CA Access Control

selang リファレンス ガイド

12.6

サードパーティに関する通知

CONTAINS IBM(R) 32-bit Runtime Environment for AIX(TM), Java(TM) 2 Technology Edition, Version 1.4 Modules

サンプル スクリプトおよびサンプル SDK コード

CA Access Control 製品に含まれているサンプル スクリプトおよびサンプル SDK コードは、情報提供のみを目的として現状有姿のまま提供されます。これらは特定の環境で調整が必要な場合があるため、テストや検証を実行せずに実稼働システムにデプロイしないでください。

CA Technologies では、これらのサンプルに対するサポートを提供していません。また、これらのスクリプトによって引き起こされるいかなるエラーにも責任を負わないものとします。

CA Technologies 製品リファレンス

このマニュアルが参照している CA Technologies の製品は以下のとおりです。

- CA Access Control Enterprise Edition
- CA Access Control
- CA Single Sign-On（CA SSO）
- CA Top Secret®
- CA ACF2™
- CA Audit
- CA Network and Systems Management（CA NSM、旧 Unicenter NSM and Unicenter TNG）
- CA Software Delivery（旧 Unicenter Software Delivery）
- CA Service Desk（旧 Unicenter Service Desk）
- CA User Activity Reporting Module（旧 CA Enterprise Log Manager）
- CA Identity Manager

ドキュメントの表記規則

CA Access Control のドキュメントには、以下の規則があります。

形式	意味
等幅フォント	コードまたはプログラムの出力
斜体	強調または新規用語
太字	表示されているとおりに入力する必要のある要素
スラッシュ（/）	UNIX および Windows のパスの記述で使用される、プラットフォームに依存しないディレクトリの区切り文字

また、本書では、コマンド構文およびユーザ入力の説明に（等幅フォントで）以下の特殊な規則を使用します。

形式	意味
斜体	ユーザが入力する必要のある情報
角かっこ（[]）で囲まれた文字列	オプションのオペランド
中かっこ（{}）で囲まれた文字列	必須のオペランド セット
パイプ（|）で区切られた選択項目	代替オペランド（1 つ選択）を区切ります。 たとえば、以下の例は「*ユーザ名*または*グループ名*のいずれか」を意味します。 {*username*|*groupname*}
…	前の項目または項目のグループが繰り返し可能なことを示します
下線	デフォルト値
スペースに続く、行末の円記号（¥）	本書では、コマンドの記述が 1 行に収まらない場合があります。このような場合、行末の空白とそれに続く円記号（¥）は、そのコマンドが次の行に続くことを示します。 **注**：このような円記号はコピーしないでください。また、改行はコマンドに含めないようにしてください。これらの文字は、実際のコマンド構文の一部ではありません。

例：コマンドの表記規則

以下のコードは、本書でのコマンド表記規則の使用方法を示しています。

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

この例の内容

- 標準的な等幅フォントで表示されているコマンド名（ruler）は表示されているとおりに入力します。
- 斜体で表示されている *className* オプションは、クラス名（USER など）のプレースホルダです。
- 2 番目の角かっこで囲まれた部分を指定しなくても、コマンドは実行できます。この部分は、オプションのオペランドを示します。
- オプションのパラメータ（props）を使用する場合は、キーワード *all* を選択するか、またはカンマで区切られたプロパティ名を 1 つ以上指定します。

ファイル ロケーションに関する規則

CA Access Control のドキュメントには、ファイル ロケーションに関する以下の規則があります。

- *ACInstallDir* -- CA Access Control のデフォルトのインストール ディレクトリ。
  - Windows -- <インストール パス>
  - UNIX -- <インストール パス 2>
- *ACSharedDir* -- CA Access Control for UNIX で使用される、デフォルトのディレクトリ。
  - UNIX -- /opt/CA/AccessControlShared
- *ACServerInstallDir* -- CA Access Control エンタープライズ管理 のデフォルトのインストール ディレクトリ。
  - /opt/CA/AccessControlServer
- *DistServerInstallDir* -- デフォルトの配布サーバ インストール ディレクトリ。
  - /opt/CA/DistributionServer
- *JBoss_HOME* -- デフォルトの JBoss インストール ディレクトリ。
  - /opt/jboss-4.2.3.GA

CA への連絡先

テクニカル サポートの詳細については、弊社テクニカル サポートの Web サイト（http://www.ca.com/jp/support/）をご覧ください。

マニュアルの変更点

以下のドキュメントのアップデートは、本書の最新のリリース以降に行われたものです。

-

目次

第 1 章: 概要

このセクションには、以下のトピックが含まれています。

本書の内容

本書では、CA Access Control の selang コマンド、データベースのクラスとプロパティ、および Windows の値について説明します。また、エンタープライズ管理機能、レポート機能、および拡張ポリシー管理機能を備えた CA Access Control Enterprise Edition についても説明します。

用語を簡潔に示すために、本書の全体を通してこの製品を CA Access Control と呼びます。

本書の対象読者

本書は、selang コマンドの実行、CA Access Control で保護される環境のメンテナンスや設定を担当するセキュリティ管理者およびシステム管理者を対象としています。

第 2 章: selang コマンド言語

このセクションには、以下のトピックが含まれています。

CA Access Control コマンド ライン インタープリタ

CA Access Control は、CA Access Control のコマンド言語である selang というコマンド シェルを使用して管理します。selang コマンド言語を使用すると、CA Access Control データベースに定義を作成することができます。 selang コマンド言語は、コマンド定義言語です。

selang ユーティリティは、CA Access Control インストールの bin ディレクトリにあります。selang シェルに切り替えると、特別な selang プロンプトが表示されます。表示されるプロンプトの形式は、作業環境によって異なります。たとえば、以下のように表示されます。

```
AC>
```

デフォルトでは、selang コマンド シェルは、ローカル データベースに対して実行されます。別の端末上の CA Access Control データベースに対してコマンドを実行する場合は、selang コマンドを入力する前に hosts コマンドを指定します。

詳細情報:

selang 環境 (P. 37)
hosts コマンド - リモート CA Access Control 端末への接続 (P. 140)

selang ユーティリティ - CA Access Control コマンドラインの実行

selang ユーティリティは、CA Access Control データベースおよびネイティブ環境にアクセスできるコマンド シェルを起動します。このコマンド シェルから selang のコマンドを発行することで、データベースが動的に更新されます。

注: -o オプションを指定した場合を除き、コマンドの実行結果は標準出力に送信されます。

UNIX でのこのコマンドの形式は、以下のようになります。

```
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 でのこのコマンドの形式は、以下のようになります。

```
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*

実行する selang コマンドを指定します。指定したコマンドの実行後に、selang が終了します。

command に空白が含まれる場合は、文字列全体を引用符で囲みます。以下に例を示します。

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

-d *path*

定義したパスのデータベースが更新されるように指定します。

注: ローカル データベースのみを指定できます。

-f *file*

端末の標準入力からではなく、指定されたファイルから selang コマンドが読み取られるように指定します。

入力ファイルのコマンドが実行されると、実行中のコマンドの行番号が画面に表示されます。selang のプロンプトは表示されません。*file* に指定されたコマンドの実行後に、selang が終了します。

-h

このユーティリティのヘルプ画面を表示します。

--l

デフォルトのローカル データベースが更新されるように指定します。通常、このデータベースは *ACInstallDir*/seosdb です（*ACInstallDir* は CA Access Control をインストールしたディレクトリです）。

このオプションを -d または -p と同時に指定する必要はありません。

注: このオプションは selang に取って代わるものです。これは seosd が実行されていないときにのみ有効です。また、データベース ファイルを更新するための十分なネイティブ権限を持った CA Access Control 管理者のみ実行できます。

-o *file*

selang の出力が指定されたファイルに書き込まれるように指定します。selang を起動するたびに、新しい空のファイルが作成されます。既存のファイル名を指定した場合は、そのファイルの現在の情報が上書きされます。

-p *pmdb*

定義した PMDB のデータベース（PMDB サブディレクトリ内のデータベース）が更新されるように指定します。この場合、PMDB はローカル端末上に存在する必要があります。このデータベースに対する変更内容は、サブスクライバには伝達されません。

注: このオプションは、指定された PMDB 上で sepmdd または seosd のいずれかが実行されている場合は無効となります。また、*hosts* コマンドの使用とは異なります。

重要: サブスクライバへの伝達が必要な変更はこのモードで行わないでください。更新の作成時にネイティブ モードを使用すると、CA Access Control 設定オプションで定義されているように、ネイティブ ホスト ファイルのみが更新されます。

-r *file*

定義したファイルからコマンドが読み取られるように指定します。このファイルでは、標準の selang 構文で記述されたコマンドがセミコロンまたは改行記号で区切られている必要があります。*file* 内のコマンドが実行された後、ユーザに入力を促すメッセージが表示されます。

このオプションでファイルを定義しない場合は、ホーム ディレクトリの .selangrc ファイルが使用されます。

-s

selang がサイレント モードで開かれるように指定します。著作権に関するメッセージは表示されません。

-u *user pass*

（UNIX のみ）selang を実行するユーザ名およびパスワードを指定します。

このオプションを使用するには、seos.ini ファイルの check_password トークンを yes に設定する必要があります。これにより、selang -u を実行するときに、「パスワードを入力してください」というメッセージが表示されます。試行することができるログインは 3 回までです。

seos.ini ファイルの [lang] セクションにある no_check_password_users トークンには、selang へのログイン中にパスワード チェックを省略するユーザのリストが含まれます。

注: check_password トークンが no（デフォルト）に設定されている場合、パスワードの入力は要求されません。

-v

（Windows のみ）出力にコマンド ラインを書き込みます。

使用上の注意

- -h が使用されると、他のオプションはすべて無視されます。
- -c オプションを -f オプションと同時に使用することはできません。
- -d オプションを -p オプションと同時に使用することはできません。
- -d または -p を指定した場合、-l を指定する必要はありません。

詳細情報:

hosts コマンド - リモート CA Access Control 端末への接続 (P. 140)

selang コマンド シェルの機能

selang コマンド シェルに切り替えると、以下のプロンプトが表示されます。

```
AC>
```

プロンプトが表示された後に、selang コマンドを入力します。複数のコマンドを入力する場合は、セミコロン(;)で区切ります。1 つのコマンドを複数の行にまたがって入力する必要がある場合は、行末に円記号(¥)を入力して、次の行に残りを入力します。コマンド ラインは編集可能です。左右の矢印キーで行内を移動します。文字を挿入するには、コマンド ラインに文字を直接入力します。文字を削除するには、標準の BackSpace キーや Del キーを押します。UNIX の場合は、Ctrl キーを押しながら D キーを押して文字を削除することもできます。

selang では、UNIX シェル tcsh およびその他のスマート シェルで使用できる多数のコマンド ライン入力機能がサポートされています。以下の機能が含まれます。

- 特殊文字
- ショートカット キー
- コマンド履歴
- 特殊機能

注: UNIX の場合には *UNIX exit* を使用できます。これは、ユーザまたはグループの追加または更新の前または後に自動的に実行されるように指定することができるプログラム(シェル スクリプトまたは実行可能ファイル)です。UNIX exit の詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

特殊文字

selang では、以下の特殊文字をサポートしています。

文字	説明	意味
# または *	ポンド(シャープ)またはアスタリスク	行頭にある場合は、その行がコメントであることを示し、その行は実行されません。コメント行は、ファイルから selang コマンドを入力する場合に有用です。
!	感嘆符(!)	行頭にある場合は、その行がシェル コマンドであることを示します。selang はコマンドをオペレーティング システムのシェル プログラムに送って実行します。CA Access Control はシェル コマンド行を実行しません。

文字	説明	意味
¥	円記号	行末の円記号は、コマンドが次の行に続くことを示します。
;	セミコロン	1 つのコマンドを終了し、同じ行に別のコマンドを指定します。
|	パイプ	前のコマンドの出力を次のコマンドの入力に送ります（*パイプの指定*）。

ショートカット キー

selang では、以下のショートカット キーをサポートしています。

キー	サポート プラットフォーム	意味
上矢印キー、下矢印キー、または ^	すべて	コマンド履歴内を移動してコマンドを取得するために使用します。
タブ	UNIX	単語補完機能を実行します。
Ctrl + D	UNIX	行末にカーソルを置いてこのキーを押すと、コマンド ラインの単語補完文字列に一致する単語のリストが表示されます。 行末以外の任意の場所にカーソルを置いてこのキーを押すと、カーソルの右側にある文字が削除されます。
Esc、Esc Ctrl + 2	UNIX	コマンド ラインのコマンドのヘルプ テキストが表示されます。コマンド ラインのテキストはすべて保存されるため、入力を中断した位置から続けてコマンドを入力できます。
F1	Windows	1 つ前のコマンドを 1 文字ずつ挿入します。
F2	Windows	ウィンドウが開き、「Enter char to copy up to:」という指示が表示されます。前回のコマンドに含まれていた文字を 1 文字入力すると、その文字が最初に出現する箇所までコマンドが自動的に入力されます。コマンド内に同じ文字が複数ある場合は、F2 キーをもう一度押すと、コマンドでその文字が 2 回目に出現するまでの部分が自動的に入力されます。 取り消すには BackSpace キーを押します。
F3	Windows	1 つ前のコマンドを入力します（上方向キーと同じ）。

キー	サポート プラット フォーム	意味
F4	Windows	1 つ前のコマンドを編集します。ウィンドウが開き、「Enter char to delete up to:」という指示が表示されます。 取り消すには BackSpace キーを押します。
F5	Windows	1 つ前のコマンドを入力します（上方向キーと同じ）。
F6	Windows	コマンド ラインに Ctrl + Z（^Z）を入力します。これにより、Enter キーを押して次の行に続けてコマンドを入力できるようになります。
F7	Windows	コマンド履歴を示すウィンドウを表示します。上下の方向キーで、前に入力した任意のコマンドを選択できます。 取り消すには Esc キーを押します。
F8	Windows	上方向キーと同様に 1 つ前のコマンドを入力します。ただし、カーソルはコマンド ラインの最後ではなく先頭に表示されます。
F9	Windows	ウィンドウが開き、「Enter command number:」という指示が表示されます。番号を入力するとコマンドが挿入されます。このコマンドは、F7 キーを押すと表示されるリストで、その番号と対応するコマンドです。 取り消すには Esc キーを押します。

