(root)
```

다음 조건이 참이어야 합니다.

- 사용자가 ADMIN 특성을 가집니다.
- /etc/passwd 레코드가 데이터베이스에 정의되어 있습니다.
- /etc/passwd 레코드의 ACL 에 항목이 없습니다.

예: 시간별로 파일에 대한 Access 제한

/home/bob/secrets 파일에 대한 액세스를 차단하고 주중에 08:00 및 18:00 사이에만 소유자가 파일에 액세스하는 것을 허용하려면 다음 명령을 입력합니다.

```
newfile /home/bob/secrets defac(none) restrictions(d(weekdays) t(0800:1800))
```

다음 조건이 참이어야 합니다.

- 사용자가 ADMIN 특성을 가집니다.
- Bob 이 CA Access Control 사용자이며 FILE 클래스에 있는 /home/bob/secrets 레코드의 소유자입니다.

예: 홈 디렉터리에 대한 Access 방지

다른 모든 사용자가 자신의 홈 디렉터리(/home/bob)에 있는 어떤 파일에도 액세스하지 못하게 하려면 UNIX 에서 다음 명령을 입력합니다.

```
newfile /home/bob/* defaccess(none)
```

Windows 에서도 다음 명령을 사용하여 이와 동일하게 할 수 있습니다.

```
newfile %userprofile%\* defaccess(none)
```

다음 조건이 참이어야 합니다.

- 사용자가 CA Access Control 에 정의되어 있습니다.
- 사용자가 파일의 기본 소유자입니다.

추가 정보:

권한 부여 명령-리소스에 대한 액세스 권한 설정(페이지 46)
rmfile 명령-파일 레코드 삭제(페이지 131)
showfile 명령-파일 속성 표시(페이지 146)
chfile 명령-UNIX 파일 설정 수정(페이지 167)
chfile 명령-Windows 파일 설정 수정(페이지 184)
클래스별 액세스 권한(페이지 29)

ch[x]grp 명령-그룹 속성 변경

AC 환경에 해당

그룹의 속성을 변경하고 필요할 경우 CA Access Control 데이터베이스에서 그룹을 작성하려면 chgrp, chxgrp, editgrp, editxgrp, newgrp 및 newxgrp 명령을 사용하십시오.

이러한 명령은 모두 다음과 같은 동의어를 갖습니다.

- chgrp-cg
- chxgrp-cxg
- editgrp-eg
- editxgrp-exg
- newgrp-ng
- newxgrp-nxg

이러한 명령은 구조가 동일하며, 다음과 같이 범위만 다릅니다.

- chgrp, editgrp 및 newgrp 명령은 GROUP 클래스의 레코드와 작동합니다. 이러한 명령을 사용하면 엔터프라이즈 사용자 저장소에 대한 참조 없이 CA Access Control 그룹을 작성 또는 수정할 수 있습니다. 이러한 명령의 차이점은 다음과 같습니다.
  - chgrp 명령은 GROUP 클래스에 있는 하나 이상의 레코드를 수정합니다.
  - editgrp 명령은 GROUP 클래스에 있는 하나 이상의 레코드를 작성 또는 수정합니다.
  - newgrp 명령은 GROUP 클래스에 있는 하나 이상의 레코드를 작성합니다.

  참고: 이러한 명령은 네이티브 환경에도 있지만 작동 방식이 다릅니다.
- chxgrp, editxgrp 및 newxgrp 명령은 XGROUP 클래스의 레코드와 작동합니다. 이러한 명령을 사용하면 엔터프라이즈 사용자 저장소에 정의된 CA Access Control 그룹을 작성 또는 수정할 수 있습니다. 이러한 명령의 차이점은 다음과 같습니다.
  - chxgrp 명령은 XGROUP 클래스에 있는 하나 이상의 레코드를 수정합니다.
  - editxgrp 명령은 XGROUP 클래스에 있는 하나 이상의 레코드를 작성 또는 수정합니다.
  - newxgrp 명령은 XGROUP 클래스에 있는 하나 이상의 레코드를 작성합니다.

필요한 권한

새로운 CA Access Control 그룹을 만들려면 최소한 다음 조건 중 하나를 충족해야 합니다.

- 사용자가 ADMIN 특성을 가집니다.
- ADMIN 클래스에 있는 GROUP 또는 XGROUP 레코드의 액세스 제어 목록에 CREATE 권한이 할당되어 있습니다.

그룹을 추가 또는 수정하려면 최소한 다음 조건 중 하나를 충족해야 합니다.

- 사용자가 ADMIN 특성을 가집니다.
- 그룹 레코드는 GROUP-ADMIN 특성을 가진 그룹 범위 내에 있습니다.
- 그룹의 소유자입니다.
- ADMIN 클래스에 있는 GROUP 또는 XGROUP 레코드의 액세스 제어 목록에 MODIFY(ch[x]grp 에 대해) 또는 CREATE(edit[x]grp 에 대해) 권한이 할당되어 있습니다.

```
{{chgrp|cg}|{chxgrp|cxg}|{editgrp|eg}|{editxgrp|exg}|{newgrp|ng}|{newxgrp|nxg}}
groupName ...
    [audit(none|all|success|failure|loginsuccess|loginfail|trace)|audit-] \
    [comment(string)|comment-] \
    [expire[(mm/dd/yy[yy[@hh:mm])]|expire-] \
    [gowner(groupName)] \
    [homedir(fullPath|nohomedir)] \
    [inactive(numInactiveDays)|inactive-] \
    [maxlogins(maximumNumberOfLogins)|maxlogins-] \
    [mem(groupName)|mem+(groupName)|mem-(groupName)]\
    [name('fullName')] \
    [nt[(comment(comment))]
    [owner(userName|groupName)] \
    [parent(groupName)|parent-] \
    [password( \
        [history(numberStoredPasswords)|history-] \
        [interval(maximumPasswordChangeInterval)|interval-] \
        [min_life(minimumPasswordChangeInterval)|min_life-] \
        [rules( \
                [alpha(minimumAlphaCharacters)] \
                [alphanum(minimumAlphanumericCharacters)] \
                [bidirectional|bidirectional-] \
                [grace(numberOfGraceLogins)] \
                [min_len(minimumPasswordLength)] \
                [max_len(maximumPasswordLength)] \
                [lowercase(minimumLowercaseCharacters)] \
                [max_rep(maxRepetitiveCharacters)] \
                [namechk|namechk-] \
                [numeric(minimumNumericCharacters)] \
                [oldpwchk|oldpwchk-] \
                [special(minimumSpecialCharacters)] \
                [uppercase(minimumUppercaseCharacters)] \
                [use_dbdict|use_dbdict-] \
        )|rules-] \
        )] \
    [pmdb(PolicyModelName)|pmdb-] \
    [restrictions( \
        [days({anyday|weekdays|{[mon] [tue] [wed] \
                [thu] [fri] [sat] [sun]}})] \
        [time(anytime|startTime:endTime) \
    |restrictions-] \
    [resume[(mm/dd/yy[yy][@hh:mm])]|resume-] \
    [shellprog(fullPath)] \
    [supgroup(superiorGroup)|supgroup-] \
    [suspend[(mm/dd/yy[yy][@hh:mm])]|suspend-] \
```

```
[unix[( \
    [appl(quotedString)] \
    [groupid(groupidNumber)] \
    [userlist(userName...)] \
)]] \
```

문자열을 사용하여 정의된 레코드 속성을 제거하려면 속성 바로 뒤에 -(빼기 기호) 또는 ()(빈 괄호)를 입력하십시오.

참고: 일부 매개 변수는 그룹이 프로필 그룹의 기능을 수행할 경우에만 적용됩니다. 프로필 그룹은 엔터프라이즈 그룹이 될 수 없습니다.

audit(mode)

명령에 대한 추적 감사를 설정합니다. 감사 모드는 none, all, success, failure, loginsuccess, loginfail 및 trace 입니다.

audit-

명령에 대한 추적 감사를 해제합니다.

comment(string)

그룹 레코드에 최대 255 자의 영숫자(단일 바이트)로 된 주석을 추가합니다. 문자열에 공백이 포함되어 있으면 전체 문자열 앞뒤에 작은따옴표를 입력하십시오. 이 문자열은 이전에 추가했던 모든 기존 문자열을 대체합니다.

참고: 독일어에서는 128 자만 기록됩니다.

comment-

그룹 레코드에서 설명 문자열을 삭제합니다. 이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 사용합니다.

expire(date)

그룹 구성원 계정이 만료되는 날짜를 설정합니다. 날짜를 지정하지 않으면 사용자가 현재 로그인하지 않은 경우 사용자 계정이 즉시 만료됩니다. 사용자가 로그인한 경우에는 사용자가 로그아웃할 때 계정이 만료됩니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

만료 날짜와 시간(선택 사항)을 mm/dd/yy [yy][@HH:MM] 형식으로 지정합니다. 연도는 2 자리수 또는 4 자리수가 될 수 있습니다.

참고: 다시 시작 날짜와 함께 resume 매개 변수를 지정하여 만료된 사용자 레코드를 활성화할 수 없습니다. 만료된 사용자 레코드를 활성화하려면 expire- 매개 변수를 사용하십시오.

expire-

newgrp 명령의 경우 만료 날짜가 없는 사용자 계정을 정의합니다. chgrp 및 editgrp 명령의 경우 사용자 계정에서 만료 날짜를 제거합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

gowner(groupName)

CA Access Control 사용자 또는 그룹을 그룹 레코드의 소유자로 할당합니다. 둘 이상의 그룹 이름을 지정하려면 괄호 안에 그룹 이름을 넣고 공백이나 쉼표로 그룹 이름을 구분하십시오. 그룹을 데이터베이스에 추가하면서 이 매개 변수를 생략하면 사용자 자신이 그룹 레코드의 소유자가 됩니다.

grace(numberOfGraceLogins)

사용자를 일시 중단시키기 전에 허용되는 최대 로그인 횟수를 설정합니다. 유예 로그인 수는 0 에서 255 사이여야 합니다. 유예 로그인 횟수에 도달하면 시스템 액세스가 거부되며 시스템 관리자에게 문의하여 새 암호를 선택해야 합니다. grace 를 0 으로 설정하면 사용자가 로그인할 수 없습니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

grace-

그룹에 대한 유예 로그인 설정을 삭제합니다. 이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 사용합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

groupName

작성할 그룹 또는 속성을 변경할 그룹의 이름을 지정합니다. new[x]grp 명령에서는 각 그룹 이름이 고유해야 하며, 데이터베이스에 현재 존재해서는 안 됩니다. 하지만 그룹과 사용자가 동일한 이름을 공유할 수 있습니다.

history

저장되는 암호의 개수를 지정합니다. history-를 사용하여 히스토리를 제거할 수 있습니다.

homedir(fullPath|nohomedir)

사용자 홈 디렉터리의 전체 경로를 지정합니다. 지정하는 경로가 슬래시로 끝나면 groupName 이 지정된 경로로 연결됩니다. nohomedir 을 지정하면 홈 디렉터리는 자동으로 설정되지 않습니다.

inactive(numInactiveDays)

시스템에서 사용자가 비활성 상태로 전환되기 전까지 대기하는 일 수를 지정합니다. 날짜 수에 도달하면 사용자는 로그인할 수 없습니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

numInactiveDays 로 양의 정수나 0 을 입력하십시오. inactive 를 0 으로 설정하면 inactive- 매개 변수를 사용하는 것과 결과적으로 동일합니다.

참고: 사용자 레코드에서 비활성 사용자는 표시되지 않습니다. 비활성 사용자를 식별하려면 마지막 액세스 시간 값을 비활성 기간 값과 비교해야 합니다.

inactive-

사용자의 상태를 비활성에서 활성으로 변경합니다. 이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 사용합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

interval(maximumPasswordChangeInterval)

암호 설정 또는 변경 이후 시스템에 새 사용자 암호를 입력해야 하는 일 수를 설정합니다. 0 또는 양수를 입력합니다. 간격 0 은 암호가 만료되지 않도록 그룹에 대한 암호 간격 검사를 비활성화합니다. setoptions 명령에서 설정된 기본값은 사용되지 않습니다. 보안 요구 사항이 낮은 사용자에 대해서만 간격을 0 으로 설정하십시오.

지정된 일 수에 도달하면 CA Access Control 은 사용자에게 현재 암호가 만료되었음을 알립니다. 사용자가 암호를 즉시 갱신하거나 유예 로그인 횟수에 도달할 때까지 기존 암호를 계속 사용할 수 있습니다. 유예 로그인 횟수에 도달하고 나면 시스템에 대한 액세스가 거부되고 시스템 관리자에게 연락하여 새 암호를 선택해야 합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

interval-

그룹에 대한 암호 간격 설정을 취소합니다. 취소된 경우 사용자 레코드의 값이 사용되고, 그렇지 않은 경우 setoptions 명령에서 설정된 기본값이 사용됩니다. 이 매개 변수를 chgrp 또는 editgrp 명령과 함께만 입력합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

maxlogins(maximumNumberOfLogins)

사용자가 동시에 로그인할 수 있는 최대 터미널 수를 설정합니다. 값 0 은 사용자가 터미널 수에 제한 없이 동시에 로그인할 수 있음을 의미합니다. 이 매개 변수를 지정하지 않은 경우 사용자 레코드의 값이 사용됩니다. 그렇지 않으면, 전역 최대 로그인 설정이 사용됩니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

참고: maxlogins 가 1 로 설정되어 있으면 selang 을 실행할 수 없습니다. CA Access Control 을 종료하고 maxlogins 설정을 1 보다 큰 수로 변경한 후 CA Access Control 을 다시 시작해야 합니다.

maxlogins-

그룹의 최대 로그인 설정을 삭제합니다. 이 매개 변수를 지정하지 않은 경우 사용자 레코드의 값이 사용됩니다. 그렇지 않으면, 전역 최대 로그인 설정이 사용됩니다. 이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 사용합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

mem(GroupName) | mem+(GroupName)

멤버 그룹(또는 자식 그룹)을 CA Access Control 에 있는 그룹에 추가합니다. 구성원 그룹(GroupName)이 이미 CA Access Control 에 정의되어 있어야 합니다. 둘 이상의 구성원 그룹을 추가하는 경우 그룹 이름을 쉼표로 구분하십시오. 그룹 이름에 공백이 있는 경우, 이름 앞뒤에 따옴표를 입력하십시오.

참고: 사용자를 내부 그룹에 추가하려면 join[x] 명령을 사용하십시오.

이 옵션은 내부 그룹에만 적용됩니다.

mem-(GroupName)

그룹에서 구성원 그룹을 제거합니다. 구성원 그룹(GroupName)이 이미 CA Access Control 에 정의되어 있어야 합니다. 둘 이상의 구성원 그룹을 제거하는 경우 그룹 이름을 쉼표로 구분하십시오. 그룹 이름에 공백이 있는 경우, 이름 앞뒤에 따옴표를 입력하십시오.

참고: 사용자를 내부 그룹에서 제거하려면 join[x]- 명령을 사용하십시오.

이 옵션은 내부 그룹에만 적용됩니다.

min_life(minimumPasswordChangeInterval)

사용자에게 암호 변경을 다시 허용하기 전에 경과해야 하는 최소 일 수입니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

min_life-

그룹의 min_life 설정을 삭제합니다. 이 매개 변수를 지정하지 않고 min_life 매개 변수가 사용자 레코드에 설정된 경우, 사용자 레코드에 있는 값이 사용됩니다. 그렇지 않으면, 전역 min_life 설정이 사용됩니다. 이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 사용합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

name(fullname)

그룹의 전체 이름을 지정합니다. 최대 47 자의 영숫자 문자열을 입력하십시오. 문자열에 공백이 포함되어 있으면 작은따옴표로 문자열을 둘러쌉니다.

nt(nt-group-attributes)

(Windows 전용) 로컬 Windows 시스템에서 그룹 정의를 추가 또는 변경합니다.

comment('comment')

기본 레코드에 설명 문자열을 추가합니다. 이전에 설명 문자열을 레코드에 추가한 경우 여기서 지정한 새 문자열이 기존 문자열을 바꿉니다.

comment 는 최대 255 자의 영숫자 문자열입니다. 문자열이 공백을 포함하는 경우, 작은 따옴표를 전체 문자열 앞뒤에 입력하십시오.

owner(Name)

CA Access Control 사용자 또는 그룹을 그룹 레코드의 소유자로 할당합니다. 데이터베이스에 그룹을 추가하고 이 매개 변수를 생략하는 경우, 사용자는 소유자가 됩니다. 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

parent(groupName)

기존 CA Access Control 그룹을 그룹 레코드의 상위 그룹으로 할당합니다. 부모 및 자식 관계에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

parent-

그룹과 해당 상위 그룹 간의 링크를 삭제합니다. 이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 사용합니다.

password

이 그룹에 암호를 할당합니다.

rules

암호에 규칙을 지정합니다.

alpha(minimumAlphaCharacters)

최소 영문자 수.

alphanum(minimumAlphanumericCharacters)

최소 문자 수.

bidirectional|bidirectional-

양방향 암호 암호화의 사용 여부를 지정합니다. 양방향 암호 암호화를 활성화하면 각각의 새 암호가 암호화되며 나중에 일반 텍스트로 다시 암호를 해독할 수 있습니다. 이 암호화는 새 암호와 이전 암호(암호 기록) 사이에 보다 폭넓은 비교를 제공합니다. 양방향 암호화가 비활성화되면, 단방향 암호 기록 암호화가 활성화되어 이전 암호를 해독할 수 없습니다.

참고: 이 기능을 사용하려면 기록을 1 보다 큰 값으로 설정해야 합니다.

참고: UNIX 에서도 이 기능을 사용하려면 구성 설정 passwd_format 을 NT 로 설정해야 합니다.

min_len(minimumPasswordLength)

최소 암호 길이.

max_len(maximumPasswordLength)

최대 암호 길이.

lowercase(minimumLowercaseCharacters)

최소 소문자 수.

max_rep(maximumRepetitiveCharacters)

최대 반복 문자 수.

namechk|namechk-

이름과 비교하여 암호 검사.

numeric(minimumNumericCharacters)

최소 숫자 수.

oldpwchk|oldpwchk-

이전 암호와 비교하여 암호 검사.

special(minimumSpecialCharacters)

최소 특수 문자 수.

uppercase(minimumUppercaseCharacters)

최소 대문자 수.

use_dbdict|use_dbdict-

암호 사전을 설정합니다. use_dbdict 는 토큰을 **db** 로 설정하고, 암호를 CA Access Control 데이터베이스에 있는 단어와 비교합니다. use_dbdict- sets 는 토큰을 **file** 로 설정하고, seos.ini 파일에 지정된 파일(UNIX 의 경우) 또는 Windows 레지스트리(Windows 의 경우)에 대해 암호를 검사합니다.

password-

그룹에 대한 암호 필요성을 삭제합니다.

pmdb(PolicyModelName)

그룹의 사용자가 sepass 유틸리티를 사용하여 암호를 변경할 경우 새 암호가 지정된 정책 모델로 전달되도록 지정합니다. PMDB 의 완전한 이름을 입력합니다.

이 암호는 seos.ini 의 [seos] 섹션에 있는 parent_pmd 토큰 또는 passwd_pmd 토큰에 정의된 정책 모델로 전달되지 않습니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

pmdb-

그룹 레코드에서 PMDB 특성을 제거합니다. 이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 사용합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

restrictions(days(dayData) time(timeData))

그룹의 구성원이 시스템에 로그인할 수 있는 요일 및 시간을 지정합니다.

CA Access Control 은 사용자가 로그인한 동안 로그인 기간이 만료될 경우 사용자를 강제로 로그오프시키지 않습니다. 또한 로그인 제한은 일괄 작업에 적용되지 않으며, 사용자는 언제라도 배경 프로세스를 실행할 수 있습니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

days 인수를 생략하고 time 인수를 지정할 경우, 시간 제한은 레코드에 이미 표시된 요일 제한에 적용됩니다. time 을 생략하고 days 를 지정할 경우 요일 제한은 레코드에 이미 표시된 시간 제한에 적용됩니다. days 와 time 을 모두 지정할 경우, 그룹 구성원은 지정된 요일에 지정된 시간 동안에만 시스템에 액세스할 수 있습니다.

days(**dayData**)

사용자가 시스템에 로그인할 수 있는 요일을 지정합니다. days 인수는 다음 하위 인수를 가집니다.

- **anyday**-사용자는 모든 요일에 로그인할 수 있습니다.
- **weekdays**-사용자는 주중(월요일부터 금요일까지)에만 로그인할 수 있습니다.
- **mon tue wed thu fri sat sun**-사용자는 지정된 요일에만 로그인할 수 있습니다. 순서에 상관 없이 요일을 지정할 수 있습니다. 둘 이상의 요일을 지정하려면 공백 또는 쉼표로 날짜를 구분하십시오.

time(**timeData**)

사용자가 시스템에 로그인할 수 있는 기간을 지정합니다. time 인수는 다음 하위 인수를 가집니다.

- **anytime**-사용자는 하루 중 언제든지 로그인할 수 있습니다.
- startTime**:**endTime-사용자는 지정된 기간 동안에만 로그인할 수 있습니다. startTime 및 endTime 형식은 모두 hhmm 이며, 여기서 hh 는 24 시간 표기(00 - 23) 시간이고 mm 은 분(00 - 59)입니다. 2400 은 올바른 시간 값이 아닙니다. endTime 이 startTime 보다 작은 숫자인 경우, 기간은 자정을 지나 연장된 것으로 간주됩니다. 그렇지 않으면 같은 날에 발생한 것으로 간주됩니다.

  참고: CA Access Control 은 프로세서의 시간대를 사용합니다. 사용자가 프로세서와 다른 시간대에 터미널에 로그인할 수 있으므로, 이런 상황을 고려해야 합니다.

restrictions-

시스템에 로그인할 수 있는 사용자의 권한을 제한하는 모든 제한을 그룹 레코드에서 삭제합니다. 이 매개 변수를 지정하지 않고 restrictions 매개 변수가 사용자 레코드에 설정된 경우, 사용자 레코드의 값이 사용됩니다. 이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 사용합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

resume(date)

suspend 매개 변수를 지정하여 비활성화된 사용자 레코드를 활성화합니다. 날짜와 시간(선택 사항)을 mm/dd/yy[@HH:MM] 형식으로 입력합니다.

suspend 매개 변수와 resume 매개 변수를 모두 지정한 경우 계속 날짜는 일시 중지 날짜 이후여야 합니다. date 를 생략하는 경우, chgrp 명령을 실행하는 즉시 사용자가 활성화됩니다. 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오. 이 매개 변수는 프로필 그룹에만 적용됩니다.

resume-

사용된 경우, 계속 날짜와 시간을 그룹 레코드에서 삭제합니다. 그 결과, 사용자의 상태가 활성(활성화됨)에서 일시 중지됨으로 변경됩니다. 이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 사용합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

shellprog(fullPath)

사용자가 login 또는 su 명령을 호출한 후 실행되는 초기 프로그램 또는 셸의 전체 경로를 지정합니다. FullPath 는 문자열입니다.

supgroup(Group'sSuperiorGroup)

슈퍼 그룹(또는 상위 그룹)을 지정합니다.

suspend(date)

사용자 레코드를 비활성화하지만 데이터베이스에 정의된 상태를 유지합니다. 날짜와 시간(선택 사항)을 mm/dd/yy[@HH:MM] 형식으로 입력합니다.

사용자는 일시 중지된 사용자 계정을 사용하여 시스템에 로그인할 수 없습니다. date 가 지정된 경우, 사용자 레코드는 지정된 날짜에 일시 중지됩니다. date 를 생략하는 경우, 사용자 레코드는 chgrp 명령을 실행하는 즉시 일시 중지됩니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

suspend-

사용자 레코드에서 일시 중지 날짜를 삭제하고 사용자 상태를 비활성화됨에서 활성(활성화된)으로 변경합니다. 이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 사용합니다. 이 매개 변수는 프로필 그룹에만 적용됩니다.

unix(groupidNumber)

(UNIX 전용) UNIX 에서 그룹 특성을 설정하거나, 그룹이 아직 존재하지 않는 경우 그룹을 작성합니다.

groupidNumber 는 십진수입니다. 그룹 ID 를 0 으로 지정할 수 없습니다. 숫자를 생략하면 CA Access Control 은 가장 큰 현재의 그룹 ID 를 찾고, 여기에 1 을 더한 숫자로 그룹의 ID 를 설정합니다. 한 번에 둘 이상의 그룹을 추가 또는 수정할 때 CA Access Control 은 이런 방법으로 그룹 ID 번호를 만듭니다. seos.ini 파일의 AllowedGidRange 토큰은 사용 불가능한 숫자를 정의할 수 있습니다.

userlist(userName)

구성원을 그룹에 할당합니다. UserName 은 하나 이상의 UNIX 사용자 이름입니다. 둘 이상의 사용자를 할당하려면 사용자 이름을 공백 또는 쉼표로 구분하십시오. chgrp 명령과 editgrp 명령의 경우 여기서 지정된 멤버 목록이 해당 그룹에 대해 현재 정의되어 있는 멤버 목록을 대체합니다.

예제

- 사용자 Bob 은 엔터프라이즈 그룹 Sales 의 부모 그룹과 소유 그룹을 ACCOUNTS 에서 PAYROLL 로 변경하려고 합니다.

  ```
  chxgrp Sales parent(PAYROLL) owner(PAYROLL)
  ```

- 사용자 Admin1 은 그룹 projectB 의 부모 그룹을 divisionA 에서 divisionB 로 변경하고 그룹 RESEARCH 를 새 소유자로 할당하려고 합니다.

  Admin1 은 ADMIN 특성을 가지고 있습니다.

  ```
  chxgrp projectB parent(divisionB) owner(RESEARCH)
  ```

- 관리자 Sally 는 그룹 프로필 NewEmployee 에 대한 홈 디렉터리와 셸 프로그램 사양을 제거하려고 합니다.

  Sally 는 NewEmployee 의 소유자입니다.

  ```
  editgrp NewEmployee homedir() shellprog()
  ```

- 사용자 Admin1 은 ProjectA 그룹을 RESEARCH 그룹의 하위 그룹으로 추가하려고 합니다. 사용자 Admin1 은 ProjectA 그룹의 소유자가 되어야 합니다.

  Admin1 은 ADMIN 특성을 가지고 있습니다.

  기본값은 owner(Admin1)입니다.

  ```
  newgrp ProjectA parent(RESEARCH)
  ```

추가 정보:

join[x] 명령-내부 그룹에 사용자 추가(페이지 125)
join[x]- 명령-그룹에서 사용자 제거(페이지 128)
rm[x]grp 명령-그룹 레코드 삭제(페이지 132)
show[x]grp 명령-그룹 속성 표시(페이지 148)
chgrp 명령-UNIX 그룹 수정(페이지 168)
chgrp 명령-Windows 그룹 수정(페이지 185)

chres 명령-리소스 레코드 수정

AC 환경에 해당

CA Access Control 클래스에 속한 리소스 레코드로 작업하려면 chres, editres 및 newres 명령을 사용하십시오. 이러한 명령은 구조가 동일하며, 다음의 내용만 다릅니다.

- chres 명령은 하나 이상의 리소스를 수정합니다.
- editres 명령은 하나 이상의 리소스를 작성 또는 수정합니다.
- newres 명령은 하나 이상의 리소스를 작성합니다.

참고: 이 명령은 네이티브 Windows 환경에도 있지만 작동 방식이 다릅니다.

newres 명령을 사용하여 리소스를 추가하려면 최소한 다음 조건 중 하나를 충족해야 합니다.

- 사용자가 ADMIN 특성을 가집니다.
- 클래스에 의한 액세스(ADMIN) 클래스에 있는 리소스 클래스 레코드의 ACL 에 CREATE 액세스 권한이 있는 경우.
- seos.ini 파일의 use_unix_file_owner 토큰이 yes 로 설정되어 있을 경우 UNIX 의 파일 소유자가 이것을 CA Access Control 의 새 리소스로 정의할 수 있습니다.

chres 또는 editres 명령을 사용하여 리소스를 추가 또는 변경하려면 리소스에 대한 충분한 권한이 있어야 합니다. CA Access Control 은 이러한 조건 중 하나에 대해 다음 순서에 따라 검사합니다.

1. 사용자가 ADMIN 특성을 가집니다.
2. 리소스 레코드는 사용자가 GROUP-ADMIN 속성을 가지는 그룹의 범위 내에 있습니다.
3. 사용자가 레코드의 소유자입니다.
4. 클래스에 의한 액세스(ADMIN) 클래스에 있는 리소스 클래스 레코드의 액세스 제어 목록에서 MODIFY(chres 의 경우) 액세스 권한이나 CREATE(editres 의 경우) 액세스 권한이 지정된 경우.

참고: 리소스 이름의 최대 길이는 255 자입니다.

다음 표에는 chres, editres 및 newres 명령을 사용하여 관리할 수 있는 각 클래스에 적용되는 명령 매개 변수가 나열되어 있습니다.

클래스	속성											
	audit	**calendar**	**category**	**comment**	**defaccess**	**label**	**level**	**notify**	**owner**	**restrictions[]**	**warning**	**other**
ACVAR				X					X			VARIABLE_TYPE, VARIABLE_VALUE
ADMIN	X	X	X	X	X	X	X	X	X	X	X	
CALENDAR				X					X			
CATEGORY				X					X			
CONNECT	X	X	X	X	X	X	X	X	X	X	X	

클래스	속성											
	audit	**calendar**	**category**	**comment**	**defaccess**	**label**	**level**	**notify**	**owner**	**restrictions[]**	**warning**	**other**
CONTAINER	X	X		X					X		X	MEM
DOMAIN	X	X	X	X	X	X	X	X	X	X	X	MEM
FILE	X	X	X	X	X	X	X	X	X	X	X	
GFILE	X	X		X				X	X		X	MEM
GHOST	X	X		X					X	X	X	MEM
GSUDO		X		X	X				X			MEM
GTERMINAL	X	X		X	X				X	X		MEM
HNODE	X	X	X	X	X	X	X	X	X	X	X	SUBSCRIBER, POLICY
HOLIDAY	X		X	X	X	X	X	X	X	X	X	DATES
HOST	X	X		X					X	X	X	
HOSTNET	X	X		X					X		X	MASK, MATCH
HOSTNP	X	X		X					X	X	X	
LOGINAPPL	X	X		X	X			X	X	X	X	LOGINFLAGS, LOGINMETHOD,

클래스	속성											
	audit	**calendar**	**category**	**comment**	**defaccess**	**label**	**level**	**notify**	**owner**	**restrictions[]**	**warning**	**other**
												LOGINPATH, LOGINSEQUENCE
MFTERMINAL	X	X	X	X		X	X	X	X		X	DAYTIME
POLICY	X	X	X	X	X	X	X	X	X	X	X	SIGNATURE, RULESET
PROCESS	X	X	X	X	X	X	X	X	X	X	X	
PROGRAM	X	X	X	X	X	X	X	X	X	X	X	TRUST
PWPOLICY				X					X			
REGKEY	X	X		X	X			X	X		X	DAYTIME
REGVAL	X	X		X	X			X	X		X	DAYTIME
RULESET	X	X	X	X	X	X	X	X	X	X	X	SIGNATURE, CMD, UNDOCMD
SECFILE				X					X			TRUST, FLAG

클래스	속성											
	audit	**calendar**	**category**	**comment**	**defaccess**	**label**	**level**	**notify**	**owner**	**restrictions[]**	**warning**	**other**
												S
SECLABEL			X	X			X		X			
SEOS		X	X	X		X	X					HOST
SPECIALPGM				X					X			
SUDO	X	X	X	X	X	X	X	X	X	X	X	TARGUID, PASSWORD
SURROGATE	X	X	X	X	X	X	X	X	X	X	X	
TCP	X		X	X	X	X	X	X	X	X	X	
TERMINAL	X	X	X	X	X	X	X	X	X	X	X	
UACC	X		X	X	X				X			
USERATTR									X		X	
USERDIR	X			X					X			

```
{{chres|cr}|{editres|er}|{newres|nr}} className resourceName \
    [ac_id(id)] \
    [audit({none|all|success|failure})] \
    [calendar[-](calendarName)] \
    [category[-](categoryName)] \
    [cmd+(selang_command_string)|cmd-] \
    [comment(string)|comment] \
    [container[-](containerName)] \
    [dates(timeperiod)] \
    [dh_dr{-|+}(dh_dr)] \
    [disable|disable-] \
    [defaccess(accessAuthority)] \
    [filepath(filePaths)] \
    [flags[-|+](flagName)] \
    [gacc(accessvalue)] \
    [gowner(groupName)] \
    [host(host-name)|host-] \
    [label(labelName)|label] \
    [level(number)|level] \
    [mask(inetAddress)|match(inetAddress)] \
    [mem(resourceName)|mem(resourceName)] \
    [node_alias{-|+}(alias)] \
    [node_ip{-|+}(ip)] \
    [notify(mailAddress)|notify] \
    [of_class(className)] \
    [owner({userName | groupName})] \
    [{password | password}] \
    [policy(name(policy-name) {{deviation+|dev+}|{deviation-|dev-}})] \
    [policy(name(policy-name) status(policy-status)
    {updator|updated_by}(user-name))] \
    [{restrictions([days({anyday|weekdays|{[mon] [tue] [wed] \
        [thu] [fri] [sat] [sun]}})] \
        [time({anytime|startTime:endTime}) \
    |restrictions}] \
    [targuid(userName)] \
    [trust | trust] \
    [value{+|-}(value)] \
    [warning | warning]
```

ac_id(id)

로컬 CA Access Control 데이터베이스 및 DMS 에 저장된 끝점(HNODE 개체)에 대해 고유한 ID 를 정의합니다. CA Access Control 은 이 ID 를 사용하여 HNODE 를 식별합니다. 따라서 끝점의 IP 주소 또는 이름을 변경해도 고급 정책 관리 기능이 영향을 받지 않으며, CA Access Control 은 여전히 끝점을 추적할 수 있습니다.

audit

기록되는 액세스 이벤트를 나타냅니다. 다음 특성 중 하나를 지정합니다.

- **all**-CA Access Control 은 권한 있는 액세스 시도와 권한 없는 액세스 시도를 모두 기록합니다.
- **failure**-CA Access Control 은 권한 없는 액세스 시도를 기록합니다. 이것이 기본값입니다.
- **none**-CA Access Control 은 레코드를 로그 파일에 기록하지 않습니다.
- **success**-CA Access Control 은 권한 있는 액세스 시도를 기록합니다.

calendar(calendarName)

Unicenter TNG 에서 시간 제한을 나타내는 Unicenter NSM 달력 레코드를 정의합니다. CA Access Control 은 이러한 개체 목록을 관리 용도로만 유지하며 보호하지는 않습니다. 둘 이상의 달력을 할당하려면 달력 이름을 공백 또는 쉼표로 구분하십시오.

calendar(calendarName)

리소스 레코드에서 하나 이상의 Unicenter NSM 달력 레코드를 삭제합니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

category(categoryName [,categoryName...])

리소스 레코드에 하나 이상의 보안 범주를 할당합니다.

CATEGORY 클래스가 활성화되어 있지 않을 때 category 매개 변수를 지정하는 경우 CA Access Control 은 데이터베이스의 리소스 정의를 업데이트합니다. 그러나 업데이트된 범주 할당은 CATEGORY 클래스가 다시 활성화될 때까지 적용되지 않습니다.

category(categoryName [,categoryName...])

리소스 레코드에서 하나 이상의 보안 범주를 삭제합니다.

지정된 보안 범주가 CATEGORY 클래스의 활성 여부에 상관 없이 리소스 레코드에서 삭제됩니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

className

리소스가 속한 클래스의 이름을 지정합니다. CA Access Control 에 정의된 리소스 클래스를 나열하려면 find 명령을 사용하십시오.

cmd+(selang_command_string)

정책을 정의하는 selang 명령의 목록을 지정합니다. 이러한 명령은 정책 배포에 사용됩니다. 예를 들면 다음과 같습니다.

```
editres RULESET IIS5#02 cmd+("nr FILE /inetpub/* defaccess(none) owner(nobody)")
```

cmd-

RULESET 개체에서 정책 배포 명령 목록을 제거합니다.

comment(string)

최대 255 자로 구성된 영숫자 문자열을 리소스 레코드에 추가합니다. 문자열이 공백을 포함하는 경우, 작은 따옴표를 전체 문자열 앞뒤에 입력하십시오. 이 문자열은 이전에 정의한 기존의 모든 문자열을 대체합니다.

참고: SUDO 클래스의 경우 이 문자열은 특별한 의미를 가지고 있습니다. SUDO 레코드 정의에 대한 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

comment

리소스 레코드에서 설명을 삭제합니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

container(containerName)

포괄적인 그룹 설정 클래스인 CONTAINER 객체를 나타냅니다.

containerName 은 CONTAINER 클래스에 정의된 하나 이상의 CONTAINER 레코드 이름입니다. 둘 이상의 CONTAINER 를 할당하려면 이름을 공백 또는 쉼표로 구분하십시오.

container(containerName)

리소스 레코드에서 하나 이상의 CONTAINER 레코드를 삭제합니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

dates(timeperiod)

휴일과 같이 사용자가 로그인할 수 없는 하나 이상의 기간을 정의합니다. 둘 이상의 기간을 지정하려면, 공백으로 각 기간을 구분하십시오. 다음 형식을 사용합니다.

```
mm/dd[/yy[yy]][@hh:mm][-mm/dd]/[/yy[yy]][@hh:mm]
```

연도를 지정하지 않은 경우(또는 1990 년 전의 연도를 지정하는 경우) 그 기간이나 휴일이 해마다 적용됨을 의미합니다. 연도는 98 또는 1998 과 같이 두 자리 또는 네 자리 숫자로 지정할 수 있습니다.

시작 시간을 지정하지 않은 경우, 그 날의 시작 시간(자정)이 사용됩니다. 종료 시간을 지정하지 않은 경우, 그 날의 종료 시간(자정)이 사용됩니다. 시간과 분의 형식은 모두 hh:mm 이며, 여기서 hh 는 24 시간 표기(00 - 23) 시간이고 mm 은 분(00 - 59)입니다.

시간 간격을 지정하지 않고(예를 들어, 12/25@14:0012/25@17:00) 월과 일(12/25)만 지정하는 경우, 휴일은 하루 종일 계속됩니다.

휴일이 발생하는 다른 시간대에 명령을 작성하는 경우, 해당 기간을 지역 시간으로 변환합니다. 예를 들어, 현재 위치가 뉴욕이고 LA 에 반나절 휴일이 있는 경우 09/14/98@18:0009/14/98@20:00 을 입력해야 합니다. 이렇게 하면 LA 에서 오후 3 시부터 오후 5 시까지 사용자가 로그인할 수 없습니다.

defaccess([accessAuthority])

리소스에 대한 기본 액세스 권한을 정의합니다. 기본 액세스 권한은 리소스 액세스 요청자 중에서 리소스 액세스 제어 목록에 없는 접근자에게 부여되는 권한입니다. 기본 액세스는 데이터베이스에 정의되지 않은 사용자에게도 적용됩니다. 유효한 액세스 권한 값은 클래스에 따라 다릅니다.

accessAuthority 를 생략하면 CA Access Control 은 UACC 클래스에서 리소스의 클래스를 나타내는 레코드의 UACC 속성에 지정된 암시적 액세스 권한을 할당합니다.

dh_dr{+|-}(dh_dr)

이 끝점이 재해 복구를 위해 사용할 배포 호스트를 정의합니다.

filepath(filePaths)

하나 이상의 절대 파일 경로를 정의합니다. 각 경로는 유효한 커널 모듈을 구성합니다. 여러 파일 경로는 콜론(:)으로 구분합니다.

flags(flagName)

리소스가 트러스트되는 방법과 리소스의 트러스트된 상태를 확인하는 방법에 대해 정의합니다. 사용 가능한 플래그는 Ctime, Mtime, Mode, Size, Device, Inode, Crc 및 Own/All/None 입니다.

gacc(accessvalue)

자주 사용하는 보호되는 파일을 프로그램이 다른 방법보다 훨씬 더 빠르게 액세스할 수 있게 해 줍니다.

gowner(groupName)

CA Access Control 그룹을 리소스 레코드의 소유자로 할당합니다. 소유자의 보안 수준, 보안 레이블 및 보안 범주 권한이 리소스에 액세스할 수 있을 만큼 충분한 경우 리소스 레코드의 소유자는 리소스에 제한 없이 액세스할 수 있습니다. 리소스의 그룹 소유자는 항상 리소스 레코드를 업데이트 및 삭제할 수 있는 권한을 가집니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

label(labelName)

리소스 레코드에 보안 레이블을 할당합니다.

label-

리소스 레코드에서 보안 레이블을 삭제합니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

level(number)

리소스 레코드에 보안 수준을 할당합니다. 1-255 사이의 양의 정수를 입력합니다.

level-

리소스에서 모든 보안 수준을 제거합니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

mask (IPv4address) match (IPv4address)

mask 및 match 매개 변수는 HOSTNET 레코드에만 적용할 수 있습니다. HOSTNET 레코드를 작성할 때는 이 두 매개 변수가 필수 사항이며 레코드를 수정할 때는 선택 사항입니다.

HOSTNET 레코드에 의해 정의된 호스트의 그룹을 정의하려면 mask 와 match 를 함께 사용하십시오. mask 주소가 있는 호스트 IP 주소의 AND 가 match 주소를 생성하는 경우 호스트는 HOSTNET 레코드 그룹의 구성원입니다.

예를 들어 mask(255.255.255.0)와 match(192.16.133.0)를 지정할 경우, 호스트 IP 주소의 범위가 192.16.133.0 ~ 192.16.133.255 이면 호스트는 그룹의 구성원이 됩니다.

mask 및 match 매개 변수에는 IPv4 주소가 필요합니다.

Mem-(resourceName)

구성원 리소스를 리소스 그룹에 추가합니다. 둘 이상의 구성원 리소스를 추가하는 경우 쉼표로 각 이름을 구분하십시오.

다음 클래스의 리소스 레코드에 대해서만 mem 매개 변수를 사용할 수 있습니다.

- CONTAINER. 이 클래스는 다른 리소스 클래스의 개체 그룹을 정의합니다.
- GFILE. 이 클래스에는 호스트 그룹을 정의하는 리소스 레코드가 있습니다.
- GHOST. 이 클래스에는 호스트 그룹을 정의하는 리소스 레코드가 있습니다.
- GSUDO. 이 클래스에는 명령 그룹을 정의하는 리소스 레코드가 있습니다.
- GTERMINAL. 이 클래스에는 터미널 그룹을 정의하는 리소스 레코드가 있습니다.
- GPOLICY. 이 클래스에는 논리 정책을 정의하는 리소스 레코드가 있습니다.
- GHNODE. 이 클래스에는 호스트 그룹을 정의하는 리소스 레코드가 있습니다.
- GDEPLOYMENT. 이 클래스에는 정책 배포를 정의하는 리소스 레코드가 있습니다.

GFILE 클래스의 리소스 그룹에 FILE 레코드를 추가하는 것과 같이 적절한 유형의 레코드를 리소스 그룹에 추가하려면 mem 매개 변수를 사용하십시오.

참고: CONTAINER 리소스에 mem 매개 변수를 사용하는 경우 of_class 매개 변수도 포함해야 합니다.

그러나 구성원 리소스와 리소스 그룹은 이미 CA Access Control 에 정의되어 있어야 합니다. 리소스 그룹을 작성하려면 원하는 클래스의 리소스를 작성하십시오. 예를 들어, 다음 명령을 사용하면 GFILE 리소스 그룹이 작성됩니다.

```
newres GFILE myfiles
```

mem(resourceName)

리소스 그룹에서 구성원 리소스를 제거합니다. 둘 이상의 구성원 리소스를 제거하는 경우 리소스 이름을 공백 또는 쉼표로 구분하십시오. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

node_alias{-|+}(alias)

끝점 별칭을 정의합니다.

끝점에 대해 별칭을 정의하면 CA Access Control 은 별칭을 기반으로 실제 끝점에 고급 정책 관리 명령을 전송할 수 있습니다.

node_ip[-|+](ip)

호스트의 IP 주소를 정의합니다. 고급 정책 관리는 끝점의 이름과 함께 IP 주소를 사용하여 필요한 끝점을 찾습니다.

notify(mailAddress)

리소스 레코드로 표시되는 리소스에 액세스할 때마다 알림 메시지를 전송하도록 CA Access Control 에 지시합니다. 사용자 이름이나 전자 메일 주소를 입력하거나 별칭이 지정된 경우 메일 그룹의 전자 메일 주소를 입력하십시오.

통지는 로그 라우팅 시스템이 활성 상태인 경우에만 발생합니다. 통지 메시지는 로그 라우팅 시스템의 설정에 따라 화면 또는 사용자의 사서함으로 전송됩니다.

통지 메시지가 전송될 때마다 감사 레코드가 감사 로그에 기록됩니다. 감사 레코드의 필터링 및 보기에 대한 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

통지 메시지의 수신인은 자주 로그인하여 각 메시지에서 설명하고 있는 무단 액세스 시도에 응답해야 합니다.

제한: 30 자.

notify

리소스 레코드로 표시된 리소스를 성공적으로 액세스하면 누구에게도 통지하지 않게 지정합니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

of_class(className)

mem 매개 변수를 사용하여 컨테이너(CONTAINER) 클래스에 추가하는 레코드의 리소스 유형을 지정합니다.

owner(Name)

CA Access Control 사용자 또는 그룹을 리소스 레코드의 소유자로 할당합니다. 소유자의 보안 수준, 보안 레이블 및 보안 범주 권한이 리소스에 액세스할 수 있을 만큼 충분한 경우 리소스 레코드의 소유자는 리소스에 제한 없이 액세스할 수 있습니다. 리소스 소유자는 항상 리소스 레코드를 업데이트 및 삭제할 수 있습니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

password

SUDO 클래스에 sesudo 명령을 사용하려면 원래 사용자의 암호를 요구하도록 지정합니다.

password-

password 매개 변수를 취소하여 sesudo 명령을 사용할 때 더 이상 원래 사용자의 암호가 필요하지 않습니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다. password 매개 변수를 이전에 사용하지 않은 경우 이 매개 변수는 필요하지 않습니다.

policy(name(name#xx) status(status) updated_by(name)) | policy(name(name#xx) deviation{+|-})

전파 트리에서 노드의 구독자를 추가하고 그 상태를 지정합니다. 또는 정책 위반의 존재 여부를 지정하기 위해 기존의 정책 버전을 업데이트합니다. 정책 상태를 업데이트할 때는 updated_by 속성을 업데이트해야 합니다. 이것은 정책 상태를 변경한 사용자의 이름을 나타내는 문자열입니다.

정책 상태는 전송됨, 배포됨, 배포 취소, 실패, 서명 실패, 큐에 추가됨, 배포 취소 실패 또는 전송 실패 중 하나가 될 수 있습니다.

policy-[(name(name#xx))]

지정된 정책 버전을 노드에서 제거합니다. 정책을 지정하지 않으면 이 노드에 배포된 모든 정책이 제거됩니다.

resourceName

수정 또는 추가할 리소스 레코드의 이름을 정의합니다. 둘 이상의 리소스를 변경 또는 추가할 경우 리소스 이름 목록을 괄호로 묶고 공백이나 쉼표로 리소스 이름을 구분합니다. 최소한 하나 이상이 리소스 이름을 지정해야 합니다.

CA Access Control 은 지정된 매개 변수에 따라 각 리소스 레코드를 독립적으로 처리합니다. 리소스를 처리할 때 오류가 발생하면 CA Access Control 은 메시지를 작성하고 목록의 다음 리소스를 계속 처리합니다.

참고: 리소스 이름에 변수를 사용하는 경우 다음 구문을 사용하여 변수를 참조하십시오: <!variable>. 예: <!AC_ROOT_PATH>\bin. 정책의 selang 규칙에서만 변수를 사용할 수 있습니다.

restrictions([days] [time])

사용자가 파일에 액세스할 수 있는 요일 및 시간을 지정합니다.

days 인수를 생략하고 time 인수를 지정할 경우, 시간 제한은 레코드에 이미 표시된 요일 제한에 적용됩니다. time 을 생략하고 days 를 지정할 경우 요일 제한은 레코드에 이미 표시된 시간 제한에 적용됩니다. days 와 time 을 모두 지정할 경우, 사용자는 지정된 요일에 지정된 시간 동안에만 시스템에 액세스할 수 있습니다.

- [Days]는 사용자가 파일에 액세스할 수 있는 요일을 지정합니다. days 인수는 다음 하위 인수를 가집니다.
  - **anyday**-사용자는 모든 요일에 파일에 액세스할 수 있습니다.
  - **weekdays**-사용자는 주중(월요일부터 금요일까지)에만 리소스에 액세스할 수 있습니다.
  - **Mon**, **Tue**, **Wed**, **Thu**, **Fri**, **Sat**, **Sun**-사용자는 지정된 요일에만 리소스에 액세스할 수 있습니다. 순서에 상관 없이 요일을 지정할 수 있습니다. 둘 이상의 요일을 지정하려면 공백 또는 쉼표로 날짜를 구분합니다.
- [Time]은 사용자가 리소스에 액세스할 수 있는 기간을 지정합니다. time 인수는 다음 하위 인수를 가집니다.
  - **anytime**-사용자는 하루 중 언제든지 리소스에 액세스할 수 있습니다.
  - **startTime:endTime**-사용자는 지정된 기간 동안에만 리소스에 액세스할 수 있습니다. startTime 및 endTime 형식은 모두 hhmm 이며, 여기서 hh 는 24 시간 표기(00 - 23) 시간이고 mm 은 분(00 - 59)입니다. 2400 은 올바른 시간 값이 아닙니다. startTime 은 endTime 보다 작아야 하며, 두 시간 모두 같은 날에 속해야 합니다. 터미널의 시간대가 프로세서와 다른 경우, 터미널의 시작 시간과 종료 시간을 프로세서에 대해 동등한 로컬 시간으로 변환하여 시간 값을 조정합니다. 예를 들어, 프로세서가 뉴욕에 있고 터미널이 LA 에 있는 경우 LA 에서 오전 8 시에서 오후 5 시까지 터미널에 대한 액세스를 허용하려면 시간(1100:2000)을 지정하십시오.

restrictions-([days] [time])

사용자의 파일 액세스 권한을 제한하는 모든 제한을 삭제합니다.

ruleset+(name)

정책과 연결할 규칙 집합을 지정합니다.

ruleset-(name)

정책에서 규칙 집합을 삭제합니다. 규칙 집합을 지정하지 않으면 정책에서 모든 규칙 집합이 제거됩니다.

signature(hash_value)

해시 값을 지정합니다. 정책에서 이 값은 정책과 연결된 RULESET 개체의 서명을 기반으로 합니다. 규칙 집합에서 이 값은 정책 배포 명령 목록 및 정책 배포 취소(제거) 명령 목록을 기반으로 합니다.

subscriber(name(sub_name) status(status))

전파 트리에서 노드의 구독자를 추가하고 그 상태를 지정합니다. 상태는 **unknown**, **available**, **unavailable** 또는 **sync** 중 하나가 될 수 있습니다.

subscriber-(name(sub_name)) | sub-

노드에서 구독자 데이터베이스를 제거합니다. 구독자를 지정하지 않으면 모든 구독자가 제거됩니다.

targuid(userName)

SUDO 클래스의 경우, 이 명령을 실행할 수 있는 권한을 빌려주는 사용자의 이름을 지정합니다. 기본값은 root 입니다.

trust

리소스가 트러스트되었음을 지정합니다. trust 매개 변수는 PROGRAM 클래스와 SECFILE 클래스의 리소스에만 적용됩니다. 사용자는 프로그램이 트러스트된 상태를 유지하는 한 프로그램을 실행할 수 있습니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

trust-

리소스가 언트러스트되었음을 지정합니다. trust 매개 변수는 PROGRAM 클래스와 SECFILE 클래스의 리소스에만 적용됩니다. 사용자는 언트러스트된 프로그램을 실행할 수 없습니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

undocmd+(selang_command_string)

정책 배포 취소를 정의하는 selang 명령의 목록을 지정합니다. 이러한 명령은 배포된 정책을 제거(배포 취소)하는 데 사용됩니다. 예:

```
editres RULESET IIS5#02 undocmd+("rr FILE /inetpub/*")
```

undocmd-

RULESET 개체에서 정책 제거 명령 목록을 제거합니다.

value+(value)

지정된 변수(ACVAR 개체)에 지정된 값을 추가합니다.

value-(value)

지정된 변수(ACVAR 개체)에서 지정된 값을 제거합니다.

warning

접근자의 권한이 리소스에 액세스하기에 충분하지 않더라도 CA Access Control 에서 리소스에 대한 액세스를 허용하도록 지정합니다. 단, CA Access Control 은 감사 로그에 경고 메시지를 기록합니다.

참고: 경고 모드에서는 CA Access Control 이 리소스 그룹에 대한 경고 메시지를 작성하지 않습니다.

warning-

접근자의 권한이 리소스에 액세스하기에 충분하지 않으면 CA Access Control 은 리소스에 대한 사용자의 액세스를 거부하고 경고 메시지를 기록하지 않습니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 사용합니다.

예제

- 사용자 admin1 은 터미널 tty30 에 대한 소유자와 기본 액세스 권한을 변경하고 주중에 일반 근무 시간(오전 8 시에서 오후 6 시까지) 동안에만 터미널을 사용할 수 있도록 제한하려고 합니다.
  - 사용자 admin1 이 ADMIN 특성을 가지고 있습니다.

```
chres TERMINAL tty30 owner(admin1) defaccess(read) restrictions \
(days(weekdays)time(0800:1800))
```

- 관리자 Sally 는 account.txt 파일의 FILE 클래스 레코드에 저장된 group 및 owner 속성을 제거하려고 합니다.
  - 사용자 Sally 는 Jared 의 사용자 레코드의 소유자입니다.

```
chres FILE /account.txt group() owner()
```

속성이 문자열에 의해 정의된 경우 레코드 속성을 제거하려면 "-" 기호 또는 빈 괄호 "()"와 함께 속성을 입력하십시오.

- 사용자 Bob 은 터미널 tty190 의 comment 필드를 삭제하는 동시에 터미널 액세스가 허가될 때마다 알림을 받으려고 합니다.
  - 사용자 Bob 은 CA Access Control 사용자이며 터미널 tty190 의 소유자입니다.

```
chres TERMINAL tty190 comment notify(Bob@athena)
```

- 사용자 Admin1 은 SURROGATE 클래스에 있는 USER.root 리소스의 보안 범주 목록에 OPERATOR 범주를 추가하려고 합니다.
  - 사용자 Admin1 이 ADMIN 특성을 가지고 있습니다.
  - OPERATOR 범주가 데이터베이스에 정의되어 있습니다.

```
chres SURROGATE USER.root category(OPERATOR)
```

- 사용자 admin1 은 /bin/su 를 EXECUTE 의 전역 액세스 권한을 가진 트러스트된 프로그램으로 정의하려고 합니다.
  - 사용자 admin1 이 ADMIN 특성을 가지고 있습니다.
  - 다음과 같은 기본값이 적용됩니다.
    - restrictions(days(anyday) time(anytime))
    - owner(admin1)
    - audit(failure)

```
newres PROGRAM /bin/su defaccess(x) trust
```

- 사용자 admin1 은 "system" 그룹의 그룹 ID 대체를, admin1 을 비롯한 어떤 사용자도 액세스할 수 없는 보호되는 리소스로 정의하려고 합니다.
  - 사용자 admin1 이 ADMIN 특성을 가지고 있습니다. 사용자 nobody 가 CA Access Control 에 정의되어 있습니다.
  - 다음과 같은 기본값이 적용됩니다.
    - restrictions(days(anyday) time(anytime))
    - audit(failure)

```
newres SURROGATE GROUP.system defaccess(n) owner(nobody)
```

- 사용자 SecAdmin 은 터미널 T1, T8 및 T11 을 포함하는 터미널 그룹 ProjATerms 를 정의하려고 합니다. 터미널 그룹은 PROJECTA 그룹에 의해서만 사용될 수 있으며, 사용 시간은 주중 일반 근무 시간(오전 8:00 부터 오후 6:00 까지)으로 제한됩니다.
  - 사용자 SecAdmin 은 ADMIN 특성을 가집니다.
  - 터미널 T1, T8 및 T11 이 CA Access Control 에 정의되어 있습니다.
  - PROJECTA 그룹이 CA Access Control 에 정의되어 있습니다.
  - audit(failure)

```
newres GTERMINAL ProjATerms mem(T1,T8,T11) owner(PROJECTA) \
restrictions(days(weekdays) time(0800:1800)) defaccess(n)
```

추가 정보:

rmres 명령-리소스 삭제(페이지 133)
showres 명령-리소스 속성 표시(페이지 150)
권한 부여 명령-리소스에 대한 액세스 권한 설정(페이지 46)
chres 명령-Windows 리소스 수정(페이지 187)
find 명령-데이터베이스 레코드 나열(페이지 116)
CONTAINER 클래스(페이지 249)
클래스별 액세스 권한(페이지 29)

ch[x]usr 명령-사용자 속성 변경

AC 환경에 해당

사용자 속성을 변경하고 필요한 경우 CA Access Control 데이터베이스에서 사용자 레코드를 정의하려면 chusr, chxusr, editusr, editxusr, newusr 및 newxusr 명령을 사용하십시오.

이러한 명령은 모두 다음과 같은 동의어를 갖습니다.

- chusr-cu
- chxusr-cxu
- editusr-eu
- editxusr-exu
- newusr-nu
- newxusr-nxu

예를 들어, cu 명령은 chusr 명령과 같다는 의미입니다.

이 모든 명령은 구조가 동일하며 범위만 다릅니다. 이러한 명령를 다음과 같이 사용하십시오.

- chusr, editusr 및 newusr 명령은 내부 사용자에 대해 사용합니다. 이러한 명령의 차이점은 다음과 같습니다.
  - chusr 명령은 하나 이상의 USER 레코드를 수정합니다.
  - editusr 명령은 하나 이상의 USER 레코드를 작성 또는 수정합니다.
  - newusr 명령은 하나 이상의 USER 레코드를 작성합니다.

  참고: 이러한 명령은 네이티브 환경에도 있지만 작동 방식이 다릅니다.
- chxusr, editxusr 및 newxusr 명령은 엔터프라이즈 사용자에 대해 사용합니다. 이러한 명령의 차이점은 다음과 같습니다.
  - chxusr 명령은 하나 이상의 XUSER 레코드를 수정합니다.
  - editxusr 명령은 하나 이상의 XUSER 레코드를 작성 또는 수정합니다.
  - newxusr 명령은 하나 이상의 XUSER 레코드를 작성합니다.

USER 및 XUSER 클래스 레코드는 모든 속성에 대해 동일합니다. 단, 엔터프라이즈 사용자 저장소에서 속성이 정의된 경우 XUSER 레코드가 이러한 속성을 다시 정의하지 않습니다.

이러한 명령을 실행하면, 해당 사용자가 현재 시스템에 로그인한 상태일지라도, 변경 사항이 사용자 레코드를 즉시 수정합니다.

필요한 권한

이러한 명령을 실행하기 위해 필요한 권한 수준은 지정할 매개 변수에 따라 다릅니다. 다음 규칙이 적용됩니다.

- ADMIN 특성을 가진 경우 audit 을 제외한 모든 매개 변수를 지정할 수 있습니다.
- AUDITOR 특성을 가지고 있는 사용자는 audit 매개 변수를 지정할 수 있습니다. auditor[-] 매개 변수를 사용하려면 ADMIN 특성이 필요합니다.
- 기존 레코드를 업데이트할 때 사용자 레코드의 소유자는 admin, auditor, server, operator, pwmanager 를 제외한 모든 매개변수를 지정할 수 있습니다. 보안 범주가 소유자의 사용자 레코드에 있는 경우 소유자는 보안 범주를 사용자 레코드에 할당할 수 있습니다. 보안 레이블이 소유자의 사용자 레코드에서 할당된 경우 소유자는 보안 레이블을 사용자 레코드에 할당할 수 있습니다. 사용자 레코드의 소유자는 소유자의 사용자 레코드에 할당된 보안 수준보다 작거나 같은 보안 수준을 할당할 수 있습니다.
- 사용자 레코드가 GROUP-ADMIN 속성을 가진 그룹의 범위 내에 속하면 레코드 소유자와 동일한 권한을 가지게 됩니다.
- 사용자 레코드가 GROUP-AUDITOR 속성을 가진 그룹의 범위 내에 속하면 audit 매개 변수를 지정할 수 있습니다.
- ADMIN 클래스에 있는 USER 레코드의 액세스 제어 목록에서 MODIFY 특성(ch[x]usr 에 대해) 또는 CREATE 특성(edit[x]usr 에 대해)을 가지고 있는 사용자는 사용자 레코드의 소유자와 동일한 권한을 갖습니다.