コマンド履歴

selang では、実行されたコマンドを*履歴リスト*に保存します。履歴リストに保存されたコマンド ラインのコマンドを表示するには、上下の矢印キーを使用します。特定の文字または文字列で始まるコマンドのみを表示するには、コマンドの先頭文字を入力した後に上下の矢印キーを使用します。*Enter* キーを押すと、コマンド ラインに現在表示されているテキストが実行されます。

以前発行したコマンドを表示するには、history コマンドを入力します。

selang コマンド シェルでは、以下のショートカットを使用して、履歴リストに保存されたコマンドを実行できます。

ショートカット	実行されるコマンド
^^［*string*］	1 つ前のコマンド。*string* を指定すると、指定された文字列が元のコマンドに追加されます。

ショートカット	実行されるコマンド
^*n* [*string*]	履歴リストの *n* 番目のコマンド（*n* は正の整数）。*string* を指定すると、指定された文字列が元のコマンドに追加されます。
^-*n* [*string*]	履歴リストの最後から *n* 番目のコマンド（*n* は正の整数）。*string* を指定すると、指定された文字列が元のコマンドに追加されます。
^*mask* [*string*]	*mask* で始まるコマンドの中で最後に発行したコマンド（*mask* はテキスト文字列）。*string* を指定すると、指定された文字列が元のコマンドに追加されます。

注: Windows の場合は、履歴リストの表示に F7 キーを使用できます。

特殊機能

selang コマンド シェルでは、入力の手間を省くさまざまなテクニックを使用できます。

注: レコード名とクラス名は、UNIX では大文字と小文字が区別されますが、Windows では区別されません。

- コマンド認識

  selang では、他の使用可能なコマンドと区別できる長さの文字列を入力すると、ただちに目的のコマンドが認識されます。たとえば、「**ho**」と入力するだけで *hosts* コマンドを実行できます。これは、「ho」で始まるコマンドが hosts だけだからです。「**ho**」と入力すると、目的のコマンドが hosts であることがただちに認識されます。一方、文字列 **new** で始まるコマンドは複数あります。このため、newusr、*newgrp*、ewfile、および newres *を区別するには、識別に必要な長さの文字列を入力する必要があります。*

- 略語

  各コマンドには 1 ～ 4 文字の省略形が関連付けられています。たとえば、文字列 new で始まるコマンドは複数あるため、**newusr** の代わりに省略形 **nu** も使用できます。このような省略形は、各コマンドの構文の一部として記載されています。コマンドは、大文字または小文字のいずれでも入力できます。

- 単語保管(UNIX のみ)

  単語の入力途中で *Tab* キーを押すと、残りの文字が自動的に入力されます。単語補完では状況に応じた処理が行われます。指定した文字列と一致する単語が複数ある場合、最も短い単語またはその文字列と一致する単語の一部が入力されます。たとえば、「*n*」と入力した場合、自動的に「ew」が追加され、単語「new」が表示されます。「new」が目的の単語ではない場合、さらに 1 文字または 2 文字入力し、*Tab* キーをもう一度押して完全な単語にします。Ctrl キーを押しながら D キーを押すと、使用できるすべての候補が表示されます。この機能は、使用するコマンドが正確にわからない場合に便利です。前のパラグラフの例では、単語「new」の次に「u」と入力して *Tab* キーを押すと、自動的に「sr」が追加され、newusr コマンドが表示されます。

  selang コマンドの一部ではない単語はメモリに保存され、後で同じセッションの単語補完に使用されます。たとえば、「newusr Mercedes」と入力し、しばらくしてから「showusr Me」と入力して *Tab* キーを押すと、以下のように省略形の「Me」から「Mercedes」に単語が補完されます。

  ```
  showusr Mercedes
  ```

  ここでは、「Me」で始まるユーザ名が以前に入力されていないことを前提としています。

ワイルドカードによる一致

selang では、以下のワイルドカード文字を使用できます。

***(アスタリスク)**

0 個以上の文字列

? (疑問符)

任意の 1 文字(ファイルのパスを区切り文字を除く)

任意の 1 文字に一致するパターンを指定するには、以下の例のように、疑問符(?)を使用します。

ワイルドカード指定	一致パターン
mmc?	mmc3、mmcx、mmc5
mmc?.t	mmc1.t、mmc2.t
mmc04.?	mmc04.a、mmc04.1

0 個以上の任意の文字列に一致するパターンを指定するには、以下の例に示すようにアスタリスク(*)を使用します。

ワイルドカード指定	一致パターン
i.c	main.c、list.c
st*.h	stdio.h、stdlib.h、string.h
*	指定されたクラスのすべてのレコード

selang コマンドの構文

selang の各コマンドは、CA Access Control データベースに対して特定のアクションを実行します。selang コマンドの構文は、以下のとおりです。

```
commandname parameters
```

commandname には、CA Access Control で実行するコマンドを指定します。通常は、コマンドの後ろに 1 つ以上のパラメータを指定します。パラメータは、コマンドの実行に必要な追加情報を CA Access Control に渡します。

selang のパラメータ構文は、以下のとおりです。

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

parameterName には、CA Access Control に渡すパラメータを指定します。多くのパラメータでは、パラメータの処理に必要な情報を CA Access Control に渡す引数を、arguments に指定する必要があります。複数の引数を指定できるパラメータもあります。複数の引数を指定する場合は、カンマまたはスペースで引数を区切ります。パラメータの引数そのものがパラメータになる場合もあります。

文字列で引数を定義する場合にレコード プロパティを削除するには、空のかっこ「**()**」を使用してプロパティを入力します。場合によっては、引数としてアスタリスク(*)を使用できます。アスタリスクは、その引数が取りうるすべての値を表すことができます。アスタリスクを使用する*前または後の*コマンドで 同じ引数に特定の値を指定した場合、その特定の値の指定は無効になりません。また、引数がファイル名の場合は、ファイル名パターンの一部としてワイルドカードを使用できます。使用できるワイルドカードは、*(0 個以上の文字)および ? (任意の 1 文字)です。

UNIX 環境では、ユーザが指定する情報で大文字と小文字が区別され、両方の文字を使用できます。たとえば、ユーザ ID が user53 のユーザのフル ネームを Mike Jones と指定できます。Windows 環境では、情報の大文字と小文字の区別は認識されませんが、*その情報は保存されます*。UNIX ワークステーションから Windows のリモート ホストを管理する場合、UNIX は保存された状態のユーザ指定情報を検索します。*たとえば、ローカル CA Access Control データベースを管理する場合、Windows 環境では Mike Jones と識別されるユーザの名前を「mike jones」と入力することができます。ただし、このデータベースをリモート UNIX マシンから管理する場合は、ユーザ名を「Mike Jones」と入力する必要があります。*

selang コマンドの権限

selang コマンドを使用して AC データベースまたはネイティブ オペレーティング システム（ネイティブ OS）環境のレコードを変更するには、適切な権限が必要です。ほとんどのコマンドの場合、実行するには以下の*いずれかの*条件を満たしている必要があります。

- リソースの所有者であること
- ADMIN 属性が割り当てられていること
- GROUPADMIN 属性で管理者権限を与えられたグループの有効範囲内に、目的のリソース レコードが含まれていること
- ADMIN クラスのレコードの ACL に、CREATE アクセス権限または MODIFY アクセス権限が設定されていること
- （Windows）ネイティブ Windows 環境の管理のみが許可されている場合は、Windows データベースの CA Access Control Administrators グループのメンバであること
- （UNIX）ネイティブ UNIX 環境の管理のみが許可されている場合は、ローカル UNIX ホストのセキュリティ ファイルの CA Access Control Administrators グループのメンバであること

注: これらの一般原則の例外については、各コマンドの説明に注記してあります。

アクセス制御リストのサポート

アクセス権限を許可または拒否するために、7 種類のアクセス制御リストを使用することができます。

ACL

標準アクセス制御リスト。リソースへのアクセスを許可されたユーザまたはグループの名前（あるいはその両方）、および各ユーザまたはグループに与えられたアクセス権のレベルが登録されています。

NACL

拒否アクセス制御リスト。リソースへのアクセスが許可されていないユーザまたはグループの名前が登録されています。

PACL

プログラム アクセス制御リスト。リストにアクセスするプログラムに依存します。各 PACL には、ユーザ名およびグループ名、アクセス権限レベル、および特定のリソースにアクセスするためにユーザが実行する必要があるプログラムやシェル スクリプトの名前が登録されています。

INET-ACL

インターネット アクセス制御リスト

CACL

条件付きアクセス制御リスト

CALACL

カレンダ アクセス制御リスト。Unicenter TNG カレンダに依存するリソース ACL です。

AZNACL

権限 ACL。リソースの説明に基づいてリソースへのアクセスを許可する ACL です。

CA Access Control では、リソースに対するユーザのアクセス権をチェックする際に、関連するすべてのリストが使用されます。

注: authorize コマンドで一度に操作できるリストは 1 つです。複数のリストを変更する場合は、authorize コマンドを繰り返し発行する必要があります。1 つの権限ルールで複数のユーザおよびグループに対する複数のアクセス権限を定義することはできません。その場合には、ルールを分割する必要があります。

以下の表に、各クラスで使用できるアクセス制御リストを示します。この表にないクラスは、アクセス制御リストがないため、authorize コマンドでは制御できません。

クラス	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			
REGVAL	X	X	X			
SUDO	X	X	X			
SURROGATE	X	X	X			
TCP	X	X	X		X	

クラス	ACL/ NACL	CALACL	PACL	INET-ACL	CACL	AZNACL
TERMINAL	X	X	X			
UACC	X	X				
USER_DIR	X					X

クラス別アクセス権限

有効なアクセス値は、リソースが属するクラスによって異なります。以下の表に、AC 環境における有効なアクセス値をクラス別に示します。

クラス	有効なアクセス値	アクセサに許可される操作
全クラス	all	そのクラスの *すべての*有効な操作を実行します。
	none	そのクラスの*どの*有効な操作も実行しません。
ADMIN	create	このクラスのレコードを作成します。
	delete	このクラスのレコードを削除します。
	join	グループを USER レコードに追加して、ユーザからグループへのリンクを完成します。 **注**: アクセサには *modify* アクセス権も必要です。
	modify	既存のレコードを変更します。 **注**: ユーザをグループにリンクする(ユーザ名を GROUP レコードに追加する)ために、アクセサには *join* アクセス権も必要です。
	password	他のユーザのパスワードを変更します。 **注**: このアクセス タイプは USER クラスにのみ適用されます。
	読み取り	このクラスのレコードを一覧表示します。
AUTHHOST	読み取り	認証されたホストからログインします。
CONNECT	読み取り	リモート ホストへ接続します。
CONTAINER	*inherited*	**注**: このクラスの有効なアクセス値は、含まれているオブジェクトのクラスの有効値です。

クラス	有効なアクセス値	アクセサに許可される操作
DOMAIN	chmod	2 つのドメイン間の信頼関係を作成および削除します。 **注**: どちらのドメインにもこのアクセス タイプが必要です。
	execute	ドメインに対するメンバの追加または削除を行います。
	読み取り	ドメイン メンバを一覧表示します。
FILE、GFILE	chdir	read および execute に相当するアクセス権限を使用して、ディレクトリへアクセスします。
	chmod	ファイル システム モードを変更します。 **注**: UNIX ホストにのみ適用されます。
	chown	レコードの所有者を変更します。
	control	*delete* と *rename* を除くすべての有効な操作を実行します。
	create	このクラスのレコードを作成します。
	delete	このクラスのレコードを削除します。
	execute	プログラムを実行します。 **注**: アクセサには *read* アクセス権も必要です。
	読み取り	ファイルまたはディレクトリを読み取り専用で使用します。 **注**: UNIX で、ファイルに関する情報を取得する操作 (ls -l など) をユーザが実行できるかどうかを制御するために *read* 権限が必要な場合は、STAT_intercept 環境設定を 1 に設定します。詳細については、「*リファレンス ガイド*」を参照してください。
	rename	このクラス内のレコードの名前を変更します。
	sec	このクラスのレコードの ACL を変更します。
	更新	*read*、*write*、および *execute* を組み合わせた操作を実行します。
	utime	ファイルの変更日時を変更します。 **注**: UNIX ホストにのみ適用されます。

クラス	有効なアクセス値	アクセサに許可される操作
	write	ファイルまたはディレクトリを変更します。
HNODE	読み取り	クラスのレコードを一覧表示します。
	write	レコードの詳細を編集します。
HOLIDAY	読み取り	指定した休日中にログインします。
KMODULE	load	カーネル モジュールをロードします。
	unload	カーネル モジュールをアンロードします。
MFTERMINAL	読み取り	メインフレーム端末からログインします。
	write	メインフレーム端末から管理を行います。
POLICY	delete	ポリシーを削除します。
	execute	ポリシーをデプロイします。
	読み取り	ポリシーの詳細を表示します。
	write	レコードの詳細を編集します。
	undeploy	*delete* と *execute* を組み合わせた操作を実行します。
PROCESS	読み取り	プロセスを強制終了します。
PROGRAM、SUDO、GSUDO	execute	プログラムを実行します。
REGKEY	delete	Windows レジストリ キーを削除します。
	読み取り	Windows レジストリ キーの内容を一覧表示します。
	write	Windows レジストリ キーを変更します。
REGVAL	delete	Windows レジストリ値を削除します。
	読み取り	Windows レジストリ値を読み取ります。
	write	Windows レジストリ値を変更します。
RULESET	読み取り	レコードの詳細を表示します。
	write	レコードの詳細を編集します。
SURROGATE	execute	別ユーザの代わりに操作を実行します。
TCP	読み取り	リモート ホストまたはホスト グループから TCP サービスへアクセスします。
TERMINAL、GTERMINAL	読み取り	端末へログインします。

クラス	有効なアクセス値	アクセサに許可される操作
	write	端末を管理します。
UACC	*inherited*	**注**: このクラスの有効なアクセス値は、定義しているクラスの有効値です。
WINSERVICE	読み取り	Windows サービスのプロパティを表示します。
	start	Windows サービスを開始します。
	modify	Windows サービスのプロパティを変更します。
	resume	一時停止された Windows サービスを再開します。
	stop	Windows サービスを停止します。
	pause	Windows サービスを一時停止します。

注: 値 none および all は全クラスで使用できます（値 all は、各クラスの none を除くアクセス値のグループ全体を表します）。アクセス権限の詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

Windows でのクラス別アクセス権限

有効なアクセス値は、リソースが属するクラスによって異なります。以下の表に、Windows（NT）環境における有効なアクセス値をクラスごとに示します。

クラス	有効なアクセス値	アクセサに許可される操作
全クラス	all	そのクラスの *すべての*有効な操作を実行します。
	none	そのクラスの*どの*有効な操作も実行しません。
COM、DISK	change	*delete*、*read*、および *write* を組み合わせた操作を実行します。
	changepermissions	リソースの ACL を変更します。
	delete	リソースを削除します。
	読み取り	リソース上のデータへ読み取り専用でアクセスします。
	takeownership、chown、owner	指定したリソースの所有者を変更します。
	write	指定したリソースへデータを書き込みます。

クラス	有効なアクセス値	アクセサに許可される操作
FILE		**注**: アクセス権限を定義できるのは NTFS ファイルに対してのみです。FAT ファイルには定義できません。
	change	*delete*、*read*、および *write* を組み合わせた操作を実行します。
	changepermissions、sec	リソースの ACL を変更します。
	chmod	*delete* を除くすべての操作を実行します。
	chown	指定したリソースの所有者を変更します。
	delete	リソースを削除します。
	execute	プログラムを実行します。 **注**: このアクセス権を使用するには、*read* アクセス権も必要です。
	読み取り	リソースへ読み取り専用でアクセスします。
	rename	リソース名を変更します。 **注**: ファイル名を変更するには、ソースに対する *delete* アクセス権とターゲットに対する *rename* アクセス権が必要です。監査ログにはイベントがこの順序で記録されます。
	write	リソースを変更します。
	更新	*read*、*write*、および execute を組み合わせた操作を実行します。
PRINTER	manage	プリンタを管理します。指定したプリンタへのデータの設定、印刷の一時停止、印刷の再開、全印刷ジョブのクリア、ACL の更新、プリンタのプロパティの変更などが挙げられます。
	print	プリンタを使用して印刷します。
REGKEY	append、create、subkey	レジストリ キーのサブキーを作成または変更します。
	takeownership、chown、owner	リソースの所有者を変更します。
	changepermissions、sec、dac、writedac	リソースの ACL を変更します。
	delete	リソースを削除します。

クラス	有効なアクセス値	アクセサに許可される操作
	enum	サブキーを列挙します。
	link	レジストリ キーへのリンクを作成します。
	notify	レジストリ キーまたはレジストリ キーのサブキーの変更通知です。
	query	レジストリ キーの値をクエリします。
	読み取り	リソースへ読み取り専用でアクセスします。
	readcontrol、manage	レジストリ キーのセキュリティ記述子の情報（システム（監査） ACL に含まれている情報を除く）の読み取り
	set	レジストリ キーの値の作成または設定します。
	write	レジストリ キーとそのサブキーを変更します。
SHARE	change	リソースのプロパティの変更、またはリソースからの共有の削除を行います。
	読み取り	リソースへ読み取り専用でアクセスします。

注: 値 *none* および *all* は全クラスで使用できます（値 *all* は、各クラスの *none* を除くアクセス値のグループ全体を表します）。アクセス権限の詳細については、「*Windows エンドポイント管理ガイド*」を参照してください。

selang 環境

selang を使用すると、ローカル CA Access Control データベースに加えて、ネイティブ（Windows または UNIX）データベース、ローカル Policy Model データベース（PMDB）、CA Access Control がインストールされているリモート ホスト（Windows または UNIX）上のデータベース、または CA Access Control 環境設定上のデータベースを変更できます。環境を切り替えるには、*env*（environment）コマンドを使用します。このコマンドはすべての環境で使用できます。

複数の環境で共通のコマンドもありますが、その場合でも、コマンドのパラメータおよび引数が異なる場合があります。そのため、新しい環境で作業を始めるときは、必ず構文を確認してください。

注: env を使用して、コマンドの *ネイティブ* プロパティを入力すると、そのコマンドが *ネイティブ*環境と現在の環境の両方に入力されます。

以下の環境がサポートされています。

環境	コマンド	プロンプト	説明
Policy Model	`env pmd`	`AC(pmd)>`	selang コマンドはすべて、ローカル PMDB に対して実行されます。
ネイティブ Windows	`env nt`	`AC(nt)>`	selang コマンドはすべて、Windows データベースを変更します。
AC	`env ac`	`AC>`	selang コマンドはすべて、CA Access Control データベースに対して実行されます。 **注**: これがデフォルトです。
ネイティブ UNIX	`env unix`	`AC(unix)>`	selang コマンドはすべて、ローカル UNIX ホストのセキュリティ ファイルに対して実行されます。
Native	`env native`	`AC(native)>`	selang コマンドはすべて、ホストのネイティブ環境で実行されます。
リモート設定	`env config`	`AC(config)>`	selang コマンドはすべて、ホストの CA Access Control 環境設定に対して実行されます。

詳細情報:

environment コマンド - セキュリティ環境の設定 (P. 131)
AC 環境のクラス (P. 258)
UNIX 環境のクラス (P. 562)
Windows 環境のクラス (P. 524)

UNIX での selang 環境設定

UNIX の場合は、selang の動作を管理できます。ほとんどのオプションが、(selang UNIX 環境の) UNIX セキュリティ システムを selang が管理する方法に関係しています。

selang ユーティリティでは、設定オプション用に以下の 2 つのファイルを使用します。

seos.ini

CA Access Control 設定オプションが格納されます。これが、CA Access Control のメイン環境設定ファイルです。

lang.ini

selang で使用する設定情報が格納されます。

selang では、以下の *一方または両方の*ディレクトリにある lang.ini ファイルを使用します。

- seos.ini ファイルが格納されているディレクトリ。
- ユーザのホーム ディレクトリ。

トークンをこれらの lang.ini ファイルの一方にのみ指定した場合は、指定されたファイルの値が使用されます。トークンを 2 つの lang.ini ファイルで異なる値に指定した場合は、ユーザのホーム ディレクトリにあるファイルの値が優先されます。

サーバの seos.ini ファイルの DefaultShell トークンおよび DefaultHome トークンの値は、lang.ini ファイルの DefaultShell トークンおよび HomeDirPrefix トークンに設定されている値より*優先されます*。

注: サンプル lang.ini ファイルは、*ACInstallDir*/samples ディレクトリの lang.init です。

ユーザ ファイルの変更

UNIX ユーザの更新時に使用されるデフォルトのファイルは /etc/passwd ですが、このデフォルトを変更できます。デフォルトの変更は、NIS で作業する場合に NIS サーバ コンピュータに対して必要になります。

ユーザ ファイルを変更するには、seos.ini ファイルの *passwd* セクションの *YpServerPasswd* がユーザ ファイルのフルパス名を示すよう変更します。

グループ更新時のファイルの変更

UNIX グループの更新時に使用されるデフォルトのファイルは /etc/group ですが、このデフォルトを変更できます。デフォルトの変更は、NIS で作業する場合に NIS サーバ コンピュータに対して必要になります。

グループを更新するためのファイルを変更するには、**seos.ini** ファイルの **passwd** セクションの **YpServerGroup** がユーザ ファイルのフルパス名を示すように変更します。

UNIX ユーザ ファイルおよびグループ ファイルの自動バックアップ

CA Access Control では、セッションでの UNIX ユーザおよび UNIX グループの初回更新前に、/etc/passwd ファイルまたは /etc/group ファイルのバックアップ コピーが作成されます。バックアップ ファイルは、それぞれ /etc/passwd.SeOS.bak および /etc/group.SeOS.bak という名前になります。UNIX システムの更新時にエラーが発生した場合は、元の情報を復元できます。バックアップが作成されるのは、selang コマンド シェルのセッションで UNIX システムに対して最初の変更を行う前のみです。

selang ヘルプの表示

対話式の selang コマンド環境では、いつでもヘルプを表示できます。

selang のオンライン ヘルプに切り替えるには、以下のいずれかを入力します。

? または help

現在の環境の selang に関するオンライン ヘルプ テキストの目次が画面に表示されます。

help topic

topic

selang コマンドまたは selang コマンド シェルに関連するその他のトピックを指定します。

指定したトピックに関するヘルプ テキストが表示されます。

help env

env

selang 環境を指定します。

指定した環境に関するヘルプ テキストの目次が画面に表示されます。

注: UNIX の場合に、コマンド ラインのテキストを削除せずに、コマンド ラインに入力したコマンドのヘルプ テキストを表示するには、Ctrl キーを押しながら 2 を押します（または Esc キーを 2 回押します）。

詳細情報:

help コマンド - selang ヘルプの表示 (P. 138)
selang 環境 (P. 37)
selang コマンド リファレンス (P. 43)

第 3 章: selang コマンド

このセクションには、以下のトピックが含まれています。

selang コマンド リファレンス

以下の表に、selang のすべてのコマンドをアルファベット順に示します。

注: すべての環境で同じように動作するコマンドは、AC 環境の説明にのみ記述があります。ただし、複数の環境で使用できても、環境ごとに動作が異なるコマンドが一部ありますので、ご注意ください。このようなコマンドには以下の表の「*説明*」欄にアスタリスク (*) が付けられており、使用可能な環境別の項目に別途説明があります。

コマンド	省略形	環境	説明
alias		AC および UNIX **注**: UNIX ホストのみ。	selang のコマンドおよびプロパティの別名を一覧表示または定義します。
authorize	auth	AC および nt	* 特定のリソースへのアクセス権を特定のアクセサに設定します。
authorize-	auth-	AC および nt	* 特定のリソースへのアクセス権を特定のアクセサから削除します。
backuppmd		pmd	PMDB データベース内のデータを指定されたディレクトリにバックアップします。
check		AC	特定のリソースへのアクセス権限がユーザにあるかどうかをチェックします。
checklogin		AC	ユーザのログイン権限、パスワード チェックが必要かどうか、および端末アクセス チェックが必要かどうかを確認します。

コマンド	省略形	環境	説明
checkpwd		AC	ユーザの新しいパスワードが、パスワード ルールに従っているかどうかをチェックします。変更はしません。
chfile	cf	AC および native	* CA Access Control データベースまたはネイティブ OS データベースのファイル レコードの定義を変更します。
chgrp	cg	AC および native	* CA Access Control データベースまたはネイティブ OS データベースの既存の内部グループ設定を変更します。
chres	cr	AC および nt	* CA Access Control データベースまたはネイティブ OS データベースの既存のリソース レコードを変更します。
chusr	cu	AC および native	* CA Access Control データベースまたはネイティブ OS データベースの既存の内部ユーザを変更します。
chxgrp	cxg	AC	CA Access Control データベースの既存のエンタープライズ グループ設定を変更します。
chxusr	cxu	AC	CA Access Control データベースの既存のエンタープライズ ユーザ設定を変更します。
createpmd		pmd	リモート ホスト上に PMDB を作成します。
deletepmd		pmd	PMDB の selang 保護ファイル、PMDB ディレクトリの内容、および PMDB ディレクトリをリモート ホストから削除します。
deploy		AC	特定の POLICY の RULESET オブジェクトに格納されている、selang のデプロイ コマンドを実行します。
deploy-		AC	特定の POLICY の RULESET オブジェクトに格納されている、selang のポリシー デプロイ解除コマンドを実行します。
editfile	ef	AC および native	* CA Access Control データベースまたはネイティブ OS データベースのファイル レコードの定義を追加または変更します。

コマンド	省略形	環境	説明
editgrp	eg	AC および native	* CA Access Control データベースまたはネイティブ OS データベースに対し、新しいグループの追加または既存のグループ設定の変更を行います。
editres	er	AC および nt	* CA Access Control データベースまたはネイティブ OS データベースに対し、新しいリソース レコードの追加または既存のリソース レコードの変更を行います。
editres config		config	指定したソースの環境設定を一覧表示します。
editusr	eu	AC および native	* CA Access Control データベースまたはネイティブ OS データベースに対し、新しいユーザの追加または既存のユーザの変更を行います。
editxgrp	exg	AC	CA Access Control データベースに対し、新しいエンタープライズ グループの追加または既存のエンタープライズ グループ プロパティの変更を行います。
editxusr	exu	AC	CA Access Control データベースに対し、新しいエンタープライズ ユーザの追加または既存のエンタープライズ ユーザ プロパティの変更を行います。
end_transaction		AC	デュアル コントロール PMDB プロセスの start_transaction コマンドを完了します。
環境	env	*all*	selang を実行するセキュリティ環境を設定します。
find	f	AC および native	環境に存在するクラスまたはクラスに含まれているレコードを一覧表示します。
findpmd		pmd	コンピュータ上のすべての PMDB を一覧表示します。
find config		config	このホストで管理できる環境設定のソース(ini ファイルまたはレジストリ エントリ)を一覧表示します。
find file		native	システム ファイルを一覧表示します。

コマンド	省略形	環境	説明
find xgroup		nt	現在のドメインまたは信頼できるドメインに存在するエンタープライズ グループの名前を一覧表示します。
find xuser		nt	現在のドメインまたは信頼できるドメインに存在するエンタープライズ ユーザの名前を一覧表示します。
get dbexport		AC	CA Access Control または PMD データベースからエクスポートされたルールを取得します。
get devcalc		AC	ポリシー偏差計算の結果を取得します。
help		*all*	selang ヘルプを表示します。
history		*all*	セッションでこれまでに発行したコマンドを表示します。
hosts		*all*	selang コマンドの送信先ホストを表示または設定します。
join	j	AC および native	* ユーザをグループに追加します。
join-	j-	AC および native	* ユーザをグループから削除します。
joinx	jx	AC	エンタープライズ ユーザをグループに追加します。
joinx-	jx-	AC	エンタープライズ ユーザをグループから削除します。
list		AC および native	*find* コマンドの別名です。
listpmd		pmd	PMDB とそのサブスクライバ、更新ファイル、およびエラー ログに関する情報を一覧表示します。
newfile	nf	AC	CA Access Control データベースのファイル レコードの定義を追加します。
newgrp	ng	AC および native	* CA Access Control データベースまたはネイティブ OS データベースに新しいグループを追加します。
newres	nr	AC および nt	* CA Access Control データベースまたはネイティブ OS データベースに新しいリソース レコードを追加します。