```
{{chusr|cu}|chxusr|cxu}|{editusr|eu}|{editxusr|eu}|{newusr|nu}| {newxusr|nxu}} \
    {userName|(userName [,userName...])} \
    [{admin | admin-}] \
    [audit({none | all | {[success][failure][loginsuccess]|[loginfail]|[trace]}})]
    \
    [{auditor | auditor-}] \
    [{category(categoryName) | category-(categoryName)}] \
    [{comment(string) | comment-}] \
    [country(string)] \
    [email(emailAddress)] \
    [enable] \
    epwasown(password) \
    [{expire[(date)] | expire-}] \
    [fullname (fullName)]
    [{gowner(groupName)] \
    [{grace(nLogins) | grace-}] \
    [{ign_hol | ign_hol-}] \
    [{inactive(nDays) | inactive-}] \
    [{interval(nDays) | interval-}] \
    [{label(labelName) | label-}] \
    [{level(number) | level-}] \
```

```
[location(string)] \
[{logical|logical-}] \
[{maxlogins(nLogins) | maxlogins-}] \
[{min_life(nDays) | min_life-}] \
[{notify(mailAddress) | notify-}] \
[{operator | operator-}] \
[organization(string)] \
[org_unit(string) \
[owner({userName | groupName})] \
[password(string)] \
[phone(string)] \
[{pmdb(pmdbName) | pmdb-}] \
[{profile(groupName) | profile-}] \
[pwasown(string)] \
[{pwmanager | pwmanager-}] \
[regular] \
[{restrictions( \
    [days({anyday|weekdays|[mon] [tue] [wed] [thu] [fri] [sat] [sun]})] \
    [time({anytime|startTime:endTime})]
    ) |restrictions-}] \
[{resume[(date)] | resume-}] \
[{server | server-}] \
[{suspend[(date)] | suspend-}] \
[nt|nt( ] \
    [admin|admin-] \
    [comment('comment')|comment- ] \
    [country('country-name')] \
    [expire|expire(mm/dd/yy[@hh:mm])|expire-] \
    [flags({account-flags)|-account-flags})] \
    [homedir(any-string)] \
    [homedrive(home-drive)] \
    [location(any-string)] \
    [logonserver(server-name)] \
    [name(full-name)] \
    [organization(name)] \
    [org_unit(name)] \
    [password(user's temporary password)] \
    [pgroup(primary-group)] \
    [phone(any-string)] \
    [privileges(privilege-list)] \
    [restrictions(days(day-data) time(hhmm:hhmm|anytime) )] \
    [script(logon-script-path)] \
    [workstation(workstation-list)] )] \
[unix({ [gecos(string)] \
    [homedir(path)] \
    [pgroup(groupName)] \
    [shellprog(fileName)] \
    [userid(number)]}]
```

admin

ADMIN 특성을 사용자에게 할당합니다. ADMIN 특성이 있는 사용자는 audit 매개 변수를 제외한 모든 매개 변수와 함께 모든 selang 명령을 실행할 수 있습니다. admin- 매개 변수를 사용하려면 ADMIN 특성이 있어야 합니다.

admin-

사용자로부터 ADMIN 특성을 제거합니다. CA Access Control 은 적어도 한 사용자는 ADMIN 특성을 갖도록 합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

audit

CA Access Control 로 보호되는 리소스에 대한 사용자 활동 중 어떤 것을 감사 로그에 기록할 것인지를 지정합니다. 둘 이상의 이벤트 유형을 지정할 경우 이벤트 유형 이름을 공백 또는 쉼표로 구분하십시오. 감사 특성은 다음과 같습니다.

- **all-**CA Access Control 은 모든 사용자 활동을 기록합니다. 모니터링되는 활동은 failure, loginfail, loginsuccess, success, 및 trace 입니다.
- **failure**-CA Access Control 은 실패한 액세스 시도를 기록합니다.
- **loginfail**-CA Access Control 은 실패한 로그인 시도를 기록합니다.
- **loginsuccess**-CA Access Control 은 성공한 로그인을 기록합니다.
- **none-**CA Access Control 은 사용자 활동을 기록하지 않습니다.
- **success**-는 성공한 액세스를 기록합니다.
- **trace**-CA Access Control 은 이러한 사용자의 활동으로 인해 추적 파일에 나타나는 모든 메시지를 기록합니다.

auditor

AUDITOR 특성을 사용자에게 할당합니다. AUDITOR 속성을 가진 사용자는 시스템 리소스 사용에 대한 감사를 수행하고 CA Access Control 권한 검사 및 데이터베이스 액세스 중에 CA Access Control 에서 보호되는 리소스에 대한 액세스가 발견될 경우 로그 파일 작성을 제어할 수 있습니다. AUDITOR 특성을 가진 사용자에게 부여된 권한에 대한 자세한 내용은 해당 OS 용 끝점 관리 안내서를 참조하십시오.

auditor-

사용자 레코드에서 AUDITOR 특성을 제거합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

auth_type

인증 방식을 지정합니다.

SSO 에서만 사용됩니다.

엔터프라이즈 사용자에 대해서는 이 매개 변수를 사용할 수 없습니다.

category(categoryName[, categoryName...])

하나 이상의 보안 범주를 사용자에게 할당합니다.

category-(categoryName[, categoryName...])

사용자 레코드에서 보안 범주를 하나 이상 삭제합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

comment(commentString)

사용자 레코드에 설명을 할당합니다.

commentString

설명을 지정합니다. commentString 은 최대 255 자의 영숫자 문자열입니다. commentString 에 공백이 포함되어 있으면 앞뒤에 작은따옴표를 입력하십시오.

comment-

사용자 레코드에서 설명을 삭제합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

country(countryName)

사용자가 있는 국가를 지정합니다. 권한 부여 프로세스 중에는 국가가 사용되지 않습니다.

countryName

국가를 정의합니다. 최대 19 자의 영숫자 문자열입니다. 문자열에 공백이 포함되어 있으면 앞뒤에 작은따옴표를 입력하십시오.

email(emailAddress)

사용자의 전자 메일 주소를 정의합니다.

emailAddress

사용자의 전자 메일 주소를 정의합니다.

제한: 최대 128 자

enable

비활성화된 사용자의 로그인을 활성화합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

epwasown(password)

마치 사용자가 자신의 암호를 바꾸는 것처럼 사용자 암호를 변경합니다. 이 암호 변경은 관리 변경이 아니므로 암호가 자동으로 만료되지 않습니다.

expire(dateTime)

사용자 계정이 만료되는 날짜를 설정합니다. 날짜를 지정하지 않는 경우, 사용자가 로그인하거나 로그아웃할 때 계정이 즉시 만료됩니다.

사용자 레코드가 이 속성에 대한 값을 가지고 있는 경우, 해당 값이 GROUP 레코드의 값보다 우선합니다.

참고: 만료된 사용자 레코드를 활성화하려면 expire- 매개 변수를 사용하십시오. 이 작업을 위해 resume 매개 변수를 사용하지 마십시오.

dateTime

날짜를 정의하고, 선택적으로 시간을 정의합니다. 형식은 다음과 같습니다.
mm/dd/[yy]yy[@HH:MM]

연도를 지정하는 데 두 자리 또는 네 자리 숫자를 사용할 수 있습니다.

expire-

new[x]usr 명령에서는 만료일이 없는 사용자 계정을 정의합니다.

ch[x]usr 및 edit[x]usr 명령에서는 사용자 계정에서 만료일을 제거합니다.

flags(accountFlags|-accountFlags)

사용자 계정의 특정 특성을 지정합니다. 유효한 플래그 값의 목록을 보려면 부록 "Windows 값"을 참조하십시오.

사용자 레코드에서 플래그를 제거하려면 빼기 부호(-)를 accountFlags 앞에 붙이십시오.

fullname(fullName)

사용자의 전체 이름을 지정합니다.

fullName

전체 이름을 정의합니다. 최대 47 자의 영숫자 문자열입니다. fullName 에 공백이 있으면 전체 문자열 앞뒤에 작은따옴표를 입력하십시오.

gecos(string)

사용자에 대한 설명 문자열을 지정합니다. 문자열을 작은따옴표로 둘러쌉니다.

gowner(groupName)

CA Access Control 그룹을 사용자 레코드의 소유자로 할당합니다. 그룹 소유자의 보안 수준 및 보안 범주 권한이 충분한 경우 사용자 레코드의 그룹 소유자는 레코드에 제한 없이 액세스할 수 있습니다. 사용자 레코드의 그룹 소유자는 항상 사용자 레코드를 업데이트하고 삭제할 수 있는 권한을 가집니다.

grace(nLogins)

사용자에게 허용되는 유예 로그인의 수를 정의합니다.

유예 로그인 횟수에 도달하고 나면 사용자는 시스템에 액세스할 수 없으므로 시스템 관리자에게 연락하여 새 암호를 선택해야 합니다. 유예 로그인 횟수가 0 으로 설정된 경우, 사용자는 로그인할 수 없습니다.

사용자 레코드에 이 매개 변수에 대한 값이 포함될 경우, 해당 값은 GROUP 레코드의 값보다 우선됩니다.

이 매개 변수를 지정하지 않고 사용자가 이 매개 변수에 대한 값이 있는 프로필 그룹을 가진 경우, GROUP 레코드에 있는 값이 사용됩니다. USER 와 GROUP 레코드에 모두 값이 없는 경우, CA Access Control 전역 유예 로그인 설정이 사용됩니다.

nLogins

유예 로그인 횟수를 정의합니다. 0 - 255 사이의 정수를 입력하십시오.

grace-

사용자의 유예 로그인 설정을 삭제합니다. CA Access Control 전역 유예 로그인 설정이 대신 사용됩니다.

이 매개 변수는 newusr 명령과 함께 사용할 수 없습니다.

homedir(path)

사용자 홈 디렉터리의 전체 경로를 지정합니다. path 가 슬래시로 끝나면 CA Access Control 은 userName 을 경로에 연결합니다.

homedrive(drive)

사용자 홈 디렉터리의 드라이브를 지정합니다.

ign_hol

사용자에게 IGN_HOL 특성을 할당합니다. IGN_HOL 특성을 가진 사용자는 휴일 레코드에 정의된 기간 중에는 언제든지 로그인할 수 있습니다.

ign_hol-

사용자로부터 IGN_HOL 특성을 제거합니다.

inactive(nDays)

시스템에서 사용자가 비활성 상태로 전환되기 전까지 대기하는 일 수를 지정합니다. 지정된 일 수에 도달하면 사용자는 로그인할 수 없습니다.

참고: 비활성 사용자는 사용자 레코드에서 표시되지 않습니다. 비활성 사용자를 식별하려면 마지막 액세스 시간 값을 비활성 기간 값과 비교해야 합니다.

nDays

날짜 수를 정의합니다. nDays 는 영(0) 또는 양의 정수입니다. nDays 가 0 인 경우 inactive- 매개 변수를 사용한 것과 효과가 동일합니다.

inactive-

사용자의 상태를 비활성에서 활성으로 변경합니다.

이 매개 변수는 newusr 명령과 함께 사용할 수 없습니다.

interval(nDays)

사용자에게 새 암호를 묻기 전에 암호가 설정되거나 변경된 후 경과해야 하는 날짜 수를 정의합니다. 영(0) 또는 양의 정수를 입력하십시오. nDays 가 0 이면 CA Access Control 은 암호 간격 검사를 비활성화하며 암호는 만료되지 않습니다. 즉, setoptions 명령으로 설정한 기본값이 사용되지 않습니다. 보안 요구 사항이 낮은 사용자에 대해서만 nDays 를 0 으로 설정하십시오.

nDays 에 도달하면 CA Access Control 은 암호가 만료되었음을 사용자에게 알립니다. 유예 로그인 횟수에 도달할 때까지 사용자는 계속해서 암호를 사용할 수 있습니다. 유예 로그인 횟수에 도달하면 시스템에 대한 사용자 액세스가 거부되므로, 시스템 관리자에게 연락하여 새 암호를 요청해야 합니다.

interval-

사용자의 암호 사용 기간 설정을 취소합니다. 사용자에게 이 매개 변수 값을 가진 프로필 그룹이 있으면 그 값이 사용됩니다. 그렇지 않은 경우 setoptions 명령에서 설정된 기본값이 사용됩니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

label(labelName)

사용자에 보안 레이블을 할당합니다.

label-

사용자 레코드에서 보안 레이블을 삭제합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

level(levelNumber)

사용자 레코드에 보안 수준을 할당합니다.

levelNumber 는 0 - 255 사이의 정수입니다.

level-

사용자 레코드에서 보안 수준을 삭제합니다.

이 매개 변수는 newusr 명령과 함께 사용할 수 없습니다.

localapps

CA SSO 에서 사용됩니다.

location(locationString)

사용자의 위치를 지정합니다. 권한 부여 프로세스 중에는 위치가 사용되지 않습니다.

locationString

위치를 정의합니다. locationString 은 최대 47 자의 영숫자 문자열입니다. locationString 에 공백이 포함되어 있으면 앞뒤에 작은따옴표를 입력하십시오.

logical

사용자에게 LOGICAL 특성을 할당합니다. LOGICAL 특성이 있는 사용자는 로그인할 수 없으며 내부 CA Access Control 용도로만 사용됩니다.

예를 들어, 리소스의 소유자조차 리소스에 액세스할 수 없도록 하기 위해 리소스의 소유자로 사용할 수 있는 사용자 nobody 는 기본적으로 논리적 사용자입니다. 즉, 이 계정을 사용하여 어떤 사용자도 로그인할 수 없습니다.

logical-

사용자의 LOGICAL 특성을 제거합니다.

logonserver(server-name)

사용자에 대한 로그인 정보를 확인하는 서버를 지정합니다. 사용자가 도메인 워크스테이션에 로그인하면 CA Access Control 은 로그인 정보를 서버로 전송하며, 이 서버는 사용자가 작업할 수 있도록 워크스테이션 권한을 부여합니다.

maxlogins(nLogins)

사용자에 대한 최대 동시 로그인 수를 설정합니다. 값 0 은 사용자가 터미널 수에 상관없이 동시에 로그인할 수 있음을 의미합니다. 이 매개 변수를 지정하지 않을 경우 전역 최대 로그인 설정이 사용됩니다.

참고: maxlogins 가 1 로 설정되어 있으면 selang 을 실행할 수 없습니다. CA Access Control 을 종료하고, maxlogins 설정을 1 보다 큰 값으로 변경하고(예: setpropadm 유틸리티 사용), CA Access Control 을 다시 시작해야 합니다.

maxlogins-

사용자의 최대 로그인 설정을 삭제합니다. 전역 설정이 대신 사용됩니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

min_life(nDays)

암호를 다시 변경하기 위해 대기해야 하는 최소 일 수입니다. 양수를 입력합니다.

min_life-

사용자의 min_life 설정을 삭제합니다. 사용자에게 이 매개 변수 값을 가진 프로필 그룹이 있으면 그 값이 사용됩니다. 그렇지 않은 경우 setoptions 명령에서 설정된 기본값이 사용됩니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

nochngpass

사용자가 다른 사용자의 암호를 변경하지 못하도록 지정합니다.

notify(notifyAddress)

사용자가 로그인할 때마다 notifyAddress 로 전자 메일을 보냅니다. 알림 메시지의 수신자는 각 메시지에 설명된 인증되지 않은 액세스 시도에 대응하기 위해 자주 로그인해야 합니다.

CA Access Control 은 알림 메시지를 전송할 때마다 감사 로그에 감사 레코드를 기록합니다.

notifyAddress

사용자 이름 또는 전자 메일 주소를 정의합니다.

제한: 30 자.

notify-

사용자가 로그인할 때 아무에게도 알리지 않도록 지정합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

nt

chusr 및 editusr 명령에서 이 매개 변수는 로컬 Windows 시스템에 있는 사용자의 정의를 변경합니다.

newusr 명령에서 이 매개 변수는 사용자를 로컬 Windows 시스템에 추가합니다.

둘 이상의 인수를 지정할 경우 공백으로 인수를 구분합니다.

CA Access Control 내의 로컬 Windows 시스템에서 작동하는 방법에 대한 자세한 내용은 environment 명령을 참조하십시오.

엔터프라이즈 사용자에 대해서는 nt 옵션 및 nt 옵션 아래의 하위 옵션이 유효하지 않습니다.

operator

사용자에게 OPERATOR 특성을 할당합니다. OPERATOR 특성을 가진 사용자는 데이터베이스에 있는 모든 리소스 레코드를 나열할 수 있으며, CA Access Control 에서 정의한 모든 파일에 대한 읽기 권한을 갖고 있습니다.

이 특성을 가진 사용자는 secons 명령의 모든 옵션도 사용할 수 있습니다. secons 유틸리티에 대한 자세한 내용은 참조 안내서를 참조하십시오.

operator-

사용자 레코드에서 OPERATOR 특성을 제거합니다.

이 매개 변수는 newusr 명령과 함께 사용할 수 없습니다.

organization(organizationString)

사용자의 조직을 지정합니다. 권한 부여 프로세스 중에는 조직이 사용되지 않습니다.

organizationString

조직을 정의합니다. organizationString 은 최대 255 자의 영숫자 문자열입니다. organizationString 에 공백이 포함되어 있으면 앞뒤에 작은따옴표를 입력하십시오.

org_unit(org_unitString)

사용자의 조직 단위를 지정합니다. 권한 부여 프로세스 중에는 조직 단위가 사용되지 않습니다.

org_unitString

조직 단위를 정의합니다. org_unitString 은 최대 255 자의 영숫자 문자열입니다. organizationString 에 공백이 포함되어 있으면 앞뒤에 작은따옴표를 입력하십시오.

owner(Name)

CA Access Control 사용자나 그룹을 사용자 레코드의 소유자로 할당합니다. 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

password(string)

사용자에게 암호를 할당합니다. 공백 또는 쉼표를 제외한 문자를 지정하십시오. 암호 검사가 활성화된 경우 암호는 하나의 로그인에 대해서만 유효합니다. 사용자가 다음번에 시스템에 로그인할 때는 새 암호를 설정해야 합니다.

자신의 암호를 변경하려면 setoptions cng_ownpwd 를 사용하여 selang 옵션을 설정하거나 sepass 를 사용해야 합니다.

pgroup(groupName)

사용자의 주 그룹 ID 를 설정합니다. groupName 은 UNIX 그룹의 이름입니다.

phone(phoneString)

사용자의 전화 번호를 정의합니다. 권한 부여 프로세스 중에는 전화 번호가 사용되지 않습니다.

phoneString

전화 번호를 정의합니다. phoneString 은 최대 255 자의 영숫자 문자열입니다. phoneString 에 공백이 포함되어 있으면 앞뒤에 작은따옴표를 입력하십시오.

pmdb(pmdbName)

사용자가 sepass 유틸리티를 사용하여 암호를 변경하면 새 암호는 지정된 PMDB 로 전파됩니다. PMDB 의 완전한 이름을 입력합니다. 이 암호는 seos.ini 의 [seos] 섹션에 있는 parent_pmd 토큰이나 passwd_pmd 토큰에 정의된 정책 모델로 전달되지 않습니다.

이 옵션은 엔터프라이즈 사용자에게 사용할 수 없습니다.

pmdb-

사용자 레코드에서 PMDB 특성을 삭제합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

privileges(privilege-list)

Windows 사용자 레코드에 특정 권한을 추가하거나, privList 앞에 빼기 부호(-)가 있는 경우 지정된 권한을 제거합니다.

이 매개 변수는 newusr 명령과 함께 사용할 수 없습니다.

profile(groupName)

사용자를 프로필 그룹에 할당합니다. 프로필 그룹에서 다음 값을 사용할 수 있습니다.

- audit
- auth_type
- expire 를
- grace
- inactive
- interval
- maxlogins
- min_life
- password rules
- pmdb
- pwd_autogen
- pwd_policy
- pwd_sync
- restrictions (days, time)
- resume
- suspend
- unix (homedir, shellprog)

profile-

프로필 그룹에서 사용자를 제거합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

pwmanager

사용자에게 PWMANAGER 특성을 할당합니다. 이 특성을 가진 사용자는 데이터베이스에서 사용자의 암호를 변경할 수 있습니다. 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

pwmanager-

사용자 레코드에서 PWMANAGER 특성을 제거합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

pwasown(string)

사용자가 암호를 변경한 것처럼 암호를 대체합니다. 이 매개 변수를 지정하면 데이터베이스의 마지막 변경 시간과 날짜가 업데이트됩니다. 유예 로그인은 종료됩니다.

regular

레코드의 OBJ_TYPE 속성을 다시 설정하고, 이에 따라 사용자로부터 권한 특성을 제거합니다.

restrictions([Days] [Time])

사용자가 로그인할 수 있는 요일 및 시간을 지정합니다. restrictions 는 [X]USER 레코드의 DAYTIME 속성에 저장됩니다.

Days 를 생략하고 Time 을 지정하면 레코드에 이미 정의된 모든 요일 제한에 시간 제한이 적용됩니다.

Time 을 생략하고 Days 를 지정하면 레코드에 이미 정의된 시간 제한에 Days 제한이 적용됩니다.

Days 와 Time 을 모두 지정하면 사용자는 지정된 요일에 지정된 시간 동안에만 시스템에 액세스할 수 있습니다.

Days

사용자가 로그인할 수 있는 요일을 지정합니다. Days 를 지정할 때 다음 키워드를 사용할 수 있습니다.

- **anyday**-사용자는 모든 요일에 파일에 액세스할 수 있습니다.
- **weekdays**-사용자는 주중(월요일부터 금요일까지)에만 리소스에 액세스할 수 있습니다.
- **Mon**, **Tue**, **Wed**, **Thu**, **Fri**, **Sat**, **Sun**-사용자는 지정된 요일에만 리소스에 액세스할 수 있습니다. 순서에 상관 없이 요일을 지정할 수 있습니다. 둘 이상의 요일을 지정하려면 공백 또는 쉼표로 날짜를 구분합니다.

Time

사용자가 로그인할 수 있는 기간을 지정합니다. time 인수는 다음 하위 인수를 가집니다.

- **anytime**-사용자는 하루 중 언제든지 리소스에 액세스할 수 있습니다.
- startTime**:**endTime-사용자는 지정된 기간 동안에만 리소스에 액세스할 수 있습니다.

  startTime 및 endTime 형식은 모두 hhmm 이며, 여기서 hh 는 24 시간 표기(00 - 23) 시간이고 mm 은 분 표기(00 - 59)입니다. 2400 은 올바른 시간 값이 아닙니다. 대신 0000 을 사용하십시오.

  startTime 은 endTime 보다 작아야 합니다.

  참고: CA Access Control 은 프로세서의 시간대를 사용합니다. 사용자가 프로세서와 다른 시간대에 터미널에 로그인할 수 있으므로, 이런 상황을 고려해야 합니다.

restrictions-([days] [time])

사용자의 로그인 능력을 제한하는 모든 제한을 삭제합니다.

resume([dateTime])

suspend 매개 변수를 지정하여 비활성화된 사용자 레코드를 활성화합니다. suspend 매개 변수와 resume 매개 변수를 모두 지정한 경우 계속 날짜는 일시 중지 날짜 이후여야 합니다. dateTime 을 생략하면 chusr 명령을 실행하는 즉시 사용자 레코드가 다시 시작됩니다. 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

dateTime 은 [m]m/[d]d/yy[@HH:MM] 형식으로 입력합니다.

resume-

사용된 경우, 계속 날짜와 시간을 사용자 레코드에서 삭제합니다. 그 결과, 사용자의 상태가 활성(활성화됨)에서 일시 중지됨으로 변경됩니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

script(logon-script-path)

사용자가 로그인할 때 자동으로 실행되는 파일의 위치를 지정합니다. 이 매개 변수는 선택 사항입니다. 일반적으로 이 로그인 스크립트는 작업 환경을 구성합니다. 사용자의 작업 환경을 설정하기 위해 profile 매개 변수를 사용할 수도 있습니다.

server

SERVER 특성을 설정합니다. 이 특성을 사용하면 현재 사용자 대신에 실행 중인 프로세스가 다른 사용자에 대한 권한 부여를 요청할 수 있습니다. 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

이 옵션은 엔터프라이즈 사용자에게 사용할 수 없습니다.

server-

SERVER 특성을 해제합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

shellprog(fileName)

사용자가 login 또는 su 명령을 호출한 후 실행되는 초기 프로그램 또는 셸의 전체 경로를 지정합니다. fileName 은 문자열입니다.

이 옵션은 엔터프라이즈 사용자에게 사용할 수 없습니다.

suspend([dateTime])

사용자 레코드를 비활성화하지만 데이터베이스에 정의된 상태를 유지합니다. 사용자는 비활성화된 사용자 계정을 사용하여 시스템에 로그인할 수 없습니다.

dateTime 이 지정되어 있으면 사용자 레코드는 지정된 날짜에 비활성화됩니다. dateTime 을 생략하면 ch[x]usr 명령을 실행하는 즉시 사용자 레코드가 비활성화됩니다.

dateTime 을 mm/dd/yy[@HH:MM] 형식으로 입력하십시오.

suspend-

사용자 레코드에서 일시 중지 날짜를 삭제하고 사용자 상태를 비활성에서 활성으로 변경합니다.

이 매개 변수는 new[x]usr 명령과 함께 사용할 수 없습니다.

unix

chusr 명령과 editusr 명령에서 이 매개 변수는 로컬 UNIX 시스템에 있는 사용자 정의를 변경합니다.

newusr 명령의 경우 이 매개 변수는 사용자를 로컬 UNIX 시스템에 추가합니다.

둘 이상의 인수를 지정할 경우 공백으로 인수를 구분합니다.

CA Access Control 내의 로컬 UNIX 시스템에서 작동하는 방법에 대한 자세한 내용은 이 장의 environment 명령을 참조하십시오.

엔터프라이즈 사용자에 대해서는 unix 옵션 및 unix 옵션 아래의 하위 옵션이 유효하지 않습니다.

userid(number)

사용자의 고유한 숫자 ID(UID)를 설정합니다. UID 는 고유한 임의 액세스 제어에 사용됩니다. number 는 십진수입니다. 기본적으로 100 보다 작은 숫자는 허용되지 않습니다. 제외된 숫자에 대한 자세한 내용은 부록 참조 안내서의 AllowedGidRange 토큰을 참조하십시오.

userName|(userName [,userName...])

사용자의 이름을 정의합니다. 각 사용자 이름은 고유해야 합니다.

newusr 명령을 사용하면 userName 은 CA Access Control 에 대해 새 사용자를 식별합니다. newusr 명령을 사용할 때 사용자가 네이티브 환경에 이미 정의되어 있으면, CA Access Control 은 해당 사용자에 대한 USER 레코드로서 이 username 을 사용합니다. 그러나 일반적으로 엔터프라이즈 사용자에 대해서는 CA Access Control 기능을 사용해야 하며, 이미 네이티브 환경에 있는 username 에 대해서는 USER 레코드 작성을 위해 newusr 을 사용하지 않아야 합니다. 대신, 해당 사용자의 CA Access Control 속성을 변경하려면 chgxusr 명령을 사용하십시오.

때로는 기본 로그인 이름이 아닌 CA Access Control 사용자 이름을 원할 수 있습니다. 이 경우, login 명령은 해당 사용자에게 작업 능력을 제공할 수 없지만 sesu 와 같은 다른 명령은 가능할 수 있습니다.

참고: 사용자 이름에 백슬래시를 포함할 수 있는 UNIX 에서는 userName 지정 시 두 개의 백슬래시를 사용하십시오.

예제

- 사용자 Bob 은 FINANCIAL 카테고리를 Jim 의 레코드에 추가하고 Jim 의 보안 레벨을 155 로 변경하고 Jim 의 시스템 접근 권한을 주중 오전 8:00 부터 오후 8:00 사이로 제한하려고 합니다.
  - 사용자 Bob 이 ADMIN 특성을 가지고 있습니다.
  - 사용자 Jim 이 CA Access Control 에 정의되어 있습니다.
  - FINANCIAL 범주가 CA Access Control 에 정의되어 있습니다.

```
chuxsr Jim category(FINANCIAL) level(155) restrictions \
(days(weekdays)time(0800:2000))
```

- 사용자 "admin"은 1995 년 8 월 5 일부터 3 주 동안 휴가를 갈 사용자 Joel 의 액세스 권한을 일시 중지하려고 합니다.
  - 사용자 admin 이 ADMIN 특성을 가지고 있습니다.
  - 사용자 Joel 이 CA Access Control 에 정의되어 있습니다.
  - 오늘의 날짜는 1994 년 8 월 3 일입니다.

```
chxusr Joel suspend(8/5/95) resume(8/26/95)
```

- 사용자 Security2 는 사용자 Bill 로부터 AUDITOR 특성을 제거하고 Bill 의 모든 활동을 감사하고자 합니다.
  - 사용자 Security2 는 ADMIN 및 AUDITOR 특성을 가지고 있습니다.
  - 사용자 Bill 이 CA Access Control 에 정의되어 있습니다.

```
chxusr Bill auditor audit(all)
```

- 사용자 Rob 은 사용자 Mary 의 레코드에 저장된 설명을 변경하고자 합니다.
  - 사용자 Rob 은 Mary 의 사용자 레코드의 소유자입니다.

```
chxusr Mary comment ('Administrator of the SALES group')
```

- admin 사용자 Sally 는 사용자 Jared 의 레코드에 저장된 국가 이름 및 위치 속성을 제거하고자 합니다.
  - 사용자 Sally 는 Jared 의 사용자 레코드의 소유자입니다.

```
chxusr Jared country() location()
```

- 사용자 Bob 은 사용자 Peter 와 Joe 를 CA Access Control 에 대해 정의하고자 합니다.
  - 사용자 Bob 이 ADMIN 특성을 가지고 있습니다.
  - 사용자 Peter 와 Joe 는 CA Access Control 에 정의되어 있지 않습니다.
  - 다음과 같은 기본값이 적용됩니다.
    - owner(Bob)
    - audit(failure,loginfailure)

```
newusr (Peter Joe)
```

- 사용자 Bob 은 사용자 Jane 을 CA Access Control 에 대해 정의하고 "payroll"을 소유 그룹으로 할당하고자 합니다.
  - 사용자 Bob 이 ADMIN 특성을 가지고 있습니다.
  - 사용자 Jane 은 CA Access Control 에 정의되어 있지 않습니다.
  - 사용자 Jane 의 이름은 JG Harris 입니다.
  - audit(failure,loginfailure)

```
newusr Jane owner(payroll) name('J.G. Harris')
```

- 사용자 Bob 은 사용자 JohnD 를 보안 범주 NewEmployee 와 보안 수준 3 으로 CA Access Control 에 대해 정의하고자 합니다. JohnD 는 평일 오전 8 시에서 오후 6 시까지만 시스템을 사용하도록 허용됩니다.
  - 사용자 Bob 이 ADMIN 특성을 가지고 있습니다.
  - NewEmployee 범주가 CA Access Control 에 정의되어 있습니다.
  - 새 사용자의 실제 이름은 John Doe 입니다.
  - 다음과 같은 기본값이 적용됩니다.
    - owner(Bob)
    - audit(failure)

```
newusr JohnD name('John Doe') category(NewEmployee) level(3) \
restrictions(days(weekdays) time(0800:1800))
```

deploy 명령-정책 배포 시작

AC 환경에 해당

정책 배포를 초기화하려면 deploy 명령을 사용하십시오. 이 명령은 배포하는 POLICY 개체와 연결된 RULESET 개체에 저장된 selang 명령을 실행합니다. 이들은 정책 배포 명령입니다.

중요! 저장된 정책을 배포할 경우에는 policydeploy 유틸리티를 사용할 것을 적극 권장합니다. deploy 명령은 정책 배포의 일부만 실행하며, 정책을 끝점에 배포할 때 DMS 를 업데이트하지 않습니다.

deploy 명령을 실행하려면 다음이 필요합니다.

- 정책을 배포하는 데이터베이스 아래 계층에 있는 각 데이터베이스에서 POLICY, HNODE 및 RULESET 클래스에 대한 하위 관리 권한.
- 정책을 배포하는 데이터베이스 아래 계층에 있는 각 데이터베이스에 대한 하위 관리 권한.

  이러한 권한은 각 해당 컴퓨터에서 정책을 구성하는 selang 명령을 실행하는 데 필요합니다.

  예를 들어, 다음과 같이 새 파일 리소스를 작성하는 경우에는 FILE 클래스에 대한 하위 관리 권한이 필요합니다.

```
nr FILE /inetpub/* defaccess(none)
```

참고: 정책 배포에 대한 자세한 내용은 엔터프라이즈 관리 안내서를 참조하십시오.

이 명령의 형식은 다음과 같습니다.

```
deploy POLICY name#xx
```

name#xx

배포하려는 정책에 대한 POLICY 개체의 이름(정책 이름 및 버전 번호)

deploy- 명령-정책 제거 시작

AC 환경에 해당

정책 배포 취소를 시작하려면 deploy-(또는 undeploy) 명령을 사용하십시오. 이 명령은 배포하는 POLICY 개체와 연결된 RULESET 개체에 저장된 selang 명령을 실행합니다. 이들은 정책 배포 취소 명령입니다.

중요! 정책의 배포를 취소할 경우에는 policydeploy 유틸리티를 사용할 것을 적극 권장합니다. deploy- 명령은 정책 배포 취소의 일부만 실행하며, 끝점에서 정책의 배포를 취소할 때 DMS 를 업데이트하지 않습니다.

이 명령을 실행하려면 다음이 필요합니다.

- 정책의 배포를 취소하는 데이터베이스 아래 계층에 있는 각 데이터베이스에서 POLICY, HNODE 및 RULESET 클래스에 대한 하위 관리 권한.
- 정책의 배포를 취소하는 데이터베이스 아래 계층에 있는 각 데이터베이스에 대한 하위 관리 권한.

  이러한 권한은 각 해당 컴퓨터에서 정책 배포 취소 스크립트를 구성하는 selang 명령을 실행하는 데 필요합니다.

참고: 정책 배포에 대한 자세한 내용은 엔터프라이즈 관리 안내서를 참조하십시오.

이 명령의 형식은 다음과 같습니다.

```
{deploy-|undeploy} POLICY name#xx
```

name#xx

배포를 취소하려는 정책에 대한 POLICY 개체의 이름(정책 이름 및 버전 번호)

editfile 명령-파일 레코드의 작성 및 수정

AC 환경에 해당

이 명령은 chfile 명령과 함께 설명되어 있습니다.

추가 정보:

chfile 명령-파일 레코드 수정(페이지 58)

edit[x]grp 명령-그룹 레코드의 작성 및 수정

AC 환경에 해당

이 명령은 ch[x]grp 명령과 함께 설명되어 있습니다.

추가 정보:

ch[x]grp 명령-그룹 속성 변경(페이지 64)

editres 명령-리소스 레코드 수정

AC 환경에 해당

이 명령은 chres 명령과 함께 설명되어 있습니다.

추가 정보:

chres 명령-리소스 레코드 수정(페이지 77)

edit[x]usr 명령-사용자 레코드 수정

AC 환경에 해당

이 명령은 chxusr 명령과 함께 설명되어 있습니다.

추가 정보:

ch[x]usr 명령-사용자 속성 변경(페이지 94)

end_transaction 명령-이중 컨트롤 트랜잭션 기록 완료

AC 환경의 UNIX 호스트에 해당

end_transaction 명령은 이중 컨트롤 PMDB 프로세스에 대한 start_transaction 명령을 완료합니다.

environment 명령-보안 환경 설정

모든 환경에 해당

environment 명령은 보안 환경을 설정합니다. CA Access Control 은 CA Access Control 및 UNIX 보안 환경을 지원합니다. selang 명령 셸을 호출하면 기본적으로 AC 환경이 선택됩니다.

이 명령의 형식은 다음과 같습니다.

```
environment {ac|config|etrust|native|nt|pmd|seos|unix}
```

ac

CA Access Control 보안 환경을 지정합니다. selang 명령은 로컬 CA Access Control 데이터베이스에 영향을 줍니다. 일부 명령은 연결된 호스트의 네이티브 OS 보안 설정에 대한 동시 업데이트를 지원합니다. CA Access Control 환경에서 selang 프롬프트는 다음과 같습니다.

```
AC>
```

config

끝점 구성 설정의 변경을 허용하는 원격 구성 환경을 지정합니다.

etrust

CA Access Control 보안 환경을 지정합니다.

참고: 이것은 AC 를 지정하는 것과 같으며, 이전 버전과의 호환성을 위해 유지됩니다.

native

연결된 로컬 또는 원격 호스트의 기본 운영 체제 보안 환경(Windows 또는 UNIX)을 지정합니다. selang 명령은 네이티브 OS 데이터베이스에 영향을 미칩니다. 네이티브 환경에서 selang 프롬프트는

```
AC(native)>
```

nt

Windows 보안 환경을 지정합니다. selang 명령은 Windows 데이터베이스에 영향을 미칩니다. 일부 명령은 CA Access Control 보안 설정에 대한 동시 업데이트를 지원합니다. Windows 환경에서 selang 프롬프트는

```
AC(nt)>
```

pmd

원격 관리 환경에 selang 명령을 지정합니다. selang 명령 셸을 pmd 환경으로 설정하면 이 명령은 선택된 호스트의 PMDB 에서 작동합니다. pmd 환경에서 selang 프롬프트는 다음과 같습니다.

```
AC(pmd)>
```

seos

CA Access Control 보안 환경을 지정합니다.

참고: 이것은 AC 를 지정하는 것과 같으며, 이전 버전과의 호환성을 위해 유지됩니다.

unix

UNIX 보안 환경을 지정합니다. selang 명령은 UNIX 보안 시스템에서 작동합니다. UNIX 환경에서 selang 프롬프트는 다음과 같습니다.

```
AC(unix)>
```

find 명령-데이터베이스 레코드 나열

AC 및 네이티브 환경에 해당

find 명령은 지정된 클래스에 있는 레코드의 이름을 나열합니다. 매개 변수를 지정하지 않으면 모든 클래스의 이름이 표시됩니다.

참고: find 명령은 list 및 search 명령과 동일합니다.

이 명령을 사용하려면 다음 조건으로 정의된 충분한 권한이 있어야 합니다.

- ADMIN, AUDITOR 또는 OPERATOR 특성을 가진 경우, find 명령을 모든 매개 변수와 함께 사용할 수 있습니다.
- ADMIN 클래스의 레코드에 대해 READ 권한이 있는 경우, 레코드로 표시되는 클래스에 대해 클래스 매개 변수를 지정할 수 있습니다.

이 명령의 형식은 다음과 같습니다.

```
{find|f|list|search} [{className|class(className)} [objName]]
```

className

find 명령으로 레코드를 검색할 클래스를 지정합니다. className 을 제공하지 않으면 find 명령은 모든 클래스를 나열합니다.

objName

CA Access Control 이 검색할 레코드를 지정합니다. objName 에는 와일드카드 문자를 포함할 수 있습니다.

예: TERMINAL 클래스의 모든 레코드 표시

TERMINAL 클래스의 모든 구성원을 표시하려면 다음 명령을 입력합니다.

```
find terminal
```

get dbexport Command - 내보낸 데이터베이스 규칙 검색

AC 환경에 해당

get dbexport 명령은 연결된 호스트에 있는 CA Access Control 데이터베이스 또는 PMD 데이터베이스에서 내보낸 규칙을 검색합니다. 내보낸 데이터베이스가 존재하려면 get dbexport 명령을 실행하기 전에 start dbexport 명령을 실행해야 합니다.

이 명령의 형식은 다음과 같습니다.

```
get dbexport [pmdname(name)] [params(OFFSET=number)]
```

pmdname(name)

(선택 사항) 내보낸 PMD 데이터베이스의 이름을 정의합니다.

params(OFFSET=number)

(선택 사항) 데이터베이스 출력에서 더 많은 행을 검색하기 위한 오프셋을 정의합니다. 각 요청마다 get dbexport 명령은 내보낸 데이터베이스에서 200 개의 행만 반환합니다. 출력에 더 많은 정보가 있으면 마지막 반환 행을 지정하는 오프셋 데이터가 반환됩니다.

예: 내보낸 데이터베이스에서 규칙 반환

다음 예제는 get dbexport 명령을 사용하여 연결된 호스트에 있는 내보낸 CA Access Control 데이터베이스에서 정보를 검색하는 방법을 보여 줍니다. 첫 번째 명령은 처음 200 개 행을 검색하고 두 번째 명령은 출력의 그 다음 200 개 행을 검색합니다.

```
AC > get dbexport
(localhost)
Data for DBEXPORT 'seosdb'
-------------------------------------------
setoptions class+(CLASS)
setoptions class+(CLASS)
setoptions class+(CLASS)
...
chres CLASS ("resource") defaccess(none)
OFFSET: 201

AC> get dbexport params("offset=201")
(localhost)
Data for DBEXPORT 'seosdb'

-------------------------------------------
chres CLASS ("resource") defaccess(none)
chres CLASS ("resource") defaccess(none)
chres CLASS ("resource") defaccess(none)
...
chres CLASS ("resource") defaccess(none)
OFFSET: 401
```

추가 정보:

start dbexport 명령 - 데이터베이스 내보내기 시작(페이지 155)

get devcalc 명령-정책 위반 데이터 검색

AC 환경에 해당

get devcalc 명령은 정책 위반 계산 결과가 포함된 정책 위반 데이터 파일(deviation.dat)에서 정보를 검색하여 한 세트 이상의 DMS 데이터베이스로 결과를 보냅니다. 데이터 파일의 존재를 확인하려면 start devcalc 명령을 먼저 실행해야 합니다.

정책 또는 호스트 보고서를 작성할 때 위반 계산 결과를 포함하도록 지정할 수도 있습니다. 그러면 보고 유틸리티가 이 명령을 실행합니다.

중요! 위반 계산에서는 기본 규칙이 적용되는지 여부를 확인하지 않습니다. 또한 데이터베이스에서 개체(사용자 또는 개체 특성, 사용자 또는 리소스 권한 부여, 실제 사용자 또는 리소스)를 제거하는 규칙을 무시합니다. 예를 들어 위반 계산에서는 다음 규칙이 적용되는지 여부를 확인할 수 없습니다. rr SUDO admCommand

참고: 정책 위반 데이터 파일 및 고급 정책 보고에 대한 자세한 내용은 엔터프라이즈 관리 안내서를 참조하십시오.

get devcalc 명령을 실행하려면 컴퓨터에 대한 터미널 액세스 권한 및 DEVCALC 하위 관리 클래스에 대한 읽기 액세스 권한이 있어야 합니다.

이 명령의 형식은 다음과 같습니다.

```
get devcalc [params("offset=number")]
```

offset=number

(선택 사항) 정책 위반 데이터 파일에서 더 많은 행을 검색하기 위한 오프셋을 정의합니다. get devcalc 명령은 정책 위반 데이터 파일에서 요청당 최대 행 수(max_lines_request 구성 설정으로 지정됨)만 반환합니다. 파일에 더 많은 정보가 있으면 마지막 반환 행을 지정하는 오프셋 데이터가 반환됩니다.

예: 정책 위반 데이터 얻기

다음 예는 max_lines_request 가 10 으로 설정되었을 때 정책 위반 데이터 파일에서 정보를 검색하기 위해 get devcalc 명령을 사용하는 방법을 보여줍니다. 첫 번째 명령은 처음 10 개 행을 검색하고 두 번째 명령은 출력의 그 다음 10 개 행을 검색합니다.

```
AC> get devcalc
(localhost)
Data for DEVCALC 'deviation'
-------------------------------------------
DATA    : DATE, Mon Mar 20 11:22:15 2006
POLICYSTART, myPolicy#01
DIFF, (FILE), (file1), (*), (*)
DIFF, (FILE), (file2), (*), (*)
DIFF, (FILE), (file3), (*), (*)
DIFF, (FILE), (file4), (*), (*)
DIFF, (FILE), (file5), (*), (*)
DIFF, (FILE), (file6), (*), (*)
DIFF, (FILE), (file7), (*), (*)
OFFSET  : 11

AC> get devcalc params("offset=11")
(localhost)
Data for DEVCALC 'deviation'
-------------------------------------------
DATA    : DIFF, (FILE), (file8), (*), (*)
DIFF, (FILE), (file9), (*), (*)
DIFF, (FILE), (file10), (*), (*)
DIFF, (FILE), (file11), (*), (*)
DIFF, (FILE), (file12), (*), (*)
DIFF, (FILE), (file13), (*), (*)
DIFF, (FILE), (file14), (*), (*)
DIFF, (FILE), (file15), (*), (*)
DIFF, (FILE), (file16), (*), (*)
DIFF, (FILE), (file17), (*), (*)
OFFSET  : 21
```

추가 정보:

start devcalc 명령-정책 위반 계산 시작(페이지 156)
setoptions 명령-CA Access Control 옵션 설정(페이지 138)

help 명령-selang 도움말 이용

모든 환경에 해당

help 명령은 selang 구문을 다음과 같은 여러 가지 방법으로 표시합니다.

- 매개 변수 없이 사용하면, selang 명령 목록을 각 명령에 대한 간략한 설명과 함께 표시합니다.
- selang 명령 이름과 함께 사용하면, 주어진 명령의 구문을 표시합니다
- access 매개 변수와 함께 사용하면, 권한 부여 명령의 access 매개 변수와 new*, ch* 및 edit* 명령의 defaccess 매개 변수에 대한 값의 목록을 표시합니다.
- lineedit 매개 변수와 함께 사용하면, selang 명령줄 조작을 위한 특수 문자 목록을 표시합니다.

참고: 명령줄의 텍스트를 삭제하지 않고 명령줄에 입력한 명령의 도움말 텍스트를 표시하려면 Ctrl+2 를 입력하십시오.

```
{help|h} [commandName|access|lineedit|className|properties|privilege]
```

access

access 매개 변수와 defaccess 매개 변수가 지정할 수 있는 접근 유형의 클래스별 목록을 요청합니다.

className

지정한 클래스에 대한 간단한 설명을 요청합니다.

command-name

지정된 명령의 구문을 요청합니다.

lineedit

selang 명령 라인 처리를 위한 특수 문자 목록을 요청합니다.

properties

(AC 환경) 사용자 정의 속성을 업데이트하는 방법에 대한 정보를 요청합니다.

privilege

(Windows 환경) ch[x]grp, ch[x]usr, edit[x]grp 및 edit[x]usr 명령에 대한 가능한 Windows 권한 목록을 요청합니다.

추가 정보:

selang 명령 참조(페이지 39)
selang 환경(페이지 34)

selang 도움말 보기(페이지 37)

history 명령-이전에 실행한 명령 표시

모든 환경에 해당

history 명령은 현재 selang 명령 셸 세션 도중에 입력한 모든 명령을 나열합니다. 명령은 시간 순으로 나열되고, 각 명령의 앞에는 명령의 번호가 표시됩니다. 예를 들어, 세 번째로 입력한 명령 앞에는 번호 3 이 옵니다.

ch[x]usr, new[x]usr 또는 edit[x]usr 명령의 일부로 암호를 입력했더라도 history 명령으로 암호를 표시할 수는 없습니다. history 명령은 일반 텍스트 암호 대신 일련의 별표(***)를 표시합니다.

이 명령의 형식은 다음과 같습니다.

```
history
```

추가 정보:

명령 기록(페이지 23)

hosts 명령-원격 CA Access Control 터미널에 연결

모든 환경에 해당

hosts 명령은 selang 명령을 보내는 호스트 또는 정책 모델을 지정합니다. 이 명령을 사용하면 다른 이름으로 원격 CA Access Control 컴퓨터에 연결하여, 로컬 CA Access Control 서비스가 실행되고 있는 동안 원격으로 컴퓨터를 관리할 수 있습니다. 기본적으로 모든 selang 명령은 로컬 호스트의 데이터베이스로 전달됩니다.

hosts 명령은 호스트로 보내야 하는 명령을 실행하기 전에 실행해야 합니다.

로컬 호스트에서 원격 호스트 데이터베이스를 관리(업데이트)하려면 다음 조건 중 하나가 충족되어야 합니다.

- 로컬 데이터베이스에서 원격 호스트 데이터베이스를 업데이트할 명시적인 권한을 갖고 있음
- 로컬 데이터베이스에서 원격 호스트 데이터베이스를 업데이트하도록 허용된 그룹의 구성원임
- 원격 호스트에서 정의된 로컬 호스트의 소유자임

현재 사용할 수 있는 모든 호스트와 PMDB 를 나열하려면 매개 변수 없이 hosts 명령을 지정합니다.

참고: CA Access Control 은 별칭이 아니라 정규 호스트 이름을 통해 호스트를 보호합니다. 별칭 이름으로 인한 혼동을 방지하기 위해 CA Access Control 은 별칭 이름에 대해 HOST 규칙이 정의될 때 경고를 내보냅니다. 마찬가지로, 정규화된 이름보다 부족한 이름으로 HOST 가 정의될 경우에도 CA Access Control 은 경고를 내보내는데, 이는 CA Access Control 이 호스트에 대해 정규화된 이름(예: mymachine.yourcompany.com)을 사용하기 때문입니다.

이 명령의 형식은 다음과 같습니다.

```
hosts [{systemIds|policyModel@[hostname]}]
```

systemIds

selang 명령을 실행할 호스트의 시스템 ID 를 지정합니다. 둘 이상의 호스트를 지정하려면 괄호 안에 시스템 ID 목록을 넣고 시스템 ID 를 공백이나 쉼표로 구분하십시오.

policyModel@[hostname]

selang 명령을 실행할 정책 모델의 주소를 지정합니다. 둘 이상의 정책 모델을 지정하려면 괄호 안에 정책 모델 주소 목록을 넣고 정책 모델 주소를 공백 또는 쉼표로 구분하십시오.

hostname 을 지정하지 않으면 CA Access Control 은 로컬 호스트의 PMDB 에 연결하려고 시도합니다.

참고: 정책 모델을 사용하는 것이 호스트를 명시적으로 지정하는 것보다 유리한 점은, 정책 모델이 있는 시스템은 정책 모델에 정의된 모든 시스템을(현재 사용할 수 없더라도) 계속 업데이트하려고 한다는 점입니다. 정책 모델에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

예: 사용자 또는 그룹이 원격 호스트를 업데이트하도록 허용

사용자에게 로컬 데이터베이스로부터 원격 호스트 데이터베이스를 업데이트할 권한을 부여하려면 원격 호스트에서 다음 명령을 입력합니다.

```
authorize TERMINAL local_host uid user_name access(write)
```

로컬 데이터베이스에서 원격 호스트 데이터베이스를 업데이트할 수 있는 권한을 그룹에 부여하려면 원격 호스트에서 다음 명령을 입력합니다.

```
authorize TERMINAL local_host gid(group_name) access(write)
```

예: 원격 정책 모델에 selang 명령 적용

모든 후속 명령을 h1 스테이션에 있는 정책 모델에 적용하려면 다음 명령을 입력합니다.

```
hosts Policy@h1
```

Policy@h1 과 성공적으로 연결되면 다음 메시지가 나타납니다.

```
Successfully connected to h1
```

이제부터 입력하는 모든 명령은 로컬 호스트가 아닌 Policy@h1 로 전송됩니다. selang 프롬프트는 다음과 같이 변경됩니다.

```
Remote_AC>
```

예: 원격 호스트에 selang 명령 적용

이후의 모든 명령을 athena 스테이션에 적용하려면 다음 명령을 입력합니다.

```
hosts athena
```

athena 와 성공적으로 연결되면 다음 메시지가 화면에 나타납니다.

```
(athena)
연결했습니다.
INFO: Target version is 2.50
```

입력하는 모든 명령은 athena 에 적용되고 로컬 호스트로 전송되지 않습니다. 새 사용자를 추가할 경우, 사용자는 다음 예제와 같이 athena 에만 추가됩니다.

```
Remote_AC>newusr steve
(athena) USER steve successfully added.
```

join[x] 명령-내부 그룹에 사용자 추가

AC 환경에 해당

join[x] 명령은 사용자를 하나 이상의 내부 그룹에 추가하거나, 그룹과 관련하여 사용자의 속성을 변경합니다. 지정된 사용자 및 그룹이 이미 CA Access Control 에 정의되어 있어야 합니다.

내부 사용자를 그룹에 추가하려면 join 을 사용하십시오.

엔터프라이즈 사용자를 그룹에 추가하려면 joinx 를 사용하십시오.

참고: 이 명령은 네이티브 환경에도 있지만 작동 방식이 다릅니다.

join 명령에서 얻어진 모든 속성 집합은 지정된 그룹에 있는 지정된 사용자에 대한 이전 속성 집합을 완전히 대체합니다. 해당 속성을 이전에 정의한 경우 새 join 명령이 해당 속성을 다시 지정하지 않으면 속성은 저장되지 않습니다.

참고: 그룹 속성에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오.

최소한 다음 조건 중 하나를 충족할 경우 join 명령을 사용할 수 있습니다.

- 사용자가 ADMIN 특성을 가집니다.

  참고: CA Access Control GROUP 레코드 및 엔터프라이즈 그룹을 수정하려면 MODIFY 및 JOIN 액세스 권한이 모두 필요합니다.
- 그룹 레코드는 GROUP-ADMIN 특성을 가진 그룹 범위 내에 있습니다.
- 그룹의 소유자입니다.
- ADMIN 클래스에 있는 GROUP 레코드의 액세스 제어 목록에 CONNECT 권한이 할당되어 있습니다.

이 명령의 형식은 다음과 같습니다.

```
{join[x]|j[x]} {userName|(userName [,userName...])} \
    group(groupName [,groupName...]) \
    [admin|admin-] \
    [auditor|auditor-] \
    [gowner(group-name)] \
    [operator|operator-] \
    [owner(userName|groupName)] \
    [pwmanager | pwmanager-] \
    [regular] \
    [nt | unix]
```

admin

userName 으로 지정된 사용자에게 GROUP-ADMIN 속성을 할당합니다.

admin-

사용자에게서 GROUP-ADMIN 속성을 삭제합니다.

auditor

userName 으로 지정된 사용자에게 GROUP-AUDIT 속성을 할당합니다.

auditor-

사용자에게서 GROUP-AUDIT 속성을 삭제합니다.

gowner(groupName)

사용자가 groupName 그룹에 추가되는 중임을 지정합니다.

group(groupName [,groupName...])

사용자를 구성원으로서 추가할 그룹을 지정합니다.

nt

userName 을 Windows 데이터베이스에 있는 그룹과 연결합니다.

operator

userName 으로 지정된 사용자에게 GROUP-OPERATOR 속성을 할당합니다.

operator-

사용자에게서 GROUP-OPERATOR 속성을 삭제합니다.

owner(Name)

CA Access Control 사용자나 그룹을 join 레코드의 소유자로 지정합니다. 연결을 설정하면서 소유자를 지정하지 않으면 사용자가 연결의 소유자가 됩니다.

pwmanager

GROUP-PWMANAGER 속성을 userName 에서 지정된 사용자에게 할당합니다.

regular

사용자에 대해 관리 플래그를 다시 설정합니다.

unix

userName 을 UNIX 보안 시스템의 그룹에 연결합니다.

userName

group 매개 변수로 지정된 그룹에 연결(또는 새로운 속성 집합으로 다시 연결)할 사용자를 지정합니다.

명령이 join 이면 userName 은 USER 레코드의 이름입니다. 명령이 joinx 이면 userName 은 엔터프라이즈 사용자의 이름입니다.

예제

- 사용자 Rorri 는 사용자 Bob 을 내부 그룹 staff 에 조인하고자 합니다.
  - Rorri 는 ADMIN 특성을 가집니다.
  - 다음과 같은 기본값이 적용됩니다.
    - admin
    - auditor
    - owner(Rorri)
    - pwmanager

  ```
  join Bob group(staff)
  ```

- 사용자 Rorri 는 그룹 staff 에서 Sue 의 정의를 변경하려고 합니다. Sue 는 현재 GROUP-AUDITOR 이며 Rorri 는 GROUP-PWMANAGER 특성을 추가하려고 합니다.
  - Rorri 는 ADMIN 특성을 가집니다.
  - 다음과 같은 기본값이 적용됩니다.
    - admin
    - owner(Rorri)

  ```
  join Sue group(staff) auditor pwmanager
  ```

  selang 에서 이 명령을 실행하면 이전 레코드가 삭제됩니다. Sue 의 이전 특성에 대한 레코드가 남아 있지 않습니다. 따라서 Rorri 는 Sue 가 현재 가져야 하는 두 개의 특성을 지정해야 합니다.

추가 정보:

join[x]- 명령-그룹에서 사용자 제거(페이지 128)
show[x]grp 명령-그룹 속성 표시(페이지 148)
show[x]usr 명령-사용자 속성 표시(페이지 152)

join[x]- 명령-그룹에서 사용자 제거

AC 환경에 해당

join[x]- 명령은 내부 그룹에서 사용자를 제거합니다.

join- 명령은 내부 그룹에서 내부 사용자를 제거합니다.

joinx- 명령은 내부 그룹에서 엔터프라이즈 사용자를 제거합니다.

참고: join[-] 명령은 네이티브 환경에도 있지만 작동 방식이 다릅니다.

join[x]- 명령을 사용하려면 다음 조건 중 하나를 충족해야 합니다.

- 사용자가 ADMIN 특성을 가집니다.

  참고: CA Access Control GROUP 레코드 및 네이티브 그룹을 수정하려면 MODIFY 및 JOIN 액세스 권한이 모두 필요합니다.
- 그룹 레코드는 GROUP-ADMIN 특성을 가진 그룹 범위 내에 있습니다.
- 그룹의 소유자입니다.
- ADMIN 클래스에 있는 GROUP 레코드의 액세스 제어 목록에 CONNECT 권한이 할당되어 있습니다.

이 명령의 형식은 다음과 같습니다.

```
{join[x]-|j[x]-} {userName|(userName [,userName...])} \
    group(groupName [,groupName...])
```

group(groupName [,groupName...])

사용자를 제거할 그룹을 지정합니다.

userName

그룹에서 제거할 사용자를 지정합니다.

명령이 join 이면 userName 은 USER 레코드의 이름입니다.

명령이 joinx 이면 userName 은 엔터프라이즈 사용자의 이름입니다.

예제

사용자 Bill 은 사용자 sales25 및 sales43 을 PAYROLL 그룹에서 제거하고자 합니다.

사용자 Bill 이 ADMIN 특성을 가지고 있습니다.

```
joinx- (sales25 sales43) group(PAYROLL)
```

추가 정보:

join[x] 명령-내부 그룹에 사용자 추가(페이지 125)
show[x]grp 명령-그룹 속성 표시(페이지 148)
show[x]usr 명령-사용자 속성 표시(페이지 152)

list 명령-데이터베이스 레코드 나열

AC 및 네이티브 환경에 해당

이것은 find 명령과 동일합니다.

추가 정보:

find 명령-데이터베이스 레코드 나열(페이지 116)

newfile 명령-파일 레코드 작성

AC 환경에 해당

이 명령은 chfile 명령과 함께 설명되어 있습니다.

추가 정보:

chfile 명령-파일 레코드 수정(페이지 58)

new[x]grp 명령-그룹 레코드 작성

AC 환경에 해당

이 명령은 chgrp 명령과 함께 설명되어 있습니다.

추가 정보:

ch[x]grp 명령-그룹 속성 변경(페이지 64)

newres 명령-리소스 레코드 작성

AC 환경에 해당

이 명령은 chres 명령과 함께 설명되어 있습니다.

추가 정보:

chres 명령-리소스 레코드 수정(페이지 77)

new[x]usr 명령-사용자 레코드 작성

AC 환경에 해당

이 명령은 ch[x]usr 명령과 함께 설명되어 있습니다.

추가 정보:

ch[x]usr 명령-사용자 속성 변경(페이지 94)

rename 명령-데이터베이스 레코드 이름 변경

AC 환경에 해당

데이터베이스에서 레코드의 이름을 바꿉니다. 이름을 바꾼 레코드는 새 이름으로만 인식됩니다.

참고: SEOS, UACC 및 ADMIN 클래스에 있는 레코드의 이름은 바꿀 수 없습니다.

rename 명령을 사용하려면 레코드에 대한 충분한 권한이 있어야 합니다. 다음의 조건 중 하나가 충족될 때까지 CA Access Control 은 다음 검사를 수행합니다.

- 사용자가 ADMIN 특성을 가집니다.
- 리소스 레코드는 사용자가 GROUP-ADMIN 속성을 가지는 그룹의 범위 내에 있습니다.
- 사용자가 레코드의 소유자입니다.
- ADMIN 클래스에 있는 리소스 클래스 레코드의 액세스 제어 목록에서 CREATE(editres 의 경우) 액세스 권한이 할당되었습니다.

이 명령의 형식은 다음과 같습니다.

```
rename className oldresourceName newresourceName
```

className

이름을 바꾸려는 레코드가 속한 클래스를 정의합니다.

oldresourceName

CA Access Control 에 있는 레코드의 현재 이름을 정의합니다.

newresourceName

레코드에 할당할 새 이름을 정의합니다.

예제

ADMIN 1 사용자가 HOST 클래스의 spree3 레코드를 spree4 로 이름을 변경하려고 합니다.

- 보안 관리자는 ADMIN 특성을 가지고 있습니다.

```
rename host spree3 spree4
```

rmfile 명령-파일 레코드 삭제

AC 환경에 해당

rmfile 명령은 FILE 클래스에 속한 레코드를 데이터베이스에서 삭제합니다.

다음 조건 중 하나가 충족될 경우 파일 레코드를 삭제할 수 있습니다.

- 사용자가 ADMIN 특성을 가집니다.
- 현재 가지고 있는 GROUP-ADMIN 특성이 속한 그룹의 범위 내에 레코드가 있습니다.
- 파일의 소유자입니다.
- ADMIN 클래스에 있는 FILE 레코드의 ACL 에 DELETE 액세스 권한이 할당되어 있습니다.

이 명령의 형식은 다음과 같습니다.

```
{rmfile|rf} {fileName | (filename [,filename...])}
```

fileName

제거할 파일을 정의합니다.

CA Access Control 은 각 파일 레코드를 독립적으로 처리합니다. 파일을 처리할 때 오류가 발생하면 CA Access Control 은 메시지를 작성하고 목록의 다음 파일을 계속 처리합니다.

예: 파일 보호 제거

ADMIN 특성을 갖는 보안 관리자가 파일에 대한 CA Access Control 보호를 제거하고자 합니다. UNIX 에서 사용하는 명령은 다음과 같습니다.

```
rmfile /etc/passwd
```

Windows 에서 사용하는 동일한 명령은 다음과 같습니다.

```
rmfile C:\temp\passwords.txt
```

추가 정보:

chfile 명령-파일 레코드 수정(페이지 58)
showfile 명령-파일 속성 표시(페이지 146)

rm[x]grp 명령-그룹 레코드 삭제

AC 환경에 해당

rmgrp 및 rmxgrp 명령은 하나 이상의 그룹을 CA Access Control 에서, 그리고 선택적으로 네이티브 환경에서 제거합니다.

참고: rmgrp 명령으로 삭제할 수 없는 경우가 그룹의 그룹 ID 데이터베이스에 있을 수 있습니다. 예를 들어 그룹이 다른 그룹의 소유자이거나, 다른 레코드의 소유자이거나, 리소스의 액세스 제어 목록에 있을 수 있습니다. chgrp, chusr, chres 및 권한 부여 명령을 필요에 따라 사용하여 소유권을 수동으로 변경하고 삭제하려는 그룹 레코드와 관련된 액세스 권한을 제거하십시오. 또는, sepurgedb 유틸리티를 사용하여 데이터베이스에 존재하는 불일치 상태를 자동으로 정리하십시오.

참고: rmgrp 명령은 네이티브 환경에도 있지만 작동 방식이 다릅니다.

rmgrp 명령을 사용하려면 적어도 다음 중 하나가 필요합니다.

- 사용자가 ADMIN 특성을 가집니다.
- 현재 가지고 있는 GROUP-ADMIN 특성이 속한 그룹의 범위 내에 삭제할 그룹이 있습니다.
- 삭제할 그룹의 소유자입니다.
- AUDIT 클래스의 GROUP 레코드에 DELETE 권한이 할당되어 있습니다.

이 명령의 형식은 다음과 같습니다.

```
{rmgrp|rg | rmxgrp|rxg} { groupName | (groupName [,groupName...]) } [unix|nt]
```

groupName

삭제할 CA Access Control 그룹을 지정합니다.

nt

(선택 사항) 그룹을 CA Access Control 데이터베이스에서 삭제할 뿐만 아니라 로컬 Windows 데이터베이스에서도 삭제합니다.

unix

(선택 사항) 그룹을 CA Access Control 데이터베이스에서 삭제할 뿐만 아니라 로컬 UNIX 시스템에서도 삭제합니다.

예제

사용자 Joe 는 데이터베이스에서 그룹 DEPT1 및 DEPT2 를 삭제하려고 합니다.

- 사용자 Joe 에게는 SALES 그룹에 대한 GROUP-ADMIN 권한이 있습니다.
- 그룹 DEPT1 및 DEPT2 는 SALES 그룹이 소유합니다.

```
rmxgrp (DEPT1, DEPT2)
```

추가 정보:

ch[x]grp 명령-그룹 속성 변경(페이지 64)
join[x] 명령-내부 그룹에 사용자 추가(페이지 125)
join[x]- 명령-그룹에서 사용자 제거(페이지 128)
show[x]grp 명령-그룹 속성 표시(페이지 148)
rmgrp 명령-UNIX 그룹 삭제(페이지 175)
rmgrp 명령-Windows 그룹 삭제(페이지 202)

rmres 명령-리소스 삭제

AC 환경에 해당

rmres 명령은 데이터베이스에서 리소스를 제거합니다. rmres 명령을 사용하여 ACVAR, ADMIN, APPL, CATEGORY, CONNECT, FILE, GAPPL, GHOST, GSUDO, GTERMINAL, HNODE, HOST, HOSTNET, HOSTNP, LOGINAPPL, MFTERMINAL, POLICY, PWPOLICY, SECFILE, SECLABEL, SPECIALPGM, SUDO, SURROGATE, TERMINAL, PROGRAM, PROCESS, RULESET, TCP, UACC 및 모든 사용자 정의 클래스에 속하는 레코드를 삭제할 수 있습니다.

참고: 이 명령은 네이티브 Windows 환경에도 있지만 작동 방식이 다릅니다.

데이터베이스에서 레코드를 제거하려면 다음 조건 중 하나를 충족해야 합니다.

- 사용자가 ADMIN 특성을 가집니다.
- 리소스 레코드는 사용자가 GROUP-ADMIN 속성을 가지는 그룹의 범위 내에 있습니다.
- 리소스 레코드의 소유자입니다.
- 클래스에 의한 액세스(ADMIN) 클래스에 있는 리소스 클래스 레코드의 액세스 제어 목록에 DELETE 권한이 할당되어 있는 경우.

이 명령의 형식은 다음과 같습니다.

```
{rmres|rr} className resourceName
```

className

리소스가 속한 클래스의 이름을 지정합니다. CA Access Control 에 정의된 리소스 클래스를 나열하려면 find 명령을 사용하십시오. 자세한 내용은 이 장에서 find 명령을 참조하십시오.

resourceName

삭제할 리소스 레코드의 이름을 지정합니다. 둘 이상의 리소스를 제거하려면 괄호 안에 리소스 이름 목록을 넣고 공백 또는 쉼표로 리소스 이름을 구분하십시오.

CA Access Control 은 각 리소스 레코드를 독립적으로 처리합니다. 리소스를 처리할 때 오류가 발생하면 CA Access Control 은 메시지를 작성하고 목록의 다음 리소스를 계속 처리합니다.

예제

사용자 Admin1 은 데이터베이스의 TERMINAL 클래스에서 TERMS 레코드를 제거하고자 합니다.

- 사용자 Admin1 이 ADMIN 특성을 가지고 있습니다.