コマンド	省略形	環境	説明
newusr	nu	AC および native	* CA Access Control データベースまたはネイティブ OS データベースに新しい内部ユーザを追加します。
newxgrp	nxg	AC	CA Access Control データベースに新しいエンタープライズ グループを追加します。
newxusr	nxu	AC	CA Access Control データベースに新しいエンタープライズ ユーザを追加します。
pmd		pmd	Policy Model のエラー ログの消去、サブスクライバ リストの更新、サブスクライバの解放、Policy Model サービスの開始と停止、更新ファイルの切り捨て、および初期化ファイルの再ロードを行います。
rename		AC	データベースのオブジェクト名を変更します。
restorepmd		pmd	ローカル ホスト上に PMDB をリストアします。
rmfile	rf	AC	CA Access Control データベースからファイル リソース レコードを削除します。
rmgrp	rg	AC および native	* CA Access Control データベースまたはネイティブ OS データベースからグループを削除します。
rmres	rr	AC および nt	* CA Access Control データベースまたはネイティブ Windows データベースからリソース レコードを削除します。
rmusr	ru	AC および native	* CA Access Control データベースまたはネイティブ OS データベースからユーザを削除します。
rmxgrp	rxg	AC	CA Access Control データベースからエンタープライズ グループを削除します。
rmxusr	rxu	AC	CA Access Control データベースからエンタープライズ ユーザを削除します。
ruler		AC および native	表示コマンドを実行したときに表示されるプロパティを設定します。
search		AC および native	*find* コマンドの別名です。

コマンド	省略形	環境	説明
setoptions	so	AC および nt	* データベースの動作を制御するグローバル オプションを設定または表示します。
showfile	sf	AC および native	* CA Access Control データベースまたはネイティブ OS データベースのファイル レコードのプロパティを一覧表示します。
showgrp	sg	AC および native	* CA Access Control データベースまたはネイティブ OS データベースのグループ レコードのプロパティを一覧表示します。
showres	sr	AC および nt	* CA Access Control データベースまたはネイティブ Windows データベースのレコードのプロパティを一覧表示します。
showres config		config	指定したソースの環境設定を一覧表示します。
showusr	su	AC および native	* CA Access Control データベースまたはネイティブ OS データベースのユーザ レコードのプロパティを一覧表示します。
showxusr	sxu	AC	CA Access Control データベースのエンタープライズ ユーザ レコードのプロパティを一覧表示します。
source		*all*	特定のファイル内のコマンドを実行します。
start dbexport		AC	CA Access Control または PMD データベースをエクスポートします。
start devcalc		AC	ポリシー偏差計算を開始します。
start_transaction		AC	1 つ以上のコマンドで構成されたデュアル コントロール PMDB プロセスの、未処理のトランザクションを保存するファイルの記録を開始します。
subs		pmd	親 PMDB にサブスクライバを追加するか、親 PMDB に対してデータベースをサブスクライブします。
subspmd		pmd	接続先ホストのデータベースの親を変更します。
unalias		AC および UNIX	selang のコマンドおよびプロパティの別名を削除します。
undeploy		AC	deploy- コマンドの別名です。

コマンド	省略形	環境	説明
unsubs		pmd	PMDB のサブスクライバ リストからサブスクライバを削除します。
xaudit		nt	監査基準を設定して、アクセス イベントの記録を開始します。
xaudit-		nt	監査基準を削除して、アクセス イベントの記録を停止します。

注: ネイティブ環境は、接続するホストのオペレーティング システムに応じて、Windows (nt) または UNIX のいずれかの環境の規則に従います。

AC 環境の selang コマンド

このセクションでは、CA Access Control データベースに対して実行される selang コマンド (AC 環境のコマンド) のすべてをアルファベット順に説明します。

alias コマンド - selang 別名の定義

UNIX ホストで有効

alias コマンドを使用すると、selang のコマンドやプロパティの別名を一覧表示または定義することができます。alias コマンドは、すべてのユーザが実行できます。

注: selang のすべてのセッションで使用する別名のセットを構築するには、それらの別名をスタートアップ ファイルに定義し、*selang-r* コマンドを使用します。

このコマンドの形式は以下のようになります。

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

aliasName

(オプション) 別名として使用する名前を指定します。

このオプションが指定されなかった場合は、定義されているすべての別名が一覧表示されます。

aliasValue

(オプション) selang コマンド シェルで *aliasName* に関連付ける内容を指定します。

このオプションが指定されなかった場合は、指定された別名の値が表示されます。

aliasValue には変数を 10 個まで ($0 ～ $9) 指定できます。*aliasValue* に変数がある場合は、alias を実行する際に各変数をかっこで囲まれた適切な値に置き換える必要があります。

例: 変数を使用した、新しい管理者の作成の簡略化

新しい管理者をデータベースに簡単に追加するための別名を作成するには、以下のコマンドを入力します。

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

この別名は、新しい管理者の名前をかっこの中に追加するだけで使用できます。以下に例を示します。

```
newadm(Terri)
```

ユーザ Terri がデータベースに追加されます。Terri には、データベースの管理に必要な ADMIN 属性が与えられます。これは以下のコマンドを入力することに相当します。

```
newusr Terri admin
```

例: プロパティ名の簡略化

プロパティ名 *access* を省略形 *acc* に置き換える別名を作成するには、以下のコマンドを入力します。

```
alias acc access
```

これにより、この別名を使用して以下のように入力できるようになります。

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

例: コンテキストに応じた別名の使用

別名は単なる拡張された変数ではありません。コマンド名またはプロパティ名を指定すべきコンテキストでのみ解釈されます。たとえば、以下の別名を定義します。

```
alias newterm newres terminal
```

その上で、以下のコマンドを入力します。

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

newterm という文字列の最初の出現は置き換えられますが、2 番目はそのままです。これは、コンテキストによって文字列の 2 番目のインスタンスが端末名であることが求められるためです。これは以下のコマンドを入力することに相当します。

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

詳細情報:

unalias コマンド - selang の別名の削除 (P. 185)
selang ユーティリティ - CA Access Control コマンドラインの実行 (P. 20)

authorize コマンド - リソースに対するアクセス権限の設定

AC 環境で有効

authorize コマンドを使用して、リソースに対するアクセサのアクセス権限を変更できます。

このコマンドにより、リソースに関連付けられているアクセス制御リストが変更されます。変更されるアクセス制御リスト エントリは一度に 1 つです。

アクセサがリソースにアクセスしようとすると、CA Access Control はアクセス権限を決定するために、適切なアクセス制御リストをチェックします。チェック対象のアクセス制御リストはリソース レコードに記録されているもので、リソース グループ レコードに記録されているものが含まれることもあります。アクセサが対象リソースをカバーする NACL のいずれかでアクセス権限を拒否されている場合、別の ACL で権限が与えられても権限は拒否されます。

リソースの所有者には常に、そのリソースに対するすべてのアクセス権限が与えられます。所有者であるユーザのアクセス権限を変更するには、リソースの所有者を別のユーザ、たとえばユーザ nobody に変更します。

注: このコマンドは Windows 環境にもありますが、動作が異なります。

authorize コマンドを使用するユーザには、適切な権限が必要です。具体的には、以下の条件を 1 つ以上満たしている必要があります。

- ADMIN 属性が割り当てられていること
- リソースがメンバであるリソース グループに対して、GROUP-ADMIN 属性が割り当てられていること
- リソースの所有者であること
- リソースに対応する ADMIN クラス レコードの変更アクセス権限があること

authorize コマンドは、クラスのグループによって形式が異なります。クラスは以下のグループに分類されます。

- TCP
- HOST、GHOST、HOSTNET、および HOSTNP
- その他すべてのクラス

TCP クラスが対象の場合のコマンド形式は以下のとおりです。

```
{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、および HOSTNP クラスタが対象の場合のコマンド形式は以下のとおりです。

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

その他すべてのクラスの場合の形式は以下のとおりです。

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

access (*accessType)*

リソース アクセス制御リスト（ACL）のアクセス権限エントリを定義します。この ACL には、どのアクセス権限がアクセサに与えられるかを指定します。

accessType

アクセス タイプ（read や write など）をリソース ACL に定義します。

注：authorize コマンドで access(*accessType*) オプションと deniedaccess(*accessType*) オプションをどちらも省略した場合、CA Access Control は、UACC クラスにあるリソース クラスのレコード（リソースがファイルの場合は UACC ファイル レコード）の暗黙のアクセス プロパティで指定されるアクセス権を割り当てます。

calendar(*calendarName*)

アクセス権限を決定するために使用するカレンダを指定します。

className

resourceName の所属先クラスを定義します。

deniedaccess(*accessType)*

リソース NACL に指定されているアクセス権限を変更します。NACL には、アクセサに対してどのアクセス タイプを拒否するかを指定します。

accessType

拒否するアクセス タイプ（read や write など）を指定します。

gid (*accessor* [,*accessor...*])

アクセス権限の設定対象である内部グループを 1 つ以上定義します。

ghost(*ghostName [,ghostName]...*)

TCP/IP サービスに対するアクセス権限の設定対象であるグループ ホストを 1 つ以上定義します。

host(*hostName [,hostName]...*)

TCP/IP サービスに対するアクセス権限の設定対象であるホストを 1 つ以上定義します。

hostnet(*hostNetName [,hostNetName]...*)

TCP/IP サービスに対するアクセス権限の設定対象である HOSTNET レコードを 1 つ以上定義します。

hostnp(*hostNamePattern [,hostNamePattern]...*)

TCP/IP サービスに対するアクセス権限の設定対象である HOSTNP レコードを 1 つ以上定義します。

nt

Windows のシステム ACL に値を追加するかどうかを指定します。

FILE クラスに対してのみ有効です。

resourceName

変更対象のアクセス制御リストを持つリソース レコードを指定します。

service(*serviceName*|*serviceNumber*|*serviceNumberRange*)

ローカル ホストがリモート ホストに提供することが許されるサービスを指定します。

serviceNumber |serviceNumberRange

サービス番号またはサービス番号の範囲を指定します。

範囲は 2 つの整数をハイフン(-)で区切って「1-99」のように指定します。

制限: 指定できる整数の範囲は 0 ～ 65535 です。

stationName

指定されたクラスに属するレコード名を以下のように指定します。

- **HOST** - 単一の端末の名前
- **GHOST** - ghost コマンドでデータベースに定義されたホスト グループの名前
- **HOSTNET** - IP アドレスのマスク値と一致値で定義されたホスト グループの名前
- **HOSTNP** - 名前パターンによって定義されたホスト グループの名前

解決できないホストについては、IP アドレスの範囲を IPv4 形式で入力します。

tcpServiceName

アクセス権限を設定する対象の CA Access Control TCP サービス レコードを指定します。

uid (*accessor* [,*accessor...*])

アクセス権限の設定対象である内部ユーザを 1 つ以上定義します。

アスタリスク(*)を使用してすべての内部ユーザを指定することができます。

unix

UNIX のシステム ACL に値を追加するかどうかを指定します。

ACL をサポートする UNIX 環境でのみ有効です。また、FILE クラスのレコードに対してのみ有効です。

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

条件付きプログラム アクセスの対象となるプログラムを 1 つ以上定義します。via パラメータには、リソースの PACL のエントリを指定します。*programName* には、リソースにアクセスできるプログラムを指定します。*programName* には、ワイルドカード文字を使用できます。プログラムが PACL の複数のエントリと一致した場合、ワイルドカードとの不一致が最も長いエントリが優先されます。

programName に PROGRAM クラスで定義されていないプログラムまたはシェル スクリプトを 指定すると、そのプログラムまたはシェル スクリプトを保護する PROGRAM クラスのレコードが自動的に作成されます。

xgid (*accessor* [,*accessor...*])

アクセス権限の設定対象であるエンタープライズ グループを 1 つ以上定義します。

xuid (*accessor* [,*accessor...*])

アクセス権限の設定対象であるエンタープライズ ユーザを 1 つ以上定義します。

例: Angela に対するファイル読み取りの許可

以下の selang コマンドは、エンタープライズ ユーザ Angela に対し、FILE リソース /projects/secrets で保護されているファイルの読み取りを許可します。

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

例: Angela のみに対するファイル読み取りの許可

以下の selang コマンドは、エンタープライズ ユーザ Angela に対してのみ、FILE リソース /projects/secrets で保護されているファイルの読み取りを許可します。

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

注: UNIX で、ファイルに関する情報を取得する操作(ls -l など)をユーザが実行できるかどうかを制御するために *read* 権限が必要な場合は、STAT_intercept 環境設定を 1 に設定します。詳細については、「*リファレンス ガイド*」を参照してください。

例: グループに属するすべてのユーザに対する端末へのログインの許可

以下の selang コマンドは、エンタープライズ グループ RESEARCH のすべてのメンバに対し、TERMINAL リソース *tty10* で保護されている端末へのログインを許可します。

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

例: Joe に対するファイルのバックアップの許可

以下の selang コマンドは、エンタープライズ ユーザ Joe に対し、GFILE リソース secret_files で保護されているファイルのバックアップを許可します。

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

Windows エンドポイントに対する同等のコマンドは以下のとおりです。

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

これらのコマンドは、Joe のアクセス権限がリソースの ACL または NACL で規定されていない場合のみ有効です。

詳細情報:

authorize- コマンド - リソースからのアクセス権限の削除 (P. 57)
authorize コマンド - Windows リソースに対するアクセサのアクセス権限の設定 (P. 208)
chfile コマンド - ファイル レコードの変更 (P. 65)
ch[x]grp コマンド - グループ プロパティの変更 (P. 72)
chres コマンド - リソース レコードの変更 (P. 89)
ch[x]usr コマンド - ユーザ プロパティの変更 (P. 107)
authorize- コマンド - Windows リソースに対するアクセサのアクセス権限の削除 (P. 210)

authorize- コマンド - リソースからのアクセス権限の削除

AC 環境で有効

authorize- コマンドを使用すると、リソースのアクセス制御リスト(ACL)からアクセサを削除できます。

注: このコマンドはネイティブ Windows 環境にもありますが、動作が異なります。

authorize- コマンドを使用するには、authorize コマンドを使用する場合と同じアクセス権限が必要です。

authorize- コマンドは、クラスのグループによって形式が異なります。クラスは以下のグループに分類されます。

- TCP
- HOST、GHOST、HOSTNET、および HOSTNP
- その他すべてのクラス

TCP クラスが対象の場合のコマンド形式は以下のとおりです。

```
{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、および HOSTNP クラスタが対象の場合のコマンド形式は以下のとおりです。

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

その他すべてのクラスの場合の形式は以下のとおりです。

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

access-

このコマンドによるアクセサ削除の対象が NACL ではなくリソース ACL（アクセス権限を付与する ACL）であることを指定します。

access- も deniedaccess- も指定されなかった場合は、両方の ACL からアクセサが削除されます。

calendar(*calendarName*)

アクセス権限の決定用のカレンダを削除します。

className

resourceName が属するクラスの名前を指定します。

deniedaccess-

このコマンドによるアクセサ削除の対象がリソース ACL ではなく NACL（アクセス権限を拒否する ACL）であることを指定します。

gid (*accessor* [,*accessor*]...)