```
rmres TERMINAL TERMS
```

추가 정보:

chres 명령-리소스 레코드 수정(페이지 77)
showres 명령-리소스 속성 표시(페이지 150)
rmres 명령-Windows 리소스 삭제(페이지 202)
find 명령-데이터베이스 레코드 나열(페이지 116)

rm[x]usr 명령-사용자 레코드 삭제

AC 환경에 해당

rmusr 및 rmxusr 명령은 CA Access Control 데이터베이스에서 사용자를 제거하며, CA Access Control 그룹 레코드에 있는 사용자 레코드에 대한 참조도 제거합니다.

rmxusr 명령은 CA Access Control 데이터베이스에서 엔터프라이즈 사용자를 제거합니다. rmusr 명령은 데이터베이스에서 내부 사용자를 제거합니다. rmusr 명령은 선택적으로 네이티브 환경에서 사용자를 제거할 수도 있습니다.

참고: rm[x]usr 명령으로 삭제할 수 없는 경우가 사용자 데이터베이스에 있을 수 있습니다. 예를 들어 사용자가 그룹의 소유자이거나, 다른 레코드의 소유자이거나, 리소스의 액세스 제어 목록에 있을 수 있습니다. ch[x]grp, ch[x]usr, ch[x]res 및 권한 부여 명령을 필요에 따라 사용하여 소유권을 수동으로 변경하고 삭제하려는 사용자 레코드와 관련된 액세스 권한을 제거하십시오. 또는, sepurgedb 유틸리티를 사용하여 데이터베이스에 존재하는 불일치 상태를 자동으로 정리하십시오.

참고: rmusr 명령은 네이티브 환경에도 있지만 작동 방식이 다릅니다.

rm[x]usr 명령을 실행하려면 다음 요구 사항 중 적어도 하나가 충족되어야 합니다.

- 사용자가 ADMIN 특성을 가집니다.
- 현재 가지고 있는 GROUP-ADMIN 특성이 속한 그룹의 범위 내에 삭제할 사용자 레코드가 있습니다.
- ADMIN 클래스에 있는 USER 레코드의 액세스 제어 목록에 DELETE 권한이 할당되어 있습니다.
- 사용자 레코드의 소유자입니다.

ru 는 rmusr 의 동의어입니다.

rxu 는 rmxusr 의 동의어입니다.

이 명령의 형식은 다음과 같습니다.

```
{rmusr|ru | rmxusr | rxu} { userName | (userName [,userName...]) } \
    [unix|nt] [appl(homedir=yes)]
```

appl(homedir=yes)

(UNIX 전용). 사용자의 홈 디렉터리를 삭제합니다.

이 인수는 /home, /tmp 또는 /users 에서 사용자 홈 디렉터리의 존재를 확인합니다. 홈 디렉터리가 다른 디렉터리에 있을 경우 S99DELETE_postrmusrdir.sh 스크립트를 편집하여 해당 디렉터리를 포함합니다.

참고: 이 옵션을 지정하기 전에 먼저 unix 옵션을 지정해야 합니다.

nt

사용자를 CA Access Control 에서 삭제할 뿐만 아니라 Windows 환경에서도 삭제합니다.

rmusr 에만 해당됩니다.

userName

사용자 레코드를 정의합니다.

unix

사용자를 CA Access Control 에서 삭제할 뿐만 아니라 UNIX 환경에서도 삭제합니다.

rmusr 에만 해당됩니다.

예제

다음 명령은 엔터프라이즈 사용자 Terry 와 Jane 을 CA Access Control 에서 삭제합니다.

```
rxu (Terry, Jane)
```

추가 정보:

ch[x]usr 명령-사용자 속성 변경(페이지 94)
show[x]usr 명령-사용자 속성 표시(페이지 152)
rmusr 명령-UNIX 사용자 삭제(페이지 176)
rmusr 명령-Windows 사용자 삭제(페이지 203)

ruler 명령-표시할 속성 선택

AC 및 네이티브 환경에 해당

ruler 명령은 클래스에 대해 ruler 를 정의하므로, 이에 따라 CA Access Control 이 표시하는 클래스의 속성 집합을 정의할 수 있습니다.

ruler 명령은 현재 세션의 호스트에만 적용됩니다. 각 호스트의 속성은 별도의 목록에 표시됩니다. 호스트를 변경할 경우, ruler 명령은 새 호스트에 있는 속성 표시를 변경하지 않습니다.

다음 사용자가 이 명령을 실행할 수 있습니다.

- ADMIN, AUDITOR 또는 OPERATOR 특성을 가진 사용자
- ruler 를 설정하는 클래스에 대한 ADMIN 클래스에서 읽기 액세스를 가진 사용자. 예를 들어, TERMINAL 클래스를 나타내는 레코드에 대한 ADMIN 클래스에서 읽기 액세스를 가진 경우 TERMINAL 클래스에 대한 ruler 를 설정할 수 있습니다.

이 명령의 형식은 다음과 같습니다.

```
ruler className [props( all| propertyName [,propertyName...])]
```

className

화면 표시를 변경하려는 클래스의 이름입니다.

[props(all | propertyName [,propertyName...])]

표시할 속성을 지정합니다.

props 매개 변수를 생략하면 CA Access Control 은 현재 ruler 에 있는 속성의 이름을 표시합니다.

all

클래스의 모든 속성이 표시되도록 지정합니다.

propName

CA Access Control 속성이 표시되도록 지정합니다. 최대 40 개의 속성을 공백 또는 쉼표로 구분하여 지정할 수 있습니다.

예제

- 사용자 admin 은 CA Access Control 에서 각 사용자에 대해 두 가지 속성 즉, 소유자 및 변경 사항을 알릴 사용자만 표시하고자 합니다.

  ```
  ruler USER props(NOTIFY, OWNER)
  ```

- 사용자 admin 은 USER 클래스에 대한 현재 ruler 의 속성을 표시하고자 합니다.

  ```
  ruler USER
  ```

- 사용자 admin 은 CA Access Control 을 기본 ruler 로 되돌려 클래스 USER 의 모든 속성을 표시하고자 합니다.

  ```
  ruler USER props(all)
  ```

추가 정보:

showfile 명령-파일 속성 표시(페이지 146)
show[x]grp 명령-그룹 속성 표시(페이지 148)
showres 명령-리소스 속성 표시(페이지 150)
show[x]usr 명령-사용자 속성 표시(페이지 152)

setoptions 명령-CA Access Control 옵션 설정

AC 환경에 해당

setoptions 명령은 실행 중인 시스템에서 시스템 전체 CA Access Control 옵션을 설정합니다. 예를 들면 setoptions 명령을 사용하여 각 클래스 또는 모든 클래스에 대한 보안 검사를 활성화 또는 비활성화하고, 암호 정책을 설정하고, CA Access Control 옵션의 현재 설정을 나열할 수 있습니다.

참고: 이 명령은 Windows 환경에도 있지만 작동 방식이 다릅니다.

setoptions 명령을 사용하려면 ADMIN 특성이 필요합니다. 단, setoptions list 명령을 사용하려면 AUDITOR 또는 OPERATOR 특성만 있어도 됩니다.

이 명령의 형식은 다음과 같습니다.

```
{setoptions|so} \
    [accgrr|accgrr-] \
    [accpacl|accpacl-] \
    [class+ (className)] \
    [class- (className)] \
    [class (className)] \
    [flags{+|-} (I|W)] \
    [cng_adminpwd|cng_adminpwd-] \
    [cng_ownpwd|cng_ownpwd-] \
    [cwarnlist] \
    [dms{+|-}(dms@hostname)] \
    [inactive(nDays)|inactive-] \
    [is_dms{+|-}] \
    [list] \
    [maxlogins(nLogins)|maxlogins-] \
    [password( \
        [{history(nStoredPasswords) | history-}] \
        [(interval(nDays) | interval-)] \
        [(min_life(nDays) | min_life-)] \
        [{rules( \
                [alpha(nCharacters)] \
                [alphanum(nCharacters)] \
                [(bidirectional) | (bidirectional-)] \
                [grace(nLogins)] \
                [lowercase(nCharacters)] \
                [min_len(nCharacters)]
                [max_len(nCharacters)] \
                [max_rep(nCharacters)] \
                [{namechk|namechk-}]
                [numeric(nCharacters)] \
                [{oldpwchk|oldpwchk-}]
                [prohibited(prohibitedCharacters)] \
                [special(nCharacters)] \
                [sub_str_len(nCharacters)] \
                [uppercase(nCharacters)] \
                [use_dbdict|use_dbdict-] \
        )|rules-}] \
    )] \
```

accgrr

ACCGRR(누적 그룹 권한) 옵션을 활성화합니다.

기본값은 enabled 입니다.

accgrr-

ACCGRR(누적 그룹 권한) 옵션을 비활성화합니다.

accpacl

모든 리소스에서 PACL 의 사용을 활성화합니다.

accpacl-

PACL 의 사용을 비활성화합니다.

class (className)

CA Access Control 클래스에 대한 설정을 설정 또는 삭제합니다.

class+(className)

하나 이상의 CA Access Control 클래스를 활성화합니다. CA Access Control 이 클래스의 리소스를 보호하려면 해당 클래스가 활성화되어야 합니다. 클래스에 속한 리소스에 대한 액세스를 허용하기 위해 필요한 레코드를 정의한 후에만 클래스를 활성화해야 합니다. CA Access Control 과 함께 제공되는 리소스 클래스에 대한 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

다음 값 중 하나를 사용하십시오.

- CA Access Control 클래스의 이름
- SECLEVEL. 보안 수준 검사를 활성화합니다.
- PASSWORD. 암호 규칙을 활성화합니다. Windows 에서는 임의로 긴 암호도 활성화합니다.

class-(className)

하나 이상의 CA Access Control 클래스를 비활성화합니다. 비활성화된 클래스에 속한 리소스는 CA Access Control 에 의해 보호되지 않습니다. 다음 값 중 하나를 사용하십시오.

- CA Access Control 클래스의 이름
- SECLEVEL. 보안 수준 검사를 비활성화합니다.
- PASSWORD. 암호 규칙을 비활성화합니다. Windows 에서 이 값은 긴 암호도 비활성화합니다.

GROUP, SECFILE, SEOS, UACC 및 USER 클래스는 비활성화할 수 없습니다.

cng_adminpwd

PWMANAGER 특성이 있는 사용자가 ADMIN 사용자의 암호를 변경할 수 있도록 합니다.

cng_adminpwd-

PWMANAGER 특성이 있는 사용자가 ADMIN 사용자의 암호를 변경할 수 없도록 합니다. 이것이 기본 설정입니다.

cng_ownpwd

사용자가 selang 을 통해 자신의 암호를 변경할 수 있습니다.

cng_ownpwd-

사용자가 selang 을 통해 자신의 암호를 변경할 수 없습니다. 이것이 기본 설정입니다.

cwarnlist

클래스가 경고 모드에 있는 데이터를 포함하는 테이블을 표시합니다.

dms{+|-}(dms@hostname)

이 데이터베이스에 대한 DMS 데이터베이스의 목록에서 DMS 데이터베이스를 추가 또는 제거합니다.

flags{+|-} (I|W)

클래스와 관련된 기능을 설정 또는 삭제합니다. 유효한 값:

I

지정된 클래스의 개체에 대한 대소문자 구분

W

지정된 클래스에 대한 경고 모드

참고: 플래그는 대소문자를 구분합니다. 대문자를 사용하십시오.

history(NStoredPasswords)

기록 목록에 저장되는 이전 암호 수를 지정합니다. 암호가 변경되면 이전 암호는 이 목록에 추가되며, 가장 오래된 암호는 필요에 따라 목록에서 제거됩니다. CA Access Control 은 사용자가 자신의 암호를 이 목록에 있는 암호로 변경하지 못하게 합니다.

1 에서 24 사이의 정수를 입력하십시오. 0 을 지정하면 암호가 하나도 저장되지 않습니다.

Windows 에서는 history 옵션이 8 자보다 긴 암호 사용을 활성화합니다. 암호 저장 시 사용되는 암호화 형식은 setoptions bidirectional 또는 bidirectional- 옵션으로 결정됩니다.

UNIX 에서는 history 옵션이 긴 암호의 활성화 여부에 영향을 주지 않습니다. 긴 암호의 활성화 여부를 결정하려면 passwd_local_encryption_method 구성 설정을 사용하십시오.

history-

암호 기록 검사를 비활성화합니다.

Windows 에서는 이 옵션이 긴 암호 사용을 비활성화합니다.

inactive(nDays)

사용자의 로그인이 일시 중단된 이후 비활성 기간을 지정합니다. 비활성 일은 사용자가 로그인하지 않는 날입니다. 양수를 입력합니다. inactive 를 0 으로 설정하면 inactive- 매개 변수를 사용하는 것과 결과적으로 동일합니다.

inactive-

비활성 로그인 검사를 비활성화합니다.

interval(nDays)

사용자에게 새 암호를 묻기 전에 암호가 설정되거나 변경된 후 경과해야 하는 일 수를 설정합니다. 0 또는 양수를 입력합니다. 간격이 0 이면 사용자에 대한 암호 간격 검사가 비활성화됩니다. 암호가 만료되지 않도록 하려면 간격을 0 으로 설정합니다.

segrace 유틸리티가 사용자의 로그인 스크립트의 일부이면 CA Access Control 은 지정된 일 수에 도달할 때 현재 암호가 만료되었음을 사용자에게 알립니다. 사용자는 암호를 즉시 갱신하거나 유예 로그인 횟수에 도달할 때까지 기존 암호를 계속 사용할 수 있습니다. 유예 로그인 횟수에 도달하면 시스템 액세스가 거부되며 시스템 관리자에게 문의하여 새 암호를 선택해야 합니다.

interval-

암호 간격 설정을 취소합니다.

is_dms+

현재 데이터베이스를 DMS 로 지정합니다.

is_dms-

현재 데이터베이스에서 DMS 지정을 제거합니다.

list

화면에 현재 CA Access Control 설정을 표시합니다.

maxlogins(nLogins)

사용자가 동시에 로그인할 수 있는 최대 터미널 수를 설정합니다. 값 0 은 사용자가 터미널 수에 상관없이 동시에 로그인할 수 있음을 의미합니다. 사용자의 사용자 레코드에 값을 할당하면 이 값은 무시됩니다.

참고: maxlogins 가 1 로 설정되어 있으면 selang 을 실행할 수 없습니다. CA Access Control 을 종료하고 maxlogins 설정을 1 보다 큰 수로 변경한 후 CA Access Control 을 다시 시작해야 합니다.

maxlogins-

전역 최대 로그인 검사를 비활성화합니다. 사용자 로그인이 사용자 레코드에 제한되지 않는다면, 사용자가 로그인할 수 있는 터미널 수는 제한되지 않습니다.

min_life(NDays)

암호를 변경하는 최소 일 수를 설정합니다. 양수를 입력합니다.

password

암호 옵션을 설정합니다.

rules

CA Access Control 에서 새 암호의 품질을 검사하기 위해 사용하는 하나 이상의 암호 규칙을 설정합니다. 규칙은 다음과 같습니다.

alpha(nCharacters)

새 암호에 포함해야 하는 최소 영문자 수를 설정합니다. 정수를 입력합니다.

alphanum(nCharacters)

새 암호에 포함해야 하는 최소 영숫자 수를 설정합니다. 정수를 입력합니다.

bidirectional

암호를 PMDB 의 일부로 다른 시스템에 전송할 때 일반 텍스트로(암호화된 메시지 내에서) 배포되도록 지정합니다.

UNIX 에서 이 옵션은 다음의 passwd 섹션 설정값을 설정하는 것과 같습니다.

```
Passwd_distribution_encryption_mode=bidirectional
```

참고: setoptions 명령을 사용하는 것보다 구성 설정을 지정하는 것이 좋습니다.

Windows 에서 암호는 다음 레지스트리 값에 지정된 암호화와 함께 기록 목록에 저장됩니다.

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\AccessControl\Encryption Package
```

bidirectional-

암호가 해시 암호화 형식으로 전송되도록 지정합니다.

Windows 에서 사용되는 해시 함수는 SHA-1 입니다.

UNIX 에서 이 옵션은 다음의 passwd 섹션 설정값을 설정하는 것과 같습니다.

```
Passwd_distribution_encryption_mode=compatibility
```

참고: setoptions 명령을 사용하는 것보다 구성 설정을 지정하는 것이 좋습니다.

이 옵션이 지정되어 있으면 서로 다른 운영 체제 간에 긴 암호를 배포할 수 없습니다.

grace(nLogins)

사용자에 대한 일시 중단 전에 허용되는 최대 유예 로그인 횟수를 설정합니다. 유예 로그인 횟수는 0 에서 255 사이여야 합니다(0 과 255 포함).

min_len(nCharacters)

최소 암호 길이를 설정합니다. 새 암호가 포함해야 하는 최소 문자 수를 입력합니다.

max_len(nCharacters)

최대 암호 길이를 설정합니다. 새 암호에 들어가야 하는 최대 총 문자 수를 입력합니다.

lowercase(nCharacters)

새 암호에 포함해야 하는 최소 소문자 수를 설정합니다. 정수를 입력합니다.

max_rep(nCharacters)

새 암호에 포함해야 하는 최대 반복 문자 수를 설정합니다. 정수를 입력합니다.

namechk

암호에 사용자 이름이 포함되어 있는지 또는 사용자 이름에 암호가 포함되어 있는지를 검사합니다. 기본적으로 CA Access Control 이 이 검사를 수행합니다.

namechk-

namechk 검사를 해제합니다.

numeric(nCharacters)

새 암호에 포함해야 하는 최소 숫자 수를 설정합니다. 정수를 입력합니다.

oldpwchk

새 암호에 교체되는 암호가 포함되어 있는지 또는 교체되는 암호에 새 암호가 포함되어 있는지를 검사합니다. 기본적으로 CA Access Control 이 이 검사를 수행합니다.

oldpwchk-

oldpwchk 를 해제합니다.

prohibited(prohibitedCharacters)

사용자가 암호에 사용할 수 없는 문자를 지정합니다. 금지된 문자를 입력합니다.

special(nCharacters)

새 암호에 포함해야 하는 최소 특수 문자 수를 설정합니다. 정수를 입력합니다.

sub_str_len(nCharacters)

새 암호가 이전 암호와 공유할 수 있는 최대 문자 수를 설정합니다. 정수를 입력합니다.

uppercase(nCharacters)

새 암호에 포함해야 하는 최소 대문자 수를 설정합니다. 정수를 입력합니다.

use_dbdict | use_dbdict-

암호 사전을 설정합니다. use_dbdict 는 토큰을 **db** 로 설정하고, 암호를 CA Access Control 데이터베이스에 있는 단어와 비교합니다. use_dbdict- sets 는 토큰을 **file** 로 설정하고, seos.ini 파일에 지정된 파일(UNIX 의 경우) 또는 Windows 레지스트리(Windows 의 경우)에 대해 암호를 검사합니다.

rules-

암호 품질 검사를 비활성화합니다. rules 인수에 의해 지정된 규칙은 암호 품질 검사에 사용되지 않습니다.

예: CA Access Control 옵션 설정

- 사용자 John 은 운영자 작업을 보호하는 데 사용되는 설치 시 정의된 클래스인 OpsAct 를 활성화하려고 합니다.

  사용자 John 은 ADMIN 특성을 가집니다.

  ```
  setoptions class+(OpsAct)
  ```

- 사용자 Mike 는 사용자에게 6 자 이상의 암호를 사용하도록 하는 암호 정책을 설정하고, 또한 암호 정책 적용 기능을 활성화하고자 합니다.

  사용자 Mike 가 ADMIN 특성을 가지고 있습니다.

  ```
  setoptions class+(PASSWORD)
  setoptions password(rules(min_len(6)))
  ```

- 사용자 SecAdmin 은 보안 수준 검사를 활성화하려고 합니다.

  사용자 SecAdmin 은 ADMIN 특성을 가집니다.

  ```
  setoptions class+(SECLEVEL)
  ```

- 사용자 Janani 는 알림을 보내기 위해 이 데이터베이스에 대해 DMS 를 설정하고자 합니다.

  사용자 Janani 는 ADMIN 특성을 가지고 있습니다.

  ```
  setoptions dms+(apache@myHost)
  ```

예: 경고 모드로 클래스 사용

클래스에 대한 경고 속성을 설정하여 클래스에 경고 모드를 적용합니다. 다음과 같이 setoptions selang 명령을 사용하여 이 작업을 수행할 수 있습니다.

```
setoptions class(classname) flags+ (W)
```

classname

경고 모드를 적용할 클래스 이름을 정의합니다.

참고: W 플래그는 대소문자를 구분하며 반드시 대문자를 사용해야 합니다.

클래스의 경고 모드를 지우려면 다음과 같이 setoptions 명령을 사용합니다.

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

추가 정보:

setoptions 명령-CA Access Control Windows 옵션 설정(페이지 203)

search 명령-데이터베이스 레코드 나열

AC 및 네이티브 환경에 해당

이것은 find 명령과 동일합니다.

추가 정보:

find 명령-데이터베이스 레코드 나열(페이지 116)

showfile 명령-파일 속성 표시

AC 환경에 해당

showfile 명령은 파일 레코드의 속성을 나열합니다. 속성은 알파벳 순으로 나열됩니다. CA Access Control 은 각 레코드를 독립적으로 처리하고 사용자가 충분한 권한을 가진 리소스에 대한 정보만 표시합니다.

참고: 이 명령은 네이티브 환경에도 있지만 작동 방식이 다릅니다.

showfile 명령을 실행하려면 다음 조건 중 적어도 하나가 충족되어야 합니다.

- ADMIN, AUDITOR 및 OPERATOR 특성 중 최소한 하나 이상을 가집니다.
- 파일의 소유자입니다.
- 사용자에게 ADMIN 클래스의 FILE 클래스 레코드를 나타내는 개체의 액세스 제어 목록에서 읽기 권한이 할당되어 있습니다.
- 파일을 소유하는 그룹이나 이 그룹의 상위 그룹에 GROUP-ADMIN 또는 GROUP-AUDITOR 특성이 있습니다.

이 명령의 형식은 다음과 같습니다.

```
{showfile|sf} {fileName |(fileName [,fileName...])} \
    [addprops(propName [,propName ...])] \
    [next] \
    [props(all | propName [,propName ...])] \
    [useprops(propName [,propName ...])] \
    [nt|unix]
```

addprops(propName [,propName ...])

이 쿼리만을 위해 클래스 ruler 에 추가할 속성을 정의합니다.

fileName

속성을 나열해야 하는 파일 레코드의 이름을 지정합니다.

CA Access Control 은 각 파일 레코드를 독립적으로 처리합니다. 파일을 처리할 때 오류가 발생하면 CA Access Control 은 메시지를 작성하고 목록의 다음 파일을 계속 처리합니다.

fileName 에는 와일드카드 문자를 포함할 수 있으므로 여러 파일 이름과 일치할 수 있습니다.

UNIX 에서 이름에 특수 문자나 공백이 포함된 파일의 속성을 표시하려면 파일 이름 앞에 슬래시(/)를 하나 더 입력하십시오.

next

요청된 데이터의 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. query_size 의 기본값은 100 입니다.

nt

CA Access Control 속성뿐 아니라 Windows 파일 특성도 표시합니다.

props(all|propName [,propName ...])

이 쿼리와 미래의 쿼리를 위해 이 클래스에 대해 새 ruler 를 정의합니다.

unix

CA Access Control 속성뿐 아니라 UNIX 파일 특성도 표시합니다.

useprops(propName [,propName ...])

이 쿼리만을 위한 ruler 를 정의합니다. 클래스 ruler 는 영향을 받지 않습니다.

예제

루트 사용자는 파일 레코드 /etc/passwd 의 속성을 나열하려고 합니다.

- 루트 사용자는 ADMIN 특성을 가집니다.

```
showfile /etc/passwd
```

추가 정보:

checklogin 명령-로그인 정보 확인(페이지 55)
chfile 명령-파일 레코드 수정(페이지 58)
rmfile 명령-파일 레코드 삭제(페이지 131)
showfile 명령-네이티브 파일 속성 표시(페이지 177)

show[x]grp 명령-그룹 속성 표시

AC 환경에 해당

how[x]grp 명령은 그룹 레코드의 모든 CA Access Control 속성에 대한 설정을 표시합니다. 또한 선택적으로 네이티브 환경 속성이 표시됩니다.

참고: showgrp 명령은 네이티브 환경에도 있지만 작동 방식이 다릅니다.

다음 조건 중 적어도 하나가 충족될 경우 show[x]grp 명령을 실행할 수 있습니다.

- ADMIN, AUDITOR 및 OPERATOR 특성 중 최소한 하나 이상을 가집니다.
- 나열할 각 그룹에 GROUP-ADMIN 또는 GROUP-AUDITOR 특성이 있거나 현재 가지고 있는 GROUP-ADMIN 특성이 속한 그룹의 범위 내에 나열할 각 그룹이 있습니다.
- 그룹의 소유자입니다.
- ADMIN 클래스에 있는 GROUP 레코드의 액세스 제어 목록에 읽기 권한이 할당되어 있습니다.

이 명령의 형식은 다음과 같습니다.

```
{showgrp|sg} {groupName |groupName [,groupName...] )} \
    [addprops(propName[,propName ...])] \
    [next] \
    [props(all | propName[,propName ...])] \
    [useprops(propName[,propName ...])] \
    [nt|unix]
```

addprops(propName [,propName ...])

이 쿼리만을 위해 ruler 에 추가할 속성을 정의합니다.

groupName

속성을 나열할 그룹을 지정합니다.

groupName 에는 와일드카드 문자를 포함할 수 있습니다.

UNIX 에서 이름에 특수 문자나 공백이 포함된 그룹의 속성을 표시하려면 그룹 이름 앞에 슬래시(/)를 하나 더 입력하십시오.

next

요청된 데이터 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. query_size 의 기본값은 100 입니다.

nt

데이터베이스의 속성 외에도 로컬 Windows 시스템에서 나온 그룹의 세부 정보를 보여줍니다.

props(all|propName [,propName ...])

이 쿼리와 미래의 쿼리를 위해 이 클래스에 대해 ruler 를 정의합니다.

useprops(propName [,propName ...])

이 쿼리만을 위한 ruler 를 정의합니다. 클래스 ruler 는 영향을 받지 않습니다.

unix

데이터베이스의 속성 외에도 로컬 UNIX 시스템에서 나온 그룹의 세부 정보를 보여줍니다.

예제

- 사용자 root 가 보안 그룹의 속성을 표시하고자 합니다.
  - root 사용자에게는 보안 그룹에 GROUP-ADMIN 속성이 있습니다.

  ```
  showgrp security
  ```
- 사용자 admin 은 모든 엔터프라이즈 그룹의 속성을 표시하고자 합니다.
  - 사용자 admin 이 ADMIN 및 AUDITOR 특성을 가지고 있습니다.

  ```
  showxgrp *
  ```

  CA Access Control 에 정의된 모든 엔터프라이즈 그룹의 속성이 나열됩니다.

추가 정보:

ch[x]grp 명령-그룹 속성 변경(페이지 64)
rm[x]grp 명령-그룹 레코드 삭제(페이지 132)
showgrp 명령-네이티브 그룹 속성 표시(페이지 178)

showres 명령-리소스 속성 표시

AC 환경에 해당

showres 명령은 데이터베이스의 클래스에 속한 리소스의 속성을 표시합니다. 속성은 알파벳 순으로 나열됩니다. showres 명령을 사용하여 ACVAR, ADMIN, CATEGORY, CONNECT, FILE, GHOST, GSUDO, GTERMINAL, HOST, HOSTNET, HOSTNP, SECFILE, SECLABEL, SUDO, SURROGATE, TERMINAL, PROGRAM, PROCESS, TCP, UACC 및 모든 사용자 정의 클래스를 나열할 수 있습니다. CA Access Control 은 각 리소스를 독립적으로 처리하고 사용자가 충분한 권한을 가진 리소스에 대한 정보만 표시합니다.

참고: 이 명령은 네이티브 Windows 환경에도 있지만 작동 방식이 다릅니다.

showres 는 또한 언트러스트된 프로그램에 대한 정보를 표시합니다. 이러한 정보는 다음과 같습니다.

- 프로그램이 언트러스트된 이유
- 프로그램에 마지막으로 액세스한 사용자의 UID(프로그램을 언트러스트된 상태로 만든 사용자일 필요는 없음)
- 이러한 마지막 사용자가 프로그램에 액세스한 날짜 및 시간

최소한 다음 조건 중 하나를 충족할 경우 showres 명령을 실행할 수 있습니다.

- ADMIN, AUDITOR 및 OPERATOR 특성 중 최소한 하나 이상을 가집니다.
- 사용자가 리소스의 소유자입니다.
- 사용자에게 ADMIN 클래스의 리소스 클래스 레코드를 나타내는 개체의 액세스 제어 목록에서 읽기 권한이 할당되어 있습니다.

이 명령의 형식은 다음과 같습니다.

```
{showres|sr} className resourceName \
    [addprops(propName [,propName...])] \
    [next] \
    [props(all | propName [,propName...])] \
    [useprops(propName [,propName...])]
```

addprops(propName [,propName...])

이 쿼리만을 위해 현재 ruler 에 추가할 속성을 정의합니다.

className

리소스가 속한 클래스의 이름을 지정합니다. CA Access Control 에 정의된 리소스 클래스를 나열하려면 find 명령을 사용하십시오.

next

요청된 데이터의 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. 쿼리 크기의 기본값은 100 으로 설정됩니다.

props(all|propName [,propName ...])

이 쿼리와 미래의 쿼리를 위해 이 클래스에 대해 새 ruler 를 정의합니다.

resourceName

속성을 나열해야 하는 리소스 레코드의 이름을 지정합니다. 여러 리소스의 속성을 나열하는 경우 리소스 이름 목록을 괄호로 묶고 공백이나 쉼표로 리소스 이름을 구분합니다.

CA Access Control 은 각 리소스 레코드를 독립적으로 처리합니다. 리소스를 처리할 때 오류가 발생하면 CA Access Control 은 메시지를 작성하고 목록의 다음 리소스를 계속 처리합니다.

resourceName 에는 와일드카드 문자를 포함할 수 있습니다.

UNIX 에서 이름에 특수 문자나 공백이 포함된 단일 리소스 레코드의 속성을 표시하려면 파일 이름 앞에 슬래시(/)를 하나 더 입력하십시오.

useprops(propName [,propName ...])

이 쿼리만을 위한 ruler 를 정의합니다. 클래스 ruler 는 영향을 받지 않습니다.

예제

사용자 Admin1 은 TERMINAL 클래스에서 마스크 ath*와 이름이 일치하는 레코드의 속성을 나열하고자 합니다.

- 사용자 Admin1 이 ADMIN 및 AUDITOR 특성을 가지고 있습니다.

```
showres TERMINAL ath*
```

추가 정보:

chres 명령-리소스 레코드 수정(페이지 77)
rmres 명령-리소스 삭제(페이지 133)
showres 명령-네이티브 리소스 속성 표시(페이지 207)
find 명령-데이터베이스 레코드 나열(페이지 116)

show[x]usr 명령-사용자 속성 표시

AC 환경에 해당

show[x]usr 명령은 CA Access Control 에 정의된 하나 이상의 사용자의 모든 속성 값을 표시합니다.

내부 사용자의 속성을 표시하려면 showusr 명령을 사용하십시오.
엔터프라이즈 사용자의 속성을 표시하려면 showxusr 명령을 사용하십시오.

참고: showusr 명령은 네이티브 환경에도 있지만 작동 방식이 다릅니다.

자신의 사용자 레코드에 대한 속성은 언제든지 나열할 수 있습니다. 다른 사용자 레코드의 속성을 나열하려면 다음 조건 중 하나가 참이어야 합니다.

- 사용자 레코드의 소유자입니다.
- ADMIN, AUDITOR 및 OPERATOR 특성 중 최소한 하나 이상을 가집니다.
- ADMIN, AUDITOR 또는 OPERATOR 그룹 특성 중 최소한 하나 이상을 가지고 있는 그룹의 범위 내에 사용자 레코드가 있습니다.
- ADMIN 클래스에 있는 USER 레코드의 액세스 제어 목록에 읽기 권한이 할당되어 있습니다.

이 명령의 형식은 다음과 같습니다.

```
{showusr|su |showxusr |sxu } [ {userName |(userName [,userName...]) } ] \
    [addprops(propName [,propName...])] \
    [next] \
    [props( all | propName [,propName...])] \
    [useprops(propName[,propName...])] \
    [nt|unix]
```

addprops(propName [,propName...])

이 쿼리만을 위해 현재 ruler 에 추가할 속성을 정의합니다.

next

요청된 데이터의 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. 쿼리 크기의 기본값은 100 으로 설정됩니다.

nt

데이터베이스의 속성 외에 사용자의 Windows 속성도 표시합니다.

props(all|propName [,propName ...])

이 쿼리와 미래의 쿼리를 위해 이 클래스에 대해 새 ruler 를 정의합니다.

unix

데이터베이스의 속성 외에 사용자의 UNIX 속성도 표시합니다.

userName

사용자 이름을 정의합니다. 이름에 와일드카드 문자를 포함할 수 있습니다.

UNIX 에서 이름에 특수 문자나 공백이 포함된 단일 사용자 레코드의 속성을 표시하려면 그룹 이름 앞에 슬래시(/)를 하나 더 입력하십시오.

userName 을 지정하지 않으면 자신의 사용자 레코드 속성이 표시됩니다.

useprops(propName [,propName ...])

이 쿼리만을 위한 ruler 를 정의합니다. 클래스 ruler 는 영향을 받지 않습니다.

예제

- 사용자 root 는 엔터프라이즈 사용자 Robin 의 속성을 나열하고자 합니다. root 가 ADMIN 및 AUDITOR 특성을 가지고 있습니다.

  ```
  showxusr Robin
  ```

- 사용자 root 는 엔터프라이즈 사용자 Robin 과 Leslie 의 사용자 속성을 나열하고자 합니다. root 가 ADMIN 및 AUDITOR 특성을 가지고 있습니다.

  ```
  showxusr (Robin,Leslie)
  ```

추가 정보:

rm[x]usr 명령-사용자 레코드 삭제(페이지 135)
ch[x]usr 명령-사용자 속성 변경(페이지 94)
showusr 명령-네이티브 사용자 속성 표시(페이지 179)

source 명령-파일에서 명령 실행

모든 환경에 해당

source 명령을 사용하면 파일에 배치된 하나 이상의 selang 명령을 실행할 수 있습니다. CA Access Control 은 지정된 파일을 읽고, 명령을 실행하고, selang 프롬프트를 반환합니다. 데이터베이스에 정의된 모든 사용자가 이 명령을 사용할 수 있습니다.

이 명령은 UNIX 에서 csh 및 tcsh 의 source 명령과 유사합니다.

이 명령의 형식은 다음과 같습니다.

```
source fileName
```

fileName

selang 명령이 포함된 파일의 이름을 지정합니다.

예제

사용자 admin 은 initf1 파일에 있는 명령을 실행하고자 합니다. 사용자는 다음 명령을 입력합니다.

```
source initf1
```

start dbexport 명령 - 데이터베이스 내보내기 시작

AC 환경에 해당

start dbexport 명령은 연결된 호스트의 CA Access Control 데이터베이스를 내보내고 출력을 버퍼에 복사합니다. PMDB 에 연결된 경우 또한 이 명령을 사용하여 PMD 데이터베이스를 내보낼 수 있습니다.

참고: 출력을 보려면 get dbexport 명령을 사용하십시오.

이 명령의 형식은 다음과 같습니다.

```
start dbexport [pmdname(name)] [filter("CLASS, CLASS...")] [param("depend=yes")]
[param("edit=yes")]
```

filter("CLASS, CLASS...")

(선택 사항) 데이터베이스에서 내보낼 클래스를 정의합니다. 클래스를 정의하지 않으면 데이터베이스에 있는 모든 규칙을 내보냅니다.

param("depend=yes")

(선택 사항) 필터 매개 변수에 정의하는 클래스와 함께 종속된 클래스를 내보내도록 지정합니다. 이 매개 변수를 지정하면 CA Access Control 은 지정된 클래스와 다음의 종속된 클래스를 내보냅니다.

- 특정 클래스의 리소스를 수정하는 규칙을 내보내고, 이 클래스에 해당 리소스 그룹이 있는 경우 CA Access Control 은 이 리소스 그룹에 있는 리소스를 수정하는 규칙도 또한 내보냅니다.
- 특정 리소스 그룹에 있는 리소스를 수정하는 규칙을 내보내면 CA Access Control 은 이 리소스 그룹의 구성원 리소스를 수정하는 규칙도 또한 내보냅니다.
- 클래스에 PACL 이 있는 특정 클래스의 리소스를 수정하는 규칙을 내보내면 CA Access Control 은 PROGRAM 클래스에 있는 리소스를 수정하는 규칙도 또한 내보냅니다.
- 클래스에 CALACL 이 있는 특정 클래스의 리소스를 수정하는 규칙을 내보내면 CA Access Control 은 CALENDAR 클래스에 있는 리소스를 수정하는 규칙도 또한 내보냅니다.
- 특정 클래스에 있는 리소스를 수정하는 규칙을 내보내고, 해당 클래스에 있는 리소스 중 하나가 CONTAINER 리소스 그룹의 구성원인 경우 CA Access Control 은 CONTAINER 클래스에 있는 리소스를 수정하고 각 CONTAINER 리소스 그룹의 구성원인 리소스를 수정하는 규칙을 내보냅니다.

param("edit=yes")

(선택 사항) CA Access Control 이 새 리소스 또는 접근자를 만드는 각 규칙을 리소스 또는 접근자를 수정하는 규칙으로 변경하도록 지정합니다.

예: 이 매개 변수를 지정하면 CA Access Control 은 모든 newres 규칙을 editres 규칙으로 변경합니다.

pmdname(name)

(선택 사항) 내보낼 PMD 데이터베이스의 이름을 정의합니다.

예: 데이터베이스 내보내기 시작

다음 예제는 FILE 및 GFILE 클래스 리소스를 수정하는 규칙의 내보내기를 시작합니다. 규칙은 연결된 호스트의 CA Access Control 데이터베이스인 seosdb 에서 내보내집니다.

```
start dbexport filter("FILE, GFILE")
```

예: 종속된 클래스와 함께 데이터베이스 내보내기 시작

다음 예제는 FILE 클래스 리소스와 FILE 클래스 리소스에 종속된 모든 클래스를 수정하는 규칙의 내보내기를 시작하고, 새 리소스 또는 접근자를 만드는 규칙을 리소스 또는 액세스를 수정하는 규칙으로 변경합니다.

```
start dbexport filter("FILE") param("depend=yes edit=yes")
```

추가 정보:

get dbexport Command - 내보낸 데이터베이스 규칙 검색(페이지 117)

start devcalc 명령-정책 위반 계산 시작

AC 환경에 해당

start devcalc 명령은 정책 위반 계산을 시작하고 위반 상태를 전송합니다. 위반 데이터는 정책 위반 데이터 파일(deviation.dat)에 저장되며 정책 위반 상태는 하나 이상의 설정된 DH 를 통해 DMS 로 전송됩니다. 실제 위반 데이터를 검색하려면 get devcalc 명령을 실행해야 합니다.

참고: 위반 계산기를 직접 실행할 필요는 없습니다. 고급 정책 관리를 사용하는 경우 일반적으로 policyfetcher 가 이 작업을 수행합니다. 엔터프라이즈 보고 기능을 사용하는 경우 보고서 에이전트도 이 작업을 수행합니다. 정책 위반 계산에 대한 자세한 내용은 엔터프라이즈 관리 안내서를 참조하십시오.

start devcalc 명령을 실행하려면 컴퓨터에 대한 터미널 액세스 권한 및 DEVCALC 하위 관리 클래스에 대한 실행 액세스 권한이 있어야 합니다.

이 명령의 형식은 다음과 같습니다.

```
start devcalc [params("-pn name#xx -strict")]
```

-pn name#xx

(선택 사항) 위반 계산기가 차이를 계산해야 하는 POLICY 개체(정책 버전)의 목록을 쉼표로 구분하여 정의합니다. 정책을 지정하지 않으면 위반 계산기는 로컬 호스트에 배포된 모든 정책에 대해 차이를 계산합니다.

-strict

(선택 사항) 로컬 HNODE 개체와 관련된 정책을 첫 번째 사용 가능한 DMS 의 HNODE 개체와 관련된 정책과 비교합니다.

일반적으로 위반 계산기는 로컬 호스트에서만 위반을 확인합니다. 이 옵션을 지정하면 위반 계산기는 로컬 정책을 목록에 있는 첫 번째 사용 가능한 DMS 에 대한 정책과 비교합니다. 이 경우 다음을 비교합니다.

1. 로컬 호스트를 나타내는 HNODE 개체에 연결된 정책 목록
2. HNODE 개체에 연결된 각 POLICY 개체의 정책 상태
3. HNODE 개체에 연결된 각 POLICY 개체의 정책 서명

위반 계산의 결과를 확인해야 할 경우 이 옵션을 사용하십시오.

참고: 위반 계산을 동시에 실행하는 다수의 끝점을 가지고 있는 경우 DMS 과부하가 초래됩니다. 따라서 DMS 목록을 사용하도록 끝점을 구성하거나, 계층을 더 작은 여러 계층으로 나누고 이러한 더 작은 계층에서 이 옵션을 사용하는 것이 좋습니다.

예: 특정 정책에 대한 정책 위반 계산 시작

다음 예에서는 start devcalc 명령을 사용하여 myPolicy 라고 하는 두 번째 정책 버전에 대해 정책 위반을 계산하고 위반 상태를 로컬 CA Access Control 데이터베이스에 지정된 DMS 목록으로 전송하는 방법을 보여줍니다.

```
AC> start devcalc params("-pn myPolicy#02")
```

start_transaction 명령-이중 컨트롤 트랜잭션 기록 시작

AC 환경의 UNIX 호스트에 해당

start_transaction 및 end_transaction 명령은 이중 컨트롤 PMDB 프로세스에서 처리되지 않은 트랜잭션(하나 이상의 명령으로)을 포함하는 파일을 만듭니다. 트랜잭션에서 명령을 입력한 관리자(ADMIN 특성을 가진 사용자)를 Maker 라고 부릅니다. PMDB 에서 명령을 실행하기 전에 Checker(Maker 가 아닌 관리자)가 명령을 허가해야 합니다.

트랜잭션을 처리할 수 있으려면 그 전에 Checker 가 트랜잭션을 잠가야 합니다. Checker 에 의해 트랜잭션이 잠기기 전까지 Marker 는 트랜잭션을 검색하거나, 명령을 변경하거나, 트랜잭션을 삭제할 수 있습니다. 자세한 내용은 참조 안내서의 sepmd 유틸리티를 참조하십시오. Maker 가 end_transaction 명령을 입력하면 트랜잭션은 고유한 id 번호를 수신합니다. Maker 가 나중에 이 트랜잭션을 편집하거나 검색할 경우 start_transaction 명령에서 트랜잭션 이름 다음에 이 식별 번호를 붙여야 합니다. Maker 가 트랜잭션을 수신할 때 Maker 의 이름, 트랜잭션의 ID 번호 및 간단한 설명이 표시됩니다(Maker 가 transactionName 매개 변수에 설명을 입력한 경우).

Maker 는 다른 Maker 의 트랜잭션을 변경할 수 없습니다. 트랜잭션에 사용된 개체는 명령이 처리될 때까지 다른 트랜잭션에서 다른 Maker 가 사용할 수 없습니다.

처리되지 않은 각 트랜잭션은 Checker 가 처리할 때까지 별개의 파일에 유지됩니다. Checker 는 트랜잭션을 허가하거나 거부할 수 있습니다. 트랜잭션이 인증되는 경우 명령이 실행되고 PMDB 는 이에 따라 변경됩니다. Checker 가 트랜잭션을 거부하는 경우 명령은 삭제되고 PMDB 는 변경되지 않습니다.

Maker 의 작업 끝에 end_transaction 명령을 입력하면 트랜잭션의 숫자 id 가 표시됩니다. 명령은 다음과 같은 이유로 인해 실패할 수 있습니다.

- 명령이 아직 처리되지 않은 다른 트랜잭션에 사용된 개체를 참조하는 경우
- 명령이 Maker 와 관련된 경우(자기 자신을 변경할 수 없음)
- 명령에 잘못된 구문이 포함된 경우
- 명령이 존재하지 않는 개체를 참조할 경우(이 경우 경고 메시지가 나타남)

- ADMIN 특성을 가진 경우 start_transaction 및 end_transaction 명령을 실행할 수 있습니다.
- start_transaction 및 end_transaction 명령을 호출하기 전에 hosts 명령을 실행해야 하므로 hosts 명령을 사용할 수 있는 권한이 있어야 합니다.

참고: 이중 컨트롤에 대한 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

참고:

- start_transaction 명령과 end_transaction 명령을 호출하기 전에 hosts 명령을 실행해야 하며, PMDB 의 이름은 "maker"여야 합니다.
- start_transaction 명령과 end_transaction 명령이 실행되려면 pmd.ini 파일과 seos.ini 파일의 [pmd] 섹션에 있는 is_maker_checker 토큰이 yes 로 설정되어야 합니다.

이 명령의 형식은 다음과 같습니다.

```
start_transaction transactionName [transactionId]
.
.
.
end_transaction
```

transactionName

트랜잭션의 이름 또는 설명을 지정합니다. 최대 256 자의 영숫자를 입력할 수 있습니다.

transactionId

트랜잭션을 만들 때 부여되는 고유한 번호를 지정합니다. 이 숫자 ID 는 트랜잭션을 만들 때 자동으로 표시됩니다. 동일한 트랜잭션을 업데이트할 때 이 ID 번호를 지정해야 합니다.

예제

- Maker Sally가 사용자 Anne을 PMDB에 추가하고 그들의 시스템 접근을 주중 오전 8:00부터 오후 8:00 사이로 제한하려고 합니다. 그런 다음 Sally는 tty30 터미널에 대한 기본 접근 권한을 읽기 전용으로 변경하려고 합니다. Sally는 이 트랜잭션을 "general"로 부르려고 합니다.

  – Maker는 ADMIN 특성을 가집니다.

  ```
  hosts maker@
  start_transaction general
  newusr anne
  (days(weekdays)time(0800:2000))
  chres TERMINAL tty30
  defaccess(read)
  end_transaction
  ```

  Sally가 end_transaction 명령을 입력하면 ID 번호(예: 7)가 트랜잭션에 할당됩니다.

- Maker Sally는 FINANCIAL 범주를 사용자 Anne에 추가하려고 합니다. Sally는 같은 날짜에 사용자 Anne 레코드를 먼저 추가했으며 명령은 아직까지 PDDB에서 처리 또는 구현되지 않습니다.

  – Maker는 ADMIN 특성을 가집니다.

  ```
  hosts maker@
  start_transaction general 7
  chusr anne category(FINANCIAL)
  end_transaction
  ```

unalias 명령-selang 별칭 제거

UNIX 호스트에 해당

unalias 명령은 alias 명령으로 정의된 별칭을 제거합니다.

참고: alias 명령을 사용하면 모든 정의된 별칭과 해당 값을 나열할 수 있습니다.

이 명령의 형식은 다음과 같습니다.

```
unalias aliasName
```

aliasName

데이터베이스에서 삭제하려는 별칭의 이름을 지정합니다.

추가 정보:

alias 명령 selang 별칭 정의(페이지 44)

undeploy 명령-정책 제거 시작

AC 환경에 해당

이 명령은 deploy- 명령과 동의어입니다.

추가 정보:

deploy- 명령-정책 제거 시작(페이지 113)

원격 구성 환경의 selang 명령

이 단원에는 CA Access Control 구성 리소스에서 작동하는 모든 selang 명령(config 환경의 명령)에 대한 완전한 참조가 알파벳순으로 포함되어 있습니다.

editres config-구성 설정 수정

config 환경에 해당

CA Access Control 구성 설정을 수정하려면 editres config 명령을 사용하십시오.

editres 구성 명령의 형식은 파일 집합에 따라 달라집니다. 이러한 집합은 다음과 같습니다.

- 감사 구성 파일(audit.cfg 및 auditrouteflt.cfg) 및 PMDB 필터 파일
- 다른 모든 파일

감사 구성 파일 및 PMDB 필터 파일에 대한 이 명령의 구문은 다음과 같습니다.

```
editres config name [line+|-(value)] [clear]
```

다른 모든 파일에 대한 이 명령의 구문은 다음과 같습니다.

```
editres config name section(path) token[-](name) value[+|-](value) data_type(type)
```

name

수정할 구성 리소스를 지정합니다. PMDB 필터 파일을 수정하려면 다음 형식으로 파일 이름을 지정하십시오: pmdname@filter. 예: master_pmdb@filter.flt

참고: 관리 중인 호스트에 대한 구성 리소스의 목록을 보려면 find config 명령을 사용하십시오.

clear

감사 구성 파일 또는 PMDB 필터 파일에서 모든 값을 삭제합니다.

참고: 이 옵션은 파일에서 주석을 삭제하지 않습니다.

data_type(type)

구성 항목의 데이터 유형을 지정합니다.

값: str, numeric, multi_str

기본값: str

참고: UNIX 에서는 data_type 이 str 만 가능합니다. UNIX 는 구성 설정을 파일(텍스트 문자열)에 저장하므로 다른 데이터 형식은 UNIX 에 적용되지 않습니다.

line+(value)

감사 구성 파일 또는 PMDB 필터 파일에 추가할 값을 정의합니다.

참고: 값은 값 또는 주석일 수 있습니다.

line-(value)

감사 구성 파일 또는 PMDB 필터 파일에서 제거할 값을 정의합니다.

참고: 값은 값 또는 주석일 수 있습니다.

section(path)

수정할 구성 리소스의 섹션을 정의합니다.

참고: Windows 레지스트리 설정의 경우 이 옵션을 지정하지 않으면, name 이 정의하는 레지스트리 키가 수정됩니다.

token(name)

수정할 구성 항목의 이름을 정의합니다.

token-(name)

제거할 구성 항목의 이름을 정의합니다.

value(value)

구성 항목에 할당할 값을 정의합니다. 구성 항목에 대한 값이 이미 있으면 CA Access Control 은 이 값을 value 로 대체합니다.

value 를 지정하지 않으면 구성 항목 값이 다시 설정됩니다.

value+(value)

(Windows REG_MULTI_SZ 레지스트리 항목에만 해당) 구성 항목에 추가할 값을 정의합니다.

(기타 모든 구성 값) 구성 항목에 할당할 값을 정의합니다. 구성 항목에 대한 값이 이미 있으면 CA Access Control 은 이 값을 value 로 대체합니다.

참고: selang 이 할당된 값을 올바로 해석할 수 있도록 값을 따옴표(" ")로 둘러싸십시오.

value-(value)

(Windows REG_MULTI_SZ 레지스트리 항목에만 해당) 구성 항목에서 제거할 값을 정의합니다.

(기타 모든 구성 값) 구성 항목에서 모든 값을 제거하도록 지정합니다.

예: Windows 에서 ACROOT 구성 설정 수정

다음 예에서는 Windows 용 CA Access Control 구성 설정을 수정하는 방법을 보여줍니다.

- 이 예에서는 감사 전용 모드를 사용하도록 CA Access Control 을 구성합니다.

  ```
  er CONFIG ACROOT section(SeOSD) token(GeneralInterceptionMode) value(1)
  ```

- 이 예에서는 호스트 이름 확인을 위해 CA Access Control 이 유지 관리하는 도메인 이름 목록에 도메인 이름을 추가합니다. domain_names 레지스트리 항목은 REG_MULTI_SZ 레지스트리 항목입니다.

  ```
  er CONFIG ACROOT section(SeOSD) token(domain_names) value+(company.com)
  ```

- 이 예에서는 호스트 이름 확인을 위해 CA Access Control 이 유지 관리하는 도메인 이름 목록에서 도메인 이름을 제거합니다. domain_names 레지스트리 항목은 REG_MULTI_SZ 레지스트리 항목입니다.

  ```
  er CONFIG ACROOT section(SeOSD) token(domain_names) value-(company.com)
  ```

- 이 예에서는 구성 설정을 제거합니다.

  ```
  er CONFIG ACROOT section(AccessControl) token-(Emulate)
  ```

- 이 예에서는 관리되는 호스트에서 정책 모델의 부모 정책 모델을 구성합니다.

  ```
  er config myPMDB@PMDROOT token(Parent_Pmd) value(topPMDB@host1.comp.ca)
  ```

예: UNIX 에서 seos.ini 구성 설정 수정

다음 예에서는 UNIX 용 CA Access Control 구성 설정을 수정하는 방법을 보여줍니다.

- 이 예에서는 PAM 인증을 활성화하도록 CA Access Control 을 구성합니다.

  ```
  er CONFIG seos.ini section(seos) token(pam_enabled) value(yes)
  ```

- 이 예는 CA Access Control 이 호스트 이름 확인을 위해 유지하는 도메인 이름을 구성합니다.

  ```
  er CONFIG seos.ini section(seosd) token(domain_names) value+(company.com)
  ```

- 이 예는 CA Access Control 이 호스트 이름 확인을 위해 유지하는 도메인 이름을 제거합니다.

  ```
  er CONFIG seos.ini section(seosd) token(domain_names) value-(company.com)
  ```

- 이 예에서는 구성 설정을 제거합니다.

  ```
  er CONFIG seos.ini section(serevu) token-(admin_user)
  ```

예: 감사 구성 파일 수정

다음 예제는 감사 구성 파일에 한 줄을 추가합니다.

```
er CONFIG audit.cfg line+("FILE;*;Administrator;*;R;P")
```

예: PMD 필터 파일 수정

다음 예제는 PMD 필터 파일에 한 줄을 추가합니다.

```
er config pmdb@filter line+("*;*;USER;*;OLD_PASSWD;PASS")
```

find config-구성 리소스 나열

config 환경에 해당

find config 명령은 관리 중인 호스트에 대한 CA Access Control 구성 리소스를 나열합니다. 이러한 리소스는 레지스트리 키 또는 구성 파일일 수 있습니다.

가능한 리소스는 호스트 유형에 따라 다릅니다.

UNIX	Windows
seos.ini	ACROOT

UNIX	Windows
pmd.ini@pmd_name	pmd_name@PMDROOT
	SEOSDRV

이 명령의 형식은 다음과 같습니다.

```
find config
```

참고: 이 명령은 audit.cfg 또는 auditrouteflt.cfg 구성 파일의 목록을 반환하지 않습니다.

예: Windows 호스트에 대한 구성 리소스 나열

다음 예에서는 pmdb 라는 정책 모델이 있는 Windows 호스트에 대한 find config 명령의 출력을 보여 줍니다.

```
AC(config)> find config
(localhost)
pmdb@PMDROOT
ACROOT
SEOSDRV
```

showres config-구성 정보 표시

config 환경에 해당

CA Access Control 구성 정보를 표시하려면 showres config 명령을 사용하십시오.

showres 구성 명령의 형식은 파일 집합에 따라 달라집니다. 이러한 집합은 다음과 같습니다.

- 감사 구성 파일(audit.cfg 및 auditrouteflt.cfg) 및 PMDB 필터 파일
- 다른 모든 파일

감사 구성 파일 및 PMDB 필터 파일에 대한 이 명령의 구문은 다음과 같습니다.

```
showres config name
```

다른 모든 파일에 대한 이 명령의 구문은 다음과 같습니다.

```
showres config name [section(path)] [token(name)] [recursive] [sectiononly]
```

name

정보를 보려는 구성 리소스를 지정합니다. PMDB 필터 파일에 대한 정보를 보려면 다음 형식으로 파일 이름을 지정하십시오: pmdname@filter. 예: master_pmdb@filter.flt

참고: 관리 중인 호스트에 대한 구성 리소스의 목록을 보려면 find config 명령을 사용하십시오.

section(path)

(선택 사항) 정보를 보려는 구성 리소스의 섹션을 정의합니다.

이 옵션을 지정하지 않으면 name 구성 리소스에 있는 모든 구성 항목 및 섹션이 나열됩니다.

token(name)

(선택 사항) 정보를 보려는 구성 항목의 이름을 정의합니다.

이 옵션을 지정하지 않으면 정의한 섹션(path)에 있는 모든 구성 항목 및 섹션이 나열됩니다.

recursive

모든 하위 섹션에 있는 모든 구성 항목 및 섹션에 대한 정보를 표시하도록 지정합니다.

sectiononly

섹션에 대한 정보만 표시하도록 지정합니다(구성 항목은 나열되지 않음).

기본 UNIX 환경의 selang 명령

이 단원에는 CA Access Control 시스템 파일에서 작동하는 모든 selang 명령(기본 UNIX 환경의 명령)에 대한 완전한 참조가 알파벳순으로 포함되어 있습니다.

chfile 명령-UNIX 파일 설정 수정

기본 UNIX 환경에 해당

chfile 및 editfile 명령은 하나 이상의 UNIX 파일의 설정을 변경합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{{chfile|cf}|{editfile|ef}} fileName \
    [owner(userName)] \
    [group(groupName)] \
    [mode( \
        [fowner(string)] \
        [fgroup(string)] \
        [fother(string)] \
    )]
```

fileName

설정을 변경할 파일의 이름을 지정합니다. 적어도 하나의 UNIX 파일 이름을 입력해야 합니다. 하나 이상의 파일을 변경하려면 괄호 안에 파일 이름 목록을 넣고 파일 이름을 공백 또는 쉼표로 분리합니다.

group(groupName)

파일이 속한 그룹을 변경합니다. 유효한 그룹 이름을 지정합니다.

mode

파일의 액세스 모드를 업데이트합니다.

fowner(string)

파일 소유자의 액세스 모드를 지정합니다. 읽기 권한, 쓰기 권한, 실행 권한을 지정하려면 string 에 각각 r, w, x 문자를 사용하십시오. 문자 s 를 사용하여 setuid 파일을 만드십시오.

기존 사용 권한에 권한을 추가하려면 string 의 첫 머리에 더하기 기호(+)를 지정하십시오. 사용 권한을 제거하려면 string 의 첫 머리에 빼기 기호(-)를 지정하십시오. 접두사를 지정하지 않으면 이전의 사용 권한이 string 으로 재설정됩니다.

fgroup(string)

파일 그룹의 액세스 모드를 지정합니다. 읽기 권한, 쓰기 권한, 실행 권한을 지정하려면 string 에 각각 r, w, x 문자를 사용하십시오. 문자 s 를 사용하여 setgid 파일을 만드십시오.

기존 사용 권한에 권한을 추가하려면 string 의 첫 머리에 더하기 기호(+)를 지정하십시오. 사용 권한을 제거하려면 string 의 첫 머리에 빼기 기호(-)를 지정하십시오. 접두사를 지정하지 않으면 이전의 사용 권한이 string 으로 재설정됩니다.

fother(string)

다른 접근자에게 적용되는 액세스 모드를 지정합니다. 읽기 권한, 쓰기 권한, 실행 권한을 지정하려면 string 에 각각 r, w, x 문자를 사용하십시오. 기존 사용 권한에 권한을 추가하려면 string 의 첫 머리에 더하기 기호(+)를 지정하십시오. 사용 권한을 제거하려면 string 의 첫 머리에 빼기 기호(-)를 지정하십시오. 접두사를 지정하지 않으면 이전의 사용 권한이 string 으로 재설정됩니다.

owner(userName)

파일의 소유자를 변경합니다. 유효한 UNIX 사용자 이름을 지정합니다.

추가 정보:

find file 명령-네이티브 파일 나열(페이지 172)
showfile 명령-네이티브 파일 속성 표시(페이지 177)
chres 명령-리소스 레코드 수정(페이지 77)

chgrp 명령-UNIX 그룹 수정

기본 UNIX 환경에 해당

UNIX 그룹으로 작업하려면 chgrp, editgrp 및 newgrp 명령을 사용하십시오. 이러한 명령은 구조가 동일하며, 다음의 내용만 다릅니다.

- chgrp 명령은 하나 이상의 UNIX 그룹을 수정합니다.
- editgrp 명령은 하나 이상의 UNIX 그룹을 작성 또는 수정합니다.
- newgrp 명령은 하나 이상의 UNIX 그룹을 작성합니다.

참고: 구성 설정(seos.ini)에 지정된 파일로부터 그룹을 읽고 추가하고 업데이트하고 삭제하게 됩니다. 기본적으로 이 파일은 /etc/group 입니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{{chgrp|cg}|{editgrp|eg}|{newgrp|ng}} groupName \
    [groupid(integer)] \
    [userlist(userNames)]
```

groupid(integer)

그룹의 그룹 ID 를 설정합니다. 그룹의 고유 번호 ID 를 나타내는 양의 정수를 입력하십시오. CA Access Control 은 그룹 ID 로 0 을 허용하지 않습니다.

groupName

수정할 그룹의 이름을 지정합니다. 기존 UNIX 그룹의 이름을 지정합니다. 둘 이상의 그룹을 변경하려면 괄호 안에 그룹 이름 목록을 넣고 그룹 이름을 공백 또는 쉼표로 분리합니다.

userlist(userNames)

새 멤버 목록을 지정합니다. 각 사용자 이름은 이미 UNIX 에 정의되어 있어야 합니다. 목록에 둘 이상의 사용자가 있으면 사용자 이름을 공백 또는 쉼표로 분리합니다. 여기서 지정한 사용자 목록이 그룹에 정의된 이전 사용자 목록을 대체합니다.

추가 정보:

rmusr 명령-UNIX 사용자 삭제(페이지 176)
showusr 명령-네이티브 사용자 속성 표시(페이지 179)
ch[x]grp 명령-그룹 속성 변경(페이지 64)

chusr 명령-UNIX 사용자 수정

기본 UNIX 환경에 해당

UNIX 사용자로 작업하려면 chgusr, editusr 및 newusr 명령을 사용하십시오. 이러한 명령은 구조가 동일하며, 다음의 내용만 다릅니다.

- chusr 명령은 하나 이상의 UNIX 사용자를 수정합니다.
- editusr 명령은 하나 이상의 UNIX 사용자를 작성 또는 수정합니다.
- newusr 명령은 하나 이상의 UNIX 사용자를 작성합니다.

참고: 구성 설정(seos.ini)에 지정된 파일로부터 사용자를 읽고 추가하고 업데이트하고 삭제하게 됩니다. 기본적으로 이 파일은 /etc/passwd 입니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

참고: 이 명령은 CA Access Control 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{{chusr|cu}|{editusr|eu}|{newusr|nu}} userName \
    [enable] \
    [gecos(string)] \
    [homedir({path|nohomedir})] \
    [password(string)] \
    [pgroup(groupName)] \
    [shellprog(path)] \
    [userid(number)]
```

enable

어떤 이유로든 비활성화된 사용자 계정의 로그인을 활성화합니다. 이것은 chusr 및 editusr 매개 변수입니다.

gecos(string)

사용자의 실제 이름과 같이 사용자에 관한 일반적인 주석이 들어 있는 문자열을 지정합니다. 문자열을 작은따옴표로 둘러쌉니다.

homedir(path|nohomedir)

사용자 홈 디렉터리의 전체 경로를 지정합니다. CA Access Control 은 디렉터리를 작성하려고 시도합니다. CA Access Control 이 홈 디렉터리를 성공적으로 작성하는지와 관계없이 UNIX 파일은 업데이트됩니다.

nohomedir 을 지정하면 UNIX 는 사용자에 대해 homedir 을 만들지 않습니다.

password(string)

사용자에게 암호를 할당합니다. 공백 이외의 문자를 지정합니다. 암호는 한 번의 로그인에만 유효합니다. 사용자가 다음번에 시스템에 로그인할 때는 새 암호를 설정해야 합니다.

pgroup(groupName)

사용자의 주 그룹 이름을 지정합니다.

shellprog(path)

사용자가 login 명령이나 su 명령을 호출한 후에 실행되는 초기 프로그램이나 셸의 전체 경로를 지정합니다.

userid(number)

사용자의 고유한 숫자 ID 를 지정합니다. 이 숫자 ID 는 고유한 임의 액세스 제어에 사용됩니다. 100 보다 큰 십진수를 입력하십시오. 100 보다 작은 값은 인정되지 않습니다.

userName

기존 UNIX 사용자의 이름. 둘 이상의 사용자를 변경하려면 괄호 안에 사용자 이름 목록을 넣고 이름을 공백 또는 쉼표로 분리합니다.

추가 정보:

rmusr 명령-UNIX 사용자 삭제(페이지 176)
showusr 명령-네이티브 사용자 속성 표시(페이지 179)
ch[x]usr 명령-사용자 속성 변경(페이지 94)

editfile 명령-UNIX 파일 설정 수정

기본 UNIX 환경에 해당

이 명령은 chfile 명령과 함께 설명되어 있습니다.

추가 정보:

chfile 명령-UNIX 파일 설정 수정(페이지 167)

editgrp 명령-UNIX 그룹 작성 및 수정

기본 UNIX 환경에 해당

이 명령은 chgrp 명령과 함께 설명되어 있습니다.

추가 정보:

chgrp 명령-UNIX 그룹 수정(페이지 168)

editusr 명령-UNIX 사용자 작성 및 수정

기본 UNIX 환경에 해당

이 명령은 chusr 명령과 함께 설명되어 있습니다.

추가 정보:

chusr 명령-UNIX 사용자 수정(페이지 170)

find file 명령-네이티브 파일 나열

네이티브 환경에 해당

문자열인 마스크와 일치하는 모든 시스템 파일을 나열하려면 find file 명령을 사용하십시오. 파일들이 한 열에 시간순으로 정렬됩니다.

이 명령의 형식은 다음과 같습니다.

```
find file [directory][/mask]
```

directory

directory 디렉터리의 모든 파일을 나열합니다.

mask

mask 변수와 일치하는 directory 디렉터리의 모든 파일을 나열합니다. mask 에는 와일드카드 문자가 포함될 수 있습니다.

예: Windows 의 특정 경로에서 실행 프로그램 파일 찾기

다음 명령은 CA Access Control bin 디렉터리에 있는 모든 실행 파일을 나열합니다.

```
find file C:\Program\Files\CA\AccessControl\bin\*.exe
```

예: UNIX 에서 패턴 일치 파일 찾기

다음 명령은 CA Access Control bin 디렉터리에 있는 파일 중 se 문자로 시작되는 모든 파일을 나열합니다.

```
find file /opt/CA/AccessControl/bin/se*
```

join 명령-사용자를 네이티브 그룹에 추가

네이티브 환경에 해당

join 명령은 사용자를 그룹에 추가합니다. 지정된 사용자 및 그룹은 이미 네이티브 OS 에 정의되어 있어야 합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

join 명령을 사용하려면 적어도 다음 중 하나가 참이어야 합니다.

- CA Access Control 사용자 레코드에 ADMIN 특성을 가지고 있습니다.
- 그룹 레코드는 GROUP-ADMIN 특성을 가진 그룹 범위 내에 있습니다.
- 데이터베이스에 있는 그룹 레코드의 소유자입니다.
- ADMIN 클래스의 GROUP 레코드의 액세스 제어 목록에 JOIN 또는 MODIFY 액세스 권한을 갖습니다.

참고: ADMIN 이 CA Access Control GROUP 레코드 및 네이티브 그룹을 수정할 권한을 가지려면 MODIFY 및 JOIN 속성이 모두 필요합니다.

이 명령의 형식은 다음과 같습니다.

```
{join|j} userName group(groupName)
```

group(groupName)

사용자가 추가되는 네이티브 그룹을 지정합니다.

userName

group 매개 변수로 지정한 그룹에 연결되어 있는 네이티브 사용자의 사용자 이름을 지정합니다. 둘 이상의 사용자를 지정하려면 괄호 안에 사용자 이름을 넣고 사용자 이름을 공백 또는 쉼표로 분리합니다.

예제

사용자 Eli 는 사용자 Bob 을 "staff" 그룹에 조인하려고 합니다.

- Eli 는 ADMIN 특성을 갖고 있으며 현재 환경은 native 입니다.

```
join Bob group(staff)
```

추가 정보:

join- 명령-네이티브 그룹에서 사용자 제거(페이지 174)
showgrp 명령-네이티브 그룹 속성 표시(페이지 178)
showusr 명령-네이티브 사용자 속성 표시(페이지 179)
join[x] 명령-내부 그룹에 사용자 추가(페이지 125)

join- 명령-네이티브 그룹에서 사용자 제거

네이티브 환경에 해당

join- 명령은 그룹에서 사용자를 제거합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

join- 명령을 사용하려면 다음 조건 중 하나가 참이어야 합니다.

- 사용자가 ADMIN 특성을 가집니다.
- 그룹 레코드는 GROUP-ADMIN 특성을 가진 그룹 범위 내에 있습니다.
- 데이터베이스에 있는 그룹 레코드의 소유자입니다.
- ADMIN 클래스의 GROUP 레코드의 액세스 제어 목록에 JOIN 또는 MODIFY 액세스 권한을 갖습니다.

사용자 프로필의 소유권만 가지고 있는 경우 사용자를 그룹에서 삭제할 수 있는 권한이 충분하지 않습니다. ADMIN 이 CA Access Control 레코드와 네이티브 그룹을 수정할 권한을 가지려면 MODIFY 및 JOIN 속성이 모두 필요합니다.

이 명령의 형식은 다음과 같습니다.

```
{join-|j-} userName group(groupName)
```

group(groupName)

사용자를 제거할 네이티브 그룹을 지정합니다.

userName

그룹에서 제거하려는 사용자의 사용자 이름을 지정합니다. 그룹에서 둘 이상의 사용자를 제거하려면 괄호 안에 사용자 이름 목록을 넣고 사용자 이름을 공백 또는 쉼표로 분리합니다.

예제

사용자 Bill 은 사용자 sales25 와 sales43 을 PAYROLL 그룹에서 제거하려고 합니다.

- 사용자 Bill 은 ADMIN 특성을 갖고 있으며 현재 환경은 native 입니다.

```
join- (sales25 sales43) group(PAYROLL)
```

추가 정보:

join 명령-사용자를 네이티브 그룹에 추가(페이지 173)
showgrp 명령-네이티브 그룹 속성 표시(페이지 178)

showusr 명령-네이티브 사용자 속성 표시(페이지 179)
join[x]- 명령-그룹에서 사용자 제거(페이지 128)

newgrp 명령-UNIX 그룹 작성

기본 UNIX 환경에 해당

이 명령은 chgrp 명령과 함께 설명되어 있습니다.

추가 정보:

chgrp 명령-UNIX 그룹 수정(페이지 168)

newusr 명령-UNIX 사용자 작성

기본 UNIX 환경에 해당

이 명령은 chusr 명령과 함께 설명되어 있습니다.

추가 정보:

chusr 명령-UNIX 사용자 수정(페이지 170)

rmgrp 명령-UNIX 그룹 삭제

기본 UNIX 환경에 해당

rmgrp 명령은 UNIX 시스템에서 하나 이상의 그룹을 삭제합니다. seos.ini 파일에 지정된 파일에서 그룹을 제거합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

참고: 구성 설정(seos.ini)에 지정된 파일로부터 그룹을 읽고 추가하고 업데이트하고 삭제하게 됩니다. 기본적으로 이 파일은 /etc/group 입니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

이 명령의 형식은 다음과 같습니다.

```
{rmgrp|rg} groupName
```

groupName

삭제할 그룹의 이름을 지정합니다. 그룹 이름은 기존의 UNIX 그룹 이름이어야 합니다. 하나 이상의 그룹 이름을 지정합니다. 둘 이상의 그룹을 제거하려면 괄호 안에 그룹 이름 목록을 넣고 그룹 이름을 공백 또는 쉼표로 분리합니다.

추가 정보:

chgrp 명령-UNIX 그룹 수정(페이지 168)
showgrp 명령-네이티브 그룹 속성 표시(페이지 178)
rm[x]grp 명령-그룹 레코드 삭제(페이지 132)

rmusr 명령-UNIX 사용자 삭제

기본 UNIX 환경에 해당

rmusr 명령은 UNIX 시스템에서 하나 이상의 사용자를 삭제합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

참고: 구성 설정(seos.ini)에 지정된 파일로부터 사용자를 읽고 추가하고 업데이트하고 삭제하게 됩니다. 기본적으로 이 파일은 /etc/passwd 입니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

이 명령의 형식은 다음과 같습니다.

```
{rmusr|ru} userName
```

userName

기존 UNIX 사용자의 사용자 이름을 지정합니다. 둘 이상의 사용자를 제거하려면 괄호 안에 사용자 이름 목록을 넣고 사용자 이름을 공백 또는 쉼표로 분리합니다.

추가 정보:

chusr 명령-UNIX 사용자 수정(페이지 170)
showusr 명령-네이티브 사용자 속성 표시(페이지 179)
rm[x]usr 명령-사용자 레코드 삭제(페이지 135)

showfile 명령-네이티브 파일 속성 표시

네이티브 환경에 해당

showfile 명령은 하나 이상의 시스템 파일에 대한 네이티브 세부 정보를 나열합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{showfile|sf} fileName [next] \
    [{props|addprops}(propNames)]
```

addprops(propName)

표시할 속성(ruler)을 설정합니다. 속성 목록이 현재 ruler 에 추가됩니다. ruler 는 현재 쿼리용으로만 설정되고 이전에 설정된 ruler 로 되돌아갑니다.

fileName

세부 정보를 나열할 파일의 이름을 지정합니다. 하나 이상의 UNIX 파일 이름을 입력합니다. 둘 이상의 파일을 지정하려면 괄호 안에 파일 이름 목록을 넣고 각 이름을 공백 또는 쉼표로 분리합니다.

next

요청된 데이터의 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. 쿼리 크기의 기본값은 100 으로 설정됩니다.

props(all|propName)

표시할 속성(ruler)을 설정합니다.

ruler 는 이후 쿼리에 대해서 설정된 상태를 유지합니다.

예: UNIX 파일의 세부 정보 표시

UNIX 파일 /tmp/foo 의 세부 정보를 나열하려고 합니다.

```
showfile /tmp/foo
```

예: Windows 파일의 소유자 표시

Windows 파일 C:\tmp\foo.exe 의 소유자가 누구인지 알고자 합니다.

```
showfile C:\tmp\foo.exe props(Owner)
```

추가 정보:

chfile 명령-UNIX 파일 설정 수정(페이지 167)
chfile 명령-Windows 파일 설정 수정(페이지 184)
showfile 명령-파일 속성 표시(페이지 146)

showgrp 명령-네이티브 그룹 속성 표시

네이티브 환경에 해당

showgrp 명령은 네이티브 운영 체제에 있는 하나 이상의 그룹에 대한 세부 정보를 표시합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

참고: UNIX 에서는 구성 설정(seos.ini)에 지정된 파일로부터 그룹을 읽고 추가하고 업데이트하고 삭제하게 됩니다. 기본적으로 이 파일은 /etc/group 입니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

이 명령의 형식은 다음과 같습니다.

```
{showgrp|sg} groupName [next] \
    [{props|addprops}(propNames)]
```

addprops(propName)

표시할 속성(ruler)을 설정합니다. 속성 목록이 현재 ruler 에 추가됩니다. ruler 는 현재 쿼리용으로만 설정되고 이전에 설정된 ruler 로 되돌아갑니다.

groupName

세부 사항이 표시되는 그룹의 이름을 지정합니다. 그룹 이름은 기존의 네이티브 그룹 이름이어야 합니다. 하나 이상의 그룹 이름을 지정합니다. 둘 이상의 그룹을 나열하려면 괄호 안에 그룹 이름 목록을 넣고 그룹 이름을 공백 또는 쉼표로 분리합니다.

next

요청된 데이터의 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. 쿼리 크기의 기본값은 100 으로 설정됩니다.

props(all|propName)

표시할 속성(ruler)을 설정합니다.

ruler 는 이후 쿼리에 대해서 설정된 상태를 유지합니다.

예제

unix 환경에 있을 때 UNIX 그룹 security 의 세부 정보를 나열하려면 다음 명령을 입력합니다.

```
showgrp security
```

추가 정보:

chgrp 명령-UNIX 그룹 수정(페이지 168)
chgrp 명령-Windows 그룹 수정(페이지 185)
show[x]grp 명령-그룹 속성 표시(페이지 148)

showusr 명령-네이티브 사용자 속성 표시

기본 UNIX 환경에 해당

showusr 명령은 기본 운영 체제에서 정의된 하나 이상의 사용자 속성을 표시합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

참고: UNIX 에서는 구성 설정(seos.ini)에 지정된 파일로부터 사용자를 읽고 추가하고 업데이트하고 삭제하게 됩니다. 기본적으로 이 파일은 /etc/passwd 입니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

이 명령의 형식은 다음과 같습니다.

```
{showusr|su} userName [next] \
    [{props|addprops}(propNames)]
```

addprops(propName)

표시할 속성(ruler)을 설정합니다. 속성 목록이 현재 ruler 에 추가됩니다. ruler 는 현재 쿼리용으로만 설정되고 이전에 설정된 ruler 로 되돌아갑니다.

userName

기본 속성이 표시되는 사용자의 이름을 지정합니다. 기존의 네이티브 사용자 이름을 지정합니다. 둘 이상의 사용자의 속성을 나열하려면 괄호 안에 사용자 이름 목록을 넣고 이름을 공백 또는 쉼표로 분리합니다.

next

요청된 데이터의 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. 쿼리 크기의 기본값은 100 으로 설정됩니다.

props(all|propName)

표시할 속성(ruler)을 설정합니다.

ruler 는 이후 쿼리에 대해서 설정된 상태를 유지합니다.

예제

unix 환경에 있을 때 UNIX 사용자 leslie 의 세부 정보를 나열하려면 다음 명령을 입력합니다.

```
showusr leslie
```

추가 정보:

chusr 명령-UNIX 사용자 수정(페이지 170)
chusr 명령-Windows 사용자 수정(페이지 190)
show[x]usr 명령-사용자 속성 표시(페이지 152)

네이티브 Windows 환경의 selang 명령

이 단원에는 네이티브 Windows 환경에서 작동하는 모든 selang 명령에 대한 완전한 참조가 알파벳순으로 나열되어 있습니다.

권한 부여 명령-Windows 리소스에 Access 하기 위한 Accessor 의 권한 설정

네이티브 Windows 환경에 해당

권한 부여 명령은 특정 리소스에 대한 액세스 권한이 있는 사용자 및 그룹 목록을 유지관리합니다. authorize 를 사용하면 다음을 수행하도록 목록을 변경할 수 있습니다.

- 특정 CA Access Control 사용자 또는 그룹의 리소스에 대한 액세스를 허용합니다.
- 특정 CA Access Control 사용자 또는 그룹의 리소스에 대한 액세스를 차단합니다.
- 특정 사용자 또는 그룹의 리소스에 대한 액세스 권한 수준을 변경합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

다음 Windows 환경 클래스는 ACL 을 지원하고 권한 부여 명령으로 제어될 수 있습니다.

- COM
- DISK
- FILE
- PRINTER
- REGKEY
- SHARE

목록에 나타나지 않는 클래스에는 액세스 제어 목록이 없으므로 권한 부여 명령으로 제어될 수 없습니다.

이 명령의 형식은 다음과 같습니다.

```
{authorize|auth} className resourceName \
    [access(accessValue)|deniedaccess(accessvalue)] \
    [gid(groupName, ...)] \
    [uid(userName, ...)]
```

access(accessValue)

uid 또는 gid 매개 변수에서 사용자가 식별하는 접근자에 부여하려는 리소스에 대한 액세스 권한을 지정합니다.

className

resourceName 이 속한 클래스의 이름을 지정합니다.

deniedaccess(accessvalue)

uid 또는 gid 매개 변수에서 사용자가 식별하는 접근자에 부여하려는 리소스에 대한 부정적인 액세스 권한을 지정합니다.

거부된 accessvalue 는 all, create, delete, join, modify, none, password 또는 read 가 될 수 있습니다.

참고: authorize-가 아닌 권한 부여 명령에서만 accessValue 를 사용할 수 있습니다.

gid(groupName)

설정 중인 리소스에 대한 액세스 권한을 가지는 Windows 그룹을 지정합니다. groupName 값은 하나 이상의 Windows 그룹 이름을 나타냅니다. 하나 이상의 그룹을 지정하는 경우 그룹 이름을 공백이나 쉼표로 구분하십시오.

resourceName

수정 또는 추가할 리소스 레코드의 이름. 둘 이상의 리소스를 변경 또는 추가할 경우 리소스 이름 목록을 괄호로 묶고 공백이나 쉼표로 리소스 이름을 구분합니다. 최소한 하나 이상이 리소스 이름을 지정해야 합니다.

CA Access Control 은 지정된 매개 변수에 따라 각 리소스 레코드를 독립적으로 처리합니다. 리소스를 처리할 때 오류가 발생하면 CA Access Control 은 메시지를 작성하고 목록의 다음 리소스를 계속 처리합니다.

uid(userName)

리소스에 대한 액세스 권한을 설정 중인 Windows 사용자를 지정합니다. userName 은 하나 이상의 Windows 사용자 이름입니다. 둘 이상의 사용자를 지정하는 경우, 사용자 이름을 공백 또는 쉼표로 구분합니다. Windows 에 정의된 모든 사용자를 지정하려면 userName 에 대해 별표(*)를 지정하십시오.

추가 정보:

authorize- 명령-Windows 리소스에 액세스하는 접근자의 권한 제거(페이지 183)
chfile 명령-Windows 파일 설정 수정(페이지 184)
chgrp 명령-Windows 그룹 수정(페이지 185)
chres 명령-Windows 리소스 수정(페이지 187)
chusr 명령-Windows 사용자 수정(페이지 190)
권한 부여 명령-리소스에 대한 액세스 권한 설정(페이지 46)
클래스별 Windows 액세스 권한(페이지 32)

authorize- 명령-Windows 리소스에 Access 하는 Accessor 의 권한 제거

네이티브 Windows 환경에 해당

authorize- 명령은 접근자를 표준 Access Control 목록에서 삭제하여 리소스에 대한 액세스 권한을 제거합니다. 이렇게 하면 특정 리소스에 액세스하는 접근자의 권한을 기본 액세스가 결정합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{authorize-|auth-} className resourceName \
    [gid(groupName, ...)] \
    [uid(userName, ...)]
```

className

resourceName 이 속한 클래스의 이름을 지정합니다.

gid(groupName)

설정 중인 리소스에 대한 액세스 권한을 가지는 Windows 그룹을 지정합니다. groupName 값은 하나 이상의 Windows 그룹 이름을 나타냅니다. 하나 이상의 그룹을 지정하는 경우 그룹 이름을 공백이나 쉼표로 구분하십시오.

resourceName

수정 또는 추가할 리소스 레코드의 이름을 지정합니다. 둘 이상의 리소스를 변경 또는 추가할 경우 리소스 이름 목록을 괄호로 묶고 공백이나 쉼표로 리소스 이름을 구분합니다. 최소한 하나 이상이 리소스 이름을 지정해야 합니다.

CA Access Control 은 지정된 매개 변수에 따라 각 리소스 레코드를 독립적으로 처리합니다. 리소스를 처리할 때 오류가 발생하면 CA Access Control 은 메시지를 작성하고 목록의 다음 리소스를 계속 처리합니다.

uid(userName)

리소스에 대한 액세스 권한을 설정 중인 Windows 사용자를 지정합니다. userName 은 하나 이상의 Windows 사용자 이름입니다. 둘 이상의 사용자를 지정하는 경우, 사용자 이름을 공백 또는 쉼표로 구분합니다. Windows 에 정의된 모든 사용자를 지정하려면 userName 에 대해 별표(*)를 지정하십시오.

추가 정보:

권한 부여 명령-Windows 리소스에 액세스하기 위한 접근자의 권한 설정(페이지 181)
chfile 명령-Windows 파일 설정 수정(페이지 184)
chgrp 명령-Windows 그룹 수정(페이지 185)

chres 명령-Windows 리소스 수정(페이지 187)
chusr 명령-Windows 사용자 수정(페이지 190)
authorize- 명령-리소스에서 액세스 권한 제거(페이지 51)

chfile 명령-Windows 파일 설정 수정

네이티브 Windows 환경에 해당

chfile 명령과 editfile 명령은 동일합니다. 이 두 명령은 하나 이상의 Windows 파일을 수정합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

NTFS 파일 시스템에 대한 이 명령의 형식은 다음과 같습니다.

```
{{chfile|cf}|{editfile|ef}} fileName \
    [attrib(attributeValue)] \
    [attrib(-attributeValue)] \
    [defaccess(accessValue)] \
    [owner(userName|groupName)]
```

FAT 파일 시스템에 대한 이 명령의 형식은 다음과 같습니다.

```
{{chfile|cf}|{editfile|ef}} fileName \
    [attrib([-]attributeValue)]
```

attrib([-]attributeValue)

파일의 문자를 결정하는 일련의 특성을 지정합니다. value 인수 앞에 빼기 부호(–)가 있으면 이 매개 변수는 특성을 제거합니다.

defaccess(accessValue)

Native 보안 내장 그룹 Everyone 에 대한 액세스 권한을 지정합니다. 모든 시스템 사용자는 Everyone 그룹의 구성원입니다. Everyone 그룹에게 액세스를 제공하는 것에는 권한이 부여된 모든 사용자 외에도 잠재적인 모든 익명의 사용자도 포함됩니다.

참고: CA Access Control 환경에서 정의된 개체에 대한 Defaccess 는 다른 의미를 가집니다. 기본 액세스 권한은, 리소스에 대한 액세스를 요청하지만 리소스의 CA Access Control 목록에 없는 접근자에 부여된 권한입니다. 또한 기본 액세스는 CA Access Control 에 정의되지 않은 사용자에게 적용됩니다.

defaccess 매개 변수는 NTFS 파일 시스템에만 적용됩니다.

owner(userName|groupName)

사용자나 그룹을 파일 레코드의 소유자로 할당합니다. 파일 레코드의 소유자는 파일에 무제한 액세스할 수 있습니다. 파일 소유자는 항상 파일 레코드를 업데이트하거나 삭제할 수 있습니다.

추가 정보:

showfile 명령-네이티브 파일 속성 표시(페이지 177)
chfile 명령-파일 레코드 수정(페이지 58)
Windows 파일 특성(페이지 485)

chgrp 명령-Windows 그룹 수정

네이티브 Windows 환경에 해당

Windows 그룹으로 작업하려면 chgrp, editgrp 및 newgrp 명령을 사용하십시오. 이러한 명령은 구조가 동일하며, 다음의 내용만 다릅니다.

- chgrp 명령은 하나 이상의 Windows 그룹을 수정합니다.
- editgrp 명령은 하나 이상의 Windows 그룹을 작성 또는 수정합니다.
- newgrp 명령은 하나 이상의 Windows 그룹을 작성합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

둘 이상의 그룹을 정의하거나 둘 이상의 그룹 속성을 변경하려면 괄호 안에 그룹 이름 목록을 넣고 공백 또는 쉼표로 그룹 이름을 구분합니다.

참고: 그룹에서 멤버를 추가하거나 제거하려면 join 또는 join- 명령을 사용하십시오.

이 명령의 형식은 다음과 같습니다.

```
{{chgrp|cg}|{editgrp|eg}|{newgrp|ng}} groupName \
    [global] \
    [comment(string)|comment-] \
    [privileges(privList)] \
    [privileges(-privList)] \
    [rename_group]
```

comment(string)

최대 255 자로 구성된 영숫자 설명 문자열을 그룹 레코드에 추가합니다. 이전에 설명 문자열을 그룹 레코드에 추가한 경우 여기서 지정한 새 문자열이 기존 문자열을 바꿉니다. 문자열이 공백을 포함하는 경우, 작은 따옴표를 전체 문자열 앞뒤에 입력하십시오.

표준 Windows 그룹은 시스템 설치 시 설명이 추가됩니다. Windows 및 AC 환경에서 모두 새 그룹을 작성할 경우 CA Access Control 은 "CA Access Control 그룹" 설명을 삽입합니다.

global

전역 그룹을 나타냅니다. 각 그룹 이름은 고유해야 하며 현재 Windows 데이터베이스에 존재할 수 없습니다. Windows 에서는 그룹 및 사용자가 동일한 이름을 공유할 수 없습니다.

참고: 전역 그룹을 작성하고 CA Access Control 버전 4.1 의 서비스를 사용할 때 ~groupName 을 사용하십시오. 버전 4.1 이상은 이전 버전과의 호환성을 위해 이 형식을 지원합니다.

groupName

newgrp 명령의 경우, 데이터베이스에 추가된 그룹 레코드의 이름을 지정합니다. 각 그룹 이름은 고유해야 하고 Windows 데이터베이스에 현재 존재하지 않아야 합니다. CA Access Control 데이터베이스와 달리 Windows 에서는 그룹 및 사용자가 동일한 이름을 공유할 수 없습니다.

chgrp 명령의 경우 변경 중인 속성을 가지는 그룹의 이름을 지정합니다.

둘 이상의 그룹을 정의하거나 둘 이상의 그룹 속성을 변경하려면 괄호 안에 그룹 이름 목록을 넣고 공백 또는 쉼표로 그룹 이름을 구분합니다.

privileges(privList|-privList)

Windows 그룹 레코드에 특정 권한을 추가합니다. 또는 privList 앞에 빼기 부호(-)가 있으면 지정된 권한을 제거합니다. 유효한 값은 네이티브 Windows 에서 사용할 수 있는 임의의 권한입니다.

이 매개 변수는 chgrp 또는 editgrp 명령과 함께만 지정할 수 있으며, 기존 그룹 레코드를 변경할 경우에만 지정할 수 있습니다. 새 그룹 레코드를 작성할 때 이 매개 변수를 사용하여 권한을 할당할 수 없습니다.

rename_group

Windows 데이터베이스의 그룹 계정 이름을 변경합니다. 이전 그룹 이름의 모든 속성이 이름을 변경한 그룹 계정에 적용됩니다. 각 그룹 이름은 고유해야 하며 Windows 데이터베이스에 존재해야 합니다. CA Access Control 데이터베이스와 달리 Windows 에서는 그룹 및 사용자가 동일한 이름을 공유할 수 없습니다.

참고: CA Access Control 을 Active Directory 와 함께 Windows 2000 에 설치하면, CA Access Control 은 이전 Windows 2000 그룹 이름을 변경합니다.

chres 명령-Windows 리소스 수정

네이티브 Windows 환경에 해당

Windows 환경에서 CA Access Control 클래스에 속한 리소스 레코드로 작업하려면 chres, editres 및 newres 명령을 사용하십시오. 이러한 명령은 구조가 동일하며, 다음의 내용만 다릅니다.

- chres 명령은 하나 이상의 리소스를 수정합니다.
- editres 명령은 하나 이상의 리소스를 작성 또는 수정합니다.
- newres 명령은 하나 이상의 리소스를 작성합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{{chres|cr}|{editres|er}|{newres|nr}} className resourceName \
    [comment(string)|comment-] \
    [defaccess(accessValue)] \
    [dword(integer)|string(string)|binary(hexastring)|multistring(string)] \
    [location(string)|location()] \
    [maxusers(integer)] \
    [owner(userName|groupName)] \
    [share_name(string)|sharename-]
```

또는

```
{{chres|cr}|{editres|er}|{newres|nr}} \
    DOMAIN resourceName \
    [computer(workstationName)|computer-(workstationName)] \
    [domainpwd(connectPassword)] \
    [trusted(domainName)|trusted-(domainName)]
```

binary(hexastring)

레지스트리 키의 값을 지정합니다(16 진수일 경우).

className

resourceName 이 속한 클래스의 이름을 지정합니다.

newres 명령의 경우 유효한 값은 REGKEY, REGVAL, OU 및 SHARE 입니다. chres 및 editres 명령의 경우 유효한 값은 COM, DISK, DOMAIN, FILE, PRINTER, REGKEY, REGVAL, SERVICE, DEVICE, SESSION, OU 및 SHARE 입니다.

comment(string)

리소스 레코드에 설명 문자열을 추가합니다. 이전에 설명 문자열을 리소스 레코드에 추가한 경우 여기서 지정한 새 문자열이 기존 문자열을 바꿉니다. 이 매개 변수는 SHARE 및 PRINTER 리소스에 대해서만 유효합니다.

computer(workstationName)|computer-(workstationName)

도메인에 추가 중인 워크스테이션의 이름을 지정합니다. 또는 인수 앞에 빼기 부호가 있으면 도메인에서 제거 중인 워크스테이션의 이름을 지정합니다. 이 매개 변수는 DOMAIN 리소스와 함께만 사용할 수 있습니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 지정할 수 있습니다.

defaccess(accessValue)

Native 보안 내장 그룹 Everyone 에 대한 액세스 권한을 지정합니다. 모든 시스템 사용자는 Everyone 그룹의 구성원입니다. Everyone 그룹에게 액세스를 제공하는 것에는 권한이 부여된 모든 사용자 외에도 잠재적인 모든 익명의 사용자도 포함됩니다.

참고: CA Access Control 환경에서 정의된 개체에 대한 Defaccess 는 다른 의미를 가집니다. 기본 액세스 권한은, 리소스에 대한 액세스를 요청하지만 리소스의 CA Access Control 목록에 없는 접근자에 부여된 권한입니다. 또한 기본 액세스는 CA Access Control 에 정의되지 않은 사용자에게 적용됩니다.

defaccess 매개 변수는 NTFS 파일 시스템에만 적용됩니다.

domainpwd(connectPassword)

트러스트 관계를 변경할 때 관리자가 입력해야 할 암호를 지정합니다.

이 매개 변수는 DOMAIN 리소스와 함께만 사용할 수 있습니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 지정할 수 있습니다.

dword(integer)

레지스트리 키의 값을 지정합니다(정수일 경우).

gen_prop(propertyName)

OU 클래스의 속성을 지정합니다.

이 매개 변수는 OU 클래스에 대해서만 유효합니다.

gen_value(valueName)

OU 클래스의 속성 값을 지정합니다.

이 매개 변수는 OU 클래스에 대해서만 유효합니다.

location(string)

프린터의 위치를 나타냅니다. 공백 있는 ( )를 사용하여 이 속성을 제거합니다.

이 매개 변수는 PRINTER 리소스에 대해서만 유효합니다.

maxusers(integer)

공유 디렉터리에 한 번에 연결할 수 있는 최대 사용자 수(integer)를 지정합니다.

이 매개 변수는 SHARE 리소스에 대해서만 유효합니다.

multistring(string)

레지스트리 키의 값을 지정합니다(다중 문자열일 경우).

owner(userName|groupName)

사용자나 그룹을 리소스 레코드의 소유자로 할당합니다. 리소스 레코드의 소유자는 리소스에 무제한 액세스할 수 있습니다. 리소스 소유자는 항상 리소스 레코드를 업데이트 및 삭제할 수 있습니다. 자세한 내용은 Windows 용 끝점 관리 안내서를 참조하십시오.

FAT 파일 시스템의 FILE 또는 SHARE 레코드에 대해서는 owner 매개 변수를 지정할 수 없습니다. 또한 이 매개 변수는 DEVICE, DOMAIN, OU, PROCESS, REGVAL, SERVICE 및 SESSION 리소스에 대해서 유효하지 않습니다.

resourceName

수정 또는 추가할 리소스 레코드의 이름. 둘 이상의 리소스를 변경 또는 추가할 경우 리소스 이름 목록을 괄호로 묶고 공백이나 쉼표로 리소스 이름을 구분합니다. 최소한 하나 이상이 리소스 이름을 지정해야 합니다.

CA Access Control 은 지정된 매개 변수에 따라 각 리소스 레코드를 독립적으로 처리합니다. 리소스를 처리할 때 오류가 발생하면 CA Access Control 은 메시지를 작성하고 목록의 다음 리소스를 계속 처리합니다.

share_name(shareName)|share_name-

프린터의 공유 지점을 식별합니다.

이 매개 변수는 PRINTER 리소스에 대해서만 유효합니다.

string(string)

레지스트리 키의 값을 지정합니다(문자열일 경우).

trusted(domainName) | trusted-(domainName)

트러스트된 도메인에 추가 중인 도메인의 이름을 지정합니다. 또는 인수 앞에 빼기 부호가 있으면 언트러스트된 도메인의 이름을 지정합니다. 이 매개 변수는 DOMAIN 리소스와 함께만 사용할 수 있습니다. 이 매개 변수는 chres 또는 editres 명령과 함께만 지정할 수 있습니다.

추가 정보:

rmres 명령-Windows 리소스 삭제(페이지 202)
showres 명령-네이티브 리소스 속성 표시(페이지 207)
chres 명령-리소스 레코드 수정(페이지 77)

chusr 명령-Windows 사용자 수정

네이티브 Windows 환경에 해당

Windows 사용자로 작업하려면 chgusr, editusr 및 newusr 명령을 사용하십시오. 이러한 명령은 구조가 동일하며, 다음의 내용만 다릅니다.

- chusr 명령은 하나 이상의 Windows 사용자를 수정합니다.
- editusr 명령은 하나 이상의 Windows 사용자를 작성 또는 수정합니다.
- newusr 명령은 하나 이상의 Windows 사용자를 작성합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{{chusr|cu}|{editusr|eu}|{newusr|nu}} userName \
    [comment(string)|comment-] \
    [country(string)] \
    [expire|expire(mm/dd/yy[@hh:mm])|expire-] \
    [flags{(accountFlags)|-(accountFlags)}] \
    [full_name(fullName)] \
    [homedir(homeDir)] \
    [homedrive(homeDrive)] \
    [location(string)] \
    [logonserver(serverName)] \
    [organization(name)] \
    [org_unit(name)] \
    [password(password)] \
    [pgroup(primaryGroup)] \
    [phone(string)] \
    [privileges(privList)] \
```

```
[profile(path)] \
[rename_user]
[restrictions( \
    days({[mon] [tue] [wed] [thu] [fri] [sat] [sun]}|anyday|weekdays) \
    time(startTime:endTime|anytime))]\
[restrictions-] \
[resume[(date)]|resume-} \
[script(logonScriptPath)] \
[suspend[(date)] | suspend-] \
[terminals(terminalList)|terminals-(terminalList)] \
[workstations(workstationList)|workstations-(workstationList)|workstations-]
```

comment(string)|comment-

사용자 레코드에 설명 문자열을 할당합니다.

인수는 최대 255 자로 구성된 영숫자 문자열입니다. 문자열이 공백을 포함하는 경우, 작은 따옴표를 전체 문자열 앞뒤에 입력하십시오.

country(string)

사용자가 있는 국가를 지정합니다. 이 문자열은 권한 부여 프로세스 중에 사용되지 않습니다.

인수는 최대 19 자로 구성된 영숫자 문자열입니다. 문자열이 공백을 포함하는 경우, 작은 따옴표를 전체 문자열 앞뒤에 입력하십시오.

expire|expire(mm/dd/yy[@hh:mm) | expire-

사용자 계정이 만료되는 날짜를 설정합니다. 사용자가 현재 로그인하지 않은 상태에서 날짜를 지정하지 않으면 계정이 즉시 만료됩니다. 사용자가 로그인한 상태이면 사용자가 로그아웃했을 때 계정이 만료됩니다.

newusr 명령과 함께 사용된 expire-는 만료일이 없는 사용자 계정을 정의합니다. chusr 및 editusr 명령의 경우 지정된 사용자 계정에서 만료일을 제거합니다.

날짜 인수의 형식은 mm/dd/yy [@hh:mm]입니다.

flags(accountFlags|- accountFlags)

사용자 계정의 특정 특성을 지정합니다. 유효한 플래그 값의 목록을 보려면 부록 "Windows 값"을 참조하십시오.

사용자 레코드에서 플래그를 제거하려면 빼기 부호(-)를 accountFlags 앞에 붙이십시오.

full_name(fullName)

사용자 레코드와 관련된 사용자의 전체 이름을 지정합니다.

인수는 최대 256 자로 구성된 영숫자 문자열입니다. 문자열이 공백을 포함하는 경우, 작은 따옴표를 전체 문자열 앞뒤에 입력하십시오.

gecos(string)

사용자의 전체 이름과 같은 사용자에 대한 주석 문자열을 지정합니다. 문자열을 작은따옴표로 둘러쌉니다.

homedir(homeDir)

사용자의 홈 디렉터리를 지정합니다. 사용자는 자신의 홈 드라이브와 홈 디렉터리에 자동으로 로그인합니다.

homedrive(homeDrive)

사용자 홈 디렉터리의 드라이브를 지정합니다. 사용자는 자신의 홈 드라이브와 홈 디렉터리에 자동으로 로그인합니다.

location(string)

사용자의 위치를 지정합니다. 이 문자열은 권한 부여 프로세스 중에 사용되지 않습니다.

인수는 최대 19 자로 구성된 영숫자 문자열입니다. 문자열이 공백을 포함하는 경우, 작은 따옴표를 전체 문자열 앞뒤에 입력하십시오.

logonserver(serverName)

사용자에 대한 로그인 정보를 확인하는 서버를 지정합니다. 사용자가 도메인 워크스테이션에 로그인하면 CA Access Control 은 로그인 정보를 서버로 전송하며, 이 서버는 사용자가 작업할 수 있도록 워크스테이션 권한을 부여합니다.

organization(name)

사용자가 근무하는 조직을 지정합니다. 이 정보는 권한 부여 프로세스 중에 사용되지 않습니다.

인수는 최대 256 자로 구성된 영숫자 문자열입니다. 문자열이 공백을 포함하는 경우, 작은 따옴표를 전체 문자열 앞뒤에 입력하십시오.

org_unit(name)

사용자가 근무하는 조직 단위를 지정합니다. 이 정보는 권한 부여 프로세스 중에 사용되지 않습니다.

인수는 최대 256 자로 구성된 영숫자 문자열입니다. 문자열이 공백을 포함하는 경우, 작은 따옴표를 전체 문자열 앞뒤에 입력하십시오.

password(password)

사용자에게 암호를 할당합니다. 암호 검사가 활성화된 경우 암호는 하나의 로그인에 대해서만 유효합니다. 사용자가 다음번에 시스템에 로그인할 때는 새 암호를 설정해야 합니다.

인수는 최대 14 자로 구성된 문자열이고, 공백 또는 쉼표가 들어갈 수 없습니다. 암호 검사가 활성화된 경우 암호는 하나의 로그인에 대해서만 유효합니다. 사용자가 다음번에 시스템에 로그인할 때 "암호 사용 기간 제한 없음" 플래그를 설정하지 않은 경우 사용자는 새 암호를 설정해야 합니다.

자신의 암호를 변경하려면 setoptions cng_ownpwd 를 사용하여 selang 옵션을 설정하거나 sepass 를 사용해야 합니다.

Windows NT 시스템에서 사용자 암호를 설정하는 경우, 다음 메시지가 나타날 수 있습니다.

암호가 요구된 것보다 짧습니다.

이 오류는 암호가 정책 요구 사항에 맞지 않는다는 의미입니다. 오류 발생 원인은 다음 중 하나입니다.

- 암호가 필요한 길이보다 짧거나 깁니다.
- 최근에 사용된 적이 있으며 Windows NT 변경 내역 필드에 존재하는 암호입니다.
- 암호의 고유 문자가 부족합니다.
- 암호가 다른 암호 정책 요구 사항(예: CA Access Control 암호 정책에 따라 설정된 요구 사항)과 맞지 않습니다.

이 오류를 방지하려면, 적용되는 모든 요구 사항에 적합한 암호를 설정해야 합니다.

pgroup(primaryGroup)

사용자의 주 그룹 ID 를 설정합니다. 주 그룹은 사용자가 정의된 그룹 중 하나이며 전역 그룹이어야 합니다.

인수는 최대 14 자로 구성된 문자열이고, 공백 또는 쉼표가 들어갈 수 없습니다.

phone(string)

사용자의 전화 번호를 지정합니다. 이 정보는 권한 부여 프로세스 중에 사용되지 않습니다.

privileges(privList)

Windows 사용자 레코드에 특정 권한을 추가하거나, privList 앞에 빼기 부호(-)가 있는 경우 지정된 권한을 제거합니다. 이 매개 변수는 chusr 또는 editusr 명령과 함께만 지정할 수 있으며, 기존 사용자 레코드를 변경할 경우에만 지정할 수 있습니다. 새 사용자 레코드를 작성할 때 이 매개 변수를 사용하여 권한을 할당할 수 없습니다.

profile(path)

데스크톱 환경에 대한 사용자의 프로필(프로그램 그룹, 네트워크 연결)을 포함하는 파일의 전체 경로 위치를 지정합니다. 사용자가 워크스테이션에 로그인할 때마다 동일한 환경이 화면에 표시됩니다.

rename_user

Windows 데이터베이스의 사용자 계정 이름을 변경합니다. 이전 사용자 이름의 모든 속성은 이름을 변경한 사용자 계정에 적용됩니다. 각 사용자 이름은 고유해야 하며 Windows 데이터베이스에 존재해야 합니다. CA Access Control 데이터베이스와 달리 Windows 에서는 그룹 및 사용자가 동일한 이름을 공유할 수 없습니다.

참고: CA Access Control 이 Active Directory 와 함께 Windows 2000 에 설치되면 CA Access Control 은 사용자 로그온 이름(이전 Windows 2000 사용자 이름)을 변경합니다. 그러나 CA Access Control 은 Windows 처럼 전체 이름을 변경하지 않습니다.

참고: 개체 이름의 최대 길이는 255 자입니다. CA Access Control 과 Windows 는 이름이 255 자가 넘는 리소스를 관리하지 않습니다.

restrictions([days] [time])|restrictions-([days] [time])

사용자가 파일에 액세스할 수 있는 요일 및 시간을 지정합니다.

days 인수를 생략하고 time 인수를 지정할 경우, 시간 제한은 레코드에 이미 표시된 요일 제한에 적용됩니다. time 을 생략하고 days 를 지정할 경우 요일 제한은 레코드에 이미 표시된 시간 제한에 적용됩니다. days 와 time 을 모두 지정할 경우, 사용자는 지정된 요일에 지정된 시간 동안에만 시스템에 액세스할 수 있습니다.

- [Days]는 사용자가 파일에 액세스할 수 있는 요일을 지정합니다. days 인수는 다음 하위 인수를 가집니다.
  - **anyday**-사용자는 모든 요일에 파일에 액세스할 수 있습니다.
- **weekdays**-사용자는 주중(월요일부터 금요일까지)에만 리소스에 액세스할 수 있습니다.
  - **Mon**, **Tue**, **Wed**, **Thu**, **Fri**, **Sat**, **Sun**-사용자는 지정된 요일에만 리소스에 액세스할 수 있습니다. 순서에 상관 없이 요일을 지정할 수 있습니다. 둘 이상의 요일을 지정하려면 공백 또는 쉼표로 날짜를 구분합니다.

- [Time]은 사용자가 리소스에 액세스할 수 있는 기간을 지정합니다. time 인수는 다음 하위 인수를 가집니다.
  - **anytime**-사용자는 하루 중 언제든지 리소스에 액세스할 수 있습니다.
  - **startTime:endTime**-사용자는 지정된 기간 동안에만 리소스에 액세스할 수 있습니다. startTime 및 endTime 형식은 모두 hhmm 이며, 여기서 hh 는 24 시간 표기(00 - 23) 시간이고 mm 은 분(00 - 59)입니다. 2400 은 올바른 시간 값이 아닙니다. startTime 은 endTime 보다 작아야 하며, 두 시간 모두 같은 날에 속해야 합니다. 터미널의 시간대가 프로세서와 다른 경우, 터미널의 시작 시간과 종료 시간을 프로세서에 대해 동등한 로컬 시간으로 변환하여 시간 값을 조정합니다. 예를 들어, 프로세서가 뉴욕에 있고 터미널이 LA 에 있는 경우 LA 에서 오전 8 시에서 오후 5 시까지 터미널에 대한 액세스를 허용하려면 시간(1100:2000)을 지정하십시오.

resume(date)|resume-

Windows 가 사용자 계정을 복원할 날짜 및 시간(선택 사항). suspend 매개 변수와 resume 매개 변수를 모두 지정할 경우 다시 시작 날짜가 일시 중단 날짜보다 뒤에 와야 합니다. 그렇지 않으면 사용자의 일시 중단 상태가 무한정 지속됩니다.

날짜와 시간(선택사항)을

mm/dd/yy[@HH:MM] 형식으로 입력합니다.

활성(활성화됨)에서 일시 중지됨으로 사용자 계정의 상태를 변경하려면 resume- 매개 변수를 사용하십시오. 이 매개 변수는 chusr 또는 editusr 명령과 함께만 사용합니다.

script(loginScriptPath)

사용자가 로그인할 때 자동으로 실행되는 파일의 위치를 지정합니다. 이 로그인 스크립트는 작업 환경을 구성합니다. profile 매개 변수 역시 사용자의 작업 환경을 설정하기 때문에 이 매개 변수는 선택 사항입니다.

suspend(date)|suspend-

사용자 계정을 비활성화합니다. 사용자는 일시 중지된 사용자 계정을 사용하여 시스템에 로그인할 수 없습니다. 날짜를 지정할 경우 Windows 는 지정된 날짜에 사용자 계정을 일시 중지합니다. 날짜를 생략할 경우 Windows 는 chusr 명령을 실행하는 즉시 사용자 계정을 일시 중단합니다.

날짜와 시간(선택 사항)을 mm/dd/yy[@HH:MM] 형식으로 입력합니다.

비활성화됨에서 활성(활성화됨)으로 사용자의 계정 상태를 변경하려면 suspend- 매개 변수를 사용하십시오. 이 매개 변수는 chusr 또는 editusr 명령과 함께만 사용합니다.

terminals(terminalList)|terminals-(terminalList)

사용자가 로그인할 수 있는 터미널을 최대 8 개까지 지정합니다. 목록은 큰따옴표로 묶고 이름은 쉼표로 구분하십시오. 예:

```
"terminal1,terminal2"
```

workstations(workstationList)|workstations-(workstationList)|workstations-

사용자가 로그인할 수 있는 워크스테이션을 최대 8 개까지 지정합니다. 목록은 큰따옴표로 묶고 이름은 쉼표로 구분하십시오. 예:

```
"workstation1,workstation2"
```

editfile 명령-Windows 파일 설정 수정

네이티브 Windows 환경에 해당

이 명령은 chfile 명령과 함께 설명되어 있습니다.

추가 정보:

chfile 명령-Windows 파일 설정 수정(페이지 184)

editgrp 명령-Windows 그룹 작성 및 수정

네이티브 Windows 환경에 해당

이 명령은 chgrp 명령과 함께 설명되어 있습니다.

추가 정보:

chgrp 명령-Windows 그룹 수정(페이지 185)

editusr 명령-Windows 사용자 작성 및 수정

네이티브 Windows 환경에 해당

이 명령은 chusr 명령과 함께 설명되어 있습니다.

추가 정보:

chusr 명령-Windows 사용자 수정(페이지 190)

editres 명령-Windows 리소스 작성 및 수정

네이티브 Windows 환경에 해당

이 명령은 chres 명령과 함께 설명되어 있습니다.

추가 정보:

chres 명령-Windows 리소스 수정(페이지 187)

find file 명령-네이티브 파일 나열

네이티브 환경에 해당

문자열인 마스크와 일치하는 모든 시스템 파일을 나열하려면 find file 명령을 사용하십시오. 파일들이 한 열에 시간순으로 정렬됩니다.

이 명령의 형식은 다음과 같습니다.

```
find file [directory][/mask]
```

directory

directory 디렉터리의 모든 파일을 나열합니다.

mask

mask 변수와 일치하는 directory 디렉터리의 모든 파일을 나열합니다. mask 에는 와일드카드 문자가 포함될 수 있습니다.

예: Windows 의 특정 경로에서 실행 프로그램 파일 찾기

다음 명령은 CA Access Control bin 디렉터리에 있는 모든 실행 파일을 나열합니다.

```
find file C:\Program\Files\CA\AccessControl\bin\*.exe
```

예: UNIX 에서 패턴 일치 파일 찾기

다음 명령은 CA Access Control bin 디렉터리에 있는 파일 중 se 문자로 시작되는 모든 파일을 나열합니다.

```
find file /opt/CA/AccessControl/bin/se*
```

find {xuser|xgroup} 명령-엔터프라이즈 사용자 또는 그룹 나열

네이티브 Windows 환경에 해당

find {xuser|xgroup} 명령은 현재 도메인 또는 트러스트된 도메인의 엔터프라이즈 사용자 또는 그룹을 나열합니다.

참고: 이 명령은 Directory Services 가 있는 지원되는 Windows 2000 운영 체제에서만 지원됩니다.

이 명령의 형식은 다음과 같습니다.

```
find {xuser|xgroup} mask [domain(domainName)] [next]
```

xgroup

엔터프라이즈 그룹을 반환하도록 지정합니다.

xuser

엔터프라이즈 사용자를 반환하도록 지정합니다.

domain(domainName)

검색을 제한하기 위한 트러스트된 도메인을 정의합니다.

이 옵션을 지정하지 않으면 현재 도메인의 사용자가 반환됩니다.

mask

엔터프라이즈 사용자에 대한 마스크를 정의합니다.

next

이전의 find xuser 또는 find xgroup 명령으로 시작된 엔터프라이즈 사용자 또는 그룹의 나열을 selang 출력이 계속 진행하도록 지정합니다.

목록에 항목이 10 개보다 많은 경우 이 옵션을 사용하십시오.

예: 엔터프라이즈 사용자 표시

다음 명령은 현재 도메인의 엔터프라이즈 사용자 중 abc 로 시작되는 처음 100 개의 사용자를 나열합니다.

```
find xuser abc*
```

join 명령-사용자를 네이티브 그룹에 추가

네이티브 환경에 해당

join 명령은 사용자를 그룹에 추가합니다. 지정된 사용자 및 그룹은 이미 네이티브 OS 에 정의되어 있어야 합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

join 명령을 사용하려면 적어도 다음 중 하나가 참이어야 합니다.

- CA Access Control 사용자 레코드에 ADMIN 특성을 가지고 있습니다.
- 그룹 레코드는 GROUP-ADMIN 특성을 가진 그룹 범위 내에 있습니다.
- 데이터베이스에 있는 그룹 레코드의 소유자입니다.
- ADMIN 클래스의 GROUP 레코드의 액세스 제어 목록에 JOIN 또는 MODIFY 액세스 권한을 갖습니다.

참고: ADMIN 이 CA Access Control GROUP 레코드 및 네이티브 그룹을 수정할 권한을 가지려면 MODIFY 및 JOIN 속성이 모두 필요합니다.

이 명령의 형식은 다음과 같습니다.

```
{join|j} userName group(groupName)
```

group(groupName)

사용자가 추가되는 네이티브 그룹을 지정합니다.

userName

group 매개 변수로 지정한 그룹에 연결되어 있는 네이티브 사용자의 사용자 이름을 지정합니다. 둘 이상의 사용자를 지정하려면 괄호 안에 사용자 이름을 넣고 사용자 이름을 공백 또는 쉼표로 분리합니다.

예제

사용자 Eli 는 사용자 Bob 을 "staff" 그룹에 조인하려고 합니다.

- Eli 는 ADMIN 특성을 갖고 있으며 현재 환경은 native 입니다.

```
join Bob group(staff)
```

추가 정보:

join- 명령-네이티브 그룹에서 사용자 제거(페이지 174)
showgrp 명령-네이티브 그룹 속성 표시(페이지 178)
showusr 명령-네이티브 사용자 속성 표시(페이지 179)
join[x] 명령-내부 그룹에 사용자 추가(페이지 125)

join- 명령-네이티브 그룹에서 사용자 제거

네이티브 환경에 해당

join- 명령은 그룹에서 사용자를 제거합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

join- 명령을 사용하려면 다음 조건 중 하나가 참이어야 합니다.

- 사용자가 ADMIN 특성을 가집니다.
- 그룹 레코드는 GROUP-ADMIN 특성을 가진 그룹 범위 내에 있습니다.
- 데이터베이스에 있는 그룹 레코드의 소유자입니다.
- ADMIN 클래스의 GROUP 레코드의 액세스 제어 목록에 JOIN 또는 MODIFY 액세스 권한을 갖습니다.

사용자 프로필의 소유권만 가지고 있는 경우 사용자를 그룹에서 삭제할 수 있는 권한이 충분하지 않습니다. ADMIN 이 CA Access Control 레코드와 네이티브 그룹을 수정할 권한을 가지려면 MODIFY 및 JOIN 속성이 모두 필요합니다.

이 명령의 형식은 다음과 같습니다.

```
{join-|j-} userName group(groupName)
```

group(groupName)

사용자를 제거할 네이티브 그룹을 지정합니다.

userName

그룹에서 제거하려는 사용자의 사용자 이름을 지정합니다. 그룹에서 둘 이상의 사용자를 제거하려면 괄호 안에 사용자 이름 목록을 넣고 사용자 이름을 공백 또는 쉼표로 분리합니다.

예제

사용자 Bill 은 사용자 sales25 와 sales43 을 PAYROLL 그룹에서 제거하려고 합니다.

- 사용자 Bill 은 ADMIN 특성을 갖고 있으며 현재 환경은 native 입니다.

```
join- (sales25 sales43) group(PAYROLL)
```

추가 정보:

join 명령-사용자를 네이티브 그룹에 추가(페이지 173)
showgrp 명령-네이티브 그룹 속성 표시(페이지 178)

showusr 명령-네이티브 사용자 속성 표시(페이지 179)
join[x]- 명령-그룹에서 사용자 제거(페이지 128)

newgrp 명령-Windows 그룹 작성

네이티브 Windows 환경에 해당

이 명령은 chgrp 명령과 함께 설명되어 있습니다.

추가 정보:

chgrp 명령-Windows 그룹 수정(페이지 185)

newres 명령-Windows 리소스 작성

네이티브 Windows 환경에 해당

이 명령은 chres 명령과 함께 설명되어 있습니다.

추가 정보:

chres 명령-Windows 리소스 수정(페이지 187)

newusr 명령-Windows 사용자 작성

네이티브 Windows 환경에 해당

이 명령은 chusr 명령과 함께 설명되어 있습니다.

추가 정보:

chusr 명령-Windows 사용자 수정(페이지 190)

rmgrp 명령-Windows 그룹 삭제

네이티브 Windows 환경에 해당

rmgrp 명령은 Windows 데이터베이스에서 하나 이상의 그룹을 삭제합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{rmgrp|rg} groupName
```

groupName

삭제할 그룹의 이름을 지정합니다. 그룹 이름은 기존 Windows 그룹 이름이어야 합니다. 하나 이상의 그룹 이름을 지정합니다. 둘 이상의 그룹을 제거하려면 괄호 안에 그룹 이름 목록을 넣고 그룹 이름을 공백 또는 쉼표로 분리합니다.

rmres 명령-Windows 리소스 삭제

rmres 명령은 Windows 시스템 데이터베이스에서 하나 이상의 리소스를 제거합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{rmres|rr} className resourceName
```

className

리소스가 속한 클래스의 이름을 지정합니다.

resourceName

className 클래스의 기존 Windows 리소스 이름을 지정합니다. 둘 이상의 리소스를 제거하려면 괄호 안에 사용자 이름 목록을 넣고 공백 또는 쉼표로 사용자 이름을 구분하십시오.

추가 정보:

chres 명령-Windows 리소스 수정(페이지 187)
showres 명령-네이티브 리소스 속성 표시(페이지 207)
rm[x]usr 명령-사용자 레코드 삭제(페이지 135)

rmusr 명령-Windows 사용자 삭제

네이티브 Windows 환경에 해당

rmusr 명령은 Windows 시스템 데이터베이스에서 하나 이상의 사용자를 삭제합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{rmusr|ru} userName
```

userName

기존 Windows 사용자의 이름을 지정합니다. 둘 이상의 사용자를 제거하려면 괄호 안에 사용자 이름 목록을 넣고 사용자 이름을 공백 또는 쉼표로 분리합니다.

추가 정보:

chusr 명령-Windows 사용자 수정(페이지 190)
showusr 명령-네이티브 사용자 속성 표시(페이지 179)
rm[x]usr 명령-사용자 레코드 삭제(페이지 135)

setoptions 명령-CA Access Control Windows 옵션 설정

setoptions 명령은 Windows 운영 체제와 관련된 시스템 전반의 CA Access Control 옵션을 동적으로 설정합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

setoptions 명령을 사용하려면 ADMIN 특성이 필요합니다. 단, setoptions list 명령을 사용하려면 AUDITOR 또는 OPERATOR 특성만 있어도 됩니다.

이 명령의 형식은 다음과 같습니다.

```
setoptions|so \
    [audit_policy( \
        [success(system|logon|access|rights \
                |process|security|manage)] \
        [failure(system|logon|access|rights \
                |process|security|manage)] \
    )]
    [password(
        [history(number-stored-passwords)]
        [interval(nDays)]
        [min_life(NDays)]
        )]
```

audit_policy{+|-}

감사의 활성화(+) 또는 비활성화(-) 여부를 지정합니다.

audit_policy(success(system|logon|access|rights|process|security|manage))

권한이 부여된 검색된 액세스 이벤트 중 어떤 것을 기록할지를 지정합니다. 액세스 유형은 다음과 같습니다.

- **system**-컴퓨터를 종료 또는 다시 시작하려고 시도합니다.
- **logon**-시스템에 로그온 또는 로그오프하려고 시도합니다.
- **access**-파일과 같은 확실한 개체에 액세스하려고 시도합니다.
- **rights**-Windows Server 권한을 사용하려고 시도합니다.
- **process**-프로그램 활성화, 일정한 형태의 핸들 중복, 개체에 대한 간접 액세스, 프로세스 종료 등의 이벤트.
- **security**-정책 개체 규칙을 변경하려고 시도합니다.
- **manage**-사용자 또는 그룹 계정을 만들거나 삭제하거나 변경하려고 시도합니다. 또한 암호 변경을 시도합니다.

audit_policy(failure(system|logon|access|rights|process|security|manage))

권한이 부여되지 않은 검색된 액세스 이벤트 중 어떤 것을 기록할지를 지정합니다. 액세스 유형은 다음과 같습니다.

- **system**-컴퓨터를 종료 또는 다시 시작하려고 시도합니다.
- **logon**-시스템에 로그온 또는 로그오프하려고 시도합니다.
- **access**-파일과 같은 확실한 개체에 액세스하려고 시도합니다.
- **rights**-Windows Server 권한을 사용하려고 시도합니다.
- **process**-프로그램 활성화, 일정한 형태의 핸들 중복, 개체에 대한 간접 액세스, 프로세스 종료 등의 이벤트.
- **security**-정책 개체 규칙을 변경하려고 시도합니다.
- **manage**-사용자 또는 그룹 계정을 만들거나 삭제하거나 변경하려고 시도합니다. 또한 암호 변경을 시도합니다.

history(number-stored-passwords)

데이터베이스에 저장되어 있는 이전 암호 수를 지정합니다. 새 암호를 제공할 때 사용자는 기록 목록에 저장된 암호를 지정할 수 없습니다. NStoredPasswords 는 1 과 24 사이의 정수입니다. 0 을 지정하면 암호가 하나도 저장되지 않습니다.

interval(nDays)

사용자에게 새 암호를 묻기 전에 암호가 설정되거나 변경된 후 경과해야 하는 일 수를 설정합니다.

nDays 값은 양수 또는 0 이어야 합니다. 간격이 0 이면 사용자에 대한 암호 간격 검사가 비활성화됩니다. 암호가 만료되지 않도록 하려면 간격을 0 으로 설정합니다.

min_life(NDays)

암호를 변경하는 최소 일 수를 설정합니다. NDays 는 양의 정수여야 합니다.

추가 정보:

showfile 명령-파일 속성 표시(페이지 146)

showfile 명령-네이티브 파일 속성 표시

네이티브 환경에 해당

showfile 명령은 하나 이상의 시스템 파일에 대한 네이티브 세부 정보를 나열합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

이 명령의 형식은 다음과 같습니다.

```
{showfile|sf} fileName [next] \
    [{props|addprops}(propNames)]
```

addprops(propName)

표시할 속성(ruler)을 설정합니다. 속성 목록이 현재 ruler 에 추가됩니다. ruler 는 현재 쿼리용으로만 설정되고 이전에 설정된 ruler 로 되돌아갑니다.

fileName

세부 정보를 나열할 파일의 이름을 지정합니다. 하나 이상의 UNIX 파일 이름을 입력합니다. 둘 이상의 파일을 지정하려면 괄호 안에 파일 이름 목록을 넣고 각 이름을 공백 또는 쉼표로 분리합니다.

next

요청된 데이터의 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. 쿼리 크기의 기본값은 100 으로 설정됩니다.

props(all|propName)

표시할 속성(ruler)을 설정합니다.

ruler 는 이후 쿼리에 대해서 설정된 상태를 유지합니다.

예: UNIX 파일의 세부 정보 표시

UNIX 파일 /tmp/foo 의 세부 정보를 나열하려고 합니다.

```
showfile /tmp/foo
```

예: Windows 파일의 소유자 표시

Windows 파일 C:\tmp\foo.exe 의 소유자가 누구인지 알고자 합니다.

```
showfile C:\tmp\foo.exe props(Owner)
```

추가 정보:

chfile 명령-UNIX 파일 설정 수정(페이지 167)
chfile 명령-Windows 파일 설정 수정(페이지 184)
showfile 명령-파일 속성 표시(페이지 146)

showgrp 명령-네이티브 그룹 속성 표시

네이티브 환경에 해당

showgrp 명령은 네이티브 운영 체제에 있는 하나 이상의 그룹에 대한 세부 정보를 표시합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

참고: UNIX 에서는 구성 설정(seos.ini)에 지정된 파일로부터 그룹을 읽고 추가하고 업데이트하고 삭제하게 됩니다. 기본적으로 이 파일은 /etc/group 입니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

이 명령의 형식은 다음과 같습니다.

```
{showgrp|sg} groupName [next] \
    [{props|addprops}(propNames)]
```

addprops(propName)

표시할 속성(ruler)을 설정합니다. 속성 목록이 현재 ruler 에 추가됩니다. ruler 는 현재 쿼리용으로만 설정되고 이전에 설정된 ruler 로 되돌아갑니다.

groupName

세부 사항이 표시되는 그룹의 이름을 지정합니다. 그룹 이름은 기존의 네이티브 그룹 이름이어야 합니다. 하나 이상의 그룹 이름을 지정합니다. 둘 이상의 그룹을 나열하려면 괄호 안에 그룹 이름 목록을 넣고 그룹 이름을 공백 또는 쉼표로 분리합니다.

next

요청된 데이터의 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. 쿼리 크기의 기본값은 100 으로 설정됩니다.

props(all|propName)

표시할 속성(ruler)을 설정합니다.

ruler 는 이후 쿼리에 대해서 설정된 상태를 유지합니다.

예제

unix 환경에 있을 때 UNIX 그룹 security 의 세부 정보를 나열하려면 다음 명령을 입력합니다.

```
showgrp security
```

추가 정보:

chgrp 명령-UNIX 그룹 수정(페이지 168)
chgrp 명령-Windows 그룹 수정(페이지 185)
show[x]grp 명령-그룹 속성 표시(페이지 148)

showres 명령-네이티브 리소스 속성 표시

Windows 리소스의 속성을 표시합니다.

이 명령의 형식은 다음과 같습니다.

```
showres|sr className resourceName [next] \
    [{props|addprops}(propNames)]
```

addprops(propName)

표시할 속성(ruler)을 설정합니다. 속성 목록이 현재 ruler 에 추가됩니다. ruler 는 현재 쿼리용으로만 설정되고 이전에 설정된 ruler 로 되돌아갑니다.

className

리소스가 속한 클래스의 이름을 지정합니다.

next

요청된 데이터의 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. 쿼리 크기의 기본값은 100 으로 설정됩니다.

props(all|propName)

표시할 속성(ruler)을 설정합니다.

ruler 는 이후 쿼리에 대해서 설정된 상태를 유지합니다.

resourceName

className 클래스의 기존 Windows 리소스 이름을 지정합니다.

showusr 명령-네이티브 사용자 속성 표시

기본 UNIX 환경에 해당

showusr 명령은 기본 운영 체제에서 정의된 하나 이상의 사용자 속성을 표시합니다.

참고: 이 명령은 AC 환경에도 있지만 작동 방식이 다릅니다.

참고: UNIX 에서는 구성 설정(seos.ini)에 지정된 파일로부터 사용자를 읽고 추가하고 업데이트하고 삭제하게 됩니다. 기본적으로 이 파일은 /etc/passwd 입니다. 자세한 내용은 UNIX 용 끝점 관리 안내서를 참조하십시오.