エントリの削除対象である内部グループを 1 つ以上指定します。各 *accessor* はカンマまたはスペースで区切ります。

ghost(*ghostName*)

GHOST クラスのオブジェクトの名前を指定します。

host(*hostName*)

HOST クラスのオブジェクトの名前を指定します。

hostnet(*hostNetName*)

HOSTNET クラスのオブジェクトの名前を指定します。

hostnp(*hostNamePattern*)

HOSTNP クラスに定義されているパターンを指定します。

nt

Windows のシステム ACL から値を削除するかどうかを指定します。

FILE クラスに対してのみ有効です。

resourceName

アクセス制御リストを変更するリソース レコードの名前を指定します。指定できるリソース レコードは 1 つのみです。

service(*serviceName*|*serviceNumber*|*serviceNumberRange*)

ACL から削除するサービスを定義します。

stationName

指定されたクラスに属するレコード名を以下のように指定します。

- **HOST** - 単一の端末の名前
- **GHOST** - ghost コマンドでデータベースに定義されたホスト グループの名前
- **HOSTNET** - IP アドレスのマスク値と一致値で定義されたホスト グループの名前
- **HOSTNP** - 名前パターンによって定義されたホスト グループの名前

解決できないホストについては、IP アドレスの範囲を入力します。

serviceNumber |serviceNumberRange

サービス番号またはサービス番号の範囲を指定します。

範囲は 2 つの整数をハイフン(-)で区切って「1-99」のように指定します。

制限: 指定できる整数の範囲は 0 ～ 65535 です。

uid (*accessor* [,*accessor*]...)

エントリの削除対象である内部ユーザを 1 つ以上指定します。各 *accessor* はカンマまたはスペースで区切ります。

uid(*) と指定すると、すべての内部ユーザを指定することができます。

unix

UNIX のシステム ACL から値を削除するかどうかを指定します。

ACL をサポートする UNIX 環境でのみ有効です。また、FILE クラスのレコードに対してのみ有効です。

xgid (*accessor* [,*accessor*]...)

エントリの削除対象であるエンタープライズ ユーザを 1 つ以上指定します。各 *accessor* はカンマまたはスペースで区切ります。

xuid (*accessor* [,*accessor*]...)

エントリの削除対象であるエンタープライズ グループを 1 つ以上指定します。各 *accessor* はカンマまたはスペースで区切ります。

例: ファイル アクセスのためのグループ権限の削除

以下のコマンドは、リソース /products/new でカバーされているファイルの ACL と NACL の両方からグループ research を削除します。

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

これにより、グループ research は対象となるファイルのデフォルトアクセスが有効となります。

詳細情報:

authorize コマンド - リソースに対するアクセス権限の設定 (P. 51)
authorize コマンド - Windows リソースに対するアクセサのアクセス権限の設定 (P. 208)
authorize- コマンド - Windows リソースに対するアクセサのアクセス権限の削除 (P. 210)
chfile コマンド - ファイル レコードの変更 (P. 65)
ch[x]grp コマンド - グループ プロパティの変更 (P. 72)
chres コマンド - リソース レコードの変更 (P. 89)
ch[x]usr コマンド - ユーザ プロパティの変更 (P. 107)

check コマンド - ユーザのアクセス権限のチェック

AC 環境で有効

check コマンドを使用すると、特定のリソースへのアクセス権限がユーザにあるかどうかをチェックできます。このコマンドは、リソースの ACL およびデフォルトのアクセス プロパティに基づいてアクセス権限を確認します。ただし、このコマンドは PACL をサポートしていません。つまり、ユーザが特定のプログラムを使用してリソースにアクセスできるかどうかはチェックされません。

注: このコマンドは、seos の停止中には使用できません。PACL の詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

このコマンドを使用するには、以下の条件のいずれかを満たす、リソースに対する適切な権限が必要です。

- このコマンドを実行するプロセスに SERVER 属性があること
- ADMIN 属性が割り当てられていること

このコマンドの形式は以下のようになります。

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

access(*authority*)

uid パラメータで指定したアクセサについてチェックするアクセス権限を指定します。

有効な値は、チェック対象のリソースによって異なります。

className

resourceName が属するクラスの名前を指定します。

resourceName

リソース レコードの名前を指定します。

uid(*userName*)

resourceName へのアクセス権限をチェックする対象の CA Access Control ユーザの名前を指定します。

例: リソースへのアクセス権がユーザにあるかどうかのチェック

ユーザ Alain に *file* クラスのリソース *testfile* への write アクセス権があるかどうかを確認するには、以下のコマンドを入力します。

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

以下に示すこのコマンドのサンプル出力は、ユーザ Alain がリソースの所有者であるため、指定したファイルに対する write アクセス権があることを示しています。

```
FILE /testfile へのアクセス GRANTED
ステージ: リソースの OWNER のチェック
```

checklogin コマンド - ログイン情報の取得

AC 環境で有効

checklogin コマンドを使用すると、ユーザのログイン権限、パスワード チェックが必要かどうか、および端末アクセス チェックが必要かどうかをチェックできます。

注: このコマンドは、seos の停止中には使用できません。

このコマンドを使用するには、以下の条件のいずれかを満たす、リソースに対する適切な権限が必要です。

- このコマンドを実行するプロセスに SERVER 属性があること
- ADMIN 属性が割り当てられていること

このコマンドの形式は以下のようになります。

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

password(*password*)

(オプション) パスワード チェックが有効な場合に、オペレーティング システムのパスワードおよびデータベースと照合してチェックするパスワードを指定します。

userName

ログイン権限のチェック対象ユーザの名前を指定します。

terminal(*terminalName*)

(オプション) ログインする権限がユーザにあるかどうかをチェックする端末を指定します。

例: ユーザにログイン権限があるかどうかのチェック

ユーザ Frank に端末 *mutra* から *localhost* にログインする権限があるかどうかをチェックするには、以下のコマンドを入力します。

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

以下のコマンド出力は、ユーザ Frank が端末 mutra からホスト winsome (localhost) にログインできることを示しています。

```
ユーザ frank のホスト winsome へのログインが許可されます。
ステージ: Resource class global universal access
```

ユーザ Frank のパスワードを検証するには、以下のコマンドを入力します。

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

ユーザ Frank のパスワードを CA Access Control データベースのパスワードと照合して検証するには、以下のコマンドを実行します。

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

上記の *so* コマンドにより、パスワード チェックが有効になります。

checkpwd コマンド - パスワードのルール遵守チェック

AC 環境で有効

checkpwd コマンドを使用すると、ユーザのパスワードがパスワード ルールを遵守しているかどうかをチェックできます。このチェックでパスワードは変更されません。

このコマンドを使用するには、ADMIN 属性を持つスーパーユーザである必要があります。

新しいパスワードは、CA Access Control パスワード ルールに従って受け付けられるか拒否されます。

- 新しいパスワードが受け付けられると、以下の成功メッセージが表示されます。

  userName のパスワードの変更が許可されます。

- 新しいパスワードが拒否されると、以下の失敗メッセージが表示されます。

  userName のパスワードの変更が拒否されます。
  denied_reason

  denied_reason は、合格しなかったパスワード ルールです。

  以下に例を示します。

  JDoe のパスワード変更が拒否されます。
  パスワードに含まれる小文字の数が少なすぎます。

denied_reason には、そのパスワードが合格しなかった最初のルールのみ表示されます。たとえば、パスワードが短すぎ、かつ、パスワードに大文字が不足している場合は、「*パスワードが短すぎます。*」とのみ表示されます。

注: このコマンドは、seos の停止中には使用できません。パスワード ルールの詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

このコマンドの形式は以下のようになります。

checkpwd *userName* password(*newPassword*)

userName

チェック対象の新しいパスワードを使用する CA Access Control ユーザの名前を指定します。

password(*newPassword)*

チェックするパスワードを指定します。

chfile コマンド - ファイル レコードの変更

AC 環境で有効

FILE クラスに属するレコードに対する作業には、chfile、editfile、および newfile コマンドを使用します。これらのコマンドは構造が同じですが、以下の点のみ異なっています。

- chfile コマンド*は、FILE クラスに属する 1 つ以上のレコードを変更します。*
- editfile コマンド*は、FILE クラスに属する 1 つ以上のレコードを作成または変更します。*
- newfile コマンド*は、FILE クラスに属する 1 つ以上のレコードを作成します。*

注: このコマンドはネイティブ環境にもありますが、動作が異なります。

FILE クラスに属するファイルのレコードを追加または変更するには、そのファイルに対する適切な権限が必要です。CA Access Control では、ユーザに対し以下の条件がチェックされ、いずれかの条件が満たされるとチェックは終了します。