이 명령의 형식은 다음과 같습니다.

```
{showusr|su} userName [next] \
    [{props|addprops}(propNames)]
```

addprops(propName)

표시할 속성(ruler)을 설정합니다. 속성 목록이 현재 ruler 에 추가됩니다. ruler 는 현재 쿼리용으로만 설정되고 이전에 설정된 ruler 로 되돌아갑니다.

userName

기본 속성이 표시되는 사용자의 이름을 지정합니다. 기존의 네이티브 사용자 이름을 지정합니다. 둘 이상의 사용자의 속성을 나열하려면 괄호 안에 사용자 이름 목록을 넣고 이름을 공백 또는 쉼표로 분리합니다.

next

요청된 데이터의 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. 쿼리 크기의 기본값은 100 으로 설정됩니다.

props(all|propName)

표시할 속성(ruler)을 설정합니다.

ruler 는 이후 쿼리에 대해서 설정된 상태를 유지합니다.

예제

unix 환경에 있을 때 UNIX 사용자 leslie 의 세부 정보를 나열하려면 다음 명령을 입력합니다.

```
showusr leslie
```

추가 정보:

chusr 명령-UNIX 사용자 수정(페이지 170)
chusr 명령-Windows 사용자 수정(페이지 190)
show[x]usr 명령-사용자 속성 표시(페이지 152)

xaudit 명령-시스템 Access 제어 목록 수정

xaudit 명령은 SACL(시스템 액세스 제어 목록)에 항목을 추가합니다. 목록의 각 항목으로 인해 지정된 사용자 또는 그룹이 리소스에 액세스하려고 시도할 때 감사 메시지가 로그 파일에 기록됩니다. xaudit- 명령은 SACL 에서 항목을 제거하고, FILE, PRINTER, REGKEY, DISK, COM 또는 SHARE 리소스 유형에 대해 유효합니다.

이 명령의 형식은 다음과 같습니다.

```
xaudit className resourceName \
    [failure(auditMode)]\
    [gid(groupName)] \
    [success(auditMode)] \
    [uid(userName)]
```

className

리소스가 속한 리소스 유형의 이름을 지정합니다.

failure(auditMode)

리소스에 대한 권한 없는 액세스 시도를 기록합니다.

값이 속하는 리소스 유형에 따라 auditmode 에 대한 올바른 값은 다음과 같습니다.

참고: NTFS 파일만 감사 모드를 사용할 수 있습니다.

- **DISK** 및 **COM**: changePermissions, delete, modify, query, read, synchronize, takeOwnership.
- **FILE**: changePermissions, delete, execute, read, takeOwnership, write.
- **PRINTER**: changePermissions, delete, print, takeOwnership.
- **REGKEY**: delete, enumerate, link, notify, queryValue, readControl, setValue, subkey, write.

모든 리소스 유형에 대해: none 또는 all.

gid(groupName)

리소스에 대한 액세스를 감사할 그룹을 지정합니다. 둘 이상의 그룹을 지정하는 경우 사용자 이름을 공백 또는 쉼표로 구분하십시오.

resourceName

SACL(시스템 액세스 제어 목록)을 수정할 리소스 레코드의 이름을 지정합니다.

success(auditMode)

리소스에 대한 권한이 부여된 액세스를 기록합니다.

값이 속하는 리소스 유형에 따라 auditmode 에 대한 올바른 값은 다음과 같습니다.

참고: NTFS 파일만 감사 모드를 사용할 수 있습니다.

- **DISK** 및 **COM**: changepermissions, delete, modify, query, read, synchronize, takeownership.
- **FILE**: changePermissions, delete, execute, read, takeOwnership, write.
- **PRINTER**: changePermissions, delete, print, takeOwnership.
- **REGKEY**: delete, enumerate, link, notify, queryValue, readControl, setValue, subkey, write.

모든 리소스 유형에 대해: none 또는 all.

uid(userName)

감사 중인 리소스에 대한 액세스를 가지는 사용자를 지정합니다. 둘 이상의 사용자를 지정하는 경우, 사용자 이름을 공백 또는 쉼표로 구분합니다. Windows NT 데이터베이스에 정의된 모든 사용자를 지정하려면 userName 에 대해 별표(*)를 지정하십시오.

추가 정보:

xaudit- 명령-시스템 액세스 제어 목록 제거(페이지 211)

xaudit- 명령-시스템 Access 제어 목록 제거

xaudit- 명령은 SACL 에서 항목을 제거하고, FILE, PRINTER, REGKEY, DISK, COM 또는 SHARE 리소스 유형에 대해 유효합니다.

이 명령의 형식은 다음과 같습니다.

```
xaudit- className, resourceName \
   [gid(groupName)] \
   [uid(userName)]
```

className

리소스가 속한 리소스 유형의 이름을 지정합니다.

gid(groupName)

감사 중인 리소스에 대한 액세스를 가지는 그룹을 지정합니다. 둘 이상의 그룹을 지정하는 경우 사용자 이름을 공백 또는 쉼표로 구분하십시오.

resourceName

SACL(시스템 액세스 제어 목록)을 제거할 리소스 레코드의 이름을 지정합니다.

uid(userName)

감사 중인 리소스에 대한 액세스를 가지는 사용자를 지정합니다. 둘 이상의 사용자를 지정하는 경우, 사용자 이름을 공백 또는 쉼표로 구분합니다. Windows NT 데이터베이스에 정의된 모든 사용자를 지정하려면 userName 에 대해 별표(*)를 지정하십시오.

추가 정보:

xaudit 명령-시스템 액세스 제어 목록 수정(페이지 209)

정책 모델 환경의 selang 명령

이 단원에는 정책 모델 환경에서 작동하는 모든 selang 명령에 대한 완전한 참조가 알파벳순으로 나열되어 있습니다.

backuppmd 명령 - PMDB 백업

pmd 환경에 해당

backuppmd 명령은 PMDB 데이터베이스의 데이터를 지정된 디렉터리에 백업합니다. 정책, 배포 정보, 구성 파일을 포함한 PMDB 데이터베이스의 모든 데이터가 백업됩니다.

DMS 에 대해 이 명령의 형식은 다음과 같습니다.

```
backup pmdName destination(path)
```

PMDB 에 대해 이 명령의 형식은 다음과 같습니다.

```
backup pmdName [destination(path)|hir_host(name)]
```

destination(path)

백업 파일이 저장될 디렉터리를 정의합니다.

참고: 경로를 지정하지 않으면 _pmd_backup_directory_ 토큰에 지정된 기본 위치에 파일이 백업됩니다.

기본값: (UNIX) ACInstallDir/data/policies_backup/pmdName

기본값: (Windows) ACInstallDir\data\policies_backup\pmdName

pmdName

백업할 PMDB 또는 DMS 의 이름을 정의합니다.

hir_host(name)

계층에 있는 모든 PMDB 를 지정된 호스트 name 에 백업하고, 백업이 'name' 호스트로 이동되었을 때 구독이 계속 유효하도록 PMDB 구독자를 수정합니다.

참고: 이 명령은 마스터 및 하위 PMDB 가 동일한 호스트에 배포된 경우에만 지원됩니다.

createpmd 명령-호스트에 PMDB 작성

pmd 환경에 해당

createpmd 명령은 원격 호스트에서 PMDB 를 정의합니다. 한 명 이상의 사용자를 PMDB 의 관리자, 감사자 및 암호 관리자로 지정할 수 있습니다. 또한 PMDB 의 부모 PMDB 와 구독자 PMDB 도 정의할 수 있습니다. 원격 호스트에서 createpmd 명령을 실행할 수 있습니다.

이 명령의 형식은 다음과 같습니다.

```
createpmd pmdname \
    [admins(user [user ...])] \
    [auditors(user [user ...])] \
    [pwman(user [user ...])] \
    [parentpmd(pmdname@host)] \
    [desktop(host-names...)] \
    [subscriber(host-names|pmdnames...)] \
    [pwdfile(file-name))] \
    [grpfile(file-name))] \
    [nis] \
    [xadmins(user [user ...])] \
    [xauditors(user [user ...])] \
```

admins(user [user ...])

PMDB 관리자가 될 하나 이상의 내부 사용자를 지정합니다. 여러 사용자가 있을 경우 공백으로 구분하십시오.

auditors(user [user ...])

PMDB 의 감사 파일을 볼 수 있는 하나 이상의 내부 사용자를 지정합니다. 여러 사용자가 있을 경우 공백으로 구분하십시오.

pwmans(user [user ...])

PMDB 암호 관리자가 될 하나 이상의 사용자를 지정합니다. 여러 사용자가 있을 경우 공백으로 구분하십시오.

parentpmd(pmdname@host)

작성 중인 PMDB 에 대해 부모가 되는 PMDB 의 이름을 지정합니다.

참고: selang remote 명령을 사용하여 여러 부모 정책 모델을 정의하려면 큰따옴표를 사용해야 합니다. 예를 들어, 정책 모델을 만든 후 상위 정책 모델을 정의하려면 다음 명령을 사용하십시오.

```
createpmd subs2 admins(abc123 root) auditors(abc123 root) desktop(pcp36949) \
parentpmd("aa@pcp36949,bb@pcp36949")
```

desktop(host [host ...])

관리자가 PMDB 를 관리할 수 있는 하나 이상의 호스트를 지정합니다. 여러 호스트가 있을 경우 공백으로 구분하십시오. 기본값은 새 PMDB 의 호스트입니다.

subscribers(host | pmd [host | pmd ...])

새 PMDB 의 구독자가 될 호스트 또는 PMDB 를 지정합니다. 여러 호스트 또는 pmd 가 있을 경우 공백으로 구분하십시오.

pwdfile(filename)

PMDB 암호 파일을 지정합니다.

grpfile(filename)

PMDB 그룹 파일을 지정합니다.

nis

새 PMDB 의 호스트에서 NIS 설치를 수행하고 모든 UNIX 업데이트를 필터링하기 위한 필터 파일을 생성합니다.

xadmins(user [user ...])

PMDB 관리자가 될 하나 이상의 엔터프라이즈 사용자를 지정합니다. 여러 사용자가 있을 경우 공백으로 구분하십시오.

xauditors(user [user ...])

PMDB 의 감사 파일을 볼 수 있는 하나 이상의 엔터프라이즈 사용자를 지정합니다. 여러 사용자가 있을 경우 공백으로 구분하십시오.

pwmans(user [user ...])

PMDB 암호 관리자가 될 하나 이상의 엔터프라이즈 사용자를 지정합니다. 여러 사용자가 있을 경우 공백으로 구분하십시오.

deletepmd 명령-호스트에서 PMDB 제거

pmd 환경에 해당

deletepmd 명령은 호스트에서 다음 항목을 제거합니다.

- PMDB 의 selang 보호 파일:
  - 데이터베이스 파일
  - 레지스트리 항목
- PMDB 디렉터리의 내용
- PMDB 디렉터리

중요! 작업 상의 심각한 문제가 발생하지 않도록 하려면 PMDB 파일을 직접 삭제하여 PMDB 를 삭제하려고 하지 마십시오. PMDB 에 대해서는 항상 deletepmd 명령을 사용하십시오.

이 명령의 형식은 다음과 같습니다.

```
deletepmd pmdname
```

findpmd 명령-호스트의 PMDB 나열

pmd 환경에 해당

findpmd 명령은 연결되어 있는 호스트의 PMDB 및 이들의 데몬이 로드되었는지 여부를 나열합니다.

이 명령의 형식은 다음과 같습니다.

```
findpmd
```

listpmd 명령-PMDB 에 대한 정보 나열

pmd 환경에 해당

listpmd 명령은 PMDB, 구독자, 업데이트 파일 및 오류 로그에 대한 정보를 나열합니다. 옵션을 사용하지 않을 경우 정책 모델 pmdName 의 모든 구독자가 나열됩니다.

이 명령의 형식은 다음과 같습니다.

```
listpmd pmdName \
    [{info|subscriber(subNames)|cmd(offset) \
    |errors|all_errors|log}] \
    [next]
```

cmd(offset)

업데이트 파일의 모든 명령과 해당 오프세트를 표시합니다.

오프셋은 파일에 있는 업데이트 위치를 표시합니다. 오프셋을 지정할 경우, 목록은 오프셋에서 시작됩니다. 오프셋을 지정하지 않을 경우, 업데이트 파일의 처음 부분부터 표시됩니다.

참고: 업데이트 파일에는 PMDB 에서 전파해야 하거나 전파한 적이 있는 업데이트가 포함되어 있습니다. 오프셋은 구독자에게 전송해야 하는 다음 업데이트의 위치를 표시합니다. 업데이트 파일의 처음 오프셋과 마지막 오프셋이 화면에 표시됩니다.

errors|all_errors

정책 모델 오류 로그를 표시합니다. errors 매개 변수는 연결 실패 오류를 제외한 모든 유형의 오류를 표시합니다. all_errors 는 모든 오류를 표시합니다.

info

정책 모델에 부모 데이터베이스가 있는지 여부를 포함하여 정책 모델 pmdName 에 대한 일반적인 정보를 표시합니다.

next

요청된 데이터 부분을 표시합니다. 이 옵션은 쿼리 데이터가 설정 쿼리 크기보다 클 때 유용합니다.

최대 쿼리 크기는 query_size 구성 설정에 의해 결정됩니다. 쿼리 크기의 기본값은 100 으로 설정됩니다.

pmdname

정보를 나열할 PMDB 의 이름을 정의합니다.

subscriber(subNames)

오류 번호, 가용성, 오프셋 및 전파할 다음 명령을 포함하여 정책 모델의 구독자와 구독자의 상태를 나열합니다. subNames 매개 변수를 사용하여 구독자의 하위 집합을 선택할 수 있습니다.

로그

정책 모델 일반 로그 파일을 표시합니다.

예: 선택한 구독자에 대해 PMDB 구독자 정보 표시

compInt 문자로 시작되는, myPMDB 정책 모델에 대한 구독자 목록을 표시하려면 다음 명령을 입력합니다.

```
listpmd myPMDB subscriber(compInt*)
```

pmd 명령-PMDB 제어

pmd 환경에 해당

pmd 명령은 정책 모델 오류 로그를 지우고, 구독자 목록을 업데이트하며, 정책 모델 서비스를 시작 및 중지하고, 업데이트 파일을 자릅니다.

이 명령의 형식은 다음과 같습니다.

```
pmd pmdName \
    {[release(subname)|start|stop|truncate(offset)|lock|unlock \
    |reloadini|startlog|killog|clrerror|backup|operation]}
```

backup

정책 모델을 백업 상태로 바꿉니다.

clrerror|clrerr

정책 모델 오류 로그를 지웁니다.

killog

정책 모델 일반 로그 파일을 비활성화합니다. 이 옵션을 지정하면 로그에 메시지가 기록되지 않습니다.

중요! PMDB 서비스를 종료하기 위해 kill 명령을 사용하지 마십시오.

lock

정책 모델을 잠금 상태로 만들고 정책 모델이 구독자에게 더 이상 업데이트를 보내지 않도록 합니다.

operation

정책 모델을 백업에서 작동 상태로 이동합니다.

pmdname

선택한 옵션을 실행할 PMDB 의 이름을 정의합니다.

release(subName)

subName 에 의해 지정된 구독자를 사용 불가능한 구독자 목록에서 제거합니다. 이는 구독자가 즉시 업데이트를 받을 수 있음을 의미합니다. subName 은 업데이트를 사용할 수 있게 될 구독자를 지정합니다.

reloadini

(UNIX 전용) 정책 모델 pmd.ini 파일 및 seos.ini 파일을 다시 읽어서, 정책 모델 데몬을 다시 로드하지 않고도 구성 설정을 변경할 수 있게 해줍니다.

startlog

정책 모델 일반 로그 파일에 쓸 수 있게 합니다. 로그 파일이 사용되어지지 않았었다면 이 옵션을 씁니다.

start

CA Access Control 정책 모델 서비스를 시작합니다. 수행할 명령이 없을 때 이 옵션을 사용합니다.

stop

CA Access Control 정책 모델 데몬/서비스를 중지합니다.

truncate|trunc[(offset)]

업데이트 파일로부터 엔트리를 제거합니다. 오프셋을 지정하지 않을 경우, 파일이 가능한 최상위 오프셋에서 잘립니다. 가능한 최상위 오프셋은 구독자를 성공적으로 업데이트한 마지막 명령의 위치입니다. offset 을 지정할 경우, 지정된 오프셋까지의 모든 항목이 삭제됩니다.

참고: 시작 오프셋에서 빼기를 하여 얻어낸 오프셋이 아니라 listpmd 명령에 의해 제공된 실제 오프셋을 사용하여 파일을 잘라내야 합니다.

unlock

정책 모델을 잠금 상태에서 잠금 해제 상태로 전환하고 정책 모델이 구독자에게 업데이트를 보내도록 합니다.

restorepmd 명령 - PMDB 복원

pmd 환경에 해당

restorepmd 명령은 로컬 호스트에서 PMDB 를 복원합니다. PMDB 를 복원하기 위해 사용하는 백업 파일은 복원 호스트와 동일한 플랫폼, 운영 체제, CA Access Control 버전을 실행하는 호스트에서 가져와야 합니다. CA Access Control 은 복원 호스트에서 반드시 실행 중이어야 합니다.

참고: 다른 터미널에서 PMDB 를 백업 및 복원하는 경우 PMDB 는 복원된 PMDB 데이터베이스에서 터미널 리소스를 자동으로 업데이트하지 않습니다. 복원된 PMDB 에 새 터미널 리소스를 추가해야 합니다. 새 터미널 리소스를 추가하려면 복원된 PMDB 를 중단하고, selang -p pmdb 명령을 실행한 다음 복원된 PMDB 시작하십시오.

이 명령의 형식은 다음과 같습니다.

```
restorepmd pmdName [source(path)] [admin(user)] [xadmin(user)] [parentpmd(name)]
```

admin(user)

(UNIX) 내부 사용자를 복원된 PMDB 의 관리자로 정의합니다.

pmdName

복원할 PMDB 의 이름을 정의합니다.

parentpmd(name)

(선택 사항) 복원된 PMDB 의 부모의 이름을 정의합니다. 이름을 pmd@host 형식으로 지정합니다.

source(path)

(선택 사항) 백업 파일이 있는 디렉터리를 정의합니다. 원본 디렉터리를 지정하지 않으면 기본 위치에 있는 파일에서 PMDB 가 복원됩니다. 기본 위치는 _pmd_backup_directory_ 토큰에 정의됩니다.

기본값: (UNIX) ACInstallDir/data/policies_backup/pmdName

기본값: (Windows) ACInstallDir\data\policies_backup\pmdName

xadmin(user)

(UNIX) 엔터프라이즈 사용자를 복원된 PMDB 의 관리자로 정의합니다.

subs 명령-구독자 추가 또는 데이터베이스 구독

pmd 환경에 해당

subs 명령은 상위 PMDB 에 구독자를 추가하거나 상위 PMDB 에 데이터베이스를 구독합니다.

호스트를 PMDB 에 구독시킬 경우:

- 호스트가 작동 중이어야 합니다.
- CA Access Control 이 해당 호스트에서 실행 중이어야 합니다.
- PMDB 는 구독된 호스트의 상위 PMDB 이어야 합니다.

PMDB 를 다른 PMDB 에 구독시킬 경우:

- 구독된 PMDB 의 parent_pmd 구성 설정에 구독 중인 PMDB(부모 PMDB)의 이름을 포함해야 합니다.
- 구독된 PMDB 가 있는 호스트에서 CA Access Control 이 실행되고 있어야 합니다.

이 명령의 형식은 다음과 같습니다.

```
subs pmdname \
    [subs(subsname)] \
    [host_type(mfHost) sysid(sysID) mf_admin(mfAdmin) port(port)] \
    {offset(offset) }
```

또는

```
subs pmdname [newsubs(subsname)]
```

또는

```
subs pmdname [parentpmd(pmdname2@host)]
```

host_type(mfhost)

구독자의 메인프레임 호스트 유형입니다.

mf_admin(mfAdmin)

구독자의 메인프레임 관리자입니다.

newsubs(subsname)

subname 을 정책 모델 pmdname 에 구독시키고, 새 구독자에게 전체 PMDB 의 내용, 암호 및 그룹 파일을 보냅니다.

parentpmd(pmdName2@host)

PMDB pmdName2@host 를 pmdName 의 부모 정책 모델로 만듭니다.

pmdname

선택한 옵션을 실행할 PMDB 의 이름을 정의합니다.

port(port)

구독자의 포트 번호입니다.

subs(subsname)

구독자를 PMDB 에 할당합니다.

sysid(sysId)

구독자의 시스템 ID 입니다.

subspmd 명령-부모 PMDB 변경

pmd 환경에 해당

subspmd 명령은 연결된 호스트에서 CA Access Control 데이터베이스의 부모를 변경합니다.

이 명령의 형식은 다음과 같습니다.

```
subspmd parentpmd(pmdname@host)
```

parentpmd(pmdname@host)

pmdname@host 를 현재 호스트의 부모 정책 모델로 만듭니다.

unsubs 명령-구독자 제거

pmd 환경에 해당

unsubs 명령은 정책 모델의 구독자 목록에서 구독자를 제거합니다.

이 명령의 형식은 다음과 같습니다.

```
unsubs pmdName subs(subName)
```

pmdname

선택한 옵션을 실행할 PMDB 의 이름을 정의합니다.

subs(subName)

pmdname 구독자 목록에서 제거할 구독자의 이름을 정의합니다.

제 4 장: 클래스 및 속성

이 단원에는 CA Access Control 데이터베이스 및 기본 운영 체제에 정의되어 있는 모든 클래스의 각 속성에 대한 설명이 포함되어 있습니다. 각 속성이 해당 환경에서 알파벳순, 클래스별로 정리되어 있는 이 장에서는 수정할 수 있는 속성, 속성을 업데이트하기 위해 사용하는 selang 매개 변수, 매개 변수가 들어 있는 명령에 대한 정보를 제공합니다.

이 장은 아래의 주제를 포함하고 있습니다.

클래스 및 속성 정보

제공되는 클래스와 속성 정보에는 다음 규칙이 적용됩니다.

- 속성 목록 앞의 설명 자료에는 클래스 레코드의 키가 정의되어 있습니다.

  키는 새 레코드를 작성할 때 지정하는 레코드 ID 입니다. 일단 생성된 키는 수정할 수 없는 속성이 됩니다.

- 매개 변수와 함께 사용된 기호 [-]는 매개 변수와 함께 빼기 기호를 입력하면 데이터베이스에서 해당 매개 변수를 삭제할 수 있음을 나타냅니다.

  예를 들어, 텍스트에 포함되어 있는 comment 는 데이터베이스 레코드에 주석을 추가하며 comment-는 데이터베이스에서 주석을 삭제합니다. 레코드를 작성할 때 매개 변수를 빼기 부호와 함께 사용할 수 없습니다.

- 데이터베이스에는 두 가지 유형의 클래스, 즉 접근자 클래스 및 리소스 클래스가 있습니다.

  접근자 클래스(USER 및 GROUP)에서는 리소스 클래스에서 사용하는 것과 다른 selang 명령 집합을 사용하여 레코드 작업을 수행합니다.

  - chusr, editusr 및 newusr 을 사용하여 USER 클래스 레코드를 처리합니다.
  - chgrp, editgrp 및 newgrp 를 사용하여 GROUP 클래스 레코드를 처리합니다.

- chres, editres 및 newres 를 사용하여 리소스 클래스의 레코드를 처리합니다. 리소스가 파일인 경우 chfile 또는 editfile 명령을 사용할 수도 있습니다.
- showgrp, showres, showfile 또는 showusr 을 사용하여 레코드 속성을 나열합니다.
- 리소스 레코드에서 ACL 을 추가, 변경 또는 제거하려면 authorize 와 authorize-를 사용합니다.

추가 정보:

selang 명령 참조(페이지 39)

AC 환경의 클래스

이 단원에는 CA Access Control 데이터베이스에 있는 모든 클래스와 속성(AC 환경의 클래스)에 대한 완전한 참조가 알파벳순으로 포함되어 있습니다.

ACVAR 클래스

ACVAR 클래스의 각 레코드는 끝점에 배포된 변수를 정의합니다. 이 클래스는 비활성화할 수 없습니다.

ACVAR 클래스의 키는 이 변수의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

POLICIES

(정보) 이 변수를 사용하는 정책(POLICY 개체)의 목록입니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

VARIABLE_TYPE

변수 유형을 정의합니다. 유효한 값:

built-in

설치 중 CA Access Control 에서 만들어진 변수를 지정합니다. 정적 변수는 끝점의 시스템 설정에 따라 확인됩니다.

참고: 기본 제공 변수는 수정 또는 삭제할 수 없습니다.

osvar

운영 체제 값에 따라 변수가 확인되도록 지정합니다.

regval

(Windows) 레지스트리 값에 따라 변수가 확인되도록 지정합니다.

참고: REG_SZ 또는 REG_EXPAND_SZ 레지스트리 유형을 가리키는 레지스트리 값만 정의할 수 있습니다.

static

정의하는 문자열 값으로 변수가 확인되도록 지정합니다.

참고: 기존 변수의 변수 유형을 변경할 수 없습니다.

VARIABLE_VALUE

변수의 값을 정의합니다.

참고: 이 속성은 변수 값 내의 어떠한 중첩 변수도 확장하지 않습니다.

VARIABLE_EXPANDED_VALUE

(정보) 변수 값을 정의하고 이 변수 값 내의 모든 중첩 변수를 확장합니다.

ADMIN 클래스

ADMIN 클래스의 각 레코드에는 비 ADMIN 사용자에게 특정 클래스의 관리를 허용하는 정의가 포함됩니다. 위임된 사용자가 관리할 각 CA Access Control 클래스를 나타내려면 ADMIN 레코드를 작성해야 합니다. 레코드는 각각의 액세스 권한이 있는 접근자 목록을 포함하며 조건부 액세스 제어 목록(CACL)도 지원합니다.

ADMIN 클래스 레코드의 키는 보호 중인 클래스의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

AAUDIT

(정보) CA Access Control 이 감사하는 활동 유형을 표시합니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

AGENT 클래스

AGENT 클래스의 각 레코드는 CA SSO 에서 에이전트로 사용되는 개체를 정의합니다.

AGENT 클래스 레코드의 키는 에이전트의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

AGENT_TYPE

에이전트 유형입니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

AGENT_TYPE 클래스

AGENT_TYPE 클래스의 각 레코드는 CA SSO 에서 사용되는 에이전트 유형을 정의합니다.

AGENT_TYPE 클래스 레코드의 키는 에이전트의 유형입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

AGENT_FLAG

특성에 대한 정보를 포함합니다. 플래그는 다음 값을 포함할 수 있습니다.

- **aznchk**-이 특성을 권한 부여에 사용할지 여부를 나타냅니다.
- **predef**(미리 정의됨), **freetext**(자유 텍스트) 또는 **userdir**(사용자 디렉터리)**-**사용자 특성의 소스를 지정합니다.
- **user** 또는 **group**-이 값은 특성(접근자)이 사용자인지 또는 그룹인지를 나타냅니다.

AGENT_LIST

이 AGENT_TYPE 개체를 agent_type 매개 변수 값으로 지정하여 작성한 AGENT 클래스의 개체 목록입니다. 예를 들어, 이 속성은 AGENT 클래스에 개체를 작성할 때 암시적으로 업데이트됩니다.

CLASSES

이 에이전트와 관련된 클래스 또는 리소스의 다중 문자열 목록입니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

APPL 클래스

APPL 클래스의 각 레코드는 CA SSO 에서 사용되는 응용 프로그램을 정의합니다.

APPL 클래스 레코드의 키는 응용 프로그램의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

APPLTYPE

CA SSO 에서 사용됩니다.

AZNACL

권한 부여 ACL 을 정의합니다. 권한 부여 ACL 은 리소스 설명을 기초로 리소스에 대한 액세스를 허용하는 ACL 입니다. 설명은 개체가 아닌 권한 부여 엔진으로 전송됩니다. 일반적으로 AZNACL 이 사용될 경우에는 개체가 데이터베이스에 없습니다.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CAPTION

바탕 화면에서 응용 프로그램 아이콘 밑에 표시되는 텍스트. 기본값은 APPL 레코드의 이름입니다.

제한: 47 자의 영숫자.

CMDLINE

응용 프로그램 실행 파일의 파일 이름입니다. CA SSO 에서 사용됩니다.

제한: 255 자.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CONTAINED_ITEMS

레코드가 컨테이너인 경우 포함된 응용 프로그램의 레코드 이름입니다.

이 속성을 수정하려면 chres, editres 및 newres 명령과 함께 item[-](applName) 매개 변수를 사용하십시오.

CONTAINERS

컨테이너 응용 프로그램의 레코드 이름입니다(레코드가 다른 응용 프로그램에 포함된 경우).

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

DIALOG_FILE

응용 프로그램에 대한 로그인 시퀀스를 포함하는 디렉터리의 CA SSO 스크립트 이름입니다. 기본 디렉터리 위치는 /usr/sso/scripts 입니다. 기본값은 "no script"입니다.

이 속성을 수정하려면 script[-](fileName) 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

GROUPS

응용 프로그램을 사용할 수 있는 사용자 그룹의 목록입니다.

HOST

응용 프로그램이 위치하는 호스트의 이름입니다.

이 속성을 수정하려면 host[-](hostName) 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

ICONFILE

바탕 화면에 응용 프로그램을 나타내는 아이콘이 포함된 파일의 이름 또는 전체 경로입니다. CA Access Control 은 사용자의 워크스테이션에 이 아이콘이 나타날 것으로 기대합니다. 파일 이름만 입력한 경우 파일의 검색 순서는 다음과 같습니다.

1. 현재 디렉터리
2. PATH 환경 변수에 나열된 디렉터리

기본값은 기본 워크스테이션 아이콘입니다.

ICONID

아이콘 파일 내의 아이콘 숫자 ID(필요한 경우)입니다. ICONID 를 지정하지 않으면 기본 아이콘이 사용됩니다.

IS_CONTAINER

응용 프로그램이 컨테이너인지 여부입니다. 기본값은 "no"입니다.

이 속성을 수정하려면 chres, editres 및 newres 명령과 함께 container[-] 매개 변수를 사용합니다.

IS_DISABLED

응용 프로그램이 비활성화되었는지 여부입니다. 응용 프로그램이 비활성화된 경우 사용자는 응용 프로그램에 로그인할 수 없습니다. 이 기능은 응용 프로그램을 변경할 때 임의의 사용자가 응용 프로그램에 로그인하지 않도록 할 때 유용합니다. 비활성화된 응용 프로그램은 응용 프로그램 메뉴 목록에 나타나지만, 사용자가 응용 프로그램을 선택하면 메시지가 표시되고 로그인이 종료됩니다. 기본값은 "not disabled"입니다.

IS_HIDDEN

응용 프로그램을 실행할 수 있는 사용자의 바탕 화면에 응용 프로그램 아이콘이 나타나는지 여부입니다. 예를 들어, 암호를 다른 응용 프로그램에 제공할 목적으로만 사용하는 응용 프로그램인 마스터 응용 프로그램을 숨기려는 경우가 있습니다. 기본값은 "not hidden"입니다.

이 속성을 수정하려면 chres, editres 및 newres 명령과 함께 hidden[-] 매개 변수를 사용합니다.

IS_SENSITIVE

미리 설정된 시간이 경과한 후 사용자가 응용 프로그램을 열 경우 재인증을 받아야 하는지 여부. 기본값은 "not sensitive"입니다.

이 속성을 수정하려면 chres, editres 및 newres 명령에서 sensitive[-] 매개 변수를 사용하십시오.

LOGIN_TYPE

사용자 암호가 제공되는 방식입니다. pwd(일반 암호), otp(1 회 암호), appticket(메인프레임 응용프로그램 인증을 위한 독점적 티켓), none(필요한 암호 없음) 또는 passticket(IBM 에서 작성하여 메인프레임 보안 패키지에 의해 사용되는 1 회 암호 교체 형식)이 있습니다. 기본값은 pwd 입니다.

이 속성을 수정하려면 login_type(value) 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

MASTER_APPL

다른 응용 프로그램에 암호를 제공하는 응용 프로그램의 레코드 이름입니다. 기본값은 no master 입니다.

이 속성을 수정하려면 master[-](applName) 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PGMDIR

응용 프로그램의 실행 파일이 존재하는 디렉터리 또는 디렉터리 목록. CA SSO 에서 사용됩니다.

PWD_AUTOGEN

응용 프로그램 암호가 CA SSO 에 의해 자동으로 생성되는지 여부를 나타냅니다. 기본값은 no 입니다.

PWD_SYNC

응용 프로그램 암호가 다른 응용 프로그램의 암호와 자동으로 동일하게 유지되는지 여부를 나타냅니다. 기본값은 no 입니다.

PWPOLICY

응용 프로그램에 대한 암호 정책의 레코드 이름입니다. 암호 정책은 새 암호의 유효성을 검사하고 암호가 만료되는 시기를 정의하는 규칙 집합입니다. 기본값은 유효성 검사 없음입니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SCRIPT_POSTCMD

로그인 스크립트 후에 하나 이상의 명령이 실행되는지 여부를 나타냅니다.

SCRIPT_PRECMD

로그인 스크립트 전에 하나 이상의 명령이 실행되는지 여부를 나타냅니다.

SCRIPT_VARS

CA SSO 에서 사용되며, 응용 프로그램별로 저장되는 응용 프로그램 스크립트의 변수 값이 있는 변수 목록입니다.

TKTKEY

CA SSO 에서만 사용됩니다.

TKTPROFILE

CA SSO 에서만 사용됩니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

AUTHHOST 클래스

AUTHHOST 클래스의 각 레코드는 CA SSO 에서 인증 호스트를 정의합니다.

AUTHHOST 클래스 레코드의 키는 권한 부여 호스트의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

AZNACL

권한 부여 ACL 을 정의합니다. 권한 부여 ACL 은 리소스 설명을 기초로 리소스에 대한 액세스를 허용하는 ACL 입니다. 설명은 개체가 아닌 권한 부여 엔진으로 전송됩니다. 일반적으로 AZNACL 이 사용될 경우에는 개체가 데이터베이스에 없습니다.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

ETHINFO

호스트의 이더넷 정보입니다.

GROUPS

리소스 레코드가 속하는 GAUTHHOST 또는 CONTAINER 레코드 목록입니다.

AUTHHOST 클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 또는 GAUTHHOST 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

KEY

CA SSO 에서만 사용됩니다.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PATH

CA SSO 에서만 사용됩니다.

PROPERTIES

UNIX dbdump 에서만 사용됩니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

SEED

CA SSO 에서만 사용됩니다.

SERNUM

인증 호스트의 일련 번호입니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UNTRUST

리소스가 언트러스트되는지, 트러스트되는지를 정의합니다. UNTRUST 속성이 설정되면 접근자는 리소스를 사용할 수 없습니다. UNTRUST 속성이 설정되지 않으면, 리소스에 대한 데이터베이스에 나열된 다른 속성이 접근자의 액세스 권한을 결정하는 데 사용됩니다. 어떤 방식으로든 트러스트된 리소스가 변경되면 CA Access Control 은 자동으로 UNTRUST 속성을 설정합니다.

이 속성을 수정하려면 chres, editres 또는 newres 명령과 함께 trust[-] 매개 변수를 사용합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

USER_DIR_PROP

(정보) 사용자 디렉터리의 이름

USER_FORMAT

CA SSO 에서만 사용됩니다.

USERALIAS

특정한 authhost 에 정의된 모든 사용자의 별칭이 들어 있습니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

CALENDAR 클래스

CALENDAR 클래스의 각 레코드는 CA Access Control 에서 시간 제한이 적용된 사용자, 그룹 및 리소스에 대해 Unicenter TNG 달력 개체를 정의합니다. CA Access Control 은 적용을 위해 Unicenter TNG 활성 달력을 지정된 시간 간격으로 검색합니다. 리소스에 달력을 할당하려면 chgrp, chres, chusr, editgrp, editres, editusr, newgrp, newres 및 newusr 명령의 달력(calendarName) 속성을 사용하십시오.

다음 클래스는 클래스 레코드에 CALENDAR 속성을 가집니다. 이러한 리소스 클래스의 각 개체에는 CALENDAR 클래스 개체를 하나만 할당할 수 있습니다.

- ADMIN
- APPL
- AUTHHOST
- CONNECT
- CONTAINER
- DOMAIN(Windows 전용)
- FILE
- GFILE
- GHOST
- GROUP

- GSUDO
- GTERMINAL
- HOST
- HOSTNET
- HOSTNP
- LOGINAPPL(UNIX 전용)
- MFTERMINAL
- PROCESS
- PROGRAM
- REGKEY(Windows 전용)
- SUDO
- SURROGATE
- TCP
- TERMINAL
- USER

CALENDAR 클래스에 대한 키는 Unicenter TNG 달력의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

CATEGORY 클래스

CATEGORY 클래스의 각 레코드는 데이터베이스의 보안 범주를 정의합니다.

CATEGORY 클래스 레코드의 키는 보안 범주의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

CONNECT 클래스

CONNECT 클래스의 각 레코드는 로컬 호스트에서 원격 호스트에 연결하기 위해 TCP over IPv4 를 사용할 수 있는 원격 호스트를 정의합니다.

참고: IP 통신에 대한 CA Access Control 액세스 규칙은 IPv4 에만 적용됩니다. CA Access Control 은 IPv6 에 의한 액세스를 제어하지 않습니다.

참고: 액세스 조건으로 CONNECT 클래스가 사용되는 경우 TCP 클래스는 액세스를 효과적으로 제어할 수 없습니다. 연결 보호를 위해 TCP 클래스와 CONNECT 클래스 중 하나를 사용하고, 둘 다 사용하지는 마십시오.

CONNECT 클래스 레코드의 키는 원격 호스트의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

CONTAINER 클래스

CONTAINER 클래스의 각 레코드는 다른 리소스 클래스의 개체 그룹을 정의하므로, 규칙을 여러 개체 클래스에 적용할 때 액세스 규칙을 정의하는 작업을 단순화합니다. CONTAINER 클래스 레코드의 구성원은 다음과 같은 클래스의 개체가 될 수 있습니다.:

- APPL
- AUTHHOST
- CONNECT
- CONTAINER
- DICTIONARY
- DOMAIN(Windows 전용)
- FILE
- GAPPL
- GAUTHHOST
- GFILE
- GHOST
- GSUDO
- GTERMINAL
- HNODE
- HOLIDAY
- HOST
- HOSTNET
- HOSTNP
- MFTERMINAL
- PARAM_DESC
- POLICY
- PROCESS
- PROGRAM
- REGKEY(Windows 전용)
- RULESET
- SPECIALPGM

- SUDO
- SURROGATE
- TCP
- TERMINAL
- WEBSERVICE

참고: CONTAINER 레코드는 다른 CONTAINER 레코드에 중첩될 수 있습니다.

개체를 CONTAINER 레코드의 구성원으로 지정하기 전에 해당 클래스에 개체에 대한 레코드를 작성해야 합니다.

컨테이너의 개체가 해당 클래스 레코드에서 ACL 을 가지지 않을 경우 이 개체는 구성원으로 속하는 CONTAINER 레코드의 ACL 을 상속합니다.

CONTAINER 클래스의 키는 CONTAINER 레코드의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

MEMBERS

그룹의 구성원인 클래스에 있는 개체 목록입니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

DEPLOYMENT 클래스

DEPLOYMENT 클래스의 각 레코드는 끝점에 대해 배포 또는 배포 취소 작업을 정의합니다. 배포 작업에는 끝점이 필요에 따라 정책을 배포 또는 배포 취소하는 데 필요한 정보가 포함됩니다.

DEPLOYMENT 클래스의 키는 배포 작업의 이름이며, 이 이름은 대개 자동으로 생성됩니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

DMS_NAME

배포 작업이 작성된 DMS 의 이름을 정의합니다.

GPOLICY

이 배포 작업이 작성된 정책의 이름을 정의합니다.

GROUPS

이 배포 작업이 구성원으로 있는 배포 패키지(GDEPLOYMENT)를 정의합니다.

HNODE

이 배포 작업이 작성된 호스트를 정의합니다.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OPERATION

이 배포 작업의 결과로서 끝점이 수행해야 하는 작업의 유형을 지정합니다. Deploy 와 Undeploy 중 하나일 수 있습니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

POLICY_VERSION

이 배포 작업이 작성된 정책 버전을 정의합니다.

RESULT_MESSAGE

배포 또는 배포 취소 selang 스크립트의 출력을 정의합니다. 이는 정책 배포 또는 배포 취소 스크립트가 실행될 때 selang 이 출력하는 메시지입니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

STATUS

배포 작업의 상태를 정의합니다. 다음 중 하나일 수 있습니다.

- **성공** -오류 없이 정책이 배포되었습니다.
- **경고** -오류와 함께 배포 스크립트가 실행되었습니다.
- **실패**-배포 작업을 실행하는 데 오류가 있었습니다.
- **작업 없음**-배포 패키지가 효과적으로 비어 있어서 수행할 일이 없습니다.

  참고: 이것은 다른 배포 경로를 통해 정책이 이미 호스트에 할당되었기 때문일 수 있습니다.
- **실행되지 않음** - 정책 확인 중 정책에서 하나 이상의 오류를 발견했습니다.
- **동기화되지 않음** - 정책에 변수가 포함되어 있으며 이 변수 값이 끝점에서 변경되었습니다.
- **보류 중인 배포** - 정책에 정의되지 않았거나 확인되지 않은 변수가 있습니다.
- **보류 중인 필수 구성 요소**-모든 필수 정책이 배포된 후에야 배포 작업이 실행됩니다.
- **보류 중인 종속 요소**-모든 종속 정책이 배포 취소된 후에야 배포 작업이 실행됩니다(배포 취소 정책).
- **픽스**-배포 작업이 다시 배포되기를 기다리고 있습니다.

TARGETYPE

policyfetcher 가 CA Access Control 배포 패키지만 처리하도록 제한하기 위해 호스트(대상) 유형을 정의합니다. 다음 중 하나일 수 있습니다: UNAB, AC, None

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

DICTIONARY 클래스

DICTIONARY 클래스의 각 레코드는 CA Access Control 데이터베이스에 저장된 공용 사전에서 암호와 비교할 단어를 정의합니다. 사용자가 암호를 변경하면 DICTIONARY 클래스의 각 레코드에 대해 암호가 확인됩니다.

DICTIONARY 클래스에 레코드(단어)를 추가하는 것 외에, 유틸리티 또는 프로그램을 실행하여 외부 파일로부터 사전 단어를 가져올 수도 있습니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

DOMAIN 클래스

Windows 에 해당

DOMAIN 클래스의 각 레코드는 Windows 네트워크의 도메인을 정의합니다.

DOMAIN 레코드에 대한 키는 도메인 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

FILE 클래스

FILE 클래스의 각 레코드는 특정 파일이나 디렉터리 또는 파일 이름 패턴과 일치하는 파일에 대해 허용된 액세스를 정의합니다. 규칙을 정의하기 위해 파일이 이미 만들어져 있어야 할 필요는 없습니다.

다른 파일과 마찬가지로 장치 파일 및 심볼릭 링크도 보호될 수 있습니다. 그러나 링크를 보호하면 링크가 가리키는 파일은 자동으로 보호되지 않습니다.

참고: NTFS 파일 시스템에서 FILE 클래스의 레코드는 파일 스트림에 대한 액세스도 정의합니다. 파일 스트림 보호에 대한 자세한 내용은 Windows 용 끝점 관리 안내서를 참조하십시오.

스크립트를 파일로 정의할 때 파일에 대한 읽기 및 실행 액세스를 모두 허용하십시오. 바이너리를 정의할 때 실행 액세스만으로 충분합니다.

특정 _restricted 그룹 외부의 사용자인 경우 FILE 클래스의 _default 레코드(또는 _default 레코드가 없으면 UACC 클래스에 있는 FILE 에 대한 레코드)는 seos.ini, seosd.trace, seos.audit 및 seos.error 파일과 같은 CA Access Control 의 부분인 파일만 보호합니다. 이런 파일은 CA Access Control 에 명시적으로 정의되어 있지 않지만 CA Access Control 에 의해 자동으로 보호됩니다.

FILE 클래스 레코드의 키는 레코드로 보호된 파일 또는 디렉터리의 이름입니다. 전체 경로를 지정해야 합니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 속하는 GFILE 또는 CONTAINER 레코드 목록입니다.

DB 속성: GROUPS

FILE 클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 또는 GFILE 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UNTRUST

리소스가 언트러스트되는지, 트러스트되는지를 정의합니다. UNTRUST 속성이 설정되면 접근자는 리소스를 사용할 수 없습니다. UNTRUST 속성이 설정되지 않으면, 리소스에 대한 데이터베이스에 나열된 다른 속성이 접근자의 액세스 권한을 결정하는 데 사용됩니다. 어떤 방식으로든 트러스트된 리소스가 변경되면 CA Access Control 은 자동으로 UNTRUST 속성을 설정합니다.

이 속성을 수정하려면 chres, editres 또는 newres 명령과 함께 trust[-] 매개 변수를 사용합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

GAPPL 클래스

GAPPL 클래스의 각 레코드는 CA SSO 에서 사용되는 응용 프로그램 그룹을 정의합니다. GAPPL 레코드에 각 응용 프로그램을 추가하기 전에 각 응용 프로그램에 대한 APPL 클래스 레코드를 작성해야 합니다. 그런 다음 레코드를 그룹화하기 위해 APPL 클래스의 레코드를 GAPPL 레코드에 명시적으로 연결해야 합니다.

GAPPL 클래스 레코드의 키는 GAPPL 레코드의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

AZNACL

권한 부여 ACL 을 정의합니다. 권한 부여 ACL 은 리소스 설명을 기초로 리소스에 대한 액세스를 허용하는 ACL 입니다. 설명은 개체가 아닌 권한 부여 엔진으로 전송됩니다. 일반적으로 AZNACL 이 사용될 경우에는 개체가 데이터베이스에 없습니다.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

GROUPS

리소스 레코드가 속하는 CONTAINER 레코드 목록입니다.

GAPPL 클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

MEMBERS

그룹의 구성원인 APPL 클래스에 있는 개체 목록입니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

GAUTHHOST 클래스

GAUTHHOST 클래스의 각 레코드는 CA SSO 에서 사용되는 인증 호스트 그룹을 정의합니다. GAUTHHOST 레코드에 각 응용 프로그램을 추가하기 전에 각 응용 프로그램에 대한 AUTHHOST 클래스 레코드를 작성해야 합니다. 그런 다음 레코드를 그룹화하기 위해 AUTHHOST 클래스의 레코드를 GAUTHHOST 레코드에 명시적으로 연결해야 합니다.

GAUTHHOST 클래스 레코드의 키는 GAUTHHOST 레코드의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

AZNACL

권한 부여 ACL 을 정의합니다. 권한 부여 ACL 은 리소스 설명을 기초로 리소스에 대한 액세스를 허용하는 ACL 입니다. 설명은 개체가 아닌 권한 부여 엔진으로 전송됩니다. 일반적으로 AZNACL 이 사용될 경우에는 개체가 데이터베이스에 없습니다.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

GROUPS

리소스 레코드가 속하는 CONTAINER 레코드 목록입니다.

GAUTHHOST 클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

MEMBERS

그룹의 구성원인 AUTHHOST 클래스에 있는 개체 목록입니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

GFILE 클래스

GFILE 클래스의 각 레코드는 특정 파일 그룹, 특정 디렉터리 또는 이름 패턴과 일치하는 파일에 대해 허용된 액세스를 정의합니다. GFILE 레코드에 각 응용 프로그램을 추가하기 전에 각 응용 프로그램에 대한 FILE 클래스 레코드를 작성해야 합니다. 그런 다음 레코드를 그룹화하기 위해 FILE 클래스의 레코드를 GFILE 레코드에 명시적으로 연결해야 합니다. FILE 클래스 레코드를 정의하기 위해 파일이 이미 만들어져 있어야 할 필요는 없습니다.

GFILE 클래스 레코드의 키는 GFILE 레코드의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

MEMBERS

그룹의 구성원인 FILE 클래스에 있는 개체 목록입니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

GDEPLOYMENT 클래스

GDEPLOYMENT 클래스의 각 레코드는 배포 패키지를 정의합니다. 배포 패키지는 동일한 트랜잭션(정책 할당, 업그레이드 등)의 결과로 특정 호스트에 대해 작성되는 모든 배포 작업과 함께 DMS 및 그룹에 자동으로 작성됩니다. 즉, 사용자가 만드는 각 트랜잭션은 필요한 수만큼의 배포 작업을 생성하며(DEPLOYMENT 개체) 이들을 호스트별로 그룹화합니다(GDEPLOYMENT 개체).

GDEPLOYMENT 클래스의 키는 배포 패키지의 이름이며, 이 이름은 대개 자동으로 생성됩니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GHNODE

이 배포 패키지가 작성된 호스트 그룹을 정의합니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

HNODE

이 배포 패키지가 작성된 호스트를 정의합니다.

MEMBERS

그룹의 구성원인 DEPLOYMENT 클래스에 있는 개체 목록입니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

POLICY

이 배포 패키지가 작성된 정책을 정의합니다.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

TRIGGER

이 배포 패키지가 작성된 이유를 지정합니다. 다음 중 하나일 수 있습니다.

- 할당-호스트에 정책을 또는 호스트 그룹에 호스트를 할당한 결과.
- 할당 취소-호스트에서 정책을 또는 호스트 그룹에서 호스트를 할당 취소한 결과.
- 직접 배포-직접 배포 작업의 결과.
- 직접 배포 취소-직접 배포 취소 작업의 결과.
- 업그레이드-업그레이드 작업의 결과.
- 복원-호스트(HNODE)에 대한 복원 작업의 결과.
- Hnode 삭제-호스트(HNODE)를 삭제한 결과.
- Ghnode 삭제-호스트 그룹(GHNODE)을 삭제한 결과.
- 다시 설정-호스트를 다시 설정한 결과.
- 다운그레이드-호스트에서 정책을 다운그레이드한 결과

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

GHNODE 클래스

GHNODE 클래스의 각 레코드는 호스트 그룹, 즉 여러 호스트(HNODE 개체)로 이루어진 그룹을 정의합니다. 각 호스트에 대한 HNODE 클래스 레코드를 먼저 작성한 후에 GHOST 레코드에 각 호스트를 추가해야 합니다.

이 클래스는 정책 배포 및 할당을 관리하는 데 사용됩니다.

GHNODE 클래스 레코드의 키는 호스트 그룹에 대한 논리 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

MEMBERS

그룹의 구성원인 HNODE 클래스에 있는 개체 목록입니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

POLICIES

이 개체에 대해 배포해야 할 정책 목록입니다.

POLICYASSIGN

이 개체에 할당되는 정책 목록을 정의합니다.

표시 이름: 할당된 정책

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

GHOST 클래스

GHOST 클래스의 각 레코드는 호스트 그룹을 정의합니다. 각 호스트에 대한 HOST 클래스 레코드를 먼저 작성한 후에 GHOST 레코드에 각 호스트를 추가해야 합니다. 서비스는 /etc/services 파일(UNIX), \system32\drivers\etc\services 파일(Windows) 또는 다른 서비스 이름 확인 방법을 사용하여 시스템에 정의되어야 합니다. 서비스를 인증할 때 서비스의 이름이 아닌 TCP/IP 프로토콜의 포트 번호로 서비스를 식별할 수 있습니다. 서비스를 추가할 때 서비스의 이름이 아닌 TCP/IP 프로토콜의 포트 번호로 서비스를 식별할 수 있습니다. 그런 다음 레코드를 그룹화하기 위해 HOST 클래스의 레코드를 GHOST 레코드에 명시적으로 연결해야 합니다.

GHOST 레코드는 호스트 그룹에 속한 다른 스테이션(호스트)이 인터넷 통신을 사용할 때 액세스를 제어하는 액세스 규칙을 정의합니다. 각 클라이언트 그룹(GHOST 레코드)에는 로컬 호스트가 클라이언트 그룹에 속한 호스트에 제공할 수 있는 서비스를 제어하는 서비스 규칙을 나열하는 INETACL 속성이 있습니다.

GHOST 클래스 레코드의 키는 GHOST 레코드의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

INETACL

로컬 호스트가 클라이언트 호스트 그룹에 제공할 수 있는 서비스 및 해당 액세스 유형을 정의합니다. 액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

Services reference

서비스에 대한 참조(포트 번호 또는 이름)입니다. 모든 서비스를 지정하려면 별표(*)를 서비스 참조로 입력하십시오.

CA Access Control 은 /etc/rpc 파일(UNIX) 또는 \etc\rpc 파일(Windows)에 지정된 대로 동적 포트 이름을 지원합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

INETACL 속성에서 접근자와 해당 액세스 유형을 수정하려면 authorize[-] 명령과 함께 access(type-of-access), service 및 stationName 매개 변수를 사용합니다.

INSERVRNGE

로컬 호스트가 클라이언트 호스트 그룹에 제공하는 서비스 범위를 지정합니다.

INETACL 속성과 유사한 기능을 수행합니다.

INSERVRANGE 속성에서 접근자와 해당 액세스 유형을 수정하려면 authorize[-] 명령과 함께 service(serviceRange) 매개 변수를 사용하십시오.

MEMBERS

그룹의 구성원인 HOST 클래스에 있는 개체 목록입니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

GPOLICY 클래스

GPOLICY 클래스의 각 레코드는 논리 정책을 정의합니다. 이 개체에는 이 정책에 속한 정책 버전(POLICY 개체)과 개체가 할당된 호스트 및 호스트 그룹에 대한 정보가 포함되어 있습니다.

GDEPLOYMENT 클래스의 키는 논리 정책의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GHNODEASSIGN

이 정책이 할당되는 호스트 그룹을 정의합니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

HNODEASSIGN

이 정책이 할당되는 호스트를 정의합니다.

LATEST_FINALIZED_VERSION

완료된 최신 정책 버전(POLICY 개체)의 이름을 정의합니다.

LATEST_VERSION

이 정책과 관련된 최신 정책 버전(POLICY 개체)의 이름을 정의합니다.

MEMBERS

그룹의 구성원인 POLICY 클래스(정책 버전)에 있는 개체 목록입니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다**.**

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

POLICY TYPE

그룹 정책 유형을 나타내는 값입니다. 유효한 값:

- 없음
- 로그인 - 정책을 UNAB 로그인 정책으로 지정합니다.
- 구성 - 정책을 UNAB 구성 정책으로 지정합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

GROUP 클래스

GROUP 클래스의 각 레코드는 데이터베이스에 있는 사용자 그룹을 정의합니다.

각 GROUP 클래스 레코드의 키는 그룹의 이름입니다.

참고: 프로필 그룹의 속성은 프로필 그룹과 관련된 각 사용자에게 적용됩니다. 그러나 사용자(USER 또는 XUSER) 레코드에 동일한 속성이 지정되어 있으면 사용자 레코드가 프로필 그룹 레코드의 속성보다 우선합니다.

CA Access Control 끝점 관리에서 또는 selang 명령 chgrp 를 사용하여 이러한 속성의 대부분을 변경할 수 있습니다.

참고: 대부분의 경우, 그리고 달리 지정되지 않은 경우, ch[x]grp 를 사용하여 속성을 변경하려면 속성 이름을 명령 매개 변수로 사용할 수 있습니다.

CA Access Control 끝점 관리에서 또는 selang 명령 showgrp 를 사용하여 모든 속성을 볼 수 있습니다.

APPLS

(정보) 접근자가 액세스할 수 있는 응용 프로그램 목록을 표시합니다. CA SSO 에서 사용됩니다.

AUDIT_MODE

CA Access Control 이 감사 로그에 기록하는 활동을 정의합니다. 다음과 같은 활동의 결합을 지정할 수 있습니다.

- 로깅 없음
- 추적 파일에 기록된 모든 활동
- 실패한 로그인 시도
- 성공한 로그인
- CA Access Control 에 의해 보호되는 리소스에 대해 실패한 액세스 시도
- CA Access Control 에 의해 보호되는 리소스에 대해 성공한 액세스

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 audit 매개 변수에 해당합니다. GROUP 또는 XGROUP 에 AUDIT_MODE 를 사용하여 그룹의 모든 구성원에 대한 감사 모드를 설정할 수 있습니다. 하지만 사용자의 감사 모드가 USER 레코드, XUSER 레코드 또는 프로필 그룹에 정의되어 있는 경우 AUDIT_MODE 를 사용하여 그룹 구성원에 대한 감사 모드를 설정할 수 없습니다.

AUTHNMTHD

(정보) 그룹 레코드와 함께 사용하는 인증 방법을 표시하며, method 1 에서 method 32 까지 또는 none 이 있습니다. CA SSO 에서 사용됩니다.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

EXPIRE_DATE

접근자가 무효화되는 날짜를 정의합니다. USER 레코드의 EXPIRE_DATE 속성 값이 GROUP 레코드의 값보다 우선합니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 expire[-] 매개 변수에 해당합니다.

FULLNAME

접근자와 관련된 전체 이름을 정의합니다. CA Access Control 은 감사 로그 메시지에서 접근자를 식별하기 위해 전체 이름을 사용합니다(권한 부여에는 사용하지 않음).

FULLNAME 은 영숫자 문자열입니다. 최대 길이가 255 자인 그룹용입니다. 최대 길이가 47 자인 사용자용입니다.

GAPPLS

그룹이 액세스할 수 있는 응용 프로그램 그룹 목록을 정의합니다. CA SSO 에서 사용됩니다.

GROUP_MEMBER

이 그룹의 구성원인 그룹을 정의합니다.

HOMEDIR

새 그룹 구성원에게 할당된 홈 디렉터리의 경로를 정의합니다.

이 속성을 수정하려면 homedir 매개 변수를 chgrp, editgrp 또는 newgrp 명령과 함께 사용하십시오.

제한: 255 자의 영숫자.

INACTIVE

사용자를 비활성 상태로 변경하기 전에 경과해야 하는 비활성 기간을 정의합니다. 계정 상태가 비활성이면 사용자는 로그인할 수 없습니다.

USER 레코드의 INACTIVE 속성 값은 GROUP 레코드의 값보다 우선됩니다. 두 가지 모두 SEOS 클래스 레코드의 INACT 속성보다 우선됩니다.

참고: CA Access Control 은 상태를 저장하지 않고 동적으로 상태를 계산합니다. 비활성 사용자를 식별하려면 INACTIVE 값을 사용자의 LAST_ACC_TIME 값과 비교해야 합니다.

INACTIVE 는 프로필 기능의 일부입니다.

MAXLOGINS

한 사용자에게 허용되는 동시 로그인의 최대 수를 정의합니다. 영(0) 값은 사용자가 동시 로그인 수로 어떤 값이든 가질 수 있음을 나타냅니다.

사용자가 로그인해서 데이터베이스를 관리하려면(예: selang 을 실행해서), MAXLOGINS 는 0 이거나 1 보다 커야 합니다. 그 이유는 CA Access Control 이 각 작업(로그인, selang, GUI 등)을 터미널 세션으로 간주하기 때문입니다.

USER 레코드의 MAXLOGINS 속성 값이 GROUP 레코드의 값보다 우선합니다. 두 가지 모두 SEOS 클래스 레코드의 MAXLOGINS 값보다 우선합니다.

MAXLOGINS 은 프로필 기능의 일부입니다.

MEMBER_OF

이 그룹이 구성원으로 있는 그룹을 정의합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PASSWDRULES

암호 규칙을 지정합니다. 이 속성에는 CA Access Control 에서 암호 보호를 처리하는 방법을 결정하는 여러 필드가 있습니다. 규칙의 전체 목록은 USER 클래스의 수정 가능한 PROFILE 속성을 참조하십시오.

이 속성을 수정하려면 setoptions 명령에서 password 매개 변수와 rules 또는 rules- 옵션을 사용하십시오.

PASSWDRULES 는 프로필 기능의 일부입니다.

POLICYMODEL

sepass 유틸리티로 사용자 암호를 변경할 때 새 암호를 수신하는 PMDB 를 지정합니다. 이 속성에 대한 값을 입력하는 경우 암호는 parent_pmd 또는 passwd_pmd 구성 설정에 의해 정의된 정책 모델로 전송되지 않습니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 pmdb[-] 매개 변수에 해당합니다.

POLICYMODEL 은 프로필 기능의 일부입니다.

PROFUSR

이 프로필 그룹과 관련된 사용자 목록을 표시합니다.

PWD_AUTOGEN

그룹 암호의 자동 생성 여부를 나타냅니다. 기본값은 "no"입니다. CA SSO 에서 사용됩니다.

PWD_SYNC

그룹 암호가 모든 그룹 응용 프로그램에 대해 자동으로 동일하게 유지되는지 여부를 나타냅니다. 기본값은 "no"입니다. CA SSO 에서 사용됩니다.

PWPOLICY

그룹에 대한 암호 정책의 레코드 이름을 정의합니다. 암호 정책은 새 암호의 유효성을 검사하고 암호가 만료되는 시기를 정의하는 규칙 집합입니다. 기본값은 유효성 검사 없음입니다. CA SSO 에서 사용됩니다.

RESUME_DATE

일시 중단된 USER 계정의 일시 중단이 해제되는 날짜를 정의합니다.

RESUME_DATE 와 SUSPEND_DATE 는 함께 작동합니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 resume[-] 매개 변수에 해당합니다.

RESUME_DATE 는 프로필 기능의 일부입니다.

REVACL

접근자의 액세스 제어 목록을 표시합니다.

SHELL

(UNIX 만) 사용자가 이 그룹의 구성원일 때 새 UNIX 사용자에게 할당된 셸 프로그램입니다.

이 속성을 수정하려면 chxgrp 명령과 함께 shellprog 매개 변수를 사용하십시오.

SUBGROUP

이 그룹을 부모 그룹으로 가지는 그룹 목록을 표시합니다.

SUPGROUP

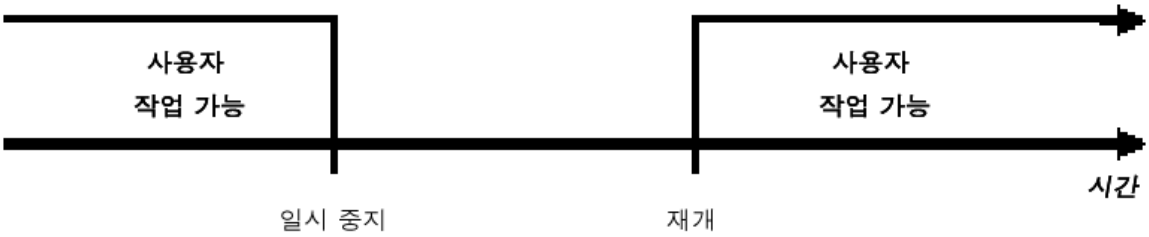
부모 그룹("superior" 그룹)의 이름을 정의합니다.

이 속성을 수정하려면 ch[x]grp 명령과 함께 parent[-] 매개 변수를 사용하십시오.

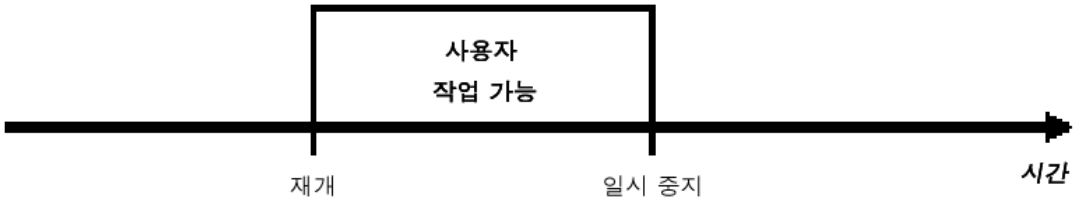
SUSPEND_DATE

사용자 계정이 일시 중단되어 무효화되는 날짜를 정의합니다.

레코드의 일시 중지 날짜가 계속 날짜보다 앞에 있는 경우 사용자는 일시 중지 날짜 전과 계속 날짜 후에 작업할 수 있습니다.



사용자가 가지고 있는 다시 시작 날짜가 일시 중단 날짜보다 빠른 경우, 다시 시작 날짜 전이라도 레코드가 무효화됩니다. 사용자는 다시 시작 날짜와 중단 날짜 사이에만 작업할 수 있습니다.



사용자 레코드의 SUSPEND_DATE 속성 값은 그룹 레코드의 값보다 우선합니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 suspend[-] 매개 변수에 해당합니다.

SUSPEND_WHO

일시 중단된 날짜를 활성화한 관리자를 표시합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

USERLIST

그룹에 소속되는 사용자를 정의합니다.

이 속성에 포함된 사용자 목록은 네이티브 환경 USERS 속성에 있는 목록과 다를 수도 있습니다.

이 속성을 수정하려면 join[x][-] 명령을 사용하십시오.

GSUDO 클래스

GSUDO 클래스에 있는 각 레코드는 작업 위임(Task Delegation), 즉 DO(sesudo)에서 사용자의 실행을 허용하거나 방지하는 작업의 그룹을 정의합니다. 각 작업에 대한 SUDO 클래스 레코드를 GSUDO 레코드에 추가하려면 SUDO 클래스 레코드를 먼저 만들어야 합니다.

GSUDO 를 사용하여 각 리소스에 대해 동일한 액세스 규칙을 지정하지 않고 SUDO 리소스 그룹의 액세스 규칙을 정의합니다. SUDO 클래스의 레코드를 GSUDO 레코드에 명시적으로 연결하여 그룹화해야 합니다.

GSUDO 클래스 레코드의 키는 그룹의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

MEMBERS

그룹의 구성원인 SUDO 클래스에 있는 개체 목록입니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

GTERMINAL 클래스

GTERMINAL 클래스의 각 레코드는 터미널 그룹을 정의합니다. 각 터미널에 대한 TERMINAL 클래스 레코드를 GTERMINAL 레코드에 추가하려면 TERMINAL 클래스 레코드를 먼저 만들어야 합니다. 그런 다음 TERMINAL 클래스의 레코드를 GTERMINAL 레코드에 명시적으로 연결하여 그룹화해야 합니다.

터미널 그룹은 액세스 규칙을 정의할 때 유용합니다. 각 터미널에 대해 동일한 액세스 규칙을 지정할 필요 없이 단일 명령을 사용하여 터미널 그룹에 대한 액세스 규칙을 지정할 수 있습니다. 마찬가지로 단일 명령을 사용하여 터미널 그룹 규칙을 사용자 그룹에도 적용할 수 있습니다.

GTERMINAL 레코드의 키는 터미널 그룹의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

MEMBERS

그룹의 구성원인 TERMINAL 클래스에 있는 개체 목록입니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

GWINSERVICE 클래스

GWINSERVICE 클래스의 각 레코드는 Windows 서비스 그룹을 정의합니다. Windows 서비스 그룹에 대해 액세스 규칙을 정의하려면 GWINSERVICE 클래스의 레코드를 사용하십시오.

GWINSERVICE 클래스 레코드의 키는 GWINSERVICE 레코드의 이름입니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

HNODE 클래스

HNODE 클래스에는 조직의 CA Access Control 호스트에 대한 정보가 포함되어 있습니다. 클래스의 각 레코드는 엔터프라이즈의 노드를 나타냅니다.

이 클래스는 여러 PMDB 및 끝점에서 업로드되고 DMS 에 저장된 정보를 관리하는 데 사용됩니다.

HNODE 클래스 레코드의 키는 끝점에 대한 실제 호스트 이름(예: myHost.ca.com) 또는 정책 모델 노드에 대한 PMDB 이름(예: myPMD@myHost.ca.com)입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

EFFECTIVE_POLICIES

이 개체에 배포해야 할 정책 버전 목록을 정의합니다.

표시 이름: 유효한 정책

GHNODES

이 개체가 구성원으로 있는 호스트 그룹의 목록을 정의합니다.

표시 이름: 노드 그룹

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

HNODE_IP

호스트의 IP 주소입니다.

표시 이름: IP

HNODE_KEEP_ALIVE

HNODE 가 배포 호스트에 하트비트를 전송한 마지막 시간을 정의합니다.

표시 이름: 마지막 하트비트

LOGIN

호스트에 대한 기본 액세스 유형을 정의합니다.

표시 이름: LOGIN

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NODE_INFO

(정보) 노드 OS 의 세부 정보를 지정합니다.

NODE_TYPE

(정보) 호스트에 설치된 CA Access Control 유형을 정의합니다. 유효한 값:

- ACU - UNIX 용 CA Access Control
- ACW - Windows 용 CA Access Control
- UNAB - UNIX 인증 브로커(UNAB)

참고: HNODE 레코드는 NODE_TYPE 속성에 대해 ACU 및 UNAB 모두의 값을 가질 수 있습니다.

NODE_VERSION

(정보) 호스트에 설치된 CA Access Control 버전을 정의합니다. 버전 번호 앞에는 NODE_TYPE 이 붙습니다.

예제: ACU:12.50-00.647

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

PARENTS

(정보) 전파 트리에 있으며 parent_pmd 구성 설정에 의해 정의된 노드의 부모인 PMDB 의 목록입니다.

POLICYASSIGN

이 개체에 할당되는 정책 목록을 정의합니다.

표시 이름: 할당된 정책

POLICY_STATUS

POLICIES 속성에 나열된 각 정책의 상태입니다. 속성의 값은 다음 필드가 있는 구조입니다.

oidPolicy

POLICY 개체의 개체 ID 이며, POLICIES 속성의 값과 동일합니다.

policy_status

다음 중 하나를 나타내는 정수입니다.

- 배포됨 - 정책이 끝점에 성공적으로 배포되었습니다.
- 배포되었지만 오류 발생-정책이 배포 스크립트에서 온 하나 이상의 규칙과 함께 배포되었지만, 끝점에서의 실행에 실패했습니다.
- 배포 취소 - 정책이 끝점에서 성공적으로 배포 취소되었습니다.

  참고: 정책이 배포 취소되면 호스트에 대한 상태가 나타나지 않습니다(즉, 상태가 비어 있게 됨).
- 배포 취소되었지만 오류 발생-정책이 배포 스크립트에서 온 하나 이상의 규칙과 함께 배포 취소되었지만, 끝점에서의 실행에 실패했습니다.
- 배포 실패 - 배포 스크립트의 오류 때문에 정책 배포에 실패했습니다.

  참고: 이 상태는 정책 확인이 활성화된 경우에만 나타날 수 있습니다. 그렇지 않으면 정책에 오류가 있더라도 policyfetcher 는 정책을 배포합니다(배포되었지만 오류 발생 상태).
- 알 수 없음-정책 상태를 알 수 없습니다.
- 배포 보류 - 필수 정책이 배포될 때까지 기다리거나 정책에 정의되지 않았거나 확인되지 않은 변수가 포함되어 있습니다.
- 배포 취소 보류-종속 정책이 배포 취소되기를 기다리고 있습니다.
- 동기화되지 않음 - 정책에 변수가 포함되어 있으며 이 변수 값이 끝점에서 변경되었습니다.
- 실행되지 않음 - 정책 확인 중 정책에서 하나 이상의 오류를 발견했습니다.
- 큐에 추가됨-사용되지 않습니다(이전 버전과의 호환성을 위해서만 사용됨).
- 전송됨-사용되지 않습니다(이전 버전과의 호환성을 위해서만 사용됨).
- 전송 실패-사용되지 않습니다(이전 버전과의 호환성을 위해서만 사용됨).

- 서명 실패-사용되지 않습니다(이전 버전과의 호환성을 위해서만 사용됨).

deviation

이 노드에 정책 위반이 있는지 여부를 나타내는 값입니다. 유효한 값:

- Yes
- No
- Unset

dev_time

마지막 위반 상태 업데이트 시간입니다.

ptime

마지막 정책 상태 업데이트 시간입니다.

updator

정책을 배포 또는 제거한 사용자의 이름입니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

SUBSCRIBER_STATUS

부모당 노드의 상태입니다. 속성의 값은 다음 필드가 있는 구조입니다.

oidSubs

HNODE 개체의 개체 ID 이며, SUBSCRIBERS 속성의 값과 동일합니다.

status

다음 상태 중 하나를 나타내는 값입니다.

- 사용 가능한 것
- 사용할 수 없음
- 동기화
- 알 수 없음

stime

마지막 상태 업데이트 시간

SUBSCRIBERS

전파 트리에 있는 노드의 구독자 목록입니다. 이 속성을 업데이트하면 HNODE 개체 이름의 값과 함께 PARENTS 속성이 암시적으로 업데이트됩니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UNAB_ID

(정보) 보고를 위한 UNAB 호스트 ID 를 표시합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

HOLIDAY 클래스

HOLIDAY 클래스의 각 레코드는 사용자가 로그인하기 위해 추가 권한이 필요할 때 하나 이상의 기간을 정의합니다.

각 사용자는 레코드의 모든 기간 동안 동일한 액세스 권한을 가집니다. 따라서 휴일 레코드에 둘 이상의 휴일 기간을 포함할 경우, 일부 기간에는 로그인을 허용할 수 없으며 다른 기간에는 로그인을 금지할 수 없습니다. 예를 들어, 특정한 사용자가 1 월 1 일에는 로그인하도록 허용하되 크리스마스에는 로그인하지 못하게 하려면 이 두 휴일을 서로 다른 레코드에 정의해야 합니다.

연도를 지정하지 않을 경우 휴일은 매년 적용되는 것으로 간주됩니다.

newusr, chusr 또는 editusr 명령에서 IGN_HOL 특성을 지정하여 개별 사용자에 대한 HOLIDAY 클래스 제한사항을 다시 정의할 수 있습니다.

HOLIDAY 클래스 레코드의 키는 HOLIDAY 레코드의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

HOL_DATE

사용자가 로그인할 수 없는 기간을 지정합니다.

다음 규칙이 HOL_DATE 속성에 적용됩니다.

- 연도를 지정하지 않으면 기간 또는 휴일이 매년 적용됨을 의미합니다. 연도는 99 또는 1999 와 같이 두 자리 또는 네 자리로 지정할 수 있습니다.
- 시작 시간을 지정하지 않은 경우, 그 날의 시작 시간(자정)이 사용됩니다. 종료 시간을 지정하지 않은 경우, 그 날의 종료 시간(자정)이 사용됩니다.
- 시간 간격을 지정하지 않고 날짜만 지정할 경우, 휴일은 지정된 하루 동안 계속됩니다.

이 속성을 수정하려면 dates 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

HOST 클래스

HOST 클래스의 각 레코드는 IPv4 로 연결되었을 때 호스트가 로컬 컴퓨터에 대해 가지는 액세스를 정의합니다.

참고: IP 통신에 대한 CA Access Control 액세스 규칙은 IPv4 에만 적용됩니다. CA Access Control 은 IPv6 에 의한 액세스를 제어하지 않습니다.

CA Access Control 은 사용자가 HOST 클래스에 추가하는 호스트 이름의 주소를 확인해야 합니다. 즉, 이름이 운영 체제 호스트 파일에 나타나거나, NIS 또는 DNS 에 정의되어 있어야 합니다.

각 HOST 레코드에 대해, INETACL 속성은 로컬 호스트가 해당 호스트에 대해 제공할 수 있는 서비스를 정의합니다.

CA Access Control 은 호스트 이름의 별칭을 허용하지만, 별칭을 나타내는 레코드는 권한 부여 검사에 사용되지 않습니다. CA Access Control 이 호스트와의 연결을 보호하도록 하려면 해당 호스트의 정규 이름을 알고 있어야 합니다.

HOST 클래스 레코드의 키는 호스트의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 속하는 GHOST 또는 CONTAINER 레코드 목록입니다.

HOST 클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 또는 GHOST 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

INETACL

로컬 호스트가 클라이언트 호스트 그룹에 제공할 수 있는 서비스 및 해당 액세스 유형을 정의합니다. 액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

Services reference

서비스에 대한 참조(포트 번호 또는 이름)입니다. 모든 서비스를 지정하려면 별표(*)를 서비스 참조로 입력하십시오.

CA Access Control 은 /etc/rpc 파일(UNIX) 또는 \etc\rpc 파일(Windows)에 지정된 대로 동적 포트 이름을 지원합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

INETACL 속성에서 접근자와 해당 액세스 유형을 수정하려면 authorize[-] 명령과 함께 access(type-of-access), service 및 stationName 매개 변수를 사용합니다.

INSERVRNGE

로컬 호스트가 클라이언트 호스트 그룹에 제공하는 서비스 범위를 지정합니다.

INETACL 속성과 유사한 기능을 수행합니다.

INSERVRANGE 속성에서 접근자와 해당 액세스 유형을 수정하려면 authorize[-] 명령과 함께 service(serviceRange) 매개 변수를 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

HOSTNET 클래스

HOSTNET 클래스의 각 레코드는 특정 네트워크의 호스트 그룹을 정의합니다. HOSTNET 레코드는, 그룹의 다른 호스트가 IPv4 통신 사용 시 로컬 호스트에 대해 갖는 액세스를 제어하는 규칙을 정의합니다.

참고: IP 통신에 대한 CA Access Control 액세스 규칙은 IPv4 에만 적용됩니다. CA Access Control 은 IPv6 에 의한 액세스를 제어하지 않습니다.

INMASKMATCH 는 HOSTNET 레코드의 영향을 받는 다른 호스트를 지정합니다. INETACL 속성은 로컬 호스트가 다른 호스트에게 제공할 수 있는 서비스를 정의합니다.

HOSTNET 클래스 레코드의 키는 HOSTNET 레코드의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

INETACL

로컬 호스트가 클라이언트 호스트 그룹에 제공할 수 있는 서비스 및 해당 액세스 유형을 정의합니다. 액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

Services reference

서비스에 대한 참조(포트 번호 또는 이름)입니다. 모든 서비스를 지정하려면 별표(*)를 서비스 참조로 입력하십시오.

CA Access Control 은 /etc/rpc 파일(UNIX) 또는 \etc\rpc 파일(Windows)에 지정된 대로 동적 포트 이름을 지원합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

INETACL 속성에서 접근자와 해당 액세스 유형을 수정하려면 authorize[-] 명령과 함께 access(type-of-access), service 및 stationName 매개 변수를 사용합니다.

INSERVRNGE

로컬 호스트가 클라이언트 호스트 그룹에 제공하는 서비스 범위를 지정합니다.

INETACL 속성과 유사한 기능을 수행합니다.

INSERVRANGE 속성에서 접근자와 해당 액세스 유형을 수정하려면 authorize[-] 명령과 함께 service(serviceRange) 매개 변수를 사용하십시오.

INMASKMATCH

이 HOSTNET 레코드가 적용되는 호스트 그룹을 정의합니다. 속성에는 마스크 값과 일치 값이 포함되며, 이러한 값은 요청 호스트가 그룹에 속하는지 여부를 결정하기 위해 요청 호스트의 IP 주소에 적용됩니다.

INMASKMATCH 속성은 IPv4 형식의 주소만 지원합니다.

참고: 이 속성은 chres 명령의 mask 및 match 매개 변수에 해당합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

HOSTNP 클래스

HOSTNP 클래스의 각 레코드는 유사한 호스트 이름을 가지는 호스트 그룹을 정의합니다. HOSTNP 레코드는, 레코드의 이름 패턴이 일치하는 다른 스테이션(호스트)이 IPv4 사용 시 로컬 호스트에 대해 갖는 액세스를 제어하는 액세스 규칙을 정의합니다. 각 마스크(HOSTNP 레코드)에는 로컬 호스트가 호스트 그룹에 제공할 수 있는 서비스를 제어하는 서비스 규칙을 나열하는 INETACL 속성이 있습니다.

HOSTNP 클래스 레코드의 키는 HOSTNP 레코드로 보호되는 호스트의 호스트 이름을 필터링하기 위해 사용된 이름 패턴입니다.

참고: IP 통신에 대한 CA Access Control 액세스 규칙은 IPv4 에만 적용됩니다. CA Access Control 은 IPv6 에 의한 액세스를 제어하지 않습니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

INETACL

로컬 호스트가 클라이언트 호스트 그룹에 제공할 수 있는 서비스 및 해당 액세스 유형을 정의합니다. 액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

Services reference

서비스에 대한 참조(포트 번호 또는 이름)입니다. 모든 서비스를 지정하려면 별표(*)를 서비스 참조로 입력하십시오.

CA Access Control 은 /etc/rpc 파일(UNIX) 또는 \etc\rpc 파일(Windows)에 지정된 대로 동적 포트 이름을 지원합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

INETACL 속성에서 접근자와 해당 액세스 유형을 수정하려면 authorize[-] 명령과 함께 access(type-of-access), service 및 stationName 매개 변수를 사용합니다.

INSERVRNGE

로컬 호스트가 클라이언트 호스트 그룹에 제공하는 서비스 범위를 지정합니다.

INETACL 속성과 유사한 기능을 수행합니다.

INSERVRANGE 속성에서 접근자와 해당 액세스 유형을 수정하려면 authorize[-] 명령과 함께 service(serviceRange) 매개 변수를 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

KMODULE 클래스

KMODULE 클래스의 각 레코드는 운영 체제의 커널 모듈을 정의합니다.

KMODULE 클래스에 모듈이 정의되어 있을 때 이 모듈의 로드 및 언로드를 위해 운영 체제에 대해 호출하면 CA Access Control 은 이 모듈에 대해 정의된 권한 부여를 확인합니다.

KMODULE 레코드의 키는 보호되고 있는 커널 모듈의 이름입니다.

각 KMODULE 레코드에는 다음 속성이 포함되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.KMODULE 레코드에 대한 올바른 액세스 권한은 로드 및 언로드입니다.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

FILEPATH

파일에 대한 절대 경로 목록을 정의합니다. 각 경로에는 커널 모듈이 포함됩니다. 각 파일 경로를 콜론(:)으로 구분하십시오.

동일한 모듈의 다른 버전을 가지고 있으면 둘 이상의 파일 경로를 사용하십시오.

파일 경로를 제공하지 않으면 CA Access Control 은 커널 모듈 로드 시 파일 경로 검사를 수행하지 않습니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다**.**

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

SIGNATURE

filepath 속성에 정의된 커널 모듈 파일에 대한 고유 값을 표시합니다.

CA Access Control 이 시작될 때와 selang 명령을 사용하여 KMODULE 레코드가 변경될 때 커널 모듈의 서명이 계산됩니다. seretrust -m 명령을 사용하여 서명을 직접 설정할 수 있습니다.

참고: CA Access Control 은 Linux 시스템에 대해서만 SIGNATURE 속성을 사용합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

LOGINAPPL 클래스

UNIX 에 해당

LOGINAPPL 클래스의 각 레코드는 로그인 응용 프로그램을 정의하고, 프로그램을 사용하여 로그인할 수 있는 사용자를 식별하며, 로그인 프로그램이 사용되는 방식을 제어합니다.

LOGINAPPL 클래스 레코드의 키는 응용 프로그램의 이름, 즉 로그인 응용 프로그램을 나타내는 논리적 이름입니다. 이 논리 이름은 LOGINPATH 속성에서 실행 파일의 전체 경로 이름과 연결되어 있습니다.

CA Access Control 은 또한 일반 로그인 응용 프로그램을 제어하고 보호할 수 있습니다. 이는 특정 규칙을 일반 패턴과 연결하는 로그인 응용 프로그램 그룹을 보호할 수 있다는 의미입니다. selang 을 사용하여 일반 응용 프로그램을 정의하려면 LOGINPATH 매개 변수의 경우를 제외하고 일반 로그인 제한을 설정할 때와 동일한 명령을 사용합니다. 이 명령에는 [,], *, ?와 같은 문자 중 하나 이상을 사용하는 정규식으로 구성된 일반 경로가 포함되어야 합니다.

CA Access Control 은 표준 로그인 프로그램의 LOGINAPPL 클래스에서 레코드의 속성 값을 미리 설정합니다. 변경하기 전에 기존 설정을 나열하고 확인해야 합니다.

중요! LOGINAPPL 은 _default 항목을 사용하지 않습니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

LOGINFLAGS

장치 번호의 변경 및 유예 로그인 횟수 감소를 포함하여 로그인 응용 프로그램의 특수 기능을 제어합니다. 유효한 값:

- **nograce**-이 응용 프로그램을 통해 사용자가 로그인할 때 유예 로그인 수가 감소하지 않음을 나타냅니다.
- **nograceroot**-이 응용 프로그램을 통해 루트가 로그인할 때 유예 로그인 수가 감소하지 않음을 나타냅니다.
- **nologin**-로그인이 사용자에게 대해서만 기록되도록 합니다. 상위 프로그램에 대한 로그인은 기록되지 않습니다.

  일부 플랫폼에서 rlogin 과 같은 프로그램으로 인해 rlogin 은 로그인을 수행하고 로그인 시퀀스 자체를 닫습니다. 결과적으로 루트에 대한 실제 로그인이 기록됩니다. 로그인 후 rlogin 은 다른 프로그램에 대한 포크를 수행하여 실제 로그인을 수행합니다.

  이 문제는 rlogin 또는 텔넷과 같은 로그인 프로그램을 사용하고 seaudit -a 를 실행할 경우 명백하게 나타납니다. uid 로서의 루트와 동일한 로그인에 대해서도 로그인 레코드가 있습니다.
- **pamlogin**-사용자가 이 응용 프로그램을 통해 로그인할 때 CA Access Control PAM 로그인 차단이 사용됨을 나타냅니다.

이 속성을 수정하려면 loginflags 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

LOGINMETHOD

로그인 응용 프로그램이 CA Access Control 보호를 목적으로 하는 유사 로그인 프로그램인지를 나타냅니다. 유효한 값:

- **normal**-이 로그인 응용 프로그램이 setuid 및 setgid 호출을 직접 실행함을 나타냅니다. seosd 는 지정된 프로그램의 규칙을 검사합니다.
- **pseudo**-이 로그인 응용 프로그램이 다른 프로그램을 호출하여 setuid 및 setgid 호출을 실행함을 나타냅니다. seosd 는 다른 프로그램에서 규칙을 검사합니다.

이 속성을 수정하려면 loginmethod 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

중요! 미리 설정된 이 속성을 수정하지 않는 것이 좋습니다.

LOGINPATH

로그인 응용 프로그램의 전체 경로(또는 일반 경로)입니다.

이 속성을 수정하려면 loginpath 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

LOGINSEQUENCE

로그인 프로세스를 시작하는 데몬의 사용자(보통 루트 아래의 inetd)를 실제 로그온한 사용자로 설정하기 위해 seosd 에서 처리하는 seteuid, setuid, setgid, setgroups 이벤트의 시퀀스를 정의합니다. 최대 8 개의 시스템 이벤트를 정의할 수 있습니다.

로그인 차단 시퀀스는 항상 트리거라는 setgid 또는 setgroups 이벤트로 시작합니다. 이 시퀀스는 사용자의 ID 를 실제 로그인한 사용자로 변경하는 setuid 이벤트로 끝납니다.

로그인을 성공적으로 수행하려면 프로그램이 setgroups 또는 setgid 에서 시작하여 setuid 또는 seteuid 로 끝나는 시퀀스로 지정된 모든 프로세스를 수행해야 합니다.

프로그램에 적합한 LoginSequence 를 설정하는 것은 어려운 작업입니다. 대부분의 로그인 프로그램은 기본 SGRP,SUID 설정에서 올바로 실행됩니다. 이 설정은 프로그램이 setgroups 시스템 호출을 실행한 다음 setuid 명령을 보내어 사용자의 ID 를 대상 사용자로 변경하는 것을 의미합니다.

그러나 SGRP, SUID 설정이 작동하지 않으면 다음 플래그를 사용하여 올바른 순서를 지정해야 합니다.

- **SEID**-첫 번째 seteuid 이벤트
- **SUID**-첫 번째 setuid 이벤트
- **SGID**-첫 번째 setgid 이벤트
- **SGRP**-첫 번째 setgroup 이벤트
- **FEID**-두 번째 seteuid 이벤트
- **FUID**-두 번째 setuid 이벤트
- **FGID**-두 번째 setgid 이벤트
- **FGRP**-두 번째 setgroup 이벤트
- **N3EID**-세 번째 seteuid 이벤트
- **N3UID**-세 번째 setuid 이벤트
- **N3GID**-세 번째 setgid 이벤트
- **N3GRP**-세 번째 setgroup 이벤트

중요! 올바른 로그인 시퀀스를 지정하려면 플래그를 사용해야 합니다. 하지만 LOGINSEQUENCE 매개 변수 내에서 임의 순서로 플래그를 지정할 수 있습니다. 예를 들어, SGRP, SEID, FEID, N3EID 는 N3EID, FEID, SGRP, SEID 와 동일합니다.

참고: 로그인 프로그램이 수행하는 시스템 호출의 시퀀스를 잘 모르는 경우에는 추적을 보고 사용자를 대상 uid 로 변경한 setuid 이벤트를 찾은 후, 첫 번째 setgid 또는 setgroup 이벤트로 시작하는 이전 추적 이벤트를 볼 수 있습니다.

예를 들어, setgroups 이벤트가 하나 있고 세번째 setuid 호출만 대상 사용자를 설정하는 경우, LOGINSEQUENCE 를 SGRP,SUID,FUID,N3UID 로 설정해야 합니다. 이러한 플래그는 임의 순서로 지정할 수 있습니다.

```
SETGRPS : P=565302 to 0,2,3,7,8,10,11,250,220,221,230

SUID > P=565302 U=0   (R=0   E=0   S=0   ) to (R=0  E=0   S=0   ) () BYPASS
SUID > P=565302 U=0   (R=0   E=0   S=0   ) to (R=0  E=0   S=-1  ) () BYPASS
LOGIN : P=565302 User=target Terminal=mercury
```

- SETGRPS 프로세스는 트리거를 나타냅니다.
- 루트가 단지 트리거 사용자가 아닌 루트로 다시 변경되었음을 알 수 있으므로 첫번째 SUID 명령은 무시됩니다. 이것은 시퀀스에서 SUID 입니다.
- 루트가 다시 루트로 변경되었음을 알 수 있으므로 두번째 SUID 명령도 무시됩니다. 이것은 시퀀스에서 FUID 입니다.
- LOGIN 이벤트는 로그인을 발생시키는 실제 SETUID 이벤트입니다. 이것은 세번째 이벤트이므로 시퀀스에서 N3UID 플래그입니다.

이 속성을 수정하려면 loginsequence 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

MFTERMINAL 클래스

MFTERMINAL 클래스의 각 레코드는 CA Access Control 관리에 사용되는 메인프레임 컴퓨터를 정의합니다. 이 클래스는 TERMINAL 클래스와 동일한 특징을 갖지만 CA Access Control 에 의해 차단되지는 않습니다.

MFTERMINAL 클래스의 키는 메인프레임 컴퓨터의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

POLICY 클래스

POLICY 클래스의 각 레코드는 정책 버전을 배포 및 배포 취소하는 데 필요한 정보를 정의합니다. 여기에는 정책을 배포 및 배포 취소하는 데 사용되는 selang 명령 목록을 포함하는 RULESET 개체에 대한 링크가 포함됩니다. 정책을 배포할 때는 deploy selang 명령이 실행됩니다. 이 명령은 정책을 정의하는 명령 및 연결된 RULESET 개체에 저장된 명령을 모두 실행합니다. 정책을 배포 취소할 때는 deploy- selang 명령이 실행됩니다. 이 명령은 정책 배포 취소를 지정하는 명령 및 연결된 RULESET 개체에 저장된 명령을 모두 실행합니다.

POLICY 클래스의 키는 정책의 이름과 그 뒤에 오는 해시 기호(#) 및 두 자리의 버전 번호입니다. 예를 들면 mypolicy#13 과 같습니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

EFFECTS_ON

이 정책이 유효한(배포되어야 하는) 호스트(HNODE 개체)의 목록을 정의합니다.

FINALIZE

이 정책 버전이 완료되었는지 여부(배포 가능한지 여부)를 지정합니다.

GROUPS

리소스 레코드가 속해 있는 CONTAINER 레코드 또는 이 정책 버전이 속해 있는 GPOLICY 개체의 목록을 정의합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

HNODES

(정보) 이 정책을 배포해야 하는 CA Access Control 노드의 목록입니다.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

POLICY_BASE_NAME

이 정책 버전이 구성원으로 있는 GPOLICY 개체의 이름을 정의합니다.

POLICY_VERSION

이 정책 버전의 버전 번호를 정의합니다.

POLICY_TYPE

정책 유형을 정의합니다. 유효한 값:

- 없음
- 로그인 - 정책을 UNAB 로그인 정책으로 지정합니다.
- 구성 - 정책을 UNAB 구성 정책으로 지정합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

RULESETS

정책을 정의하는 RULESET 개체의 목록입니다.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

SIGNATURE

이 정책과 연결된 RULESET 개체의 서명을 기반으로 하는 해시 값입니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

변수

(정보) 정책에 있는 변수의 모든 버전을 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

PROCESS 클래스

PROCESS 클래스에 있는 각 레코드는 자체 주소 공간에서 실행되며 중단되지 않도록 보호해야 하는 프로그램(실행 파일)을 정의합니다. 주요 유틸리티와 데이터베이스 서버는 서비스 거부(DoS) 공격의 주요 대상이 되기 때문에 이러한 보호가 필요한 경우가 많습니다.

참고: PROCESS 클래스에서 프로그램을 정의할 때 FILE 클래스에서도 같은 프로그램을 정의하는 것이 좋습니다. 이 작업은 누군가가 권한 없이 실행 파일을 수정(바꾸기 또는 손상시키기)하는 행위를 방지함으로써 실행 파일을 보호합니다.

CA Access Control 은 세 가지 종료(kill) 신호, 즉 일반 종료 신호(SIGTERM) 및 응용 프로그램에서 마스크할 수 없는 두 개의 신호(SIGKILL 및 SIGSTOP)로부터 보호할 수 있습니다.

환경	신호	숫자
Windows	종료	Win32 API
UNIX	프로세스 종료	9
UNIX 및 Windows	(STOP)	시스템에 따라 다름
UNIX 및 Windows	종료	15

SIGHUP 또는 SIGUSR1 과 같은 기타 신호는 목표하는 프로세스로 전달되며 해당 프로세스는 terminate 신호를 무시할지 또는 일정한 방식으로 반응할지 여부를 결정합니다.

PROCESS 클래스 레코드의 키는 레코드가 보호하는 프로그램의 이름입니다. 전체 경로를 지정합니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 정보로 표시된 속성은 수정할 수 없습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

PROGRAM 클래스

PROGRAM 클래스의 각 레코드는 TCB(Trusted Computing Base)의 일부로 간주되는 프로그램을 정의합니다. 이 클래스의 프로그램은 수정되지 않도록 감시 장치에서 모니터링하므로 보안을 위반하지 않아 트러스트할 수 있습니다. 트러스트된 프로그램이 수정되면 CA Access Control 이 자동으로 프로그램을 언트러스트된 상태로 표시하므로, 해당 프로그램은 실행되지 않습니다. 선택적으로 BLOCKRUN 속성을 사용하여, 언트러스트된 프로그램의 실행을 허용 또는 방지할 수 있습니다.

각 PROGRAM 레코드에는 트러스트된 프로그램 파일에 대한 정보를 정의하는 몇 가지 속성이 있습니다.

참고:

- UNIX 에서는 PROGRAM 클래스가 setuid 또는 setgid 로 표시되지 않은 프로그램도 포함할 수 있습니다.
- UNIX 에서 setuid 및 setgid 가 아닌 프로그램에 대해 프로그램 실행의 권한을 부여하려면 FILE 클래스의 트러스트된 실행 파일에 대해 레코드를 만들어야 합니다.

  이렇게 하면 마지막 접근자 정보(ACCSTIME 및 ACCSWHO 속성)를 추적할 수 있습니다. CA Access Control 은 권한 부여를 위해 FILE 클래스를 먼저 검사하고, 그런 다음 PROGRAM 클래스를 검사합니다.
- 어떤 프로그램이든 CA Access Control 내에서 트러스트된 프로그램으로 정의할 수 있습니다.

  프로그램이 PROGRAM 클래스에 정의되지 않은 경우 PACL(프로그램 액세스 제어 목록)에서 사용할 수 없습니다. 하지만 프로그램은 PACL 에 추가될 때 자동으로 PROGRAM 클래스에 추가됩니다.
- 디렉터리는 PROGRAM 클래스에 정의될 수 없습니다.

PROGRAM 클래스 레코드의 키는 레코드가 보호하는 프로그램의 파일 이름입니다. 파일의 전체 경로를 개체 이름으로 지정해야 합니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 정보로 표시된 속성은 수정할 수 없습니다.

ACCSTIME

(정보) 레코드에 마지막으로 액세스한 날짜 및 시간입니다.

ACCSWHO

(정보) 레코드에 마지막으로 액세스한 관리자입니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

BLOCKRUN

프로그램이 트러스트되었는지 확인하고 언트러스트된 프로그램의 실행을 차단할지 여부를 지정합니다. 실행 차단은 프로그램이 setuid 인지 일반 프로그램인지에 관계없이 수행됩니다.

리소스에 대해 이 속성을 수정하려면 chres, editres 및 newres 명령과 함께 blockrun[-] 매개 변수를 사용합니다.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

MD5

(정보) 파일의 RSA-MD5 서명입니다.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다**.**

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

참고: PROGRAM 클래스에 있는 리소스의 경우, PACL 은 setuid/setgid 프로그램(UNIX) 또는 file 리소스가 있는 프로그램(Windows)에만 적용됩니다. CA Access Control 은 먼저 파일 리소스 레코드를 검사하고, 액세스가 허용되는 경우 프로그램 리소스 레코드를 검사합니다.

PGMINFO

CA Access Control 에 의해 자동으로 생성되는 프로그램 정보를 정의합니다.

Watchdog 은 이 속성에 저장된 정보를 자동으로 확인합니다. 이 정보가 변경되면 CA Access Control 은 해당 프로그램을 언트러스트된 것으로 정의합니다.

다음 플래그 중 하나를 선택하여 확인 프로세스에서 연관된 정보를 제외할 수 있습니다.

crc

주기적 중복 검사 및 MD5 서명입니다.

ctime

(UNIX 전용) 마지막으로 파일 상태가 변경된 시간입니다.

device

UNIX 에서는 파일이 상주하는 논리 디스크입니다. Windows 에서는 파일을 포함하는 디스크의 드라이브 번호입니다.

group

프로그램 파일을 소유하고 있는 그룹입니다.

inode

UNIX 에서는 프로그램 파일의 파일 시스템 주소입니다. Windows 에서는 의미가 없습니다.

mode

프로그램 파일에 대한 관련 보안 보호 모드입니다.

mtime

프로그램 파일이 마지막으로 수정된 시간입니다.

owner

프로그램 파일을 소유하는 사용자입니다.

sha1

SHA1 서명입니다. 프로그램 또는 중요한 파일에 적용할 수 있는 보안 해시 알고리즘이라고 하는 디지털 서명 방법입니다.

size

프로그램 파일의 크기입니다.

이 속성에서 플래그를 수정하려면 chres, editres 또는 newres 명령과 함께 flags, flags+ 또는 flags- 매개 변수를 사용합니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UNTRUST

리소스가 언트러스트되는지, 트러스트되는지를 정의합니다. UNTRUST 속성이 설정되면 접근자는 리소스를 사용할 수 없습니다. UNTRUST 속성이 설정되지 않으면, 리소스에 대한 데이터베이스에 나열된 다른 속성이 접근자의 액세스 권한을 결정하는 데 사용됩니다. 어떤 방식으로든 트러스트된 리소스가 변경되면 CA Access Control 은 자동으로 UNTRUST 속성을 설정합니다.

이 속성을 수정하려면 chres, editres 또는 newres 명령과 함께 trust[-] 매개 변수를 사용합니다.

UNTRUSTREASON

(정보) 프로그램이 언트러스트된 이유입니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

PWPOLICY 클래스

PWPOLICY 클래스의 각 레코드는 암호 정책을 정의합니다. 이러한 정책은 새 암호의 유효성과 암호의 유효 기간을 모두 설정하는 일련의 규칙입니다.

PWPOLICY 클래스의 키는 암호 정책의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

APPLS

(정보) 암호 정책에 연결된 CA SSO 응용 프로그램의 목록입니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PASSWDRULES

암호 규칙을 지정합니다. 이 속성에는 CA Access Control 에서 암호 보호를 처리하는 방법을 결정하는 여러 필드가 있습니다. 규칙의 전체 목록은 USER 클래스의 수정 가능한 PROFILE 속성을 참조하십시오.

이 속성을 수정하려면 setoptions 명령에서 password 매개 변수와 rules 또는 rules- 옵션을 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

REGKEY 클래스

Windows 에 해당

REGKEY 클래스의 각 레코드는 Windows 레지스트리의 키를 정의합니다.

REGKEY 레코드에 대한 키는 키의 전체 레지스트리 경로입니다.

참고: 경로의 일부로 와일드카드 문자를 사용할 수 있습니다.

기본적으로 CA Access Control 은 CA Access Control 레지스트리 항목을 보호합니다. 이 레지스트리 항목의 루트는 다음과 같습니다.

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

CA Access Control 은 또한 다음 키를 보호합니다.

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
```

REGKEY 클래스와 REGVAL 클래스는 동일한 속성을 갖습니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

REGVAL 클래스

Windows 에 해당

REGVAL 클래스의 각 레코드는 Windows 레지스트리의 값을 정의합니다.

REGKEY 레코드에 대한 키는 값의 전체 레지스트리 경로입니다.

참고: 경로의 일부로 와일드카드 문자를 사용할 수 있습니다.

REGVAL 클래스는 NONE, READ, WRITE, DELETE 의 액세스 유형을 허용합니다.

REGVAL 클래스와 REGKEY 클래스는 동일한 속성을 갖습니다. 이러한 속성은 다음과 같습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

RESOURCE_DESC 클래스

RESOURCE_DESC 클래스의 각 레코드는 새로운 사용자 정의 클래스 개체가 CA SSO 에서 액세스할 수 있는 모든 이름을 정의합니다. RESOURCE_DESC 클래스에서 새 개체를 작성할 수 없으며 기존 개체를 수정만 할 수 있습니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

CLASS_RIGHT

선택적인 32 개의 액세스 권한 속성으로, 모두 수정 가능합니다. 처음 4 개의 권한에 대한 기본값은 다음과 같습니다.

- CLASS_RIGHT1-read
- CLASS_RIGHT2-write
- CLASS_RIGHT3-execute
- CLASS_RIGHT4-rename

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

RESPONSE_LIST

RESPONSE_TAB 클래스에서 이 객체의 이름을 포함하고 있는 객체의 이름

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

RESPONSE_TAB 클래스

RESPONSE_TAB 클래스의 각 레코드는 다른 권한 부여 결정에 대한 CA SSO 응답 테이블을 정의합니다.

응답은 권한 부여 요청이 부여되거나 거부된 후 응용 프로그램에 반환되는 개인화된 대답이며, 특정 응용 프로그램에서 인식되는 KEY=VALUE 쌍으로 구성됩니다. 응답 기능을 이용하면 사용자의 특정한 필요와 권한 부여 권한에 따라 포털 사이트를 개인화할 수 있습니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

CLASS_RIGHT

선택적인 32 개의 응답 속성은 KEY=VALUE 쌍(예: button1=yes, picture2=no 등)을 포함하는 문자열의 목록입니다. 각 액세스 값에는 하나의 속성이 있습니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

OF_RESOURCE

RESOURCE_DESC 클래스에서 사용자가 정의한 동일한 클래스를 가리키는 객체의 이름.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

RULESET 클래스

RULESET 클래스의 각 레코드는 정책을 정의하는 규칙 집합을 나타냅니다.

RULESET 클래스 레코드의 키는 레코드가 연결된 정책의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

EXPANDED COMMANDS

(정보) 배포된 정책에 있는 명령의 변수 값을 표시합니다.

EXPANDED UNDO COMMANDS

(정보) 배포된 정책에 있는 실행 취소 명령의 변수 값을 표시합니다.

FINALIZE

selang 스크립트가 완료되었는지, 그래서 정책 버전을 배포할 수 있는지 여부를 지정합니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

RULESET_DOCMD_IDX

(정보) 명령 인덱스, 즉 RULESET_DOCMDS 목록에 있는 명령 개수의 카운터입니다.

RULESET_DOCMDS

정책을 함께 정의하는 selang 명령의 목록입니다. 이러한 명령은 정책 배포를 위해 실행됩니다.

중요! 정책 배포에서는 사용자 암호를 설정하는 명령을 지원하지 않으므로 배포 스크립트 파일에 이러한 명령을 포함시키지 마십시오. UNIX(기본) selang 명령은 지원되지만 위반 보고서에는 표시되지 않습니다.

RULESET_POLICIES

(정보) 이 규칙 집합을 사용하는 정책(POLICY 개체)의 목록입니다.

RULESET_UNDOCMD_IDX

(정보) 명령 인덱스, 즉 RULESET_UNDOCMDS 목록에 있는 명령 개수의 카운터입니다.

RULESET_UNDOCMDS

정책 배포 취소 스크립트를 함께 정의하는 selang 명령의 목록입니다. 이러한 명령은 정책 배포 취소를 위해 실행됩니다.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

SIGNATURE

RULESET_DOCMDS 및 RULESET_UNDOCMDS 속성을 기반으로 하는 해시 값입니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

SECFILE 클래스

SECFILE 클래스의 각 레코드는 모니터될 파일을 정의합니다. SECFILE 클래스 레코드는 시스템에서 중요한 파일에 대한 확인 기능을 제공하지만, 하지만 조건부 액세스 제어 목록에는 나타날 수 없습니다.

자주 수정하지 않는 중요한 시스템 파일을 이 클래스에 추가하여 권한 없는 사용자가 이러한 파일을 변경하지 않도록 하십시오. SECFILE 클래스에 포함할 파일 유형의 몇 가지 예는 다음과 같습니다.

UNIX 의 경우	**Windows 의 경우**
/.rhosts	\system32\drivers\etc\hosts
/etc/services	\system32\drivers\etc\services
/etc/*	\system32\drivers\etc\protocols
/etc/hosts	
/etc/hosts.equiv	

Watchdog 은 이 파일을 검사하고 파일 정보가 변경되지 않았는지 확인합니다.

참고: SECFILE 클래스에는 디렉터리를 정의할 수 없습니다.

SECFILE 클래스 레코드의 키는 SECFILE 레코드로 보호된 파일의 이름입니다. 전체 경로를 지정합니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

AIXACL

AIX 시스템 ACL 입니다.

AICEXTI

AIX 시스템 확장 정보입니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

HPUXACL

HP-UX 시스템 ACL 입니다.

MD5

(정보) 파일의 RSA-MD5 서명입니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PGMINFO

CA Access Control 에 의해 자동으로 생성되는 프로그램 정보를 정의합니다.

Watchdog 은 이 속성에 저장된 정보를 자동으로 확인합니다. 이 정보가 변경되면 CA Access Control 은 해당 프로그램을 언트러스트된 것으로 정의합니다.

다음 플래그 중 하나를 선택하여 확인 프로세스에서 연관된 정보를 제외할 수 있습니다.

crc

주기적 중복 검사 및 MD5 서명입니다.

ctime

(UNIX 전용) 마지막으로 파일 상태가 변경된 시간입니다.

device

UNIX 에서는 파일이 상주하는 논리 디스크입니다. Windows 에서는 파일을 포함하는 디스크의 드라이브 번호입니다.

group

프로그램 파일을 소유하고 있는 그룹입니다.

inode

UNIX 에서는 프로그램 파일의 파일 시스템 주소입니다. Windows 에서는 의미가 없습니다.

mode

프로그램 파일에 대한 관련 보안 보호 모드입니다.

mtime

프로그램 파일이 마지막으로 수정된 시간입니다.

owner

프로그램 파일을 소유하는 사용자입니다.

sha1

SHA1 서명입니다. 프로그램 또는 중요한 파일에 적용할 수 있는 보안 해시 알고리즘이라고 하는 디지털 서명 방법입니다.

size

프로그램 파일의 크기입니다.

이 속성에서 플래그를 수정하려면 chres, editres 또는 newres 명령과 함께 flags, flags+ 또는 flags- 매개 변수를 사용합니다.

UNTRUST

리소스가 언트러스트되는지, 트러스트되는지를 정의합니다. UNTRUST 속성이 설정되면 접근자는 리소스를 사용할 수 없습니다. UNTRUST 속성이 설정되지 않으면, 리소스에 대한 데이터베이스에 나열된 다른 속성이 접근자의 액세스 권한을 결정하는 데 사용됩니다. 어떤 방식으로든 트러스트된 리소스가 변경되면 CA Access Control 은 자동으로 UNTRUST 속성을 설정합니다.

이 속성을 수정하려면 chres, editres 또는 newres 명령과 함께 trust[-] 매개 변수를 사용합니다.

UNTRUSTREASON

(정보) 프로그램이 언트러스트된 이유입니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

SECLABEL 클래스

SECLABEL 클래스의 각 레코드는 보안 수준을 보안 범주와 연결합니다. SECLABEL 클래스가 활성인 경우 보안 레이블은 USER 레코드의 특정 보안 수준 및 보안 범주 할당을 다시 정의합니다. 보안 레이블을 할당하는 것은 보안 레이블의 보안 수준 및 보안 범주를 사용자에게 명시적으로 할당하는 것과 같습니다.

USER 레코드가 보안 레이블을 포함할 때 다음과 같은 조건이 모두 해당될 경우에만 사용자에게 리소스에 대한 액세스가 부여됩니다.

- 보안 레이블에 지정된 사용자 보안 수준은 리소스 보안 수준보다 높거나 같습니다.
- 리소스 레코드에 지정된 모든 범주는 사용자 보안 레이블의 보안 범주 목록에 포함됩니다.

참고: Windows 에서 CA Access Control 에 정의되어 있는 각 보안 레이블은 SECLABEL 클래스에 레코드를 가지고 있어야 합니다.

SECLABEL 클래스 레코드의 키는 보안 레이블의 이름입니다. 이 이름은 사용자 또는 리소스에 보안 레이블을 할당할 때 식별하기 위해 사용됩니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

SEOS 클래스

SEOS 클래스는 CA Access Control 권한 부여 시스템의 동작을 제어합니다.

클래스에는 일반 보안 및 권한 부여 옵션을 지정하는 하나의 SEOS 레코드만 있습니다. SEOS 클래스 속성의 상태를 보거나 변경하려면 setoptions 명령을 사용하십시오.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACCPACL

권한 부여 과정 중에 UACC(defaccess) 및 PACL 목록을 검사하는 순서를 나타냅니다.

ACCPACL 이 활성화되고 ACL 을 통해 사용자에게 명시적인 액세스가 제공되면, 해당 접근자는 허용되는 액세스입니다. ACL 을 통한 명시적 액세스는 없지만 PACL 을 통해 명시적 액세스가 정의된 경우 PACL 액세스는 허용된 액세스입니다. ACL 및 PACL 이 모두 명시적 액세스를 포함하지 않으면 defaccess 가 액세스 정의에 대해 검사됩니다.

ACCPACL 이 활성화되지 않은 경우에도 ACL 이 명시적 액세스에 대해 먼저 검사됩니다. ACL 이 검사 중인 리소스에 대한 액세스 정의를 포함하지 않는 경우 다음으로 defaccess 정의가 검사됩니다. defaccess 에 명시적 액세스가 정의되지 않은 경우 PACL 액세스 정의가 검사됩니다.

CA Access Control 이 설치되어 있으면 이 속성 값이 yes 로 설정됩니다.

이 속성을 수정하려면 setoptions 명령에서 accpacl 이나 accpacl- 매개 변수를 사용하십시오.

ADMIN

ADMIN 클래스가 활성 상태인지 여부를 나타냅니다. 일반적으로 ADMIN 클래스는 활성 상태이며 보안 관리 작업을 수행하는 권한을 제어합니다. ADMIN 클래스가 비활성화된 경우에는 모든 사용자가 CA Access Control 관리자로서 작업할 수 있습니다.

APPL

APPL 클래스가 활성 상태인지 여부를 나타냅니다.

AUTHHOST

AUTHHOST 클래스가 활성 상태인지 여부를 나타냅니다.

CALENDAR

CALENDAR 클래스가 활성 상태인지 여부를 나타냅니다.

CATEGORY

CATEGORY 클래스가 활성 상태인지 여부를 나타냅니다.

CNG_ADMIN_PWD

PWMANAGER 특성을 가진 사용자가 selang 을 사용하여 ADMIN 사용자 암호를 변경할 수 있는지 여부를 나타냅니다. 기본값은 yes 입니다.

이 속성을 활성화 또는 비활성화하려면 setoptions 명령과 함께 class+ 또는 class- 매개 변수와 cng_adminpwd 옵션을 사용합니다.

CNG_OWN_PWD

사용자가 selang 을 사용하여 자신의 암호를 변경할 수 있는지 여부를 나타냅니다.

이 속성을 활성화 또는 비활성화하려면 setoptions 명령과 함께 class+ 또는 class- 매개 변수와 cng_ownpwd 옵션을 사용합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CONNECT

CONNECT 클래스가 활성 상태인지 여부를 나타냅니다. CONNECT 클래스가 활성일 때 클래스의 레코드는 발송 연결을 보호합니다.

HOST 클래스가 활성인 경우 CONNECT 클래스는 활성화된 경우라도 활성 클래스로 사용되지 않습니다.

TCP 클래스가 활성인 경우 CONNECT 클래스는 활성 클래스로 사용되지 않습니다.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIMERES

(UNIX 전용) CA Access Control 이 리소스에 대해 daytime 제한을 검사할지 여부를 나타냅니다.

DMS

이 데이터베이스가 알림을 보내야 하는 DMS 서버 목록입니다.

DOMAIN

(Windows 전용) DOMAIN 클래스가 활성 상태인지 여부를 나타냅니다.

ENDTIME

(정보) 데이터베이스 파일이 지정된 순서로 마지막에 닫힌 날짜 및 시간입니다.

FILE

FILE 클래스가 활성 상태인지 여부를 나타냅니다. FILE 클래스가 활성 상태일 때 클래스의 레코드는 파일 및 디렉터리를 보호합니다.

ACCGRR

ACCGRR(누적된 그룹 권한) 옵션은 CA Access Control 이 리소스 ACL 을 검사하는 방법에 영향을 미칩니다. ACCGRR 이 활성화되면 CA Access Control 은 사용자가 속한 모든 그룹에서 부여된 권한의 ACL 을 확인합니다. ACCGRR 이 비활성화되면 CA Access Control 은 ACL 에서 적용 가능한 항목에 값 none 이 포함되어 있는지 확인합니다. 값 none 이 포함되어 있으면 액세스가 거부됩니다. 그렇지 않으면 CA Access Control 은 액세스 제어 목록의 첫 번째 적용 가능한 항목을 제외하고 모든 그룹 항목을 무시합니다.

이 속성을 활성화 또는 비활성화하려면 setoptions ACCGRR 명령을 사용하십시오.

HOLIDAY

HOLIDAY 클래스가 활성 상태인지 여부를 나타냅니다. HOLIDAY 클래스가 활성 상태일 때 사용자는 정의된 Holiday 기간 중 로그인할 수 있는 추가 권한이 필요합니다.

HOST

HOST 클래스가 활성 상태인지 여부를 나타냅니다. HOST 클래스가 활성 상태일 때 CA Access Control 은 원격 호스트에서 들어오는 TCP/IP 서비스 요청을 보호합니다.

HOST 클래스가 활성 상태일 경우 TCP 및 CONNECT 클래스는 활성화되어도 활성 클래스로 사용되지 않습니다.

HOST 클래스의 기본값은 활성입니다.

INACT

사용자 로그인을 일시 중지한 후 비활성 기간을 나타냅니다. 비활성 일은 사용자가 로그인하지 않는 날입니다.

USER 레코드의 INACTIVE 속성 값은 GROUP 레코드의 값보다 우선됩니다. 두 가지 모두 SEOS 클래스 레코드의 INACT 속성보다 우선됩니다.

이 속성을 업데이트하려면 setoptions 명령에서 inactive 또는 inactive-매개 변수를 사용하십시오.

ISDMS

PMDB 가 DMS 역할을 할 경우 참입니다.

LOGINAPPL

(UNIX 전용) LOGINAPPL 클래스가 활성 상태인지 여부를 나타냅니다.

MAXLOGINS

사용자에게 허용된 최대 동시 로그인(터미널 세션) 수입니다. 이 값을 초과하면 사용자의 액세스가 거부됩니다. 0 값은 최대값이 없음을 나타내고 사용자는 터미널 세션 수에 상관없이 동시에 로그인할 수 있습니다. 사용자가 로그인하여 selang 을 실행하거나 데이터베이스를 관리하려는 경우 이 값은 0 이거나 1 보다 커야 합니다. 그 이유는 CA Access Control 이 각 작업(로그인, selang, GUI 등)을 터미널 세션으로 간주하기 때문입니다.

USER 레코드의 MAXLOGINS 속성 값이 GROUP 레코드의 값보다 우선합니다. 두 가지 모두 SEOS 클래스 레코드의 MAXLOGINS 속성보다 우선됩니다. SEOS 레코드의 값은 접근자 레코드에 명시적 값이 없을 때 사용되는 기본값입니다.

SEOS 클래스에 대한 이 속성을 수정하려면 maxlogins 매개 변수를 chres, editres 및 newres 명령과 함께 사용하십시오.

MFTERMINAL

MFTERMINAL 클래스가 활성 상태인지 여부를 나타냅니다.

PASSWDRULES

암호 규칙을 나타냅니다. 이 속성에는 CA Access Control 에서 암호 보호를 처리하는 방법을 결정하는 여러 필드가 있습니다. 규칙의 전체 목록은 USER 클래스의 수정 가능한 PROFILE 속성을 참조하십시오.

이 속성을 수정하려면 setoptions 명령에서 password 매개 변수와 rules 또는 rules- 옵션을 사용하십시오.

PASSWORD

암호 검사가 활성 상태인지 여부를 나타냅니다.

이 속성을 활성화 또는 비활성화하려면 setoptions 명령과 함께 class+ 또는 class- 매개 변수와 PASSWORD 옵션을 사용합니다.

PROCESS

PROCESS 클래스가 활성 상태인지 여부를 나타냅니다. PROCESS 클래스가 활성 상태일 때 클래스의 레코드는 정의된 프로세스가 종료되지 않도록 보호합니다.

또한 파일이 FILE 클래스에 정의되어야 합니다.

PROGRAM

PROGRAM 클래스가 활성 상태인지 여부를 나타냅니다. PROGRAM 클래스가 활성 상태일 때 클래스의 레코드는 트러스트됨으로 표시되어 있는 정의된 프로그램을 보호합니다.

PWPOLICY

PWPOLICY 클래스가 활성 상태인지 여부를 나타냅니다.

REGKEY

(Windows 전용) REGKEY 클래스가 활성 상태인지 여부를 나타냅니다.

REGVAL

(Windows 전용) REGVAL 클래스가 활성 상태인지 여부를 나타냅니다.

RESOURCE_DESC

RESOURCE_DESC 클래스가 활성 상태인지 여부를 나타냅니다.

RESPONSE_TAB

RESPONSE_TAB 클래스가 활성 상태인지 여부를 나타냅니다.

SECLABEL

SECLABEL 클래스가 활성 상태인지 여부를 나타냅니다.

SECLEVEL

SECLEVEL 클래스가 활성 상태인지 여부를 나타냅니다.

STARTTIME

(정보) 데이터베이스 파일이 마지막으로 열린 날짜 및 시간입니다.

SUDO

sesudo 에서 사용된 SUDO 클래스가 활성 상태인지 여부를 나타냅니다.

SYSTEM_AAUDIT_MODE

사용자 및 엔터프라이즈 사용자에 대한 기본 감사 모드(시스템 전체 감사 모드)를 지정합니다.

기본값: Failure LoginSuccess LoginFailure

SURROGATE

SURROGATE 클래스가 활성 상태인지 여부를 나타냅니다. SURROGATE 클래스가 활성 상태일 때 CA Access Control 은 surrogate 요청을 보호합니다.

TCP

TCP 클래스가 활성 상태인지 여부를 나타냅니다. TCP 클래스가 활성 상태일 때 CA Access Control 은 메일, ftp 및 http 등의 들어오는 TCP 서비스와 나가는 TCP 서비스를 보호합니다.

HOST 클래스가 활성 상태인 경우 TCP 클래스는 활성화되어도 활성 클래스로 사용되지 않습니다.

TCP 클래스가 활성인 경우 CONNECT 클래스는 활성 클래스로 사용되지 않습니다.

TERMINAL

TERMINAL 클래스가 활성 상태인지 여부를 나타냅니다. TERMINAL 클래스가 활성 상태일 때 CA Access Control 은 사인온 중에 터미널 액세스 검사를 수행하고 X-window 세션을 보호합니다.

USER_ATTR

USER_ATTR 클래스가 활성 상태인지 여부를 나타냅니다.

USER_DIR

USER_DIR 클래스가 활성 상태인지 여부를 나타냅니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

SPECIALPGM 클래스

SPECIALPGM 클래스는 지정된 프로그램에 특수 보안 권한을 부여합니다.

SPECIALPGM 클래스의 각 레코드는 다음과 같은 두 가지 기능 중 하나를 가집니다.

- Windows 에서 백업, DCM, PBF, PBN, STOP, SURROGATE, REGISTRY 및 KILL 프로그램을 등록하거나 UNIX 에서 xdm, backup, mail, DCM, PBF, PBN, stop 및 surrogate 프로그램 등록
- 특수 CA Access Control 권한 보호가 필요한 응용 프로그램을 논리적 사용자 ID 와 연결합니다. 이렇게 하면 수행하는 사람이 아니라 작업의 내용에 따라 액세스 권한을 설정할 수 있습니다.

참고: SPECIALPGM 클래스에서 프로그램을 정의할 때 FILE 클래스에서도 같은 프로그램을 정의하는 것이 좋습니다. FILE 리소스는 누군가가 권한 없이 실행 파일을 수정(바꾸기 또는 손상시키기)하는 행위를 방지함으로써 실행 파일을 보호하며, PROGRAM 리소스는 CA Access Control 이 실행되고 있지 않을 때 프로그램이 수정된 경우 이 프로그램이 실행되지 않도록 합니다.

시스템 서비스, 데몬 또는 다른 특수 프로그램을 등록하려면 PGMTYPE 속성을 사용하십시오.

논리적 사용자를 프로그램에 할당하려면 SEOSUID 및 NATIVEUID 속성을 사용하십시오.

SPECIALPGM 클래스 레코드의 키는 특수 프로그램 또는 특수 프로그램의 범위나 패턴에 대한 경로입니다.

참고: specialpgm 클래스 테이블에 넣을 수 있는 최대 규칙 수는 512 개입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

NATIVEUID

프로그램 또는 프로세스를 호출하는 사용자를 나타냅니다. 모든 CA Access Control 사용자를 지정하려면 *를 사용하십시오.

이 속성을 수정하려면 nativeuid 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

참고: CA Access Control 이전 버전과의 호환성을 유지하기 위해 NATIVUID 속성 대신 UNIXUID 속성을 사용할 수 있습니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PGMTYPE

액세스 허용 시 CA Access Control 이 바이패스하는 액세스 검사 유형을 결정합니다.

backup

READ, CHDIR 및 UTIME 액세스를 바이패스합니다.

참고: 성공적인 백업을 실행하는 방법에는 두 가지가 있습니다. 백업 프로그램이 루트가 아닌 사용자에 의해 실행된 경우 이 사용자를 OPERATOR 로 정의해야 합니다. 백업 프로그램이 루트에 의해 실행된 경우 SPECIALPGM 클래스에 백업 프로그램을 pgmtype(backup)으로 등록할 수 있습니다.

changed

(UNIX 에만 해당) PAM 활성화된 대리 ID 변경 도구(예: su)를 바이패스합니다.

예: er specialpgm /bin/su pgmtype(changeid)

dcm

STOP 이벤트를 제외한 모든 이벤트에 대한 모든 보안 검사를 바이패스합니다.

kill

(Windows 에만 해당) 프로세스에 대해 프로그램 종료를 바이패스합니다.

예를 들어 다음 규칙은 이 프로세스에서 액세스 마스크 KILL 을 사용하여 CA Access Control 서비스(프로세스)에 대한 핸들을 여는 경우 services.exe 를 바이패스합니다.

```
nr specialpgm c:\Windows\system32\services.exe pgmtype(kill)
```

Windows Server 2008 에서 서비스 중지 및 시작을 관리하는 services.exe 프로세스는 액세스 유형 KILL 을 사용하여 CA Access Control 서비스에 대한 핸들을 열고 프로세스 종료 및 시작을 관리합니다. Windows Server 2008 에 설치하는 동안 CA Access Control 은 검색 프로세스를 실행하여 services.exe 를 찾고 해당 바이패스 규칙을 만듭니다. 이 바이패스가 없으면 services.exe 에서 CA Access Control 서비스의 핸들을 열 때 "거부됨" CA Access Control 감사 이벤트가 수신됩니다.

mail

(UNIX 에만 해당) setuid 및 setgid 이벤트에 대한 데이터베이스 검사를 바이패스합니다. 메일 바이패스를 통해 메일 액세스 시도를 추적할 수 있습니다.

none

이전에 설정된 PGMTYPE 을 모두 제거합니다.

pbf

파일 처리 이벤트에 대한 데이터베이스 검사를 바이패스합니다.

pbn

네트워크 관련 이벤트에 대한 데이터베이스 검사를 바이패스합니다.

propagate

(UNIX 에만 해당) 이 PGMTYPE 을 사용하여 프로그램에서 호출된 모든 프로그램에 고유한 보안 권한을 전파합니다. 지정하지 않으면 SPECIALPGM 권한은 상위 프로그램에만 영향을 줍니다.

참고: 보안 권한은 PBF, PBN, DCM, SURROGATE 권한을 사용할 때만 전파할 수 있습니다.

registry

(Windows 에만 해당) Windows 레지스트리를 수정하는 프로그램에 대한 데이터베이스 검사를 바이패스합니다.

stop

STOP 기능에 대한 데이터베이스 검사를 바이패스합니다.

surrogate

커널에서 이벤트를 변경 중인 ID 에 대한 데이터베이스 검사를 바이패스합니다. surrogate 바이패스를 사용할 경우 추적할 수 없습니다.

xdm

(UNIX 에만 해당) 제한된 네트워크 범위(6000-6010)에 대한 네트워크 이벤트(TCP, HOST, CONNECT 클래스 등)를 바이패스합니다.

이 속성을 수정하려면 pgmtype 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

SEOSUID

이러한 특수 프로그램을 실행할 권한을 부여 받은 대리 논리적 사용자를 정의합니다. 논리적 사용자는 USER 레코드와 함께 데이터베이스에 정의되어야 합니다.

이 속성을 수정하려면 seosuid 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

예: UNIX 파일 보호

/DATABASE/data/*에 있는 파일을 보호하기 위해 데이터베이스 관리자는 firmdb_filemgr 이라는 파일 서버 데몬을 사용합니다. 이 파일 서버는 /opt/dbfirm/bin/firmdb_filemgr 에 위치합니다. 일반적으로 이 데몬은 루트 아래에서 실행되어 루트 쉘 핵에서 데이터에 액세스할 수 있도록 합니다.

다음 예제에서 논리적 사용자는 이러한 파일의 접근자로만 정의되고 다른 사용자에 의한 액세스는 제한됩니다.

1. 다음 명령을 사용하여 "중요한" 파일을 CA Access Control 에 정의합니다.

   ```
   newres file /DATABASE/data/* defaccess(NONE)owner(nobody)
   ```

2. 파일을 액세스하는 논리적 사용자를 정의합니다.

   ```
   newusr firmDB_mgr
   ```

3. 논리적 사용자 firmDB_mgr 에게만 파일 액세스를 허용합니다.

   ```
   authorize file /DATABASE/data/* uid(firmDB_mgr) access(ALL)
   ```

4. 마지막으로, firmdb_filemgr 이 논리적 사용자 firmDB_mgr 과 함께 실행되게 합니다.

   ```
   newres SPECIALPGM /opt/dbfirm/bin/firmdb_filemgr unixuid(root) \
   seosuid(firmDB_mgr)
   ```

따라서 데몬이 파일에 액세스할 때 CA Access Control 은 논리적 사용자를 루트가 아닌 파일의 접근자로 인식합니다. 루트로 파일에 액세스를 시도하는 해커는 실패합니다.

예: Windows 파일 보호

C:\DATABASE\data 에 있는 파일을 보호하기 위해 데이터베이스 관리자는 firmdb_filemgr.exe 라는 파일 서버 서비스를 사용합니다. 이 파일 서버 서비스는 C:\Program Files\dbfirm\bin\firmdb_filemgr.exe 에 있습니다. 이 서비스는 대개 시스템 계정으로 실행되므로, 시스템 해커가 데이터에 액세스할 수 있습니다.

다음 예제에서 논리적 사용자는 이러한 파일의 접근자로만 정의되고 다른 사용자에 의한 액세스는 제한됩니다.

1. 다음 명령을 사용하여 CA Access Control 에 "중요한" 파일을 정의합니다.

   ```
   newres file C:\DATABASE\data\* defaccess(NONE)owner(nobody)
   ```

2. 파일을 액세스하는 논리적 사용자를 정의합니다.

   ```
   newusr firmDB_mgr
   ```

3. 논리적 사용자 firmDB_mgr 에게만 파일 액세스를 허용합니다.

   ```
   authorize file C:\DATABASE\data\* uid(firmDB_mgr) access(ALL)
   ```

4. 마지막으로, firmdb_filemgr 이 논리적 사용자 firmDB_mgr 과 함께 실행되게 합니다.

   ```
   newres SPECIALPGM ("C:\Program Files\dbfirm\bin\firmdb_filemgr.exe") \
   nativeuid(system) seosuid(firmDB_mgr)
   ```

따라서 서비스가 파일에 액세스할 때 CA Access Control 은 논리적 사용자를 시스템 계정이 아닌 파일의 접근자로 인식합니다. 시스템 계정으로 파일에 액세스를 시도하는 해커는 실패합니다.

SUDO 클래스

SUDO 클래스의 각 레코드는 한 사용자가 sesudo 명령을 사용하여 다른 사용자로부터 사용 권한을 빌려 올 수 있는 명령을 식별합니다.

SUDO 클래스 레코드의 키는 SUDO 레코드 이름입니다. 이 이름은 사용자가 SUDO 레코드에서 명령을 실행할 때 명령 이름 대신 사용됩니다.

참고: Windows 의 경우, 터미널 서비스가 설치된 컴퓨터에서 SYSTEM 계정이 아닌 사용자 계정으로 SeOS Task Delegation 서비스가 실행되면 sesudo 명령으로 대화형 프로세스를 실행할 수 없습니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

sesudo 가 실행하는 명령입니다.

영숫자 문자열에는 명령과 허용 및 금지된 매개 변수를 포함하여 최대 255 자가 들어갈 수 있습니다.

예를 들어, 다음 프로필 정의는 COMMENT 속성을 적절하게 사용합니다.