1. ADMIN 属性が割り当てられていること
2. GROUPADMIN 属性で管理者権限を与えられたグループの有効範囲内に、目的のリソース レコードが含まれていること
3. レコードを変更する場合は、対象レコードの所有者であること
4. ADMIN クラスの FILE レコードの ACL に CREATE アクセス権限（newfile または editfile の場合）または MODIFY アクセス権限（chfile の場合）が割り当てられていること
5. seos.ini ファイルのトークン use_unix_file_owner が yes に設定されている場合は、ファイルの所有者であること（ネイティブ OS に存在する CA Access Control にファイルを定義する場合）。

```
{{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}

ログに記録するアクセス イベントを指定します。アクセス タイプは以下のとおりです。

- **all** - 許可されたアクセスと検出された不正アクセスの試みの両方がログに記録されます。
- **failure** - 検出された不正アクセスの試みがログに記録されます デフォルト値です。
- **none** - レコードは一切ログ ファイルに記録されません。
- **success** - リソースに対して許可されたアクセスを記録します。

注: audit パラメータを指定するには、AUDITOR 属性が必要です。

category(*categoryName*)

ファイルに割り当てる、(CATEGORY クラスに定義されている) セキュリティ カテゴリ レコードのスペースまたはカンマで区切られたリストを定義します。

CATEGORY クラスがアクティブでない場合に category パラメータを指定すると、データベース内のファイルの定義が更新されます。ただし、更新されたカテゴリの割り当ては、CATEGORY クラスを再度アクティブにするまでは有効になりません。

注: セキュリティ カテゴリ チェックの詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

category-(*categoryName*)

リソース レコードから 1 つ以上のセキュリティ カテゴリを削除します。複数のセキュリティ カテゴリを削除する場合は、各セキュリティ カテゴリ名をスペースまたはカンマで区切ります。

指定したセキュリティ カテゴリは、CATEGORY クラスがアクティブかどうかに関係なく、リソース レコードから削除されます。

注: このパラメータは、レコードを変更する場合にのみ有効です。

comment(*string*)

最大 255 文字の英数字から成る文字列をファイル レコードに追加します。文字列に空白が含まれる場合は、文字列を一重引用符で囲みます。以前に定義した既存のコメントがある場合、この文字列に置き換えられます。

comment-

ファイル レコードからコメント文字列を削除します。

注: このパラメータは、レコードを変更する場合にのみ有効です。

defaccess(*accessAuthority*)

ファイルに対するデフォルトのアクセス権限を指定します。デフォルトのアクセス権限は、ファイルのアクセス制御リストに含まれていないアクセサがファイルへのアクセスを要求した場合に与えられる権限です。デフォルトのアクセス権限は、データベースに定義されていないユーザにも適用されます。

fileName

ファイル レコードの名前を指定します。ファイル名は、少なくとも 1 つ指定する必要があります。

汎用ファイル名を使用して FILE クラスにレコードを追加する場合、またはレコードを変更する場合は、selang で許可されているワイルドカード式を使用します。複数のレコードを定義または変更する場合は、ファイル名のリストを丸かっこで囲み、各ファイル名をスペースまたはカンマで区切ります。

注: 複数のファイル名が指定されている場合は、指定されたパラメータに基づいて各ファイル レコードが個別に処理されます。ファイルの処理中にエラーが発生すると、メッセージが発行され、リストの次のファイルから処理が続行されます。

gowner(*groupName*)

ファイル レコードの所有者として CA Access Control グループを割り当てます。ファイル レコードのグループ所有者には、ファイルに対する無制限のアクセス権が与えられます。ただし、前提として、グループ所有者のセキュリティレベル、セキュリティ ラベル、およびセキュリティ カテゴリに、ファイルへのアクセスを許可する適切な権限が設定されている必要があります。ファイルのグループ所有者には、ファイル レコードを更新および削除する許可が常に与えられます。

label(*labelName*)

SECLABEL クラスに定義されているセキュリティ ラベルをファイルに割り当てます。セキュリティ ラベルは、特定のセキュリティ レベルと 0 個以上のセキュリティ カテゴリとの関係を表します。リソース レコードに現在セキュリティ ラベルが含まれている場合、現在のセキュリティ ラベルは、ここで指定したセキュリティ ラベルに置き換えられます。

注: セキュリティ ラベル チェックの詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

label-

ファイル レコードに定義されているセキュリティ ラベルを削除します。

注: このパラメータは、レコードを変更する場合にのみ有効です。

level(*number*)

リソース レコードにセキュリティ レベルを割り当てます。1 ～ 255 の正の整数を入力します。リソース レコードにすでにセキュリティ レベルが割り当てられている場合、既存の値は新しい値に置き換えられます。

注: セキュリティ レベル チェックの詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

level-

CA Access Control によるリソースのセキュリティ レベル チェックを停止します。

注: このパラメータは、レコードを変更する場合にのみ有効です。

notify(*mailAddress*)

リソース レコードが示すファイルへのアクセスが成功するたびに通知メッセージを送信するよう CA Access Control に指示します。ユーザ名またはユーザの電子メール アドレスを入力します。また、別名が指定されている場合は、メール グループの電子メール アドレスも入力できます。

通知は、ログ ルーティング システムがアクティブな場合にのみ行われます。通知メッセージは、ログ ルーティング システムの設定に基づいて、ユーザの画面またはメールボックスに送信されます。

通知メッセージが送信されるたびに、監査ログに監査レコードが書き込まれます。

通知メッセージの受信者は、頻繁にログインして、各メッセージに示された不正アクセスの試みに対処する必要があります。

制限: 30 文字。

注: 監査レコードのフィルタ処理と表示の詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

notify-

レコードが示すファイルへのアクセスを CA Access Control が許可する際に誰にも通知しないように指定します。

注: このパラメータは、レコードを変更する場合にのみ有効です。

owner(*Name*)

ファイル レコードの所有者として CA Access Control ユーザまたはグループを割り当てます。ファイル レコードの所有者には、ファイルに対する無制限のアクセス権が与えられます。ただし、前提として、所有者のセキュリティ レベル、セキュリティ ラベル、およびセキュリティ カテゴリに、ファイルへのアクセスを許可する適切な権限が設定されている必要があります。ファイルの所有者には、ファイル レコードを更新および削除する許可が常に与えられます。

restrictions(days(*dayData*) time(*timeData*))

ユーザがファイルにアクセスできる曜日と時間帯を指定します。

days 引数を指定せずに time 引数を指定した場合、レコード内にすでに設定されている曜日制限に対して、指定した時間帯制限が適用されます。
time 引数を指定せずに days 引数を指定した場合、レコード内にすでに設定されている時間帯制限に対して、指定した曜日制限が適用されます。
days 引数と time 引数の両方を指定した場合、指定した曜日の指定した時間帯にのみユーザはシステムにアクセスできます。

days(*dayData*)

ユーザがファイルにアクセスできる曜日を指定します。days 引数には次のサブ引数があります。

- **anyday** - ユーザは曜日を問わずファイルにアクセスできます。
- **weekdays** - ユーザは月曜から金曜までの平日に限りリソースにアクセスできます。
- **mon**、**tue**、**wed**、**thu**、**fri**、**sat**、**sun** - 指定した曜日にのみリソースにアクセスできます。曜日は任意の順で指定できます。複数の曜日を指定する場合は、各曜日をスペースまたはカンマで区切ります。

time(*timeData*)

ユーザがファイルにアクセスできる時間帯を指定します。time 引数には次のサブ引数があります。

- **anytime** - 特定の曜日の任意の時間帯にリソースにアクセスできます。
- *startTime*:*endTime* - 指定した時間帯にのみリソースにアクセスできます。*startTime* および *endTime* は両方とも *hhmm* の形式で指定します。*hh* は 24 時間表記の時間（00 から 23）、*mm* は分（00 から 59）を表します。2400 は有効な time 値ではないことに注意してください。*startTime* が *endTime* より小さいこと、および両方が同じ日の時間であることが必要です。端末がホストと異なるタイム ゾーンにある場合は、端末の開始時間と終了時間をホストのローカル時間に相当する時間に変換し、時間の値を調整してください。たとえば、ホストがニューヨークにあり、端末がロサンゼルスにある場合、ロサンゼルスの端末からのアクセスを午前 8 時から午後 5 時まで許可するには、「time(1100:2000)」と指定します。

restrictions-

ファイルに対するアクセス権限を限定するすべての曜日および時間帯の制限を削除します。

注: このパラメータは、レコードを変更する場合にのみ有効です。

warning

ファイルを警告モードにします。

warning-

ファイルの警告モードを解除します。

例: スーパーユーザ以外のすべてのユーザに対し、ファイルへのアクセスを制限

スーパーユーザ以外のすべてのユーザに対する/etc/passwd ファイルへのアクセスを READ アクセスに制限するには、以下のコマンドを入力します。

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

以下の条件が満たされている必要があります。

- ADMIN 属性が割り当てられていること
- データベースに /etc/passwd レコードが定義されていること
- /etc/passwd レコードの ACL にエントリがないこと

例: 時間を指定してファイルへのアクセスを制限

/home/bob/secrets ファイルへのアクセスを防ぎ、所有者によるアクセスを平日の 08:00 ～ 18:00 に制限するには、以下のコマンドを入力します。

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

以下の条件が満たされている必要があります。

- ADMIN 属性が割り当てられていること
- Bob が CA Access Control ユーザであり、FILE クラスの /home/ bob/secrets レコードの所有者であること

例: ホーム ディレクトリへのアクセスの制限

自分以外のユーザがホーム ディレクトリ(/home/bob)のどのファイルにもアクセスできないようにするには、UNIX で以下のコマンドを入力します。

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

同じことを Windows では以下のコマンドで実行できます。

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

以下の条件が満たされている必要があります。

- 自分自身が CA Access Control に定義されていること
- ファイルのネイティブ所有者であること

詳細情報:

authorize コマンド - リソースに対するアクセス権限の設定 (P. 51)
rmfile コマンド - ファイル レコードの削除 (P. 150)
showfile コマンド - ファイルのプロパティの表示 (P. 168)
chfile コマンド - UNIX ファイル設定の変更 (P. 192)
chfile コマンド - Windows ファイル設定の変更 (P. 212)
クラス別アクセス権限 (P. 32)

ch[x]grp コマンド - グループ プロパティの変更

AC 環境で有効

chgrp、chxgrp、editgrp、editxgrp、newgrp、および newxgrp の各コマンドは、グループのプロパティを変更するため、および必要に応じて CA Access Control データベースにグループを作成するために使用します。

各コマンドには以下のような省略形があります。

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

これらのコマンドの構造は同じで、機能だけが以下のように異なっています。

- GROUP クラスのレコードに対する作業には、chgrp、editgrp、および newgrp コマンドを使用します。これらのコマンドを使用すると、エンタープライズ ユーザ ストアを参照せずに CA Access Control グループを作成または変更できます。これらのコマンド間の相違点は以下のとおりです。
  - chgrp コマンド*は、GROUP クラスに属する 1 つ以上のレコードを変更します。*
  - editgrp コマンド*は、GROUP クラスに属する 1 つ以上のレコードを作成または変更します。*
  - newgrp コマンド*は、GROUP クラスに属する 1 つ以上のレコードを作成します。*

  注: このコマンドはネイティブ環境にもありますが、動作が異なります。
- XGROUP クラスのレコードに対する作業には、chxgrp、editxgrp、および newxgrp コマンドを使用します。これらのコマンドを使用すると、エンタープライズ ユーザ ストア定義されている CA Access Control グループを作成または変更できます。これらのコマンド間の相違点は以下のとおりです。
  - chxgrp コマンド*は、XGROUP クラスに属する 1 つ以上のレコードを変更します。*
  - editxgrp コマンド*は、XGROUP クラスに属する 1 つ以上のレコードを作成または変更します。*
  - newxgrp コマンド*は、XGROUP クラスに属する 1 つ以上のレコードを作成します。*

必要な権限

新しい CA Access Control グループを作成するには、以下の条件が少なくとも 1 つ満たされている必要があります。

- ADMIN 属性が割り当てられていること
- ADMIN クラスの GROUP または XGROUP レコードのアクセス制御リストに CREATE アクセス権が割り当てられていること

グループを追加または変更するには、以下の条件が少なくとも 1 つ満たされている必要があります。

- ADMIN 属性が割り当てられていること
- GROUP-ADMIN 属性で管理者権限を与えられたグループの適用範囲にグループ レコードが含まれていること
- グループの所有者であること
- ADMIN クラスの GROUP または XGROUP レコードのアクセス制御リストに MODIFY アクセス権（ch[x]grp の場合）または CREATE アクセス権（edit[x]grp の場合）が割り当てられていること

```
{{chgrp|cg}|{chxgrp|cxg}|{editgrp|eg}|{editxgrp|exg}|{newgrp|ng}|{newxgrp|nxg}}
groupName ...
    [{admin | admin-}] ¥
    [audit(none|all|success|failure|loginsuccess|loginfail|trace|interactive)|aud
    it-] ¥
    [{auditor | auditor-}] ¥
    [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))]
    [{operator | operator-}] ¥
    [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-] ¥
[{pwmanager | pwmanager-}] ¥
[restrictions( ¥
    [days({anyday|weekdays|{[mon] [tue] [wed] ¥
            [thu] [fri] [sat] [sun]}})] ¥
    [time(anytime|startTime:endTime) ¥
|restrictions-] ¥
[resume[(mm/dd/yy[yy][@hh:mm])]|resume-] ¥
[{server | server-}] ¥
[shellprog(fullPath)] ¥
[supgroup(superiorGroup)|supgroup-] ¥
[suspend[(mm/dd/yy[yy][@hh:mm])]|suspend-] ¥
[unix[( ¥
    [appl(quotedString)] ¥
    [groupid(groupidNumber)] ¥
    [userlist(userName...)] ¥
)]] ¥
```

文字列でプロパティが定義されているレコード プロパティを削除するには、プロパティに続けて、- (マイナス記号) または () (空の丸かっこ) を入力します。

注: 一部のパラメータは、グループがプロファイル グループとして機能する場合のみ有効です。プロファイル グループはエンタープライズ グループにはなれません。

admin

グループに ADMIN 属性を割り当てます。ADMIN 属性を持つグループのメンバであるユーザは、audit パラメータ以外のすべてのパラメータを使用して selang のすべてのコマンドを発行できます。admin パラメータを使用するには ADMIN 属性が必要です。

admin-

グループから ADMIN 属性を削除します。(CA Access Control は少なくとも 1 人のユーザが ADMIN 属性を持つようにします)。

このパラメータは、new[x]grp コマンドでは使用できません。

audit(*mode*)

このコマンドのトレース監査を有効にします。監査モードには、none、all、success、failure、loginsuccess、loginfail、trace、および interactive があります。

audit-

このコマンドのトレース監査を無効にします。

auditor

グループに AUDITOR 属性を割り当てます。AUDITOR 属性を持つグループのメンバであるユーザは、システム リソースの使用状況を監査できます。また、CA Access Control の権限チェックで検出された CA Access Control の保護対象であるすべてのリソースへのアクセス、およびデータベースへのアクセスに対するログの記録を制御できます。AUDITOR 属性を持つユーザに与えられる権限の詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

auditor-

グループ レコードから AUDITOR 属性を削除します。

このパラメータは、new[x]grp コマンドでは使用できません。

comment(*string*)

最大 255 文字の英数字(シングル バイト文字)から成るコメント文字列をグループ レコードに追加します。文字列にスペースが含まれる場合は、文字列全体を一重引用符で囲みます。以前に追加した既存の文字列がある場合、この文字列に置き換えられます。

注: ドイツ語の場合は、128 文字しか記録されません。

comment-

グループ レコードからコメント文字列（ある場合）を削除します。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ使用できます。

expire(*date*)

グループ メンバのアカウントが失効する日付を設定します。date を指定しなかった場合、現在ログインしていないユーザのユーザ アカウントはただちに失効します。ユーザがログインしていた場合、アカウントはユーザがログアウトすると失効します。このパラメータは、プロファイル グループにのみ適用されます。

失効の日付と時刻は、以下の形式で指定します。時刻は省略可能です。*mm/dd/yy [yy][@HH:MM]* 年は、下 2 桁または 4 桁のどちらでも指定できます。

注: 失効したユーザ レコードは、resume パラメータに再開日を指定しても有効にできません。失効したユーザ レコードを有効にするには、expire- パラメータを使用します。

expire-

newgrp コマンドの場合は、有効期限のないユーザ アカウントを定義します。chgrp コマンドおよび editgrp コマンドの場合は、ユーザ アカウントから有効期限を削除します。このパラメータは、プロファイル グループにのみ適用されます。

gowner(*groupName*)

グループ レコードの所有者として CA Access Control ユーザまたはグループを割り当てます。複数のグループ名を指定する場合は、グループ名を丸かっこで囲み、各グループ名をスペースまたはカンマで区切ります。このパラメータを省略した場合、データベースにグループを追加したユーザがグループ レコードの所有者になります。

grace(*numberOfGraceLogins*)

ユーザのアカウントが一時停止になるまでにログインできる最大回数を設定します。猶予ログイン回数には、0 ～ 255 の値を指定する必要があります。猶予ログイン回数に達すると、ユーザはシステムへのアクセスを拒否されるため、システム管理者に連絡して新しいパスワードを設定する必要があります。猶予回数が 0 に設定されている場合、ユーザはログインできません。このパラメータは、プロファイル グループにのみ適用されます。

grace-

グループの猶予ログイン設定を削除します。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ使用できます。このパラメータは、プロファイル グループにのみ適用されます。

groupName

作成するグループの名前またはプロパティの変更対象のグループの名前を指定します。new[x]grp コマンドの場合、データベースに存在しない一意なグループ名を指定する必要があります。ただし、グループとユーザには重複する名前を使用できます。

history

保存するパスワードの数を指定します。history- を使用して履歴ファイルを削除できます。

homedir(*fullPath*|nohomedir)

ユーザのホーム ディレクトリの完全パスを指定します。指定するパスの末尾にスラッシュを指定すると、*groupName* が指定されたパスに追加されます。nohomedir を指定すると、ホーム ディレクトリは自動的には設定されません。

inactive(*numInactiveDays*)

ユーザのステータスがシステムによって非アクティブに変更されるまでの必要経過日数を指定します。指定した日数が経過すると、ユーザはログインできなくなります。このパラメータは、プロファイル グループにのみ適用されます。

numInactiveDays には正の整数または 0 を入力します。inactive を 0 に設定すると、inactive- パラメータを使用した場合と同じ結果になります。

注: ユーザ レコードには、アクティブでないユーザのマークが設定されません。アクティブでないユーザを識別するには、Inactive Days 値と Last Accessed Time 値を比較する必要があります。

inactive-

ユーザのステータスを非アクティブからアクティブに変更します。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ使用できます。このパラメータは、プロファイル グループにのみ適用されます。

interval(*maximumPasswordChangeInterval*)

パスワードの設定または変更後、ユーザに対して新しいパスワードの入力を促すメッセージを表示するまでの経過日数を設定します。正の整数または 0 を入力します。interval に 0 を設定すると、グループに対するパスワード期間のチェックが無効になり、パスワードが失効しません。setoptions コマンドで設定したデフォルト値は使用されません。interval を 0 に設定するのは、セキュリティ要件が厳しくないユーザに限定してください。

指定した日数が経過すると、CA Access Control は、現在のパスワードが期限切れになったことをユーザに通知します。通知を受けたユーザは、ただちにパスワードを更新するか、猶予ログイン回数に達するまで古いパスワードを引き続き使用することができます。猶予ログイン回数に達するとユーザはシステムへのアクセスを拒否されるため、システム管理者に連絡して新しいパスワードを設定する必要があります。このパラメータは、プロファイル グループにのみ適用されます。

間隔-

グループに対するパスワード期間の設定を取り消します。この設定を取り消すと、ユーザ レコードの任意の値が使用されます。それ以外の場合は、setoptions コマンドで設定したデフォルト値が使用されます。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ入力できます。このパラメータは、プロファイル グループにのみ適用されます。

maxlogins(*maximumNumberOfLogins*)

ユーザが同時にログインできる端末の最大数を設定します。値 0（ゼロ）は、ユーザが任意の数の端末から同時にログインできることを意味します。このパラメータを指定しない場合は、ユーザ レコードの任意の値が使用されます。それ以外の場合は、グローバルに設定されているログインの最大数が使用されます。このパラメータは、プロファイル グループにのみ適用されます。

注: maxlogins を 1 に設定すると、selang を実行できません。この場合、CA Access Control を停止し、maxlogins の設定を 2 以上の値に変更し、CA Access Control を再起動する必要があります。

maxlogins-

グループの最大ログイン数の設定を削除します。このパラメータを指定しない場合は、ユーザ レコードの任意の値が使用されます。それ以外の場合は、グローバルに設定されているログインの最大数が使用されます。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ使用できます。このパラメータは、プロファイル グループにのみ適用されます。

mem(*GroupName*) | mem+(*GroupName*)

CA Access Control のグループにメンバ グループ（子グループ）を追加します。メンバ グループ（*GroupName*）は、CA Access Control にあらかじめに定義しておく必要があります。複数のメンバ グループを追加する場合は、各グループ名をカンマで区切ります。グループ名にスペースが含まれている場合は、一重引用符で囲みます。

注: 内部グループにユーザを追加するには、join[x] コマンドを使用します。

このオプションは、内部グループにのみ適用されます。

mem-(*GroupName*)

指定のグループからメンバ グループを削除します。メンバ グループ（*GroupName*）は、CA Access Control にあらかじめに定義しておく必要があります。複数のメンバ グループを削除する場合は、各グループ名をカンマで区切ります。グループ名にスペースが含まれている場合は、一重引用符で囲みます。

注: 内部グループからユーザを削除するには、join[x]- コマンドを使用します。

このオプションは、内部グループにのみ適用されます。

min_life(*minimumPasswordChangeInterval*)

ユーザが再びパスワードを変更できるようになるまでの最短経過日数を指定します。このパラメータは、プロファイル グループにのみ適用されます。

min_life-

グループの min_life 設定を削除します。min_life- パラメータが指定されてなく、min_life パラメータがユーザ レコードに設定されている場合は、ユーザ レコードの値が使用されます。それ以外の場合は、グローバルに設定されている min_life が使用されます。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ使用できます。このパラメータは、プロファイル グループにのみ適用されます。

name(*fullname*)

グループのフル ネームを指定します。最大 47 文字の英数字から成る文字列を入力します。文字列に空白が含まれる場合は、文字列を一重引用符で囲みます。

nt(*nt-group-attributes*)

(Windows のみ)ローカル Windows システムにグループ定義を追加するか、ローカル Windows システムのグループ定義を変更します。

comment('*comment*')

コメント文字列をネイティブ レコードに追加します。レコードにすでにコメント文字列が追加されている場合、既存の文字列はここで指定した新しい文字列で置き換えられます。

comment は、最大 255 文字の英数字から成る文字列です。文字列に空白が含まれる場合は、文字列全体を一重引用符で囲みます。

operator

グループに OPERATOR 属性を割り当てます。OPERATOR 属性を持つグループのメンバであるユーザは、データベースのすべてのリソース レコードを一覧表示できます。また、このユーザには CA Access Control で定義されたすべてのファイルに対する読み取り権限が与えられます。

この属性をもつグループのメンバであるユーザは、secons コマンドのオプションをすべて使用することもできます。secons ユーティリティの詳細については、「*リファレンス ガイド*」を参照してください。

operator-

グループ レコードから OPERATOR 属性を削除します。

このパラメータは、new[x]grp コマンドでは使用できません。

owner(*Name*)

グループ レコードの所有者として CA Access Control ユーザまたはグループを割り当てます。このパラメータを省略した場合、データベースにグループを追加したユーザが所有者になります。詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

parent(*groupName*)

既存の CA Access Control グループをグループ レコードの親グループとして割り当てます。親子関係の詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

parent-

グループとその親グループの間のリンクを削除します。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ使用できます。

password

指定されたグループにパスワードを割り当てます。

password-

このグループのパスワードの入力を不要にします。

pmdb(*PolicyModelName*)

グループ内のユーザが sepass ユーティリティを使用してパスワードを変更した場合に、指定された Policy Model に新しいパスワードを伝達するように指定します。PMDB の完全修飾名を入力します。

パスワードは、seos.ini の [seos] セクションの parent_pmd トークンまたは passwd_pmd トークンに定義されている Policy Model には送信されません。このパラメータは、プロファイル グループにのみ適用されます。

pmdb-

グループ レコードから PMDB 属性を削除します。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ使用できます。このパラメータは、プロファイル グループにのみ適用されます。

pwmanager

グループに PWMANAGER 属性を割り当てます。この属性をもつグループのメンバであるユーザは、データベース内のユーザのパスワードを変更できます。詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

pwmanager-

グループ レコードから PWMANAGER 属性を削除します。

このパラメータは、new[x]grp コマンドでは使用できません。

restrictions(days(*dayData*) time(*timeData*))

グループのメンバがシステムにログインできる曜日と時間帯を指定します。

ユーザがログイン中にログイン期間が過ぎたとしても、CA Access Control がユーザをシステムから強制ログオフすることはありません。また、このログイン制限はバッチ ジョブには適用されません。ユーザはいつでもバックグラウンド プロセスを実行することができます。このパラメータは、プロファイル グループにのみ適用されます。

days 引数を指定せずに time 引数を指定した場合、レコード内にすでに設定されている曜日制限に対して、指定した時間帯制限が適用されます。time 引数を指定せずに days 引数を指定した場合、レコード内にすでに設定されている時間帯制限に対して、指定した曜日制限が適用されます。days 引数と time 引数の両方を指定した場合、指定した曜日の指定した時間帯にのみグループのメンバはシステムへのアクセスを許可されます。

days(*dayData*)

ユーザがシステムにログインできる曜日を指定します。days 引数には次のサブ引数があります。

- **anyday** - ユーザは曜日を問わずログインできます。
- **weekdays** - ユーザは月曜日から金曜日までの平日に限りログインできます。
- **mon**、**tue**、**wed**、**thu**、**fri**、**sat**、**sun** - ユーザは指定した曜日にのみログインできます。曜日は任意の順で指定できます。複数の曜日を指定する場合は、各曜日をスペースまたはカンマで区切ります。

time(*timeData***)**

ユーザがシステムにログインできる時間帯を指定します。time 引数には次のサブ引数があります。

- **anytime** - ユーザは特定の曜日の任意の時間帯にログインできます。
- *startTime*:*endTime* - ユーザは指定した時間帯にのみログインできます。*startTime* および *endTime* は両方とも *hhmm* の形式で指定します。*hh* は 24 時間表記の時間（00 から 23）、*mm* は分（00 から 59）を表します。2400 は有効な time 値ではないことに注意してください。*endTime* の値が *startTime* の値より小さい場合、時間帯の終了時刻は翌日の時刻と見なされます。それ以外の場合、指定した時間帯は同じ日の時間であると見なされます。

  注: CA Access Control では、プロセッサのタイム ゾーンを使用します。プロセッサと異なるタイム ゾーンの端末にログインする際には注意が必要です。

restrictions-

システムにログインするユーザの権限を限定する、すべての曜日および時間帯の制限を、グループ レコードから削除します。restrictions- パラメータが指定されてなく、restrictions パラメータがユーザ レコードに設定されている場合は、ユーザ レコードの値が使用されます。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ使用できます。このパラメータは、プロファイル グループにのみ適用されます。

resume(*date*)

suspend パラメータを指定して無効にしたユーザ レコードを有効にします。日付と時刻は、*mm/dd/yy[@HH:MM]* 形式で指定します。時刻は省略可能です。

suspend パラメータと resume パラメータの両方を指定する場合、再開日を一時停止日より後に設定する必要があります。*date* を省略すると、chgrp コマンドの実行直後にユーザが有効になります。詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。このパラメータは、プロファイル グループにのみ適用されます。

resume-

再開日および再開時間（指定されている場合）をグループ レコードから消去します。これにより、ユーザのステータスがアクティブ（有効）から一時停止に変更されます。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ使用できます。このパラメータは、プロファイル グループにのみ適用されます。

rules

パスワードのルールを以下のように指定します。

alpha(*minimumAlphaCharacters*)

最低英文字数です。

alphanum(*minimumAlphanumericCharacters*)

最低文字数です。

bidirectional|bidirectional-

双方向パスワード暗号化を使用するかどうかを指定します。双方向パスワード暗号化が有効の場合、パスワードは新しくなるたび暗号化され、解読してクリア テキストに戻すことができます。この暗号化により、新しいパスワードと古いパスワードを幅広く比較できるようになります(パスワード履歴)。双方向パスワード暗号化が無効の場合、一方向パスワード履歴暗号化が有効になり、古いパスワードを解読することはできなくなります。

注: この機能を使用するには、history を 1 より大きい値に設定する必要があります。

注: UNIX でこの機能を使用する場合は、passwd_format 環境設定を NT に設定する必要もあります。

重要: seos.ini ファイルのトークン「passwd_format」([passwd] セクション)を「NT」に設定している場合、selang でユーザを作成するには、「native」オプション(「unix」ではなく)を使用する必要があります。例:

```
nu uSr_1026 native password(uSr_1026)
```

または、以下のように、作業環境がネイティブ環境(UNIX 環境ではなく)であることを確認します。