```
newres SUDO profile_name comment('command;;NAME')
```

참고: COMMENT 속성의 이 사용법은 다른 클래스의 경우와 다릅니다. SUDO 레코드 정의에 대한 자세한 내용은 해당 OS 의 끝점 관리 안내서를 참조하십시오. 이 속성은 CA Access Control 이전 버전에서 DATA 로도 알려졌습니다.

제한: 255 자.

이 속성을 수정하려면 chres, editres 및 newres 명령과 함께 comment[-] 매개 변수를 사용합니다.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 속해 있는 GSUDO 또는 CONTAINER 레코드의 목록입니다.

SUDO 클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 또는 GSUDO 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

INTERACTIVE

(Windows 전용) sesudo 를 통해 실행하려는 응용 프로그램이 서비스 응용 프로그램이 아니라 notepad.exe 나 cmd.exe 등의 대화형 Windows 응용 프로그램인 경우 이 스위치를 표시해야 합니다. sesudo 를 사용하여, interactive 로 표시되지 않은 대화형 응용 프로그램을 실행하려고 하면 해당 응용 프로그램은 대화형 기능 없이 배경에서 실행됩니다.

참고: 일부 Windows 응용 프로그램은 Windows 제한으로 인해 전경에서 실행할 수 없습니다.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

PASSWORDREQ

(UNIX 전용) 실행하기 전에 sesudo 명령이 대상 사용자 암호를 요청하는지 여부를 나타냅니다.

이 속성을 수정하려면 password 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

POLICYMODEL

sepass 유틸리티로 사용자 암호를 변경할 때 새 암호를 수신하는 PMDB 를 지정합니다. 이 속성에 대한 값을 입력하는 경우 암호는 parent_pmd 또는 passwd_pmd 구성 설정에 의해 정의된 정책 모델로 전송되지 않습니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 pmdb[-] 매개 변수에 해당합니다.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

TARGUSR

(UNIX 전용) 명령을 실행하기 위해 권한을 빌리는 사용자를 표시하는 대상 uid 를 나타냅니다. 기본값은 root 입니다.

이 속성을 수정하려면 targuid 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

SURROGATE 클래스

SURROGATE 클래스의 각 레코드는, 다른 사용자들이 한 사용자에 대해 자신의 ID 를 변경하려고 할 때 한 사용자를 다른 사용자들로부터 보호하는 제한을 정의합니다. CA Access Control 은 ID 변경 요청을 권한이 부여된 사용자만 액세스할 수 있는 추상 개체로 처리합니다.

SURROGATE 클래스의 레코드는 surrogate 보호를 가진 각 사용자 또는 그룹을 나타냅니다. 두 개의 특수 레코드, 즉 USER._default 와 GROUP._default 는 개별 SURROGATE 레코드가 없는 사용자와 그룹을 나타냅니다. 사용자의 기본값과 그룹의 기본값을 구별할 필요가 없는 경우 SURROGATE 클래스에 대해 _default 레코드를 대신 사용할 수 있습니다.

참고: 대부분의 Windows 유틸리티와 서비스(예: 다음 계정으로 실행)는 이를 실행하는 원래 사용자가 아니라 NT AUTHORITY\SYSTEM 사용자로 식별됩니다. 이러한 유틸리티와 서비스를 사용하는 사용자가 다른 사용자를 가장(impersonate)하도록 허용하려면 CA Access Control 데이터베이스에 이 SYSTEM 사용자를 생성한 다음 대상 사용자를 가장할 수 있는 권한을 부여해야 합니다.

SURROGATE 클래스 레코드의 키는 SURROGATE 레코드의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

TCP 클래스

TCP 클래스의 각 레코드는 메일, ftp 및 http 와 같은 TCP/IP 서비스를 정의합니다. TCP 클래스가 권한 부여에 사용되고 있으면, TCP 리소스가 액세스를 허용하는 경우에만 호스트는 로컬 호스트에서 서비스를 가져올 수 있습니다. 또한 TCP 리소스가 액세스를 허용하는 경우에만 로컬 호스트의 사용자나 그룹은 TCP/IP 서비스를 사용하여 원격 호스트에 액세스할 수 있습니다.

TCP 레코드의 ACL 은 호스트(HOST), 호스트 그룹(GHOST), 네트워크(HOSTNET) 및 호스트 집합(HOSTNP)에 대한 액세스 유형을 지정할 수 있습니다.

TCP 레코드의 CACL 은 호스트(HOST), 호스트 그룹(GHOST), 네트워크(HOSTNET) 및 호스트 집합(HOSTNP)에 대한 액세스 유형을 지정할 수 있으며, 사용자 및 그룹에 대한 액세스 유형도 지정할 수 있습니다.

호스트 이름뿐 아니라 IPv4 주소를 기반으로 하여 규칙을 설정할 수 있습니다. 이는 도메인 이름 변경에 대비할 수 있음을 의미합니다.

참고: IP 통신에 대한 CA Access Control 액세스 규칙은 IPv4 에만 적용됩니다. CA Access Control 은 IPv6 에 의한 액세스를 제어하지 않습니다.

참고: 액세스 조건으로 CONNECT 클래스가 사용되는 경우 TCP 클래스는 액세스를 효과적으로 제어할 수 없습니다. 연결 보호를 위해 TCP 클래스와 CONNECT 클래스 중 하나를 사용하고, 둘 다 사용하지는 마십시오.

TCP 레코드의 키는 TCP/IP 서비스를 나타내는 이름입니다. TCP 클래스는 나가는 서비스와 들어오는 서비스를 모두 제어합니다.

다음 정의는 TCP 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

로컬 호스트가 서비스를 제공하는 호스트 및 허용되는 액세스 유형을 정의합니다.

액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

호스트 참조

HOST, GHOST, HOSTNET 또는 HOSTNP 레코드를 정의합니다.

허용된 액세스

참조된 호스트가 리소스에 대해 갖는 액세스 권한입니다. 유효한 액세스 권한은 다음과 같습니다.

- **none**-호스트가 어떤 작업도 수행할 수 없습니다.
- **read**-호스트가 로컬 호스트에서 TCP 서비스를 가져올 수 있습니다.

이 속성을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CACL

액세스할 수 있는 리소스 및 호스트에 대한 액세스가 허용된 접근자(사용자 및 그룹) 목록 CACL(조건부 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

호스트 참조

HOST, GHOST, HOSTNET 또는 HOSTNP 레코드를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다. 유효한 액세스 유형은 다음과 같습니다.

- **write**-접근자가 이 서비스를 사용하여 호스트 또는 호스트 그룹에 액세스할 수 있습니다.
- **none**-접근자가 이 서비스를 사용하여 호스트 또는 호스트 그룹에 액세스할 수 없습니다.

이 속성을 수정하려면 authorize 또는 authorize- 명령을 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

TERMINAL 클래스

TERMINAL 클래스의 각 레코드는 로컬 호스트의 터미널, 네트워크의 다른 호스트 또는 로그인 세션이 작성될 수 있는 X 터미널을 정의합니다. 또한 터미널 이름이나 IP 주소 패턴(와일드카드 사용)과 일치하는 터미널을 정의합니다. 사용자 로그인 프로시저 중 터미널 권한을 확인하여 사용자가 사용할 수 있는 권한이 없는 터미널에서 로그인할 수 없도록 합니다.

TERMINAL 클래스는 관리 액세스도 제어합니다. ADMIN 사용자만이 올바른 액세스 권한을 가진 터미널에서 CA Access Control 을 관리할 수 있습니다.

새 TERMINAL 레코드를 정의하면 CA Access Control 은 사용자가 제공하는 이름을 정규화된 이름으로 변환하려고 시도합니다. 변환이 되면 데이터베이스에 정규화된 이름을 저장합니다. 변환이 실패할 경우 지정된 이름을 저장합니다. 이 레코드(chres, showres, rmres, authorize 등)를 참조하는 후속 명령을 실행할 때 데이터베이스에 나타나는 이름을 사용해야 합니다.

TERMINAL 레코드의 키는 터미널의 이름입니다. 이 이름을 통해 터미널이 CA Access Control 에서 식별됩니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 속하는 GTERMINAL 또는 CONTAINER 레코드 목록입니다.

TERMINAL 클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 또는 GTERMINAL 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다**.**

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

UACC 클래스

UACC 클래스의 각 레코드는 리소스 클래스에 허용된 기본 액세스를 정의합니다. UACC 레코드는 CA Access Control 의 보호를 받지 못하는 클래스의 리소스에 허용된 액세스 수준도 결정합니다.

UACC 는 대부분의 클래스에 적용되지만 모든 클래스에 적용되는 것은 아닙니다. 다음 표에서는 각 클래스가 UACC 클래스를 사용하는 방법을 보여줍니다.

UACC 사용	클래스
표준	ADMIN, APPL, AUTHHOST, CALENDAR, CONNECT, CONTAINER, DOMAIN, GAPPL, GAUTHHOST, GHOST, GSUDO, GTERMINAL, HOLIDAY, HOST, HOSTNET, HOSTNP, MFTERMINAL, POLICY,

UACC 사용	클래스
	PROCESS, PROGRAM, REGKEY, REGVAL, RULESET, SUDO, SURROGATE, TCP, TERMINAL, USER_DIR, 사용자 정의 클래스
비표준	FILE, GFILE
없음	AGENT, AGENT_TYPE, CATEGORY, GROUP, PWPOLICY, RESOURCE_DESC, RESPONSE_TAB, SECFILE, SECLABEL, SEOS, SPECIALPGM, USER, USER_ATTR

special _restricted 그룹 외부에 있는 사용자의 경우, UACC 클래스의 FILE 에 대한 레코드는 seos.ini, seosd.trace, seos.audit 및 seos.error 파일과 같이 CA Access Control 의 일부인 파일만 보호합니다. 이런 파일은 CA Access Control 에 명시적으로 정의되어 있지 않지만 CA Access Control 에 의해 자동으로 보호됩니다.

UACC 클래스 레코드의 키는 UACC 속성을 정의하는 클래스의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개변수를 사용하십시오.

ALLOWACCS

클래스에 대해 허용된 모든 액세스 목록입니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

USER 클래스

USER 클래스의 각 레코드는 CA Access Control 데이터베이스의 사용자를 정의합니다.

USER 레코드의 키는 사용자의 이름으로, 시스템에 로그인할 때 사용자가 입력하는 이름입니다.

CA Access Control 끝점 관리에서 또는 selang 명령 chusr 을 사용하여 대부분의 USER 속성을 변경할 수 있습니다. chusr 을 사용하여 변경할 수 없는 속성은 정보로 표시되어 있습니다.

참고: 대부분의 경우, 그리고 달리 지정되지 않은 경우, chusr 을 사용하여 속성을 변경하려면 속성 이름을 명령 매개 변수로 사용할 수 있습니다.

CA Access Control 끝점 관리에서 또는 selang 명령 showusr 을 사용하여 모든 속성을 볼 수 있습니다.

APPLIST

CA SSO 에서 사용됩니다.

APPLIST_TIME

CA SSO 에서 사용됩니다.

APPLS

(정보) 접근자가 액세스할 수 있는 응용 프로그램 목록을 표시합니다. CA SSO 에서 사용됩니다.

AUDIT_MODE

CA Access Control 이 감사 로그에 기록하는 활동을 정의합니다. 다음과 같은 활동의 결합을 지정할 수 있습니다.

- 로깅 없음
- 추적 파일에 기록된 모든 활동
- 실패한 로그인 시도
- 성공한 로그인
- CA Access Control 에 의해 보호되는 리소스에 대해 실패한 액세스 시도
- CA Access Control 에 의해 보호되는 리소스에 대해 성공한 액세스

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 audit 매개 변수에 해당합니다.

AUTHNMTHD

(정보) 그룹 레코드와 함께 사용하는 인증 방법을 표시하며, method 1 에서 method 32 까지 또는 none 이 있습니다. CA SSO 에서 사용됩니다.

BADPASSWD

CA SSO 에서 사용됩니다.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

COUNTRY

사용자에 대한 국가 설명자를 지정하는 문자열입니다. 이 문자열은 X.500 명명 체계의 일부입니다. CA Access Control 은 권한 부여를 위해 이 문자열을 사용하지 않습니다.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

EMAIL

최대 128 자로 구성된 사용자의 전자 메일 주소를 정의합니다.

EXPIRE_DATE

접근자가 무효화되는 날짜를 정의합니다. USER 레코드의 EXPIRE_DATE 속성 값이 GROUP 레코드의 값보다 우선합니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 expire[-] 매개 변수에 해당합니다.

FULLNAME

접근자와 관련된 전체 이름을 정의합니다. CA Access Control 은 감사 로그 메시지에서 접근자를 식별하기 위해 전체 이름을 사용합니다(권한 부여에는 사용하지 않음).

FULLNAME 은 영숫자 문자열입니다. 최대 길이가 255 자인 그룹용입니다. 최대 길이가 47 자인 사용자용입니다.

GAPPLS

(정보) 사용자가 액세스할 수 있는 응용 프로그램 그룹 목록을 나타냅니다. CA SSO 에서 사용됩니다.

GRACELOGIN

암호가 만료된 후 사용자가 갖는 유예 로그인 수를 정의합니다. 유예 로그인 횟수를 초과하면 시스템에 대한 액세스가 거부되고 시스템 관리자에게 새 암호를 문의해야 합니다.

유예 로그인 수는 0 에서 255 사이여야 합니다. 이 값이 0 이면 사용자는 로그인할 수 없습니다.

USER 레코드의 GRACELOGIN 속성 값은 GROUP 레코드의 NGRACE 값보다 우선됩니다. 두 가지 모두 SEOS 클래스 레코드의 PASSWDRULES 속성보다 우선됩니다.

참고: 이 속성은 ch[x]usr 명령의 grace 매개 변수에 해당합니다.

GROUPS

(정보) 사용자가 속해 있는 사용자 그룹의 목록을 표시합니다. 이 속성에는 사용자가 속해 있는 각 그룹의 사용자에게 할당된 모든 그룹 권한(예: 그룹 관리 권한(GROUP-ADMIN))도 포함되어 있습니다.

이 속성에 포함된 그룹 목록은 네이티브 환경 GROUPS 속성에 있는 목록과 다를 수도 있습니다.

참고: 이 속성은 ch[x]usr 명령으로 수정할 수 없습니다. 이 속성을 수정하려면 대신 join[-] 또는 joinx[-] 명령을 사용하십시오.

HOMEDIR

(UNIX 전용) 사용자의 홈 디렉터리를 정의합니다. CA SSO 에서 사용됩니다.

INACTIVE

사용자를 비활성 상태로 변경하기 전에 경과해야 하는 비활성 기간을 정의합니다. 계정 상태가 비활성이면 사용자는 로그인할 수 없습니다.

USER 레코드의 INACTIVE 속성 값은 GROUP 레코드의 값보다 우선됩니다. 두 가지 모두 SEOS 클래스 레코드의 INACT 속성보다 우선됩니다.

참고: CA Access Control 은 상태를 저장하지 않고 동적으로 상태를 계산합니다. 비활성 사용자를 식별하려면 INACTIVE 값을 사용자의 LAST_ACC_TIME 값과 비교해야 합니다.

LAST_ACC_TERM

마지막으로 로그인한 터미널을 표시합니다.

LAST_ACC_TIME

최종 로그인의 날짜 및 시간을 표시합니다.

LOCALAPPS

CA SSO 에서 사용됩니다.

LOCATION

사용자 위치를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

LOGININFO

사용자가 특정 응용 프로그램 및 감사 데이터에 로그인하기 위해 필요한 정보를 정의합니다. LOGININFO 에는 사용자에게 액세스 권한이 부여된 각 응용 프로그램에 대한 각 목록이 있습니다. CA SSO 에서 사용됩니다.

LOGSHIFT

근무 시간 이외에도 로그인을 허용할지 여부를 나타냅니다. CA Access Control 은 감사 로그에 이 이벤트에 대한 감사 레코드를 기록합니다.

MAXLOGINS

한 사용자에게 허용되는 동시 로그인의 최대 수를 정의합니다. 영(0) 값은 사용자가 동시 로그인 수로 어떤 값이든 가질 수 있음을 나타냅니다.

사용자가 로그인해서 데이터베이스를 관리하려면(예: selang 을 실행해서), MAXLOGINS 는 0 이거나 1 보다 커야 합니다. 그 이유는 CA Access Control 이 각 작업(로그인, selang, GUI 등)을 터미널 세션으로 간주하기 때문입니다.

USER 레코드의 MAXLOGINS 속성 값이 GROUP 레코드의 값보다 우선합니다. 두 가지 모두 SEOS 클래스 레코드의 MAXLOGINS 값보다 우선합니다.

MIN_TIME

사용자의 암호 변경 간격으로 허용된 최소 시간(일)을 정의합니다.

USER 레코드의 MIN_TIME 속성 값이 GROUP 레코드의 값보다 우선됩니다. 두 가지 모두 SEOS 클래스 레코드의 PASSWDRULES 속성보다 우선됩니다.

참고: 이 속성은 ch[x]usr 명령의 min_life 매개 변수에 해당합니다.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OBJ_TYPE

사용자 권한 특성을 지정합니다. 이러한 각각의 특성은 ch[x]usr 명령에서 동일한 이름을 갖는 매개 변수에 해당합니다. 사용자는 다음의 권한 특성 중 하나 이상을 가질 수 있습니다.

ADMIN

사용자가 UNIX 환경의 루트와 유사하게 관리 기능을 수행할 수 있는지 여부를 지정합니다.

AUDITOR

사용자가 시스템을 모니터하고, 데이터베이스의 정보를 나열하고, 기존 레코드에 대해 감사 모드를 설정할 수 있는지 여부를 지정합니다.

IGN_HOL

사용자가 HOLIDAY 레코드에 정의된 기간 중 언제든지 로그인할 수 있는지 여부를 지정합니다.

LOGICAL

사용자가 내부 CA Access Control 용도로만 사용되며 실제 사용자가 로그인하기 위해 사용할 수 없도록 지정합니다.

예를 들어, 리소스의 소유자조차 리소스에 액세스할 수 없도록 하기 위해 리소스의 소유자로 사용할 수 있는 사용자 nobody 는 기본적으로 논리적 사용자입니다. 즉, 이 계정을 사용하여 어떤 사용자도 로그인할 수 없습니다.

OPERATOR

사용자가 데이터베이스의 모든 정보를 나열하고 secons 유틸리티를 사용할 수 있는지 여부를 지정합니다.

PWMANAGER

사용자가 다른 사용자의 암호 설정을 수정하고 serevu 유틸리티에 의해 비활성화된 사용자 계정을 활성화할 수 있는지 여부를 지정합니다.

SERVER

프로세스가 사용자에게 권한 부여를 요청하고 SEOSROUTE_VerifyCreate API 호출을 발행할 수 있는지 여부를 지정합니다.

OIDCRDDATA

CA SSO 에서 사용됩니다.

OLD_PASSWD

사용자 이전 암호의 암호화된 목록을 포함합니다. 사용자는 이 목록에서 새 암호를 선택할 수 없습니다. OLD_PASSWD 에 저장되는 최대 암호 수는 setoptions 명령으로 결정됩니다.

ORG_UNIT

사용자가 근무하는 조직 단위에 대한 정보를 저장하는 문자열입니다. 이 문자열은 X.500 명명 체계의 일부입니다. CA Access Control 은 권한 부여를 위해 이 문자열을 사용하지 않습니다.

ORGANIZATION

사용자가 근무하는 조직을 정의합니다. 이 문자열은 X.500 명명 체계의 일부입니다. CA Access Control 은 권한 부여를 위해 이 문자열을 사용하지 않습니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PASSWD_A_C_W

이 레코드에 대한 사용자 암호를 마지막으로 변경한 ADMIN 사용자를 나타냅니다.

PASSWD_INT

사용자 암호 변경 사이의 최대 시간(일 수)을 정의합니다.

USER 레코드의 PASSWD_INT 속성 값은 GROUP 레코드의 값보다 우선됩니다. 두 가지 모두 SEOS 클래스 레코드의 PASSWDRULES 속성보다 우선됩니다.

참고: 이 속성은 ch[x]usr 명령의 interval 매개 변수에 해당합니다.

PASSWD_L_A_C

관리자가 마지막으로 암호를 업데이트한 날짜 및 시간을 표시합니다.

PASSWD_L_C

사용자가 마지막으로 암호를 업데이트한 날짜 및 시간을 표시합니다.

PGMINFO

CA Access Control 에 의해 자동으로 생성되는 프로그램 정보를 정의합니다.

Watchdog 은 이 속성에 저장된 정보를 자동으로 확인합니다. 이 정보가 변경되면 CA Access Control 은 해당 프로그램을 언트러스트된 것으로 정의합니다.

다음 플래그 중 하나를 선택하여 확인 프로세스에서 연관된 정보를 제외할 수 있습니다.

crc

주기적 중복 검사 및 MD5 서명입니다.

ctime

(UNIX 전용) 마지막으로 파일 상태가 변경된 시간입니다.

device

UNIX 에서는 파일이 상주하는 논리 디스크입니다. Windows 에서는 파일을 포함하는 디스크의 드라이브 번호입니다.

group

프로그램 파일을 소유하고 있는 그룹입니다.

inode

UNIX 에서는 프로그램 파일의 파일 시스템 주소입니다. Windows 에서는 의미가 없습니다.

mode

프로그램 파일에 대한 관련 보안 보호 모드입니다.

mtime

프로그램 파일이 마지막으로 수정된 시간입니다.

owner

프로그램 파일을 소유하는 사용자입니다.

sha1

SHA1 서명입니다. 프로그램 또는 중요한 파일에 적용할 수 있는 보안 해시 알고리즘이라고 하는 디지털 서명 방법입니다.

size

프로그램 파일의 크기입니다.

이 속성에서 플래그를 수정하려면 chres, editres 또는 newres 명령과 함께 flags, flags+ 또는 flags- 매개 변수를 사용합니다.

PHONE

사용자의 전화 번호를 정의합니다. 이 정보는 권한 부여에 사용되지 않습니다.

POLICYMODEL

sepass 유틸리티로 사용자 암호를 변경할 때 새 암호를 수신하는 PMDB 를 지정합니다. 이 속성에 대한 값을 입력하는 경우 암호는 parent_pmd 또는 passwd_pmd 구성 설정에 의해 정의된 정책 모델로 전송되지 않습니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 pmdb[-] 매개 변수에 해당합니다.

PROFILE

사용자의 프로필 경로를 정의합니다. 이 문자열은 로컬 절대 경로 또는 UNC 경로를 포함할 수 있습니다.

PWD_AUTOGEN

사용자 암호의 자동 생성 여부를 표시합니다. CA SSO 에서 사용됩니다.

기본값은 no 입니다.

PWD_SYNC

사용자 암호가 모든 사용자 응용 프로그램에 대해 자동으로 동일하게 유지되는지 여부를 표시합니다. CA SSO 에서 사용됩니다.

기본값은 no 입니다.

RESUME_DATE

일시 중단된 USER 계정의 일시 중단이 해제되는 날짜를 정의합니다.

RESUME_DATE 와 SUSPEND_DATE 는 함께 작동합니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 resume[-] 매개 변수에 해당합니다.

REVACL

접근자의 액세스 제어 목록을 표시합니다.

REVOKE_COUNT

CA SSO 에서 사용됩니다.

SCRIPT_VARS

CA SSO 에서 사용되며, 응용 프로그램별로 저장되는 응용 프로그램 스크립트의 변수 값이 있는 변수 목록을 정의합니다.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

SESSION_GROUP

사용자에 대해 SSO 세션 그룹을 정의합니다. SESSION_GROUP 속성의 최대 길이는 16 자입니다.

Windows 의 경우 원하는 이름이 드롭다운 목록에 없을 경우 관리자가 세션 그룹의 새 이름을 입력할 수 있습니다.

CA SSO 에서 사용됩니다.

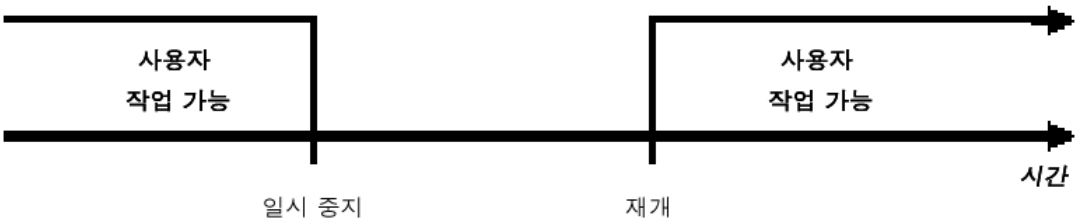
SHIFT

CA SSO 에서 사용됩니다.

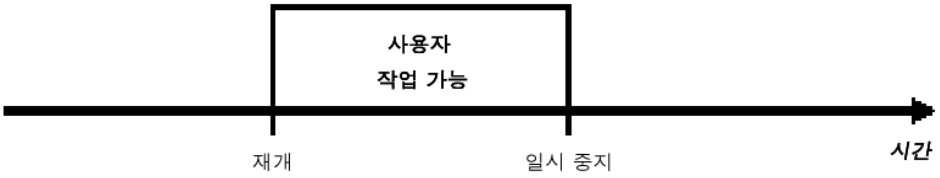
SUSPEND_DATE

사용자 계정이 일시 중단되어 무효화되는 날짜를 정의합니다.

레코드의 일시 중지 날짜가 계속 날짜보다 앞에 있는 경우 사용자는 일시 중지 날짜 전과 계속 날짜 후에 작업할 수 있습니다.



사용자가 가지고 있는 다시 시작 날짜가 일시 중단 날짜보다 빠른 경우, 다시 시작 날짜 전이라도 레코드가 무효화됩니다. 사용자는 다시 시작 날짜와 중단 날짜 사이에만 작업할 수 있습니다.



사용자 레코드의 SUSPEND_DATE 속성 값은 그룹 레코드의 값보다 우선합니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 suspend[-] 매개 변수에 해당합니다.

SUSPEND_WHO

일시 중단된 날짜를 활성화한 관리자를 표시합니다.

참고: 이 속성은 ch[x]usr 명령의 suspend[-] 매개 변수에 해당합니다.

UALIAS

하나 이상의 인증 호스트에 정의되어 있는 특정 사용자의 별칭을 표시합니다. CA SSO 에서 사용됩니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

USER_ATTR 클래스

USER_ATTR 클래스의 각 레코드는 CA SSO 사용자 디렉터리의 유효한 사용자 특성을 정의합니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ATTR_PREDEFS

특정 특성에 대해 허용된 값의 목록입니다.

ATTRNAME

(정보) 특성의 이름입니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DBFIELD

userdir 데이터베이스에 있는 필드의 이름입니다. 각 데이터베이스는 서로 다른 특성을 포함할 수 있으므로, 특성 필드를 동기화해야 합니다.

FIELDID

(정보) DB 필드의 ID 입니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PARAMETER_TYPE

사용자 특성이 문자열인지 숫자인지를 나타냅니다.

PRIORITY

사용자 특성의 우선 순위: PARAM_RULE 개체(예: APPL, URL)에 대해 권한 부여 규칙을 설정할 때, 이 규칙은 사용자 특성이 참조하는 우선 순위로 정의됩니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다.
유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

USER_DIR_PROP

(정보) 사용자 디렉터리의 이름

USERATTR_FLAGS

특성에 대한 정보를 포함합니다. 플래그는 다음 값을 포함할 수 있습니다.

- **aznchk**-이 특성을 권한 부여에 사용할지 여부를 나타냅니다.
- **predef**(미리 정의됨), **freetext**(자유 텍스트) 또는 **userdir**(사용자 디렉터리)-이 세 값은 사용자 특성의 소스를 지정합니다.
- **user** 또는 **group-**이 값은 특성(접근자)이 사용자인지 또는 그룹인지를 나타냅니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

USER_DIR 클래스

USER_DIR 클래스의 각 레코드는 CA SSO 사용자 디렉터리를 정의합니다.

USER_DIR 레코드의 키는 디렉터리의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ADMIN_NAME

디렉터리 관리자의 로그인 이름입니다.

ADMIN_PWD

디렉터리 관리자의 암호입니다. 암호는 일반 텍스트 형식으로 저장됩니다. 암호는 selang 으로 표시되지 않지만 seadmapi 함수를 사용하여 얻을 수 있습니다.

AZNACL

권한 부여 ACL 을 정의합니다. 권한 부여 ACL 은 리소스 설명을 기초로 리소스에 대한 액세스를 허용하는 ACL 입니다. 설명은 개체가 아닌 권한 부여 엔진으로 전송됩니다. 일반적으로 AZNACL 이 사용될 경우에는 개체가 데이터베이스에 없습니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CONTOBJ_CLS

컨테이너 개체가 상속하는 클래스의 이름입니다(LDAP 에 새 로그인 정보 컨테이너 작성 시 필요함).

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DIR_TYPE

디렉터리 유형입니다. 유효한 값은 ETRUST_AC, LDAP, ODBC, NT_Domain 또는 none 입니다.

GRPOBJ_CLS

그룹 개체가 상속하는 클래스의 이름입니다(LDAP 에 새 그룹 작성 시 필요함).

LICONTOBJ_CLS

로그인 정보 컨테이너 개체가 상속하는 클래스의 이름입니다(LDAP 에 새 로그인 정보 컨테이너 작성 시 필요함).

LIOBJ_CLS

로그인 정보 개체가 상속하는 클래스의 이름입니다(LDAP 에 새 로그인 정보 작성 시 필요함).

MAX_RET_ITEMS

검색되는 최대 항목 수입니다. 기본값은 디렉터리 유형에 따라 달라집니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PATH

LDAP 트리에서 모든 쿼리를 시작하는 관련 구분 이름입니다.

PORT_NUM

호스트 컴퓨터가 디렉터리에 액세스하기 위해 사용한 포트 번호입니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

TIMEOUT_CON

시간 초과 오류 메시지를 작성하기 전에 시스템이 디렉터리 연결을 대기하는 시간(초)입니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

USERATTR_LIST

USER_DIR 개체를 USER_DIR 매개 변수의 값으로 사용하여 작성된 USER_ATTR 클래스에 있는 개체의 목록입니다.

USERDIR_HOST

디렉터리에 대한 호스트 컴퓨터의 이름입니다. 이 속성은 클래스 레코드에 정의되어야 합니다.

USROBJ_CLS

사용자 개체가 상속하는 클래스의 이름입니다(LDAP 에 새 사용자 작성 시 필요함).

VERSION

디렉터리의 버전 번호입니다.

WEBSERVICE 클래스

WEBSERVICE 클래스는 사용되지 않습니다. CA Access Control 에서 이 클래스를 사용하지 않습니다.

WINSERVICE 클래스

WINSERVICE 클래스의 각 레코드는 Windows 서비스를 정의합니다. Windows 서비스에 대한 액세스 규칙을 정의하려면 WINSERVICE 클래스의 레코드를 사용하십시오.

WINSERVICE 클래스 레코드의 키는 서비스의 Windows 이름입니다.

참고: 대부분의 경우, 그리고 달리 지정되지 않은 경우, selang chres 명령을 사용하여 속성을 변경하려면 속성 이름을 명령 매개 변수로 사용할 수 있습니다.

CA Access Control 끝점 관리에서 또는 selang 명령 showres WINSERVICE 를 사용하여 모든 속성을 볼 수 있습니다.

ACL

리소스에 대해 액세스가 허용된 접근자(사용자 및 그룹) 목록 및 접근자의 액세스 유형을 정의합니다.

ACL(액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

ACL 을 수정하려면 authorize 또는 authorize- 명령과 함께 access 매개 변수를 사용하십시오.

CALACL

리소스에 대한 액세스가 허용되는 접근자(사용자 및 그룹) 목록 및 해당 액세스 유형을 Unicenter NSM 달력 상태에 따라 정의합니다.

CALACL(달력 액세스 제어 목록)에 있는 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Calendar

Unicenter TNG 의 달력에 대한 참조를 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

액세스는 달력이 ON 일 때만 허용되고, 다른 모든 경우에는 액세스가 거부됩니다.

달력 ACL 에 정의된 액세스에 따라 사용자 또는 그룹에게 리소스에 대한 액세스를 허용하려면 calendar 매개 변수와 함께 권한 부여 명령을 사용하십시오.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

GROUPS

리소스 레코드가 소속되는 CONTAINER 레코드 목록을 정의합니다.

클래스 레코드에서 이 속성을 수정하려면 해당 CONTAINER 레코드에서 MEMBERS 속성을 변경해야 합니다.

이 속성을 수정하려면 mem+ 또는 mem- 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

NACL

리소스의 NACL 속성은 리소스에 대한 권한이 거부되는 접근자를 거부되는 액세스 유형(예: 쓰기)과 함께 정의하는 액세스 제어 목록입니다. ACL, CALACL, PACL 을 참조하십시오. NACL 의 각 항목은 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Access

접근자에게 거부되는 액세스 유형을 정의합니다.

이 속성을 수정하려면 authorize deniedaccess 명령 또는 authorize-deniedaccess- 명령을 사용하십시오.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다**.**

제한: 30 자.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PACL

특정 프로그램(또는 이름 패턴이 일치하는 프로그램)이 액세스를 요청할 때 리소스에 대한 액세스가 허용되는 접근자 목록 및 해당 액세스 유형을 정의합니다. PACL(프로그램 액세스 제어 목록)의 각 요소는 다음 정보를 포함합니다.

Accessor

접근자를 정의합니다.

Program

PROGRAM 클래스의 레코드에 대한 참조를 명시적으로 정의하거나 와일드카드 패턴 일치를 사용하여 정의합니다.

Access

접근자가 리소스에 대해 갖는 액세스 권한을 정의합니다.

참고: PACL 의 리소스를 지정하기 위해 와일드카드 문자를 사용할 수 있습니다.

PACL 에 프로그램, 접근자 및 해당 액세스 유형을 추가하려면 selang 권한 부여 명령과 함께 via(pgm) 매개 변수를 사용하십시오. PACL 에서 접근자를 제거하려면 authorize- 명령을 사용할 수 있습니다.

RAUDIT

CA Access Control 이 감사 로그에 기록하는 액세스 이벤트의 유형을 정의합니다. RAUDIT 의 이름은 Resource AUDIT 에서 온 것입니다. 유효한 값:

all

모든 액세스 요청.

success

허용된 액세스 요청.

failure

거부된 액세스 요청(기본값).

none

액세스 요청 없음.

CA Access Control 은 리소스에 액세스하려는 각 시도에 대한 이벤트를 기록합니다. 그러나 액세스 규칙이 리소스에 직접 적용되었는지, 아니면 구성원으로서 리소스를 가지고 있던 그룹이나 클래스에 적용되었는지는 기록하지 않습니다.

chres 및 chfile 명령의 audit 매개 변수를 사용하여 감사 모드를 수정하십시오.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

UACC

리소스에 대한 기본 액세스 권한을 정의합니다. 이 액세스 권한은 CA Access Control 에 정의되어 있지 않거나 리소스 ACL 에 나타나지 않는 접근자에게 허용된 액세스를 나타냅니다.

이 속성을 수정하려면 defaccess 매개 변수를 chres, editres 또는 newres 명령과 함께 사용하십시오.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

WARNING

경고 모드의 활성화 여부를 지정합니다. 리소스에 대해 경고 모드가 활성화되면 리소스에 대한 모든 액세스 요청이 허용됩니다. 액세스 요청이 액세스 규칙을 위반하는 경우 감사 로그에 레코드가 기록됩니다.

XGROUP 클래스

XGROUP 클래스의 각 레코드는 데이터베이스의 사용자 그룹을 정의합니다.

각 XGROUP 클래스 레코드의 키는 그룹의 이름입니다.

참고: 프로필 그룹의 속성은 프로필 그룹과 관련된 각 사용자에게 적용됩니다. 그러나 사용자(USER 또는 XUSER) 레코드에 동일한 속성이 지정되어 있으면 사용자 레코드가 프로필 그룹 레코드의 속성보다 우선합니다.

CA Access Control 끝점 관리에서 또는 selang 명령 chxgrp 를 사용하여 이러한 속성의 대부분을 변경할 수 있습니다.

참고: 대부분의 경우, 그리고 달리 지정되지 않은 경우, chxgrp 를 사용하여 속성을 변경하려면 속성 이름을 명령 매개 변수로 사용할 수 있습니다.

CA Access Control 끝점 관리에서 또는 selang 명령 showxgrp 를 사용하여 모든 속성을 볼 수 있습니다.

APPLS

(정보) 접근자가 액세스할 수 있는 응용 프로그램 목록을 표시합니다. CA SSO 에서 사용됩니다.

AUDIT_MODE

CA Access Control 이 감사 로그에 기록하는 활동을 정의합니다. 다음과 같은 활동의 결합을 지정할 수 있습니다.

- 로깅 없음
- 추적 파일에 기록된 모든 활동
- 실패한 로그인 시도
- 성공한 로그인
- CA Access Control 에 의해 보호되는 리소스에 대해 실패한 액세스 시도
- CA Access Control 에 의해 보호되는 리소스에 대해 성공한 액세스

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 audit 매개 변수에 해당합니다. GROUP 또는 XGROUP 에 AUDIT_MODE 를 사용하여 그룹의 모든 구성원에 대한 감사 모드를 설정할 수 있습니다. 하지만 사용자의 감사 모드가 USER 레코드, XUSER 레코드 또는 프로필 그룹에 정의되어 있는 경우 AUDIT_MODE 를 사용하여 그룹 구성원에 대한 감사 모드를 설정할 수 없습니다.

AUTHNMTHD

(정보) 그룹 레코드와 함께 사용하는 인증 방법을 표시하며, method 1 에서 method 32 까지 또는 none 이 있습니다. CA SSO 에서 사용됩니다.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

EXPIRE_DATE

접근자가 무효화되는 날짜를 정의합니다. USER 레코드의 EXPIRE_DATE 속성 값이 GROUP 레코드의 값보다 우선합니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 expire[-] 매개 변수에 해당합니다.

FULLNAME

접근자와 관련된 전체 이름을 정의합니다. CA Access Control 은 감사 로그 메시지에서 접근자를 식별하기 위해 전체 이름을 사용합니다(권한 부여에는 사용하지 않음).

FULLNAME 은 영숫자 문자열입니다. 최대 길이가 255 자인 그룹용입니다. 최대 길이가 47 자인 사용자용입니다.

GAPPLS

그룹이 액세스할 수 있는 응용 프로그램 그룹 목록을 정의합니다. CA SSO 에서 사용됩니다.

GROUP_MEMBER

이 그룹의 구성원인 그룹을 정의합니다.

MEMBER_OF

이 그룹이 구성원으로 있는 그룹을 정의합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

PROFUSR

이 프로필 그룹과 관련된 사용자 목록을 표시합니다.

PWD_AUTOGEN

그룹 암호의 자동 생성 여부를 나타냅니다. 기본값은 "no"입니다. CA SSO 에서 사용됩니다.

PWD_SYNC

그룹 암호가 모든 그룹 응용 프로그램에 대해 자동으로 동일하게 유지되는지 여부를 나타냅니다. 기본값은 "no"입니다. CA SSO 에서 사용됩니다.

PWPOLICY

그룹에 대한 암호 정책의 레코드 이름을 정의합니다. 암호 정책은 새 암호의 유효성을 검사하고 암호가 만료되는 시기를 정의하는 규칙 집합입니다. 기본값은 유효성 검사 없음입니다. CA SSO 에서 사용됩니다.

REVACL

접근자의 액세스 제어 목록을 표시합니다.

SHELL

(UNIX 만) 사용자가 이 그룹의 구성원일 때 새 UNIX 사용자에게 할당된 셸 프로그램입니다.

이 속성을 수정하려면 chxgrp 명령과 함께 shellprog 매개 변수를 사용하십시오.

SUBGROUP

이 그룹을 부모 그룹으로 가지는 그룹 목록을 표시합니다.

SUPGROUP

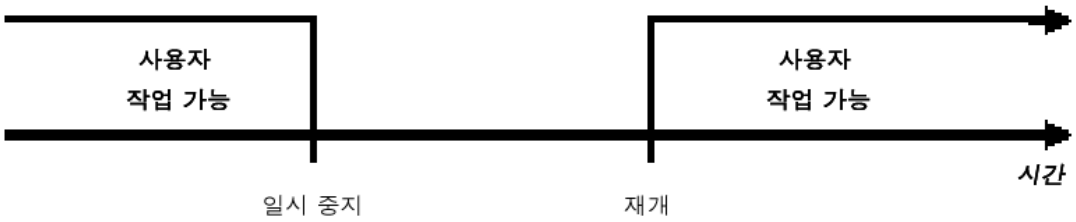
부모 그룹("superior" 그룹)의 이름을 정의합니다.

이 속성을 수정하려면 ch[x]grp 명령과 함께 parent[-] 매개 변수를 사용하십시오.

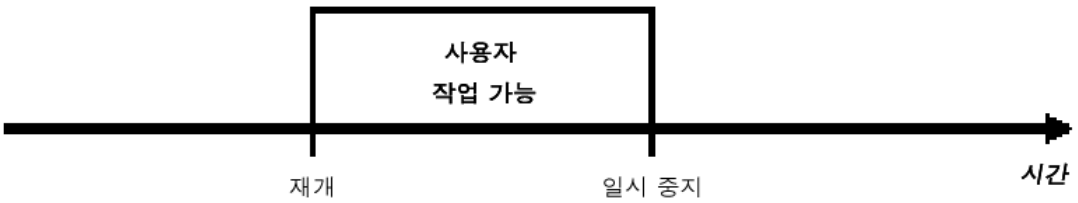
SUSPEND_DATE

사용자 계정이 일시 중단되어 무효화되는 날짜를 정의합니다.

레코드의 일시 중지 날짜가 계속 날짜보다 앞에 있는 경우 사용자는 일시 중지 날짜 전과 계속 날짜 후에 작업할 수 있습니다.



사용자가 가지고 있는 다시 시작 날짜가 일시 중단 날짜보다 빠른 경우, 다시 시작 날짜 전이라도 레코드가 무효화됩니다. 사용자는 다시 시작 날짜와 중단 날짜 사이에만 작업할 수 있습니다.



사용자 레코드의 SUSPEND_DATE 속성 값은 그룹 레코드의 값보다 우선합니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 suspend[-] 매개 변수에 해당합니다.

SUSPEND_WHO

일시 중단된 날짜를 활성화한 관리자를 표시합니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

USERLIST

그룹에 속해 있는 사용자를 표시합니다.

이 속성에 포함된 사용자 목록은 네이티브 환경 USERS 속성에 있는 목록과 다를 수도 있습니다.

XUSER 클래스

XUSER 클래스의 각 레코드는 데이터베이스의 엔터프라이즈 사용자를 정의합니다.

XUSER 레코드의 키는 사용자의 이름, 즉 시스템에 로그인할 때 사용자가 입력하는 이름입니다.

CA Access Control 끝점 관리에서 또는 selang 명령 chxusr 을 사용하여 이러한 속성의 대부분을 변경할 수 있습니다.

참고: 대부분의 경우, 그리고 달리 지정되지 않은 경우, chxusr 을 사용하여 속성을 변경하려면 속성 이름을 명령 매개 변수로 사용할 수 있습니다.

CA Access Control 끝점 관리에서 또는 selang 명령 showxusr 을 사용하여 모든 속성을 볼 수 있습니다.

APPLIST

CA SSO 에서 사용됩니다.

APPLIST_TIME

CA SSO 에서 사용됩니다.

APPLS

(정보) 접근자가 액세스할 수 있는 응용 프로그램 목록을 표시합니다. CA SSO 에서 사용됩니다.

AUDIT_MODE

CA Access Control 이 감사 로그에 기록하는 활동을 정의합니다. 다음과 같은 활동의 결합을 지정할 수 있습니다.

- 로깅 없음
- 추적 파일에 기록된 모든 활동
- 실패한 로그인 시도
- 성공한 로그인
- CA Access Control 에 의해 보호되는 리소스에 대해 실패한 액세스 시도
- CA Access Control 에 의해 보호되는 리소스에 대해 성공한 액세스

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 audit 매개 변수에 해당합니다.

AUTHNMTHD

(정보) 그룹 레코드와 함께 사용하는 인증 방법을 표시하며, method 1 에서 method 32 까지 또는 none 이 있습니다. CA SSO 에서 사용됩니다.

BADPASSWD

CA SSO 에서 사용됩니다.

CALENDAR

CA Access Control 의 사용자, 그룹 및 리소스 제한을 위한 Unicenter TNG 달력 개체를 나타냅니다. CA Access Control 은 지정된 시간 간격으로 Unicenter TNG 활성 달력을 가져옵니다.

CATEGORY

사용자 또는 리소스에 할당된 하나 이상의 보안 범주를 정의합니다.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

COUNTRY

사용자에 대한 국가 설명자를 지정하는 문자열입니다. 이 문자열은 X.500 명명 체계의 일부입니다. CA Access Control 은 권한 부여를 위해 이 문자열을 사용하지 않습니다.

CREATE_TIME

(정보) 레코드가 작성된 날짜 및 시간을 표시합니다.

DAYTIME

접근자가 리소스에 액세스할 수 있는 시기를 관리하는 날짜 및 시간 제한을 정의합니다.

이 속성을 수정하려면 chres, ch[x]usr 또는 ch[x]grp 명령과 함께 restrictions 매개 변수를 사용하십시오.

daytime restrictions 의 확인은 1 분입니다.

EMAIL

최대 128 자로 구성된 사용자의 전자 메일 주소를 정의합니다.

FULLNAME

접근자와 관련된 전체 이름을 정의합니다. CA Access Control 은 감사 로그 메시지에서 접근자를 식별하기 위해 전체 이름을 사용합니다(권한 부여에는 사용하지 않음).

FULLNAME 은 영숫자 문자열입니다. 최대 길이가 255 자인 그룹용입니다. 최대 길이가 47 자인 사용자용입니다.

GAPPLS

(정보) 사용자가 액세스할 수 있는 응용 프로그램 그룹 목록을 나타냅니다. CA SSO 에서 사용됩니다.

GRACELOGIN

암호가 만료된 후 사용자가 갖는 유예 로그인 수를 정의합니다. 유예 로그인 횟수를 초과하면 시스템에 대한 액세스가 거부되고 시스템 관리자에게 새 암호를 문의해야 합니다.

유예 로그인 수는 0 에서 255 사이여야 합니다. 이 값이 0 이면 사용자는 로그인할 수 없습니다.

USER 레코드의 GRACELOGIN 속성 값은 GROUP 레코드의 NGRACE 값보다 우선됩니다. 두 가지 모두 SEOS 클래스 레코드의 PASSWDRULES 속성보다 우선됩니다.

참고: 이 속성은 ch[x]usr 명령의 grace 매개 변수에 해당합니다.

GROUPS

(정보) 사용자가 속해 있는 사용자 그룹의 목록을 표시합니다. 이 속성에는 사용자가 속해 있는 각 그룹의 사용자에게 할당된 모든 그룹 권한(예: 그룹 관리 권한(GROUP-ADMIN))도 포함되어 있습니다.

이 속성에 포함된 그룹 목록은 네이티브 환경 GROUPS 속성에 있는 목록과 다를 수도 있습니다.

참고: 이 속성은 ch[x]usr 명령으로 수정할 수 없습니다. 이 속성을 수정하려면 대신 join[-] 또는 joinx[-] 명령을 사용하십시오.

INACTIVE

사용자를 비활성 상태로 변경하기 전에 경과해야 하는 비활성 기간을 정의합니다. 계정 상태가 비활성이면 사용자는 로그인할 수 없습니다.

USER 레코드의 INACTIVE 속성 값은 GROUP 레코드의 값보다 우선됩니다. 두 가지 모두 SEOS 클래스 레코드의 INACT 속성보다 우선됩니다.

참고: CA Access Control 은 상태를 저장하지 않고 동적으로 상태를 계산합니다. 비활성 사용자를 식별하려면 INACTIVE 값을 사용자의 LAST_ACC_TIME 값과 비교해야 합니다.

LAST_ACC_TERM

마지막으로 로그인한 터미널을 표시합니다.

LAST_ACC_TIME

최종 로그인의 날짜 및 시간을 표시합니다.

LOCALAPPS

CA SSO 에서 사용됩니다.

LOCATION

사용자 위치를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

LOGININFO

사용자가 특정 응용 프로그램 및 감사 데이터에 로그인하기 위해 필요한 정보를 정의합니다. LOGININFO 에는 사용자에게 액세스 권한이 부여된 각 응용 프로그램에 대한 각 목록이 있습니다. CA SSO 에서 사용됩니다.

LOGSHIFT

근무 시간 이외에도 로그인을 허용할지 여부를 나타냅니다. CA Access Control 은 감사 로그에 이 이벤트에 대한 감사 레코드를 기록합니다.

MAXLOGINS

한 사용자에게 허용되는 동시 로그인의 최대 수를 정의합니다. 영(0) 값은 사용자가 동시 로그인 수로 어떤 값이든 가질 수 있음을 나타냅니다.

사용자가 로그인해서 데이터베이스를 관리하려면(예: selang 을 실행해서), MAXLOGINS 는 0 이거나 1 보다 커야 합니다. 그 이유는 CA Access Control 이 각 작업(로그인, selang, GUI 등)을 터미널 세션으로 간주하기 때문입니다.

USER 레코드의 MAXLOGINS 속성 값이 GROUP 레코드의 값보다 우선합니다. 두 가지 모두 SEOS 클래스 레코드의 MAXLOGINS 값보다 우선합니다.

MIN_TIME

사용자의 암호 변경 간격으로 허용된 최소 시간(일)을 정의합니다.

USER 레코드의 MIN_TIME 속성 값이 GROUP 레코드의 값보다 우선됩니다. 두 가지 모두 SEOS 클래스 레코드의 PASSWDRULES 속성보다 우선됩니다.

참고: 이 속성은 ch[x]usr 명령의 min_life 매개 변수에 해당합니다.

NOTIFY

리소스 또는 사용자가 감사 이벤트를 생성할 때 알림을 받을 사용자를 정의합니다. CA Access Control 은 지정된 사용자에게 감사 레코드를 전자 메일로 보낼 수 있습니다.

제한: 30 자.

OBJ_TYPE

사용자 권한 특성을 지정합니다. 이러한 각각의 특성은 ch[x]usr 명령에서 동일한 이름을 갖는 매개 변수에 해당합니다. 사용자는 다음의 권한 특성 중 하나 이상을 가질 수 있습니다.

ADMIN

사용자가 UNIX 환경의 루트와 유사하게 관리 기능을 수행할 수 있는지 여부를 지정합니다.

AUDITOR

사용자가 시스템을 모니터하고, 데이터베이스의 정보를 나열하고, 기존 레코드에 대해 감사 모드를 설정할 수 있는지 여부를 지정합니다.

IGN_HOL

사용자가 HOLIDAY 레코드에 정의된 기간 중 언제든지 로그인할 수 있는지 여부를 지정합니다.

LOGICAL

사용자가 내부 CA Access Control 용도로만 사용되며 실제 사용자가 로그인하기 위해 사용할 수 없도록 지정합니다.

예를 들어, 리소스의 소유자조차 리소스에 액세스할 수 없도록 하기 위해 리소스의 소유자로 사용할 수 있는 사용자 nobody 는 기본적으로 논리적 사용자입니다. 즉, 이 계정을 사용하여 어떤 사용자도 로그인할 수 없습니다.

OPERATOR

사용자가 데이터베이스의 모든 정보를 나열하고 secons 유틸리티를 사용할 수 있는지 여부를 지정합니다.

PWMANAGER

사용자가 다른 사용자의 암호 설정을 수정하고 serevu 유틸리티에 의해 비활성화된 사용자 계정을 활성화할 수 있는지 여부를 지정합니다.

SERVER

프로세스가 사용자에게 권한 부여를 요청하고 SEOSROUTE_VerifyCreate API 호출을 발행할 수 있는지 여부를 지정합니다.

OIDCRDDATA

CA SSO 에서 사용됩니다.

OLD_PASSWD

사용자 이전 암호의 암호화된 목록을 포함합니다. 사용자는 이 목록에서 새 암호를 선택할 수 없습니다. OLD_PASSWD 에 저장되는 최대 암호 수는 setoptions 명령으로 결정됩니다.

ORG_UNIT

사용자가 근무하는 조직 단위에 대한 정보를 저장하는 문자열입니다. 이 문자열은 X.500 명명 체계의 일부입니다. CA Access Control 은 권한 부여를 위해 이 문자열을 사용하지 않습니다.

ORGANIZATION

사용자가 근무하는 조직을 정의합니다. 이 문자열은 X.500 명명 체계의 일부입니다. CA Access Control 은 권한 부여를 위해 이 문자열을 사용하지 않습니다.

PASSWD_A_C_W

이 레코드에 대한 사용자 암호를 마지막으로 변경한 ADMIN 사용자를 나타냅니다.

PASSWD_INT

사용자 암호 변경 사이의 최대 시간(일 수)을 정의합니다.

USER 레코드의 PASSWD_INT 속성 값은 GROUP 레코드의 값보다 우선됩니다. 두 가지 모두 SEOS 클래스 레코드의 PASSWDRULES 속성보다 우선됩니다.

참고: 이 속성은 ch[x]usr 명령의 interval 매개 변수에 해당합니다.

PASSWD_L_A_C

관리자가 마지막으로 암호를 업데이트한 날짜 및 시간을 표시합니다.

PASSWD_L_C

사용자가 마지막으로 암호를 업데이트한 날짜 및 시간을 표시합니다.

PGMINFO

CA Access Control 에 의해 자동으로 생성되는 프로그램 정보를 정의합니다.

Watchdog 은 이 속성에 저장된 정보를 자동으로 확인합니다. 이 정보가 변경되면 CA Access Control 은 해당 프로그램을 언트러스트된 것으로 정의합니다.

다음 플래그 중 하나를 선택하여 확인 프로세스에서 연관된 정보를 제외할 수 있습니다.

crc

주기적 중복 검사 및 MD5 서명입니다.

ctime

(UNIX 전용) 마지막으로 파일 상태가 변경된 시간입니다.

device

UNIX 에서는 파일이 상주하는 논리 디스크입니다. Windows 에서는 파일을 포함하는 디스크의 드라이브 번호입니다.

group

프로그램 파일을 소유하고 있는 그룹입니다.

inode

UNIX 에서는 프로그램 파일의 파일 시스템 주소입니다. Windows 에서는 의미가 없습니다.

mode

프로그램 파일에 대한 관련 보안 보호 모드입니다.

mtime

프로그램 파일이 마지막으로 수정된 시간입니다.

owner

프로그램 파일을 소유하는 사용자입니다.

sha1

SHA1 서명입니다. 프로그램 또는 중요한 파일에 적용할 수 있는 보안 해시 알고리즘이라고 하는 디지털 서명 방법입니다.

size

프로그램 파일의 크기입니다.

이 속성에서 플래그를 수정하려면 chres, editres 또는 newres 명령과 함께 flags, flags+ 또는 flags- 매개 변수를 사용합니다.

PHONE

사용자의 전화 번호를 정의합니다. 이 정보는 권한 부여에 사용되지 않습니다.

PWD_AUTOGEN

사용자 암호의 자동 생성 여부를 표시합니다. CA SSO 에서 사용됩니다.

기본값은 no 입니다.

PWD_SYNC

사용자 암호가 모든 사용자 응용 프로그램에 대해 자동으로 동일하게 유지되는지 여부를 표시합니다. CA SSO 에서 사용됩니다.

기본값은 no 입니다.

RESUME_DATE

일시 중단된 USER 계정의 일시 중단이 해제되는 날짜를 정의합니다.

RESUME_DATE 와 SUSPEND_DATE 는 함께 작동합니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 resume[-] 매개 변수에 해당합니다.

REVACL

접근자의 액세스 제어 목록을 표시합니다.

REVOKE_COUNT

CA SSO 에서 사용됩니다.

SCRIPT_VARS

CA SSO 에서 사용되며, 응용 프로그램별로 저장되는 응용 프로그램 스크립트의 변수 값이 있는 변수 목록을 정의합니다.

SECLABEL

사용자나 리소스의 보안 레이블을 정의합니다.

참고: SECLABEL 속성은 chres 및 ch[x]usr 명령의 label[-] 매개 변수에 해당합니다.

SECLEVEL

접근자 또는 리소스의 보안 수준을 정의합니다.

참고: 이 속성은 ch[x]usr 및 chres 명령의 level[-] 매개 변수에 해당합니다.

SESSION_GROUP

사용자에 대해 SSO 세션 그룹을 정의합니다. SESSION_GROUP 속성의 최대 길이는 16 자입니다.

Windows 의 경우 원하는 이름이 드롭다운 목록에 없을 경우 관리자가 세션 그룹의 새 이름을 입력할 수 있습니다.

CA SSO 에서 사용됩니다.

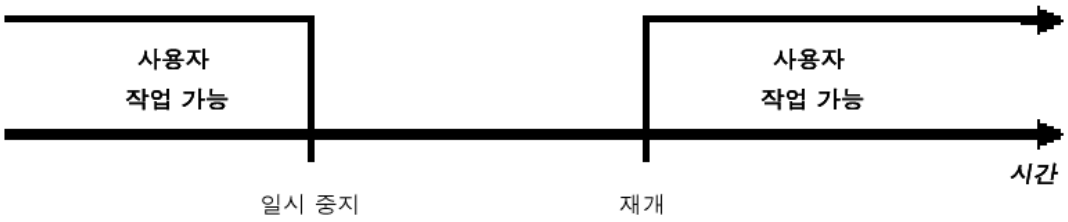
SHIFT

CA SSO 에서 사용됩니다.

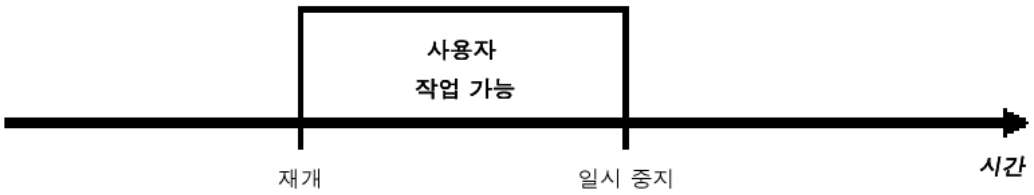
SUSPEND_DATE

사용자 계정이 일시 중단되어 무효화되는 날짜를 정의합니다.

레코드의 일시 중지 날짜가 계속 날짜보다 앞에 있는 경우 사용자는 일시 중지 날짜 전과 계속 날짜 후에 작업할 수 있습니다.



사용자가 가지고 있는 다시 시작 날짜가 일시 중단 날짜보다 빠른 경우, 다시 시작 날짜 전이라도 레코드가 무효화됩니다. 사용자는 다시 시작 날짜와 중단 날짜 사이에만 작업할 수 있습니다.



사용자 레코드의 SUSPEND_DATE 속성 값은 그룹 레코드의 값보다 우선합니다.

참고: 이 속성은 ch[x]usr 및 ch[x]grp 명령의 suspend[-] 매개 변수에 해당합니다.

SUSPEND_WHO

일시 중단된 날짜를 활성화한 관리자를 표시합니다.

UALIAS

하나 이상의 인증 호스트에 정의되어 있는 특정 사용자의 별칭을 표시합니다. CA SSO 에서 사용됩니다.

UPDATE_TIME

(정보) 레코드를 마지막으로 수정한 날짜와 시간을 표시합니다.

UPDATE_WHO

(정보) 업데이트를 수행한 관리자를 표시합니다.

Windows 환경의 클래스

이 단원에는 Windows 데이터베이스에 있는 모든 Windows 클래스와 속성(nt 환경의 클래스)에 대한 완전한 참조가 알파벳순으로 포함되어 있습니다.

참고: nt 환경이란 용어는 selang 명령 env nt 로 액세스할 수 있는 데이터베이스를 가리킵니다. 이 데이터베이스는 Windows 운영 체제가 사용자, 그룹 및 리소스에 대해 유지관리하는 데이터베이스와 동일합니다.

COM 클래스

COM 클래스의 각 레코드는 Windows 제어판의 포트에 나열된 직렬 포트(COM) 또는 병렬 포트(LPT)를 지정하여 장치를 정의합니다.

참고: CA Access Control 을 사용하여 COM 클래스의 새 개체를 작성할 수 없습니다.

COM 클래스의 키는 제어 중인 포트의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

DEV

(정보) 장치의 일련 번호를 나타내는 문자열입니다.

DACL

리소스에 액세스할 수 있는 사용자 이름 및 그룹 이름과 각 이름에 부여된 액세스 수준을 포함하는 표준 액세스 제어 목록을 정의합니다.

이 속성을 수정할 사용자는 리소스 소유자이거나 리소스에 대한 특수 액세스(ACL 수정)를 가지고 있어야 합니다.

액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

Access Type

리소스에 대한 권한을 지정합니다.

- **Allowed(허용됨)**-리소스에 대한 특수 액세스를 허용합니다.
- **Denied(거부됨)**-리소스에 대한 특수 액세스를 거부합니다.

Accessor

액세스 권한이 허용되거나 거부된 사용자 또는 그룹입니다.

Access

접근자가 리소스에 대해 가지는 액세스 권한입니다.

참고: 빈 ACL 의 경우 명시적으로 허용된 액세스가 없으므로 액세스는 암시적으로 거부됩니다. ACL이 없는 리소스의 경우 개체에 할당된 보호가 없으므로 모든 액세스 요청이 허용됩니다.

이 속성을 수정하려면 auth 또는 auth- 명령을 사용하십시오.

GID

파일 또는 장치에 대한 그룹 정보를 표시합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

SACL

Windows 시스템 액세스 제어 목록입니다 감사 지시어를 표시합니다.

DEVICE 클래스

DEVICE 클래스의 각 레코드는 Windows 제어판의 장치에 나열된 Windows 하드웨어 장치를 정의합니다.

DEVICE 클래스 레코드의 키는 제어 중인 장치의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

STARTUPTYPE

장치가 (언제) 어떻게 시작되는지 정의합니다. 옵션은

automatic

시스템이 시작되는 동안 자동으로 장치를 시작합니다.

boot

시스템을 시작할 때마다 다른 장치가 시작되기 전에 해당 장치를 시작합니다. 시스템 작동에 필수적인 중요한 장치인 경우 이 옵션을 선택합니다.

disabled

사용자가 장치를 시작하는 것을 금지합니다. 시스템은 여전히 비활성화된 장치를 시작할 수 있습니다.

manual

사용자 또는 종속 장치에 의해 장치가 시작되는 것을 허용합니다.

system

시스템을 시작할 때마다 Boot 장치가 시작된 후에 해당 장치를 시작합니다. 시스템 작동에 필수적인 중요한 장치인 경우 이 옵션을 선택합니다.

이 속성을 수정하려면 starttype 매개 변수를 chres 또는 editres 명령과 함께 사용하십시오.

STATUS

현재 서비스 상태를 변경합니다. 옵션에는 started, stopped 및 paused 가 있습니다.

이 속성을 수정하려면 status 매개 변수를 chres 또는 editres 명령과 함께 사용하십시오.

IMAGEPATH

지정된 장치의 정규화된 경로입니다.

PROFILE

사용자의 프로필에 대한 경로를 지정하는 문자열입니다. 이 문자열은 로컬 절대 경로 또는 UNC 경로를 포함할 수 있습니다.

이 속성을 수정하려면 profile 매개 변수를 chusr, editusr 또는 newusr 명령과 함께 사용하십시오.

예: 모뎀 활성화

모뎀 상태를 표시하려면 다음 selang 명령을 입력합니다.

```
showres DEVICE modem
```

모뎀을 활성화하려면 다음 명령을 입력합니다.

```
chres device modem status(started)
```

DISK 클래스

DISK 클래스의 각 레코드는 시스템 볼륨을 정의합니다. 볼륨이란 Windows 운영 체제(Server 버전)를 실행하는 컴퓨터에서 사용자가 작성하고 사용할 수 있는 모든 엔터티를 나타내는 일반적인 용어입니다. 예를 들어, 주 파티션, 확장 파티션의 논리적 드라이브, 볼륨 집합, 스트라이프 집합, 미러 집합이거나 패리티가 있는 스트라이프 집합입니다. 볼륨에는 단일 드라이브 문자가 할당되며, 파일 시스템에서 사용할 수 있도록 형식이 지정됩니다.

참고: CA Access Control 을 사용하여 DISK 클래스의 새 개체를 작성할 수 없습니다.

DISK 클래스의 키는 할당된 드라이브 문자(C:, D: 등)입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ATIME

(정보) 마지막으로 레코드에 액세스한 시간입니다.

CTIME

(정보) 만든 시간입니다.

DACL

리소스에 액세스할 수 있는 사용자 이름 및 그룹 이름과 각 이름에 부여된 액세스 수준을 포함하는 표준 액세스 제어 목록을 정의합니다.

이 속성을 수정할 사용자는 리소스 소유자이거나 리소스에 대한 특수 액세스(ACL 수정)를 가지고 있어야 합니다.

액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

Access Type

리소스에 대한 권한을 지정합니다.

- **Allowed(허용됨)**-리소스에 대한 특수 액세스를 허용합니다.
- **Denied(거부됨)**-리소스에 대한 특수 액세스를 거부합니다.

Accessor

액세스 권한이 허용되거나 거부된 사용자 또는 그룹입니다.

Access

접근자가 리소스에 대해 가지는 액세스 권한입니다.

참고: 빈 ACL 의 경우 명시적으로 허용된 액세스가 없으므로 액세스는 암시적으로 거부됩니다. ACL이 없는 리소스의 경우 개체에 할당된 보호가 없으므로 모든 액세스 요청이 허용됩니다.

이 속성을 수정하려면 auth 또는 auth- 명령을 사용하십시오.

FILE_SYSTEM

(정보) 파일 시스템을 지정하는 이름(예: FAT 또는 NTFS)입니다.

FREE_SPACE

(정보) 디스크에서 사용 가능한 총 공간(KB)입니다.

GID

파일 또는 장치에 대한 그룹 정보를 표시합니다.

LABEL

(정보) 지정된 볼륨의 이름입니다.

LINK_NUMB

(정보) 링크 수를 지정합니다. 비 NTFS 파일 시스템의 경우, 이 속성은 항상 1 입니다.

MTIME

(정보) 레코드가 마지막으로 수정된 시간입니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

SACL

Windows 시스템 액세스 제어 목록입니다 감사 지시어를 표시합니다.

TYPE

(정보) 디스크가 이동 디스크인지, 고정 디스크인지, CD-ROM 인지, RAM 디스크인지, 아니면 네트워크 드라이브인지 지정합니다.

USED_SPACE

(정보) 디스크에서 사용된 총 공간(KB)입니다.

DOMAIN 클래스

DOMAIN 클래스의 각 레코드는 공용 데이터베이스와 보안 정책(도메인)을 공유하는 컴퓨터 집합을 정의합니다. 도메인은 도메인 관리자에 의해 유지관리되는 중앙 사용자 계정과 그룹 계정에 대한 액세스를 제공합니다. 각 도메인에는 고유한 이름이 있습니다.

참고: CA Access Control 을 사용하여 DOMAIN 클래스의 새 개체를 작성할 수 없습니다.

DOMAIN 레코드에 대한 키는 도메인 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

BDC

(정보) 도메인의 디렉터리 데이터베이스 사본을 수신하며 도메인에 대한 모든 계정 및 보안 정책 정보를 포함하는 컴퓨터의 이름입니다. 사본은 주기적으로 주 도메인 컨트롤러(PDC)에 있는 마스터 사본과 자동 동기화됩니다. 백업 도메인 컨트롤러(BDC)도 사용자 로그인을 인증하며, 필요에 따라 PDC 의 역할을 수행하도록 수준을 올릴 수 있습니다. 도메인에는 여러 BDC 가 있을 수 있습니다.

COMPUTERS

지정된 도메인의 구성원인 컴퓨터를 나열합니다.

이 속성을 수정하려면 computer 또는 computer- 매개 변수를 chres 및 editres 명령과 함께 사용하십시오.

DOMAIN_NAME

도메인 이름을 정의합니다.

DOMAIN_USERS

(정보) 지정된 도메인의 구성원인 사용자 및 그룹 계정을 나열합니다.

PDC

(정보) 도메인에서 처음으로 작성된 컴퓨터의 이름이며 이 컴퓨터에는 도메인 데이터를 위한 주 저장소가 있습니다. 이것은 도메인 로그인을 인증하고 도메인의 디렉터리 데이터베이스를 유지관리합니다. 주 도메인 컨트롤러(PDC)는 도메인에 있는 모든 컴퓨터의 계정에 대한 변경 사항을 추적합니다. 이러한 변경 사항을 직접 수신하는 유일한 컴퓨터입니다. 도메인에는 하나의 PDC 만 있습니다.

TRUSTED

트러스트된 도메인과 트러스팅 도메인을 나열합니다.

트러스트 관계는 경유 인증을 허용하는 도메인 간의 링크이며, 이 관계에서는 트러스트하는 도메인이 트러스트된 도메인의 로그인 인증을 준수합니다. 트러스트 관계를 통해 하나의 도메인에 하나의 계정만 가진 사용자가 전체 네트워크를 액세스할 수도 있습니다. 트러스트된 도메인에 정의되어 있는 사용자 계정과 전역 그룹에게, 이런 계정이 트러스팅 도메인의 디렉터리 데이터베이스에 존재하지 않더라도, 트러스팅 도메인 권한 및 리소스 사용 권한을 부여할 수 있습니다.

이 속성을 수정하려면 trusted 또는 trusting- 매개 변수를 chres 및 editres 명령과 함께 사용하십시오. 이 명령에 대한 암호를 지정해야 합니다.

TRUSTING

트러스팅 도메인은 대상 도메인을 트러스트하는 도메인입니다.

FILE 클래스

Windows 환경에 해당

FILE 클래스의 각 레코드는 컴퓨터의 실제 드라이브 또는 논리 드라이브에 있는 파일 시스템(예: FAT, NTFS 또는 CDFS)에서 파일을 정의합니다.

참고: CA Access Control 을 사용하여 디스크에 실제로 파일을 만들 수는 없습니다.

FILE 클래스 레코드의 키는 레코드로 보호된 파일 또는 디렉터리의 이름입니다. 전체 경로를 지정해야 합니다.

다음 정의는 FILE 레코드에 포함된 속성을 설명합니다. selang 또는 웹 기반 GUI 를 사용하여 레코드의 수정 가능한 속성을 변경할 수 있습니다.

ATIME

마지막으로 파일에 액세스한 시간을 표시합니다.

ATTRIB

파일 또는 디렉터리에 대한 특성을 표시합니다. 특성은 다음 중 하나 이상이 될 수 있습니다.

- ARCHIVE
- COMPRESSED
- DIRECTORY
- HIDDEN
- NORMAL
- OFFLINE
- READONLY
- SYSTEM
- TEMPORARY

CTIME

만든 시간을 표시합니다.

DACL

리소스에 액세스할 수 있는 사용자 이름 및 그룹 이름과 각 이름에 부여된 액세스 수준을 포함하는 표준 액세스 제어 목록을 정의합니다.

이 속성을 수정할 사용자는 리소스 소유자이거나 리소스에 대한 특수 액세스(ACL 수정)를 가지고 있어야 합니다.

액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

Access Type

리소스에 대한 권한을 지정합니다.

- **Allowed(허용됨)**-리소스에 대한 특수 액세스를 허용합니다.
- **Denied(거부됨)**-리소스에 대한 특수 액세스를 거부합니다.

Accessor

액세스 권한이 허용되거나 거부된 사용자 또는 그룹입니다.

Access

접근자가 리소스에 대해 가지는 액세스 권한입니다.

참고: 빈 ACL 의 경우 명시적으로 허용된 액세스가 없으므로 액세스는 암시적으로 거부됩니다. ACL이 없는 리소스의 경우 개체에 할당된 보호가 없으므로 모든 액세스 요청이 허용됩니다.

이 속성을 수정하려면 auth 또는 auth- 명령을 사용하십시오.

DEV

파일이 있는 볼륨의 일련 번호를 표시합니다.

FILE_SYSTEM

파일이 있는 파일 시스템의 이름을 표시합니다.

GID

파일 또는 장치에 대한 그룹 정보를 표시합니다.

INDEX

파일과 연결된 고유한 식별자를 표시합니다.

ISDIR

파일이 디렉터리인지 여부를 나타냅니다.

LINKS_NUMB

파일에 대한 링크 수를 표시합니다. FAT 파일 시스템의 경우, 이 속성은 항상 1 입니다. NTFS 의 경우, 1 보다 클 수 있습니다.

MTIME

파일이 마지막으로 수정된 시간을 표시합니다.

NAME

파일 이름을 표시합니다.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

SACL

Windows 시스템 액세스 제어 목록입니다 감사 지시어를 표시합니다.

SIZE

파일 크기(바이트)를 표시합니다.

추가 정보:

Windows 파일 특성(페이지 485)
chfile 명령-Windows 파일 설정 수정(페이지 184)

GROUP 클래스

GROUP 클래스에는 Windows 운영 체제에 정의된 모든 그룹 레코드가 있습니다. GROUP 클래스에 있는 레코드는 모든 사용자의 그룹을 표시합니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있으며, 이러한 속성은 수정이 불가능합니다.

COMMENT

레코드에 포함할 추가 정보입니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

이 속성을 수정하려면 chgrp, editgrp 및 newgrp 명령과 함께 comment[-] 매개 변수를 사용합니다.

제한: 255 자.

FULL_NAME

사용자와 연결된 전체 이름입니다. CA Access Control 은 감사 로그 메시지에서 사용자를 식별하기 위해 전체 이름을 사용하지만, 권한 부여를 위해서는 사용하지 않습니다.

이 속성을 수정하려면 name 매개 변수를 chusr, editusr 또는 newusr 명령과 함께 사용하십시오.

GID

(정보) 그룹의 RID 가 있는 값입니다. RID 는 그룹이 작성될 때 계정 데이터베이스에 의해 결정됩니다. 이것은 도메인 내에서 계정 관리자에게 그룹을 고유하게 나타냅니다.

GLOBAL

전역 그룹을 나타냅니다. 이 속성은 Windows 그룹에만 적용됩니다. 이 값은 이전 CA Access Control 버전의 ISGLOBAL 속성을 대체합니다.

이 속성을 추가하려면 global 매개 변수를 newgrp 명령과 함께(만) 사용하십시오.

USERLIST

그룹에 속한 사용자 및 전역 그룹의 목록(로컬 그룹에만 해당)입니다. 이 속성에 포함된 목록은 CA Access Control 데이터베이스에 있는 목록과 다를 수도 있습니다.

이 속성을 수정하려면 join[-] 명령과 함께 username(groupname) 매개 변수를 사용하십시오.

PRIVILEGES

그룹에 할당된 Windows 권한입니다.

이 속성을 수정하려면 privileges 매개 변수를 chgrp, editgrp 또는 newgrp 명령과 함께 사용하십시오.

추가 정보:

chgrp 명령-Windows 그룹 수정(페이지 185)
Windows 권한(페이지 489)

OU 클래스

OU(Organizational Unit) 클래스에는 사용자, 그룹 또는 컴퓨터와 같은 개체가 있습니다. OU 클래스의 개체는 주 도메인 컨트롤러에서 작성될 수 있고, 그룹과 같은 다른 개체를 자식 개체로 가질 수 있습니다. 따라서 OU 클래스의 개체는 컨테이너 개체입니다.

참고: OU 클래스는 Active Directory 가 설치된 Windows 2000 Advanced Server 에서만 사용 가능합니다.

OU 클래스에는 다른 클래스에 있는 속성과 같은 미리 정의된 속성이 없습니다. 그러나 다음 OU 속성을 업데이트할 수 있습니다.

- 국가/지역
- 설명
- 데스크톱
- 시/군/구
- 표시 이름
- 폴더(읽기 전용 속성)
- 팩스 번호
- 관리 개체 (읽기 전용 속성)
- 소속 그룹(읽기 전용 속성)
- 이름(읽기 전용 속성)
- 우편 주소
- 우편 번호
- 사서함
- 시/도

- 상세 주소
- 전화 번호
- 개체 변경(읽기 전용 속성)
- 개체 작성(읽기 전용 속성)
- 웹 페이지

PRINTER 클래스

PRINTER 클래스의 각 레코드는 Windows 컴퓨터 시스템에 연결된 장치 중 Printers 폴더에 나열된 매체의 시각적 이미지를 다시 출력할 수 있는 장치를 정의합니다.

참고: CA Access Control 을 사용하여 PRINTER 클래스의 새 개체를 작성할 수 없습니다.

PRINTER 클래스 레코드의 키는 로컬 프린터의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

DACL

리소스에 액세스할 수 있는 사용자 이름 및 그룹 이름과 각 이름에 부여된 액세스 수준을 포함하는 표준 액세스 제어 목록을 정의합니다.

이 속성을 수정할 사용자는 리소스 소유자이거나 리소스에 대한 특수 액세스(ACL 수정)를 가지고 있어야 합니다.

액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

Access Type

리소스에 대한 권한을 지정합니다.

- **Allowed(허용됨)**-리소스에 대한 특수 액세스를 허용합니다.
- **Denied(거부됨)**-리소스에 대한 특수 액세스를 거부합니다.

Accessor

액세스 권한이 허용되거나 거부된 사용자 또는 그룹입니다.

Access

접근자가 리소스에 대해 가지는 액세스 권한입니다.

참고: 빈 ACL 의 경우 명시적으로 허용된 액세스가 없으므로 액세스는 암시적으로 거부됩니다. ACL이 없는 리소스의 경우 개체에 할당된 보호가 없으므로 모든 액세스 요청이 허용됩니다.

이 속성을 수정하려면 auth 또는 auth- 명령을 사용하십시오.

COMMENT

레코드에 포함할 추가 정보를 정의합니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

제한: 255 자.

LOCATION

프린터 위치를 나타내는 문자열입니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

이 속성을 수정하려면 location 매개 변수를 chres 또는 editres 명령과 함께 사용하십시오. 이 속성을 삭제하려면 ()를 공백과 함께 사용하십시오.

OWNER

레코드를 소유하는 사용자 또는 그룹을 정의합니다.

SHARE

프린터에 대한 공유 지점을 나타내는 이름입니다. 프린터를 액세스하려는 사용자 또는 그룹은 이 공유 이름을 사용할 수 있습니다.

이 속성을 수정하려면 share_name 또는 share_name- 매개 변수를 chres 또는 editres 명령과 함께 사용하십시오.

NAME

프린터 이름입니다.

SACL

Windows 시스템 액세스 제어 목록입니다 감사 지시어를 표시합니다.

SERVER

(정보) 프린터를 제어하는 서버를 식별하는 문자열입니다. 이런 속성이 없을 경우, 프린터는 로컬로 제어됩니다.

PROCESS 클래스

PROCESS 클래스의 각 레코드는 Windows 작업 관리자 프로그램에 나열된 실행 프로그램, 가상 메모리 주소 집합 및 스레드로 구성된 개체를 정의합니다.

참고: CA Access Control 을 사용하여 PROCESS 클래스의 새 개체를 만들 수 없습니다.

PROCESS 클래스 레코드의 키는 실행 프로그램의 실행 가능한 모듈 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 이 클래스에는 수정할 수 있는 속성이 없습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

IMAGE_PATH

(정보) 지정된 실행 모듈의 완전한 경로입니다.

PROCESS_ID

(정보) 프로세스의 고유 식별자입니다. 프로세스 ID 번호는 재사용되므로 해당 프로세스의 수명이 다될 때까지만 프로세스를 식별합니다.

REGKEY 클래스

REGKEY 클래스의 각 레코드는 Windows 레지스트리의 키를 정의합니다.

REGKEY 레코드에 대한 키는 Windows 레지스트리 키의 전체 레지스트리 경로입니다.

참고: 경로의 일부로 와일드카드 문자를 사용할 수 있습니다.

다음 정의는 REGKEY 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

DACL

리소스에 액세스할 수 있는 사용자 이름 및 그룹 이름과 각 이름에 부여된 액세스 수준을 포함하는 표준 액세스 제어 목록입니다.

이 속성을 수정할 사용자는 리소스 소유자이거나 리소스에 대한 특수 액세스(ACL 수정)를 가지고 있어야 합니다.

액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

Access Type

리소스에 대한 권한을 지정합니다.

- **Allowed(허용됨)**-리소스에 대한 특수 액세스를 허용합니다.
- **Denied(거부됨)**-리소스에 대한 특수 액세스를 거부합니다.

Accessor

액세스 권한이 허용되거나 거부된 사용자 또는 그룹의 이름입니다.

Access

접근자가 리소스에 대해 가지는 액세스 권한입니다. REGKEY 클래스에 대한 올바른 액세스 권한은 다음과 같습니다.

- **all**-접근자가 클래스에 대해 수행할 수 있는 모든 작업을 수행하도록 허용하거나 거부합니다.
- **append/create/subkey**-접근자가 레지스트리 키의 하위 키를 작성 또는 수정하는 것을 허용하거나 거부합니다.
- **changeperm/sec/dac/writedac/perm**-접근자가 리소스의 ACL 을 수정(접근자 추가 또는 제거)하는 것을 허용하거나 거부합니다.
- **chown/owner/takeownership**-접근자가 리소스의 소유자를 변경하는 것을 허용하거나 거부합니다.
- **delete**-접근자가 리소스를 삭제하는 것을 허용하거나 거부합니다.
- **enum**-접근자가 레지스트리 키의 하위 키를 열거하는 것을 허용하거나 거부합니다.
- **link**-접근자가 레지스트리 키와 연결되는 링크를 만드는 것을 허용하거나 거부합니다.
- **notify**-접근자가 레지스트리 키나 레지스트리 키의 하위 키에 대한 변경 통지를 요청하는 것을 허용하거나 거부합니다.
- **query**-접근자가 레지스트리 키의 값을 쿼리하는 것을 허용하거나 거부합니다.
- **read**-접근자가 키의 내용을 읽는 것을 허용하거나 거부하지만, 변경 사항이 저장되는 것을 방지합니다.
- **readcontrol/manage**-접근자가 레지스트리 키의 보안 설명자에 있는 정보(시스템(감사) ACL 에 있는 정보 제외)를 읽는 것을 허용하거나 거부합니다.
- **set**-접근자가 레지스트리 키의 값을 작성 또는 설정하는 것을 허용하거나 거부합니다.

- **write**-접근자가 레지스트리 키와 그 하위 키를 변경하는 것을 허용하거나 거부합니다.

참고: 빈 ACL(입력된 값이 없는 ACL)과 ACL 이 없는 리소스의 차이에 특히 주의해야 합니다. 빈 ACL 의 경우 명시적으로 부여된 액세스가 없으므로 액세스는 암시적으로 거부됩니다. ACL 이 없는 리소스의 경우 개체에 할당된 보호가 없으므로 모든 액세스 요청이 허용됩니다.

이 속성을 수정하려면 auth 또는 auth- 명령을 사용하십시오.

OWNER

리소스의 소유자로 지정된 사용자 또는 그룹입니다.

이 속성을 수정하려면 owner 매개 변수를 newres, chres 및 editres 명령과 함께 사용하십시오.

SACL

Windows 시스템 액세스 제어 목록은 감사 지시어를 지정합니다.

SUBKEYS

(정보) 키 아래에 있는 레지스트리 키(하위 키)의 목록입니다.

SUBVALUES

(정보) 현재 레지스트리 키에 설명된 레지스트리 값의 목록입니다.

REGVAL 클래스

REGVAL 클래스의 각 레코드는 레지스트리 키를 설명하는 데이터를 정의합니다. 이 데이터는 하나 이상의 사용자, 응용 프로그램 및 하드웨어 장치를 위해 시스템을 구성하는 데 필요한 정보를 저장합니다. 레지스트리 값에는 작업 중 계속 참조되는 정보가 들어 있습니다. 예를 들어 다음 항목이 포함됩니다.

- 각 사용자에 대한 프로필
- 컴퓨터에 설치된 응용 프로그램과 각 응용 프로그램에서 작성할 수 있는 파일 유형
- 폴더 및 응용 프로그램 아이콘에 대한 속성 시트 설정
- 하드웨어 구성
- 사용된 포트

REGVAL 레코드에 대한 키는 전체 레지스트리 키 이름과 값입니다.

참고: 레지스트리 키와 그 값을 잘못 변경하거나 삭제하면 이 문제를 해결하기 위해 Windows를 재설치해야 하는 심각한 시스템 문제가 발생할 수 있습니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

TYPE

데이터를 저장하는 형식입니다. 레지스트리 값 아래에 데이터를 저장할 때 다음 값 중 하나를 지정하여 저장하는 데이터 유형을 나타낼 수 있습니다.

참고: 레지스트리 값을 작성하거나 수정할 때 유형을 지정하십시오.

DWORD

4 바이트 길이의 숫자로 표시되는 데이터입니다. 장치 드라이버와 서비스에 대한 많은 매개 변수는 이 유형이며 이진수, 16 진수 또는 10 진수 형식으로 표시할 수 있습니다.

STRING

읽을 수 있는 텍스트를 표시하는 문자 시퀀스입니다.

MULTISTRING

복수 문자열입니다. 읽을 수 있는 텍스트 목록 또는 복수 값이 포함된 값입니다. 항목은 null 문자에 의해 구분됩니다.

BINARY

원시 이진 데이터입니다. 대부분의 하드웨어 구성 요소 정보는 바이너리 데이터로 저장되며, 16 진수 형식이나 읽기 쉬운 형식으로 표시될 수 있습니다.

이 속성을 수정하려면 위에서 설명한 유형 중 하나를 매개 변수로 newres, chres 또는 editres 와 함께 사용하십시오.

VALUE

Windows 레지스트리 값이 보유하는 값입니다.

SEOS 클래스

SEOS 클래스는 기본 로컬 보안 시스템의 동작을 관리합니다.

이 클래스는 일반 기본 보안 옵션을 지정하는, SEOS 라고 하는 하나의 레코드만 포함합니다. SEOS 클래스 속성의 상태를 보거나 변경하려면 setoptions 명령을 사용하십시오.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

AuditCategory

권한이 부여된 검색된 이벤트와 권한이 부여되지 않은 검색된 이벤트 중 어떤 것을 감사할지를 지정합니다.

AccountLogon

계정의 유효성 확인을 위해 이 컴퓨터를 사용하는 다른 컴퓨터에서 로그온 또는 로그아웃하는 사용자의 각 인스턴스를 감사할지 여부를 지정합니다.

AccountManagement

컴퓨터의 각 계정 관리 이벤트를 감사할지 여부를 지정합니다. 계정 관리 이벤트의 예는 다음과 같습니다.

- 사용자 계정 또는 그룹을 만들거나 변경하거나 삭제합니다.
- 사용자 계정의 이름을 바꾸거나 비활성화 또는 활성화합니다.
- 암호를 설정하거나 변경합니다.

DirectoryAccess

정의된 고유의 SACL(시스템 액세스 제어 목록)을 갖는 Active Directory 개체에 액세스하는 사용자 이벤트를 감사할지 여부를 지정합니다.

Logon

컴퓨터에서 로그온 또는 로그아웃하는 사용자의 각 인스턴스를 감사할지 여부를 지정합니다.

ObjectAccess

개체에 액세스하는 사용자의 이벤트를 감사할지 여부를 지정합니다. 이러한 개체에는 정의된 고유의 SACL(시스템 액세스 제어 목록)을 갖는 파일, 폴더, 레지스트리 키, 프린터 등이 있습니다.

PolicyChange

사용자 권한 할당 정책, 감사 정책 또는 트러스트 정책에 대한 모든 변경 사항을 감사할지 여부를 지정합니다.

PrivilegeUse

사용자 권한을 행사하는 사용자의 각 인스턴스를 감사할지 여부를 지정합니다.

DetailedTracking

프로그램 활성화, 프로세스 종료, 핸들 중복, 간접 개체 액세스 등의 이벤트에 대한 세부 추적 정보를 감사할지 여부를 지정합니다.

System

사용자가 컴퓨터를 다시 시작하거나 종료할 때 또는 시스템 보안이나 시스템 로그에 영향을 주는 이벤트가 발생할 때 감사를 수행할지 여부를 지정합니다.

기록

이전 암호를 다시 사용할 수 있기 전에 사용자 계정과 연결해야 할 고유한 새 암호의 수를 정의합니다.

제한: 1 과 24 사이의 정수. 0 을 지정하면 암호가 하나도 저장되지 않습니다.

간격

시스템이 사용자에게 변경을 요구하기 전에 암호를 사용할 수 있는 기간(일)을 정의합니다.

최소 수명

사용자가 암호를 변경할 수 있기 전에 암호를 사용해야 하는 기간(일)을 정의합니다.

최소 길이

사용자 계정의 암호에 포함할 수 있는 최소 문자 수를 정의합니다.

암호 실패

사용자 계정을 잠그게 되는 실패 로그온 시도 횟수를 정의합니다.

다음 시간 이후에 횟수 다시 계산

실패한 로그온 시도 이후, 실패한 로그온 시도 카운터에서 잘못된 로그온 시도가 0 으로 다시 설정되기 전에 경과해야 하는 시간을 분으로 정의합니다.

SERVICE 클래스

SERVICE 클래스의 각 레코드는 Windows 제어판의 서비스에 나열된 Windows 서비스를 정의합니다.

SERVICE 클래스 레코드의 키는 제어 중인 서비스의 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

ACCOUNT

서비스에 대해 사용하는 로그인 계정을 변경합니다. 대부분의 서비스는 시스템 계정에 로그인해야 하지만 일부 서비스는 특수 사용자 계정에 로그인하도록 구성할 수 있습니다. 자세한 내용은 관련 Microsoft Windows 설명서를 참조하십시오. 기본값은 LocalSystem 입니다.

이 속성을 수정하려면 account 매개 변수를 chres 또는 editres 명령과 함께 사용하십시오.

BINARY_NAME

서비스 실행 파일의 위치를 가리키는 전체 경로입니다.

IMAGEPATH

지정된 실행 모듈의 완전한 경로입니다.

INTERACTIVE

서비스가 시작되었을 때 로그인한 사용자가 사용할 수 있는 데스크톱 사용자 인터페이스를 제공합니다. 이 기능은 서비스가 LocalSystem 계정으로 실행 중인 경우에만 사용할 수 있습니다.

이 속성을 수정하려면 interactive 매개 변수를 chres 또는 editres 명령과 함께 사용하십시오.

PROFILE

사용자 프로필의 경로를 지정하는 문자열입니다. 이 문자열은 로컬 절대 경로 또는 UNC 경로를 포함할 수 있습니다.

이 속성을 수정하려면 profile 매개 변수를 chusr, editusr 또는 newusr 명령과 함께 사용하십시오.

REG_KEY

이 속성은 Windows 레지스트리에 있는 서비스 정의의 위치를 가리킵니다.

STARTUPTYPE

서비스가 (언제) 어떻게 시작되는지 정의합니다. 옵션은

- **automatic**-시스템이 시작되는 동안 자동으로 시작됩니다.
- **disabled**-사용자 또는 종속 서비스가 서비스를 시작하지 못하게 합니다.
- **manual**-사용자 또는 종속 서비스에 의해 서비스가 시작되는 것을 허용합니다.
- 이 속성을 수정하려면 chres 또는 editres 명령과 함께 startuptype 매개 변수를 사용하십시오.

STATUS

현재 서비스 상태를 변경합니다. 옵션에는 started, stopped 및 paused 가 있습니다.

이 속성을 수정하려면 status 매개 변수를 chres 또는 editres 명령과 함께 사용하십시오.

예: 서비스가 수동으로 시작되도록 구성

SeOSAgent 서비스가 수동으로 시작되도록 변경하려면 다음 selang 명령을 입력하십시오.

```
chres SERVICE "SeosAgent" starttype(manual)
```

예: 디렉터리 로그인 계정 변경

abcde 라는 암호와 함께 Directory Replicator 의 로그인 계정을 ReplAdmin 으로 변경하려면 다음 selang 명령을 입력하십시오.

```
chres SERVICE directory replicator account(repladmin) domainpwd(abcde)
```

SESSION 클래스

SESSION 클래스의 각 레코드는 로컬 호스트에 있는 사용자 세션을 정의합니다. 레코드에는 사용자 이름, 컴퓨터 이름, 연결 경과 시간 및 사용 중인 리소스가 있습니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

CNAME

세션이 설정된 호스트 이름입니다.

GUEST

게스트 계정에 대해 세션이 만들어졌는지 여부를 나타냅니다.

IDLE

서버와 워크스테이션 간의 네트워크 세션을 종료합니다.

이 속성을 수정하려면 disconnect 매개 변수를 chres 또는 editres 명령과 함께 사용하십시오.

OPENS

열린 세션 수를 나타냅니다.

RESOURCES

서버에 있는 공유 파일에 대한 정보를 제공하는 속성입니다. 이러한 정보에는 열린 공유 리소스 경로와 리소스를 연 사용자 또는 컴퓨터가 있습니다.

TIME

세션이 설정된 이후 경과한 시간입니다.

USER

사용자의 RID(Relative ID)가 있는 값입니다. RID 는 사용자가 작성될 때 SAM(보안 계정 관리자)에 의해 결정됩니다. RID 는 도메인 내 SAM 에 대한 사용자 계정을 고유하게 정의합니다.

예: 로컬 세션에서 사용자 연결 끊기

사용자 ZORRO 와 로컬 호스트 세션과의 연결을 끊으려면 다음 selang 명령을 입력하십시오.

```
chres SESSION zorro disconnect
```

참고: 사용자의 연결을 끊으면 데이터가 손실될 수도 있으므로, 연결을 끊기 전에 해당 사용자에게 알리는 것이 좋습니다.

SHARE 클래스

SHARE 클래스의 각 레코드는 하나 이상의 장치나 프로그램에서 사용하는 모든 장치, 데이터 또는 프로그램 등의 공유 리소스를 정의합니다. Windows 의 경우, 공유 리소스란 디렉터리, 파일, 프린터 및 명명된 파이프와 같이 네트워크 사용자가 사용할 수 있는 모든 리소스를 나타냅니다. 공유는 서버에 있는 리소스 중 네트워트 사용자가 사용할 수 있는 리소스도 나타냅니다.

SHARE 클래스 레코드의 키는 리소스의 공유 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

CURR_USERS

(정보) 리소스에 현재 연결된 사용자 수입니다.

DACL

리소스에 액세스할 수 있는 사용자 이름 및 그룹 이름과 각 이름에 부여된 액세스 수준을 포함하는 표준 액세스 제어 목록을 정의합니다.

이 속성을 수정할 사용자는 리소스 소유자이거나 리소스에 대한 특수 액세스(ACL 수정)를 가지고 있어야 합니다.

액세스 제어 목록에 있는 각 요소는 다음 정보를 포함합니다.

Access Type

리소스에 대한 권한을 지정합니다.

- **Allowed(허용됨)**-리소스에 대한 특수 액세스를 허용합니다.
- **Denied(거부됨)**-리소스에 대한 특수 액세스를 거부합니다.

Accessor

액세스 권한이 허용되거나 거부된 사용자 또는 그룹입니다.

Access

접근자가 리소스에 대해 가지는 액세스 권한입니다.

참고: 빈 ACL 의 경우 명시적으로 허용된 액세스가 없으므로 액세스는 암시적으로 거부됩니다. ACL이 없는 리소스의 경우 개체에 할당된 보호가 없으므로 모든 액세스 요청이 허용됩니다.

이 속성을 수정하려면 auth 또는 auth- 명령을 사용하십시오.

MAX_USERS

공유 리소스가 수용할 수 있는 동시 연결의 최대 수입니다.

참고: 이 속성 값으로 0 을 지정할 수 없습니다. Windows 는 이 값을 무시합니다.

이 속성을 수정하려면 max_users 매개 변수를 newres, chres 또는 editres 명령과 함께 사용하십시오.

NAME

공유의 이름을 정의합니다.

PATH

공유 리소스에 대한 로컬 경로를 지정하는 문자열입니다. 디스크의 경우 공유되는 경로이고, 인쇄 큐의 경우 공유되는 인쇄 큐의 이름입니다.

이 속성을 수정하려면 path 매개 변수를 newres, chres 또는 editres 명령과 함께 사용하십시오.

PERMISSION

(정보) 공유 수준 보안으로 실행되는 서버에 대한 공유 리소스의 권한을 나타내는 값입니다. 이 속성은 다음 표에 있는 값 중 하나가 될 수 있습니다.

ACCESS_READ

리소스에서 데이터를 읽은 후 리소스를 실행할 수 있는 권한(기본값)입니다.

ACCESS_WRITE

리소스에 데이터를 쓸 수 있는 권한입니다.

ACCESS_CREATE

리소스의 인스턴스(예: 파일)를 작성할 수 있는 권한입니다. 데이터는 리소스가 작성될 때 리소스에 기록될 수 있습니다.

ACCESS_EXEC

리소스를 실행할 수 있는 권한입니다.

ACCESS_DELETE

리소스를 삭제할 수 있는 권한입니다.

ACCESS_ATRIB

리소스 특성(예: 파일이 마지막으로 수정된 날짜 및 시간)을 수정할 수 있는 권한입니다.

ACCESS_PERM

리소스에 대해 사용자 또는 응용 프로그램에 할당된 권한(읽기, 쓰기, 작성, 실행 및 삭제)을 수정할 수 있는 권한입니다.

ACCESS_ALL

리소스에 대한 읽기, 쓰기, 작성, 실행 및 삭제 권한 및 해당 특성 및 권한을 수정할 수 있는 권한입니다.

ACCESS_NONE

사용 권한을 거부합니다.

REMARK

레코드에 포함할 추가 정보입니다. 이 문자열에는 최대 255자의 영숫자를 사용할 수 있습니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

이 속성을 수정하려면 comment 또는 comment- 매개 변수를 newres, chres 또는 editres 명령과 함께 사용하십시오.

RESOURCES

(정보) 서버에 있는 공유 파일에 대한 정보를 제공하는 속성입니다. 이러한 정보에는 열린 공유 리소스 경로와 리소스를 연 사용자 또는 컴퓨터가 있습니다.

TYPE

(정보) 공유 유형입니다. 공유된 리소스에 대해 다음 유형 중 하나를 사용하십시오.

File Folder

디스크 드라이브. 서버의 원격 관리(ADMIN$)와 C$, D$ 등과 같은 관리 공유를 참조할 수도 있습니다.

Print Queue

인쇄 큐

Communication device

통신 장치

Interprocess Communication (IPC)

프로세스 간 통신(IPC$)을 위해 예약된 특수 공유

USERS

현재 공유 리소스를 액세스하는 사용자에 대한 정보입니다. 이 정보에는 연결한 사용자의 이름(USER), 서버의 공유 리소스의 공유 이름 또는 클라이언트의 컴퓨터 이름(MACHINE)이 포함되어 있습니다. 또한 사용자가 연결된 초 단위의 시간(TIME)과 연결 결과로 현재 열려 있는 파일의 수(INUSE)도 포함됩니다.

USER 클래스

USER 클래스에는 Windows 운영 체제에 정의된 모든 사용자 레코드가 있습니다. USER 레코드의 키는 사용자의 이름, 즉 시스템에 로그인할 때 사용자가 입력하는 이름입니다.

다음 정의는 이 클래스 레코드에 포함된 속성을 설명합니다. 대부분의 속성은 수정 가능하며 selang 또는 관리 인터페이스를 사용하여 조작할 수 있습니다. 수정할 수 없는 속성은 정보로 표시되어 있습니다.

BAD_PW_COUNT

(정보) 사용자가 잘못된 암호를 사용하여 계정에 로그인하려고 시도한 횟수입니다. 값이 -1 이면 알 수 없는 값임을 의미합니다.

COMMENT

레코드에 포함할 추가 정보입니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

이 속성을 수정하려면 chusr, editusr 및 newusr 명령과 함께 comment[-] 매개 변수를 사용합니다.

제한: 255 자.

COUNTRY

사용자에 대한 국가 설명자를 지정하는 문자열입니다. 이 문자열은 X.500 명명 체계의 일부입니다. CA Access Control 은 권한 부여를 위해 이 문자열을 사용하지 않습니다.

이 속성을 수정하려면 country 매개 변수를 chusr, editusr 및 newusr 명령과 함께 사용하십시오.

DAYTIME

사용자가 리소스를 액세스할 수 있는 시기를 제어하는 요일 및 시간 제한입니다.

이 속성을 수정하려면 chusr, editusr 및 newusr 명령과 함께 restrictions 매개 변수를 사용하십시오.

참고: 분으로 입력한 값이 잘린다는 점을 제외하면 이 속성의 정보는 AC 환경의 DAYTIME 속성과 동일합니다.

DIAL_CALLBACK

사용자에게 제공되는 콜백 권한 유형입니다. 다음 옵션이 정의됩니다.

NoCallBack

사용자는 콜백 권한을 갖지 못합니다.

SetByCaller

원격 사용자는 전화를 걸 때 콜백 전화 번호를 지정할 수 있습니다.

Call-back Phone Number

관리자가 콜백 번호를 설정합니다.

이 속성을 수정하려면 gen_prop 또는 gen_val 매개 변수를 chusr 또는 editusr 명령과 함께 사용하십시오.

DIAL_PERMISSION

RAS 서버에 전화를 걸 수 있는 권한입니다. 값을 0 으로 지정하면 사용자는 RAS 서버에 전화를 걸 수 없습니다.

이 속성을 수정하려면 gen_prop 또는 gen_val 매개 변수를 chusr 또는 editusr 명령과 함께 사용하십시오.

EXPIRE_DATE

USER 레코드가 만료되어 사용할 수 없게 되는 날짜입니다. USER 레코드의 EXPIRE_DATE 속성 값이 GROUP 레코드의 값보다 우선합니다. 만료된 레코드를 다시 복원하려면 chusr 명령과 함께 expire- 매개 변수를 사용합니다. 만료된 사용자를 계속 진행할 수 없습니다. 계속 날짜를 지정하여 일시 중지된 사용자를 계속 진행할 수 있습니다.

이 속성을 수정하려면 expire 또는 expire- 매개 변수를 chusr, editusr 또는 newusr 명령과 함께 사용하십시오.

FLAGS

특별한 특성을 지정하기 위해 사용자 계정에 할당할 수 있는 플래그입니다. 둘 이상의 플래그를 각 계정에 적용할 수 있습니다.

이 속성을 수정하려면 flags 매개 변수를 chusr, editusr 및 newusr 명령과 함께 사용하십시오.

FULL_NAME

사용자와 연결된 전체 이름입니다. CA Access Control 은 감사 로그 메시지에서 사용자를 식별하기 위해 전체 이름을 사용하지만, 권한 부여를 위해서는 사용하지 않습니다.

이 속성을 수정하려면 name 매개 변수를 chusr, editusr 또는 newusr 명령과 함께 사용하십시오.

GID

그룹의 RID 가 있는 값입니다. RID 는 그룹이 작성될 때 계정 데이터베이스에 의해 결정됩니다. 이것은 도메인 내에서 계정 관리자에게 그룹을 고유하게 나타냅니다.

GROUPS

사용자가 속해 있는 그룹 목록입니다. 이 속성에 포함된 그룹 목록은 AC 환경 GROUPS 속성에 있는 목록과 다를 수 있습니다.

이 속성을 수정하려면 join[-] 명령과 함께 그룹 매개 변수를 사용합니다.

HOME

홈 디렉터리는 사용자가 액세스할 수 있는 폴더로서 해당 사용자에 대한 파일과 프로그램을 포함합니다. 홈 디렉터리를 개별 사용자에게 할당할 수도 있고 여러 사용자가 공유할 수도 있습니다.

HOMEDIR

사용자의 홈 디렉터리를 지정하는 문자열입니다. 사용자가 자신의 홈 디렉터리로 자동 로그인합니다.

이 속성을 수정하려면 homedir 매개 변수를 chusr, editusr 또는 newusr 명령과 함께 사용하십시오.

HOME_DRIVE

사용자 홈 디렉터리의 드라이브를 지정하는 문자열입니다. 사용자는 자신의 홈 드라이브와 홈 디렉터리에 자동으로 로그인합니다.

이 속성을 수정하려면 homedrive 매개 변수를 chusr, editusr 또는 newusr 명령과 함께 사용하십시오.

ID

사용자의 RID(Relative ID)가 있는 값입니다. RID 는 사용자가 작성될 때 SAM(보안 계정 관리자)에 의해 결정됩니다. RID 는 도메인 내 SAM 에 대한 사용자 계정을 고유하게 정의합니다.

LAST_ACC_TIME

(정보) 최종 로그인의 날짜 및 시간입니다.

LAST_LOGOFF

(정보) 최종 로그오프의 날짜 및 시간입니다.

LOCATION

사용자 위치를 저장하는 데 사용되는 문자열입니다. CA Access Control 은 권한 부여를 위해 이 정보를 사용하지 않습니다.

이 속성을 수정하려면 location 매개 변수를 chusr, editusr 및 newusr 명령과 함께 사용하십시오.

LOGON_SERVER

사용자에 대한 로그인 정보를 확인하는 서버를 지정하는 문자열입니다. 사용자가 도메인 워크스테이션에 로그인하면 CA Access Control 은 로그인 정보를 서버로 전송하며, 이 서버는 사용자가 작업할 수 있도록 워크스테이션 권한을 부여합니다.

MAX_LOGINS

(정보) 사용자가 계정에 성공적으로 로그인한 횟수입니다. 값이 -1 이면 알 수 없는 값임을 의미합니다.

NAME

사용자 이름입니다.

ORGANIZATION

사용자가 근무하는 조직에 대한 정보를 저장하는 문자열입니다. 이 문자열은 X.500 명명 체계의 일부입니다. CA Access Control 은 권한 부여를 위해 이 문자열을 사용하지 않습니다.

이 속성을 수정하려면 organization 매개 변수를 chusr, editusr 및 newusr 명령과 함께 사용하십시오.

ORG_UNIT

사용자가 근무하는 조직 단위에 대한 정보를 저장하는 문자열입니다. 이 문자열은 X.500 명명 체계의 일부입니다. CA Access Control 은 권한 부여를 위해 이 문자열을 사용하지 않습니다.

이 속성을 수정하려면 org_unit 매개 변수를 chusr, editusr 및 newusr 명령과 함께 사용하십시오.

PASSWD_EXPIRED

사용자 계정의 만료 날짜입니다.

PGROUP

사용자의 주 그룹 ID 입니다. 주 그룹은 사용자가 정의되어 있는 그룹 중 하나입니다. 주 그룹은 전역 그룹이어야 합니다. 이 문자열에는 공백이나 쉼표가 들어갈 수 없습니다.

이 속성을 수정하려면 pgroup 매개 변수를 chusr, editusr 또는 newusr 명령과 함께 사용하십시오.

PHONE

사용자 전화 번호를 저장하기 위해 사용할 수 있는 문자열입니다. 이 정보는 권한 부여에 사용되지 않습니다.

이 속성을 수정하려면 phone 매개 변수를 chusr, editusr 및 newusr 명령과 함께 사용하십시오.

PRIVILEGES

사용자에게 할당된 Windows 권한입니다.

이 속성을 수정하려면 privileges 매개 변수를 chusr, editusr 또는 newusr 명령과 함께 사용하십시오.

PROFILE

사용자의 프로필에 대한 경로를 지정하는 문자열입니다. 이 문자열은 로컬 절대 경로 또는 UNC 경로를 포함할 수 있습니다.

이 속성을 수정하려면 profile 매개 변수를 chusr, editusr 또는 newusr 명령과 함께 사용하십시오.

PW_LAST_CHANGE

(정보) 암호가 업데이트된 날짜 및 시간입니다.

RESUME_DATE

일시 중지된 USER 계정이 사용 가능해지는 날짜입니다.

RESUME_DATE 와 SUSPEND_DATE 가 함께 작동하는 방법에 대한 설명은 SUSPEND_DATE 를 참조하십시오.

SCRIPT

사용자의 로그온 스크립트 파일의 경로를 지정하는 문자열입니다. 스크립트 파일은 .CMD, .EXE 또는 .BAT 파일이 될 수 있습니다.

TERMINALS

사용자가 로그인하기 위해 사용할 수 있는 터미널 목록을 지정하는 문자열입니다.

이 속성을 수정하려면 terminals 매개 변수를 chusr, editusr 및 newusr 명령과 함께 사용하십시오.

TS_CONFIG_PGM

클라이언트가 초기 프로그램을 지정할 수 있는지 여부를 나타내는 값입니다.

TS_INITIAL_PGM 사용자 속성은 초기 프로그램을 나타냅니다. 사용자의 초기 프로그램을 지정할 경우, 이것은 사용자가 실행할 수 있는 유일한 프로그램이 됩니다. 사용자가 해당 프로그램을 종료할 때 터미널 서버는 사용자를 로그오프합니다.

이 값을 1 로 설정하면 클라이언트가 초기 프로그램을 지정할 수 있습니다. 이 값을 0 으로 설정하면 클라이언트가 초기 프로그램을 지정할 수 없습니다.

이 속성을 수정하려면 gen_prop 및 gen_val 매개 변수를 chusr 및 editusr 명령과 함께 사용하십시오.

TS_HOME_DIR

터미널 서버 로그온에 사용할 사용자 홈 디렉터리의 경로입니다. 이 문자열은 로컬 경로 또는 UNC 경로(\\machine\share\path)를 지정할 수 있습니다.

이 속성을 수정하려면 gen_prop 및 gen_val 매개 변수를 chusr 및 editusr 명령과 함께 사용하십시오.

TS_HOME_DRIVE

TS_HOME_DIR 속성에서 UNC 경로가 지정된 드라이브 사양(드라이브 문자 다음에 콜론 입력)입니다.

이 속성을 수정하려면 gen_prop 및 gen_val 매개 변수를 chusr 및 editusr 명령과 함께 사용하십시오.

TS_INITIAL_PGM

사용자가 로그온할 때 터미널 서비스가 실행하는 초기 프로그램의 경로입니다.

사용자의 초기 프로그램을 지정할 경우 사용자는 지정된 프로그램만 실행할 수 있습니다. 사용자가 해당 프로그램을 종료할 때 터미널 서버는 사용자를 로그오프합니다.

TS_CONFIG_PGM 속성을 1 로 설정하면 클라이언트는 초기 프로그램을 지정할 수 있습니다.

이 속성을 수정하려면 gen_prop 및 gen_val 매개 변수를 chusr 및 editusr 명령과 함께 사용하십시오.

TS_PROFILE_PATH

터미널 서버 로그온을 위한 사용자 프로필의 경로입니다. 경로로 식별되는 디렉터리는 수동으로 작성해야 하며 로그온 전에 존재해야 합니다.

이 속성을 수정하려면 gen_prop 및 gen_val 매개 변수를 chusr 및 editusr 명령과 함께 사용하십시오.

TS_WORKING_DIR

사용자가 로그온할 때 터미널 서비스가 실행하는 초기 프로그램의 작업 디렉터리 경로입니다.

이 속성을 수정하려면 gen_prop 및 gen_val 매개 변수를 chusr 및 editusr 명령과 함께 사용하십시오.

WORKSTATIONS

사용자가 로그인할 수 있는 워크스테이션 목록입니다.

이 속성을 수정하려면 workstations 매개 변수를 chusr, editusr 및 newusr 명령과 함께 사용하십시오.

추가 정보:

chusr 명령-Windows 사용자 수정(페이지 190)
Windows 계정 플래그(페이지 486)
Windows 권한(페이지 489)

UNIX 환경의 클래스

이 단원에는 UNIX 시스템 파일에 존재하는 모든 UNIX 클래스(unix 환경의 클래스)에 대한 완전한 참조가 알파벳순으로 포함되어 있습니다. 이러한 기본 클래스의 속성은 운영 체제에 의해 관리되며 시스템마다 다릅니다.

참고: unix 환경이란 용어는 selang 명령 env unix 로 액세스할 수 있는 시스템 파일을 가리킵니다. 이러한 파일은 UNIX 운영 체제가 사용자와 그룹에 대해 유지 관리하는 동일한 시스템 파일이며 시스템에 있는 파일입니다.

FILE 클래스

FILE 클래스의 각 레코드는 컴퓨터 파일 시스템의 실제 드라이브 또는 논리 드라이브에 있는 파일을 정의합니다.

참고: CA Access Control 을 사용하여 디스크에 실제로 파일을 만들 수는 없습니다.

FILE 클래스 레코드의 키는 레코드로 보호된 파일 또는 디렉터리의 이름입니다. 전체 경로를 지정해야 합니다.

이 기본 클래스의 속성은 운영 체제에 의해 관리되며 시스템마다 다릅니다. chfile 명령은 selang 을 사용하여 수정할 수 있는 기본 속성을 나열합니다.

GROUP 클래스

GROUP 클래스에는 UNIX 운영 체제에 정의되어 있는 모든 그룹 레코드가 포함됩니다. GROUP 클래스에 있는 레코드는 모든 사용자의 그룹을 표시합니다.

이 기본 클래스의 속성은 운영 체제에 의해 관리되며 시스템마다 다릅니다. chgrp 명령은 selang 을 사용하여 수정할 수 있는 기본 속성을 나열합니다.

USER 클래스

USER 클래스에는 UNIX 운영 체제에 정의되어 있는 모든 사용자 레코드가 포함됩니다. USER 레코드의 키는 사용자의 이름, 즉 시스템에 로그인할 때 사용자가 입력하는 이름입니다.

이 기본 클래스의 속성은 운영 체제에 의해 관리되며 시스템마다 다릅니다. chusr 명령은 selang 을 사용하여 수정할 수 있는 기본 속성을 나열합니다.

사용자 지정을 위한 클래스

이 단원에는 참조 사용자 정의 클래스 및 속성이 포함되어 있습니다.

사용자 정의 클래스

사용자 정의 클래스의 각 레코드는 사용자 자신의 필요에 맞는 맞춤형 클래스의 접근 권한을 정의합니다. 사용자 정의 클래스 이름에 대한 유일한 제한 사항은 이름이 모두 대문자이면 안된다는 것입니다.

사용자 정의 클래스 레코드의 키는 레코드의 이름입니다.

Unicenter TNG 사용자 정의 클래스

CA Access Control 에서는 Unicenter TNG 자산 클래스를 리소스로 정의할 수 있습니다. Unicenter TNG 사용자 정의 클래스를 만들고, 삭제하고, 활성화 및 비활성화할 수 있습니다.

Unicenter TNG 사용자 정의 클래스는 UACC 클래스에 있습니다.

참고: 일반 CA Access Control 클래스에 정의된 속성은 모두 사용자 정의 클래스에 사용할 수 있습니다.

부록 A: Windows 값

이 장은 아래의 주제를 포함하고 있습니다.

Windows 파일 특성

chfile, editfile 및 newfile 명령을 사용하여 특성을 파일에 할당할 수 있습니다. 특성은 파일의 특성을 결정합니다.

참고: 이러한 파일 특성의 전체 이름은 FILE_ATTRIBUTE_name 이지만 CA Access Control 에서는 name 부분(예: ARCHIVE 또는 COMPRESSED)만 입력하면 됩니다.

다음은 Windows 에서 수정할 수 있는 파일 특성을 나열하고 설명한 것입니다.

FILE_ATTRIBUTE_ARCHIVE

보관 파일이며, 백업 또는 제거하도록 표시된 파일입니다.

FILE_ATTRIBUTE_HIDDEN

숨겨진 파일이며, 숨겨진 파일은 보통 일반적인 디렉터리 목록에 포함되지 않습니다.

FILE_ATTRIBUTE_NORMAL

다른 특성이 없는 파일이며, 이 값은 단독으로 사용할 때에만 유효합니다.

FILE_ATTRIBUTE_READONLY

읽기 전용 파일이며, 응용 프로그램이 파일을 읽을 수 있지만 파일에 쓰거나 삭제할 수는 없습니다.

FILE_ATTRIBUTE_SYSTEM

운영 체제 파일 또는 운영 체제에서만 사용되는 파일입니다.

FILE_ATTRIBUTE_TEMPORARY

임시 저장을 위해 사용되는 파일입니다.

다음은 Windows 에서 수정할 수 없는 파일 특성을 나열하고 설명한 것입니다.

FILE_ATTRIBUTE_COMPRESSED

압축된 파일 또는 디렉터리입니다. 파일의 경우 파일에 있는 모든 데이터가 압축되었음을 의미하고, 디렉터리의 경우 기본적으로 새로 작성된 파일과 하위 디렉터리가 모두 압축되었음을 의미합니다.

FILE_ATTRIBUTE_DIRECTORY

디렉터리입니다.

추가 정보:

chfile 명령-Windows 파일 설정 수정(페이지 184)

Windows 계정 플래그

계정의 특별한 특성을 지정하려면 chusr, editusr 및 newusr 명령을 사용하여 사용자 계정에 플래그를 할당할 수 있습니다. 둘 이상의 플래그를 각 계정에 적용할 수 있습니다.

참고: CA Access Control 에서는 플래그의 전체 이름을 입력할 필요가 없습니다. 사용자는 표에서 설명하는 바로 가기를 사용할 수 있습니다.

다음은 Windows 에서 사용 가능한 계정 플래그입니다.

바로 가기	플래그	설명
blank	UF_PASSWRD_NOTREQD	사용자의 계정에 암호가 필요하지 않음을 나타냅니다.
cant_change	UF_PASSWORD_CANT_CHANGE	사용자가 계정 암호를 변경할 수 없음을 나타냅니다.
disable	UF_ACCOUNTDISABLE	사용자의 계정이 비활성화되었음을 나타냅니다.
dont_expire	UF_DONT_EXPIRE_PASSWORD	계정의 암호가 절대 만료되지 않음을 나타냅니다.
homedir	UF_HOMEDIR_REQUIRED	홈 디렉터리가 필요함을 나타냅니다. Windows 에서 이 값은 무시됩니다.
interdomain	UF_INTERDOMAIN_TRUST_ACCOUNT	계정을 트러스트할 수 있는 권한을 나타냅니다.
lockout	UF_LOCKOUT	사용자의 계정이 현재 잠겨 있음을

바로 가기	플래그	설명
		나타냅니다. 잠긴 계정의 잠금을 해제하려면 이 플래그를 제거하십시오.
normal	UF_NORMAL_ACCOUNT	일반 사용자를 표시하는 기본 계정 유형을 나타냅니다.
notreq	UF_PASSWRD_NOTREQD	사용자의 계정에 암호가 필요하지 않음을 나타냅니다.
protect	UF_PASSWORD_CANT_CHANGE	사용자가 계정 암호를 변경할 수 없음을 나타냅니다.
script	UF_SCRIPT	사용자가 응용 프로그램을 시작할 때 디스크 매핑을 실행하는 로그인 스크립트가 활성화됨을 나타냅니다. 이 플래그는 LAN Manager 2.0 또는 Windows 를 사용할 경우 반드시 설정해야 합니다.
server	UF_SERVER_TRUST_ACCOUNT	도메인에 있는 Windows NT 백업 도메인 컨트롤러의 계정을 나타냅니다.
temp	UF_TEMP_DUPLICATE_ACCOUNT	다른 도메인에 계정을 갖고 있는 사용자를 나타냅니다. 이 계정의 도메인에 대한 액세스는 제공하지만 트러스트 계정의 도메인에 대한 액세스는 제공하지 않습니다.
trust	UF_INTERDOMAIN_TRUST_ACCOUNT	계정을 트러스트할 수 있는 권한을 나타냅니다.
workstation	UF_WORKSTATION_TRUST_ACCOUNT	도메인의 구성원인 워크스테이션이나 서버의 계정을 나타냅니다.

추가 정보:

chusr 명령-Windows 사용자 수정(페이지 190)

Windows Access 권한

SHARE 리소스 유형에서 액세스 권한을 접근자에게 부여할 수 있습니다.

다음은 Windows 에서 사용 가능한 액세스 권한입니다.

ACCESS_ALL

리소스에 대한 읽기, 쓰기, 작성, 실행 및 삭제 권한 및 해당 특성과 권한을 수정할 수 있는 권한입니다.

ACCESS_ATTRIB

리소스의 특성을 수정할 수 있는 권한입니다.

ACCESS_CREATE

리소스가 작성될 때 리소스에 데이터를 기록할 수 있는 권한을 포함하여 리소스를 작성할 수 있는 권한입니다.

ACCESS_DELETE

리소스를 삭제할 수 있는 권한입니다.

ACCESS_EXEC

리소스를 실행할 수 있는 권한입니다.

ACCESS_NONE

액세스 없음

ACCESS_PERM

사용자 또는 응용 프로그램에게 리소스에 대해 할당된 권한을 수정할 수 있는 권한입니다.

ACCESS_READ

리소스에서 데이터를 읽은 후 리소스를 실행할 수 있는 권한(기본값)입니다.

ACCESS_WRITE

리소스에 데이터를 쓸 수 있는 권한입니다.

추가 정보:

SHARE 클래스(페이지 474)

Windows 권한

Windows 권한은 개별 사용자 계정이나 그룹에 할당할 수 있습니다. 관리자는 chusr 또는 editusr 명령을 사용하여 사용자에게 권한을 할당하거나, chgrp 또는 editgrp 명령을 사용하여 그룹에게 권한을 할당할 수 있습니다. 그룹에 추가된 사용자는 그룹에게 할당된 모든 권한을 자동으로 얻게 됩니다.

권한 또는 사용자 권한의 이름은 목록에 나타난 이름과 동일한 이름을 사용할 수 있으며, 이름 앞에는 Se 를 추가하고 이름 끝에는 Privilege 를 추가할 수 있습니다. 그러나 BatchLogon, InteractiveLogon, NetworkLogon 및 ServiceLogon 의 경우에는 Privilege 대신 Right 를 추가합니다.

다음은 Windows 에서 사용 가능한 권한입니다.

권한	기본 할당	설명
AssignPrimaryToken	없음	사용자가 프로세스의 보안 액세스 토큰을 수정할 수 있습니다.
Audit	없음	보안 감사를 생성합니다.
Backup	관리자 백업 운영자	사용자가 파일과 디렉터리를 백업할 수 있습니다. 이 권한은 모든 파일 및 디렉터리 권한을 대체합니다.
BatchLogon	없음	사용자가 배치 작업으로 로그인할 수 있습니다.
ChangeNotify	모든 사용자	일반적으로 파일과 하위 디렉터리에 대한 권한은 하향식으로 적용됩니다. 다시 말하면, 특정 디렉터리에 대한 권한이 없는 사용자는 해당 디렉터리 아래의 하위 디렉터리를 액세스할 수 있는 권한도 가지지 않습니다. 이 권한으로 사용자는 상위 디렉터리에 대한 권한이 없더라도 하위 디렉터리를 액세스할 수 있습니다.
CreatePagefile	없음	사용자가 페이지 파일을 작성할 수 있습니다. 보안은 다음 키에 대한 사용자의 액세스에 의해 결정됩니다. \CurrentControlSet\Control\SessionManagement
CreatePermanent	없음	사용자가 \\Device 와 같은 특수 영구 개체를 작성할 수 있습니다.
CreateToken	없음	토큰 개체를 작성합니다. 로컬 보안 기관(LSA)만 이 작업을 수행할 수 있습니다. 로컬 보안 기관(LSA)은 사용자가 시스템을 액세스할 수 있는 권한을 가지고 있는지 확인합니다. 이 권한의

권한	기본 할당	설명
		사용을 감사할 수 없습니다. C2 인증의 경우 어떤 사용자에게도 할당하지 않을 것을 권장합니다.
Debug	관리자	스레드와 같은 개체 또는 프로그램을 디버그합니다. 이 권한을 감사할 수 없습니다. C2 인증의 경우 시스템 관리자를 포함한 어떤 사용자에게도 할당하지 않을 것을 권장합니다.
IncreaseBasePriority	관리자 고급 사용자	사용자가 프로세스의 실행 우선 순위를 높일 수 있습니다.
IncreaseQuota	없음	사용자가 개체 할당량을 증가시킬 수 있습니다.
InteractiveLogon	대부분의 그룹	사용자가 대화식으로 로그인할 수 있습니다.
LoadDriver	관리자	사용자가 장치 드라이버를 설치하고 제거할 수 있습니다.
LockMemory	없음	사용자가 PAGEFILE.SYS 와 같은 백업 저장 파일에 페이지가 자동으로 백업될 수 없도록 컴퓨터의 메모리에서 페이지를 잠글 수 있습니다.
MachineAccount	없음	사용자가 도메인에 새 컴퓨터를 추가할 수 있습니다.
NetworkLogon	모든 사용자	사용자가 네트워크의 모든 장소에서 컴퓨터에 연결할 수 있습니다. 이것은 사용자가 자신의 컴퓨터에 로그인하기 위해 정해진 위치나 터미널에 없어도 된다는 의미입니다.
ProfileSingleProcess	관리자 고급 사용자	사용자가 단일 프로세스의 성능을 모니터하기 위해 성능 모니터 도구를 사용할 수 있습니다.
RemoteShutdownPrivilege	관리자 고급 사용자	사용자가 Windows 시스템을 원격으로 종료할 수 있습니다.
Restore	관리자 백업 운영자	사용자가 백업 파일 및 디렉터리를 복원할 수 있습니다. 이 권한은 모든 파일 및 디렉터리 권한을 대체합니다.
Security	관리자	사용자가 감사할 리소스 액세스(예: 파일 액세스)의 유형을 지정하고 보안 로그를 보거나 삭제할 수 있습니다. **참고:** 이 권한으로는 사용자가 Microsoft 의 사용자 관리자에 있는 정책 메뉴에서 감사 명령을 사용하여 시스템 감사 정책을 설정할 수 없습니다. 관리자는 항상 보안 로그를 보고 삭제할 수 있습니다.

권한	기본 할당	설명
ServiceLogon	없음	프로세스를 서비스로 시스템에 등록할 수 있습니다.
Shutdown	관리자 백업 운영자 모든 사용자 고급 사용자 사용자	사용자가 시스템 콘솔에서 시스템을 종료할 수 있습니다.
SystemEnvironment	관리자	사용자가 시스템 환경 변수를 수정할 수 있습니다. 이렇게 하면 사용자는 자신의 워크스테이션에서 시스템 환경을 설정하고 동일한 워크스테이션에서 작업하는 다른 모든 사용자가 동일한 설정을 사용할 수 있습니다.
SystemProfile	관리자	사용자가 시스템에서 프로필링(성능 샘플링)을 수행할 수 있습니다.
SystemTime	관리자 고급 사용자	사용자가 컴퓨터 시계의 시간을 설정할 수 있습니다.
TakeOwnership	관리자	사용자가 컴퓨터의 파일, 디렉터리, 프린터 및 기타 개체의 소유자가 될 수 있습니다. 이 권한은 개체를 보호하는 모든 권한을 대체합니다.
Tcb	없음	트러스트되는 안전한 운영 체제의 일부로 프로세스를 실행할 수 있습니다. 일부 하위 시스템에 이 권한이 부여됩니다.

추가 정보:

chusr 명령-Windows 사용자 수정(페이지 190)
chgrp 명령-Windows 그룹 수정(페이지 185)