```
env native
chusr usr_1 password(mypassword)
```

min_len(*minimumPasswordLength*)

パスワードの最小文字数です。

max_len(*maximumPasswordLength*)

パスワードの最大文字数です。

lowercase(*minimumLowercaseCharacters*)

小文字の最低数です。

max_rep(*maximumRepetitiveCharacters*)

文字の繰り返しの最大数です。

namechk|namechk-

パスワードと名前を照合して確認します。

numeric(*minimumNumericCharacters*)

数字の最低数です。

oldpwchk|oldpwchk-

パスワードと古いパスワードを照合して確認します。

注: Unix と Linux のオペレーティング システム上でのみ有効です。

special(*minimumSpecialCharacters*)

特殊文字の最低数です。

uppercase(*minimumUppercaseCharacters*)

大文字の最低数です。

use_dbdict|use_dbdict-

パスワード辞書を設定します。use_dbdict はトークンを **db** に設定し、パスワードを CA Access Control データベースの単語と照合して比較します。use_dbdict- トークンを **file** に設定し、UNIX の場合は seos.ini ファイル、Windows の場合は Windows レジストリに指定されたファイルとパスワードを照合して比較します。

server

SERVER 属性を設定します。現在のユーザが SERVER 属性を持つグループのメンバである場合、現在のユーザの名前で実行されているプロセスによって他のユーザの権限を確認することができるようにします。 詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

server-

SERVER 属性の設定を解除します。

このパラメータは、new[x]grp コマンドでは使用できません。

shellprog(fullPath)

ユーザが login コマンドまたは su コマンドを起動した後に実行される初期プログラムまたはシェルの完全パスを指定します。*FullPath* は文字列です。

supgroup(*Group'sSuperiorGroup*)

スーパーグループ（親グループ）を指定します。

suspend(*date*)

ユーザ レコードを無効にします。ただし、データベースには定義を残します。日付と時刻は、*mm/dd/yy[@HH:MM]* 形式で指定します。時刻は省略可能です。

ユーザは一時停止されたユーザ アカウントを使用してシステムにログインすることはできません。*date* を指定すると、指定した日にユーザ レコードが一時停止されます。*date* を省略すると、chgrp コマンドの実行直後にユーザ レコードが一時停止されます。このパラメータは、プロファイル グループにのみ適用されます。

suspend-

一時停止日をユーザ レコードから消去し、ユーザのステータスを無効からアクティブ（有効）に変更します。このパラメータは chgrp コマンドまたは editgrp コマンドにのみ使用できます。このパラメータは、プロファイル グループにのみ適用されます。

unix(*groupidNumber*)

（UNIX のみ）UNIX のグループ属性を設定するか、グループがまだ存在していない場合はグループを作成します。

groupidNumber は 10 進数です。グループ ID に 0 を指定することはできません。この数値を省略すると、その時点で最大のグループ ID が検出され、その値がグループの ID として設定されます。一度に複数のグループを追加または変更する場合も、同様の方法でグループ ID の番号が生成されます。seos.ini ファイルのトークン AllowedGidRange を使用して、特定の番号を利用できないようにすることができます。

userlist*(userName)*

グループにメンバを割り当てます。*userName* は、1 人以上の UNIX ユーザのユーザ名を表します。複数のユーザを割り当てる場合は、各ユーザ名をスペースまたはカンマで区切ります。chgrp コマンドまたは editgrp コマンドで使用する場合、グループにすでに定義されているメンバ リストはすべて、ここで指定したメンバ リストに置き換えられます。

例

- ユーザ Bob が、エンタープライズ グループ Sales の親グループおよび Sales グループを所有するグループを、ACCOUNTS から PAYROLL に変更します。

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

- ユーザ Admin1 がグループ projectB の親を divisionA から divisionB に変更し、新しい所有者としてグループ RESEARCH を指定します。

  Admin1 には ADMIN 属性があるとします。

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

- admin ユーザ Sally が、グループ プロファイル NewEmployee に対して、ホーム ディレクトリとシェル プログラムの指定を削除する操作を実行します。

  Sally は NewEmployee の所有者だとします。

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

- ユーザ Admin1 が、グループ ProjectA を、グループ RESEARCH の子グループとして追加します。ユーザ Admin1 がグループ ProjectA の所有者になります。

  Admin1 には ADMIN 属性があるとします。

  デフォルトは owner(Admin1) です。

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

詳細情報:

join[x] コマンド - ユーザの内部グループへの追加 (P. 142)
join[x]- コマンド - ユーザのグループからの削除 (P. 146)
rm[x]grp コマンド - グループ レコードの削除 (P. 151)
show[x]grp コマンド - グループ プロパティの表示 (P. 170)
chgrp コマンド - UNIX グループの変更 (P. 194)
chgrp コマンド - Windows グループの変更 (P. 213)

chres コマンド - リソース レコードの変更

AC 環境で有効

chres、editres、および newres コマンドを使用して、CA Access Control クラスに属するリソース レコードに対する作業を行います。これらのコマンドは構造が同じですが、以下の点のみ異なっています。

- chres コマンド*は、1 つ以上のリソースを*変更します。
- editres コマンド*は、1 つ以上のリソースを*作成または変更します。
- newres コマンド*は、1 つ以上のリソースを*作成します。

注: このコマンドはネイティブ Windows 環境にもありますが、動作が異なります。

newres コマンドを使用してリソースを追加するには、以下の条件が少なくとも 1 つ満たされている必要があります。

- ADMIN 属性が割り当てられていること
- ADMIN クラスにあるリソース クラスのレコードの ACL に CREATE アクセス権限が設定されていること
- seos.ini ファイルのトークン use_unix_file_owner が yes に設定されている場合、UNIX のファイルの所有者がそのファイルを新しいリソースとして CA Access Control に定義できること

chres または editres コマンドを使用してリソースを追加または変更するには、リソースに対する適切な権限が必要です。CA Access Control では、*以下の条件*をこの順序でチェックします。

1. ADMIN 属性が割り当てられていること
2. GROUPADMIN 属性で管理者権限を与えられたグループの有効範囲内に、目的のリソース レコードが含まれていること
3. レコードの所有者であること
4. ADMIN クラスにあるリソース クラスのレコードのアクセス制御リストに MODIFY アクセス権限（chres の場合）または CREATE アクセス権限（editres の場合）が割り当てられていること

注: リソース名の最大文字数は、シングル バイト文字で 255 文字です。

次の表は、chres、editres、および newres コマンドを使用して管理できる各クラスに対して使用できるコマンド パラメータの一覧です。

クラス	Properties											
	audit	calendar	category	comment	defaccess	label	level	notify	owner	restrictions[-]	warning	その他
ACVAR				X					X			VARIABLE_TYPE、VARIABLE_VALUE
ADMIN	X	X	X	X	X	X	X	X	X	X	X	
CALENDAR				X					X			
カテゴリ				X					X			
CONNECT	X	X	X	X	X	X	X	X	X	X	X	
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

クラス	Properties											
	audit	calendar	category	comment	defaccess	label	level	notify	owner	restrictions[-]	warning	その他
HOSTNP	X	X		X					X	X	X	
LOGINAPPL	X	X		X	X			X	X	X	X	LOGINFLAGS, LOGINMETHOD, 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、FLAGS
SECLABEL			X	X			X		X			
SEOS		X	X	X		X	X					HOST
SPECIALPGM				X					X			

クラス	Properties											
	audit	calendar	category	comment	defaccess	label	level	notify	owner	restrictions[-]	warning	その他
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			
USER-ATTR									X		X	
USER-DIR	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(time-period)] ¥
    [dh_dr{-|+}(dh_dr)] ¥
    [disable|disable-] ¥
    [defaccess(accessAuthority)] ¥
    [filepath(filePaths)] ¥
    [flags[-|+](flagName)] ¥
    [gacc(access-value)] ¥
    [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*)

ローカル CA Access Control データベースおよび DMS に保存されるエンドポイント（HNODE オブジェクト）の一意の ID を定義します。CA Access Control ではこの ID を使用して HNODE を識別し、エンドポイントの IP アドレスや名前の変更が拡張ポリシー管理機能に影響しないようにします。CA Access Control によるエンドポイントのトレースは引き続き可能です。

audit

ログに記録するアクセス イベントを指定します。以下のいずれかの属性を指定します。

- **all** - 許可されたアクセスと不正アクセスの試みの両方がログに記録されます。
- **failure** - 不正アクセスの試みがログに記録されます デフォルト値です。
- **none** - レコードは一切ログ ファイルに記録されません。
- **success** - 許可されたアクセスの試みがログに記録されます

calendar(*calendarName*)

Unicenter TNG の時間帯制限を表す Unicenter NSM カレンダ レコードを指定します。CA Access Control では、これらのオブジェクトのリストを管理目的にのみ使用し、オブジェクトの保護は行いません。複数のカレンダを割り当てる場合は、各カレンダ名をスペースまたはカンマで区切ります。

calendar-(*calendarName*)

リソース レコードから 1 つ以上の Unicenter NSM カレンダ レコードを削除します。このパラメータは chres コマンドまたは editres コマンドでのみ使用できます。

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

リソース レコードに 1 つ以上のセキュリティ カテゴリを割り当てます。

CATEGORY クラスがアクティブでない場合に category パラメータを指定すると、データベース内のリソースの定義が更新されます。ただし、更新されたカテゴリの割り当ては、CATEGORY クラスが再度アクティブになるまでは有効になりません。

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

リソース レコードから 1 つ以上のセキュリティ カテゴリを削除します。

指定したセキュリティ カテゴリは、CATEGORY クラスがアクティブかどうかに関係なく、リソース レコードから削除されます。このパラメータは chres コマンドまたは editres コマンドにのみ使用できます。

className

リソースが属するクラスの名前を指定します。CA Access Control に定義されているリソース クラスを一覧表示するには、find コマンドを実行します。

cmd+(*selang_command_string*)

ポリシーを定義する selang コマンドのリストを指定します。これが、ポリシーのデプロイに使用するコマンドです。例を以下に示します。

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

cmd-

ポリシー デプロイ コマンド リストを RULESET オブジェクトから削除します。

comment(*string*)

最大 255 文字の英数字から成る文字列をリソース レコードに追加します。文字列に空白が含まれる場合は、文字列全体を一重引用符で囲みます。以前に定義した既存の文字列は、この文字列に置き換えられます。

注: SUDO クラスの場合、この文字列は特別な意味を持ちます。SUDO レコードの定義の詳細については、「*CA Access Control for UNIX エンドポイント管理ガイド*」を参照してください。

comment-

リソース レコードからコメントを削除します。このパラメータは chres コマンドまたは editres コマンドにのみ使用できます。

container(*containerName*)

CONTAINER（汎用グループ化クラス）のオブジェクトを表します。

containerName は、CONTAINER クラスに定義された 1 つ以上の CONTAINER クラスのレコードの名前です。CONTAINER クラスのレコードを複数割り当てる場合は、名前をスペースまたはカンマで区切ります。

container-(*containerName*)

リソース レコードから 1 つ以上の CONTAINER クラスのレコードを削除します。このパラメータは chres コマンドまたは editres コマンドでのみ使用できます。

dates(*time-period*)

休日などユーザがログインできない期間を 1 つ以上定義します。複数の期間を指定する場合は、各期間をスペースで区切ります。以下の形式を使用します。

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

特定の年を指定しない場合、または 1990 年より前の年を指定した場合、期間または休日は毎年適用されると見なされます。年は、98 または 1998 のように、2 桁または 4 桁で指定できます。

開始時刻を指定しない場合、その日の開始時刻（午前 0 時）が使用されます。終了時刻を指定しない場合、その日の終了時刻（午前 0 時）が使用されます。時間および分の形式は *hh:mm* で指定します。*hh* は 24 時間表記の時間（00 から 23）、*mm* は分（00 から 59）を表します。

時間（例：12/25@14:00-12/25@17:00）を指定せずに、月と日のみ（12/25）を指定すると、その日 1 日が休日と見なされます。

休日を迎えるタイム ゾーンとは異なるタイム ゾーンでコマンドを発行する場合は、指定する期間をユーザのローカル時間に変換します。たとえば、ニューヨークにいて、ロサンゼルスが半日の休日となる場合、「09/14/98@18:00-09/14/98@20:00」と入力する必要があります。このように指定すると、ロサンゼルスにいるユーザは午後 3 時から午後 5 時までの間ログインできなくなります。

defaccess([*accessAuthority*])

指定したリソースのデフォルトのアクセス権限を指定します。デフォルトのアクセス権限とは、リソースのアクセス制御リストに含まれていないアクセサがリソースへのアクセスを要求した場合に与えられる権限です。デフォルトのアクセス権限は、データベースに定義されていないユーザにも適用されます。有効なアクセス権限値はクラスによって異なります。

accessAuthority を省略すると、CA Access Control では、UACC クラスにあるリソースのクラスを表すレコードの UACC プロパティに指定された、暗黙のアクセス権が割り当てられます。

dh_dr{+|-}(*dh_dr*)

このエンドポイントが惨事復旧に使用する分散ホストを定義します。

filepath(*filePaths*)

1 つ以上の絶対ファイル パスを定義します。それぞれが有効なカーネル モジュールである必要があります。複数のファイル パスはコロン（:）で区切ります。

flags(*flagName*)

リソースを trusted にする方法およびリソースのステータスが trusted であるかどうかをチェックする方法を定義します。有効なフラグは、Ctime、Mtime、Mode、Size、Device、Inode、Crc、および Own/All/None です。

gacc(*access-value*)

頻繁に開かれる保護されたファイルに対するプログラムからのアクセス速度を向上させます。

gowner(*groupName*)

リソース レコードの所有者として CA Access Control グループを割り当てます。リソース レコードのグループ所有者には、リソースに対する無制限のアクセス権が与えられます。ただし、前提として、グループ所有者のセキュリティ レベル、セキュリティ ラベル、およびセキュリティ カテゴリに、リソースへのアクセスを許可する適切な権限が設定されている必要があります。リソースのグループ所有者には、リソース レコードを更新および削除する許可が常に与えられます。詳細については、「*CA Access Control for UNIX* エンドポイント管理ガイド」を参照してください。

label(*labelName*)

リソース レコードにセキュリティ ラベルを割り当てます。

label-

リソース レコードからセキュリティ ラベルを削除します。このパラメータは chres コマンドまたは editres コマンドにのみ使用できます。

level(*number*)

リソース レコードにセキュリティ レベルを割り当てます。1 ～ 255 の正の整数を入力します。

level-

リソースからセキュリティ レベルをすべて削除します。このパラメータは chres コマンドまたは editres コマンドにのみ使用できます。

mask (*IPv4-address*) match (*IPv4-address*)

mask パラメータと *match* パラメータは、HOSTNET レコードにのみ適用されます。これらは、HOSTNET レコードを作成するときに必要です。また、レコードを変更するときにオプションで必要です。

mask と match を組み合わせて使用すると、HOSTNET レコードで定義されるホストのグループを定義できます。ホスト IP アドレスと mask アドレスの AND によって match アドレスが生成される場合、ホストは HOSTNET レコード グループのメンバです。

たとえば、mask(255.255.255.0) および match(192.16.133.0) と指定した場合、IP アドレスが 192.16.133.0 ～ 192.16.133.255 の範囲にあるホストはこのグループのメンバです。

mask パラメータと match パラメータには、IPv4 アドレスを指定する必要があります。

mem(*resourceName*)

メンバ リソースをリソース グループに追加します。複数のメンバ リソースを追加する場合は、名前をカンマで区切ります。

mem パラメータは、以下のクラスのリソース レコードとのみ組み合わせて使用できます。

- CONTAINER。このクラスでは、他のリソース クラスに属するオブジェクトのグループを定義します。
- GFILE。このクラスには、ファイルのグループを定義するリソース レコードが含まれています。
- GHOST。このクラスには、ホストのグループを定義するリソース レコードが含まれています。
- GSUDO。このクラスには、コマンドのグループを定義するリソース レコードが含まれています。
- GTERMINAL。このクラスには、端末のグループを定義するリソース レコードが含まれています。
- GPOLICY。 このクラスには、論理ポリシーを定義するリソース レコードが含まれています。
- GHNODE。このクラスには、ホスト グループを定義するリソース レコードが含まれています。
- GDEPLOYMENT。このクラスには、ポリシー デプロイを定義するリソース レコードが含まれています。

mem パラメータは、適切なタイプのレコードをリソース グループに追加するため、たとえば FILE レコードを GFILE クラスのリソース グループに追加するために使用します。

注: CONTAINER リソースに対して mem パラメータを使用する場合、of_class パラメータも併用する必要があります。

メンバ リソースとリソース グループがどちらも CA Access Control にすでに定義されている必要があります。リソース グループを作成するには、目的のクラスのリソースを作成します。たとえば、以下のコマンドを実行すると GFILE リソース グループが作成されます。

```
newres GFILE myfiles
```

mem-(*resourceName*)

リソース グループからメンバ リソースを削除します。複数のメンバ リソースを削除する場合は、各リソース名をスペースまたはカンマで区切ります。このパラメータは chres コマンドまたは editres コマンドにのみ使用できます。

node_alias{-|+}(*alias*)

エンドポイントの別名を定義します。

エンドポイントの別名を定義すると、CA Access Control で拡張ポリシー管理コマンドを実際のエンドポイントに別名を使用して送信できるようになります。

node_ip[-|+](*ip*)

ホストの IP アドレスを定義します。拡張ポリシー管理では、エンドポイントの名前と併せて IP アドレスを使用して、必要なエンドポイントを特定します。

notify(*mailAddress*)

リソース レコードが示すリソースへのアクセスが実行されるたびに通知メッセージを送信するよう CA Access Control に指示します。ユーザ名またはユーザの電子メール アドレスを入力します。また、別名が指定されている場合は、メール グループの電子メール アドレスも入力できます。

通知は、ログ ルーティング システムがアクティブな場合にのみ行われます。通知メッセージは、ログ ルーティング システムの設定に基づいて、ユーザの画面またはメールボックスに送信されます。

通知メッセージが送信されるたびに、監査ログに監査レコードが書き込まれます。監査レコードのフィルタ処理および表示の詳細については、「*CA Access Control for UNIX エンドポイント管理ガイド*」を参照してください。

通知メッセージの受信者は、頻繁にログインして、各メッセージに示された不正アクセスの試みに対処する必要があります。

制限: 30 文字。

notify-

リソース レコードが示すリソースへのアクセスが成功した場合、誰にも通知を行わないように指定します。このパラメータは chres コマンドまたは editres コマンドにのみ使用できます。

of_class(*className*)

mem パラメータを使用して CONTAINER クラスに追加するレコードのリソース タイプを指定します。

owner(*Name*)

リソース レコードの所有者として CA Access Control ユーザまたはグループを割り当てます。リソース レコードの所有者には、リソースに対する無制限のアクセス権が与えられます。ただし、前提として、所有者のセキュリティ レベル、セキュリティ ラベル、およびセキュリティ カテゴリに、リソースへのアクセスを許可する適切な権限が設定されている必要があります。リソースの所有者には、リソース レコードを更新および削除する権限が常に与えられます。詳細については、「*A Access Control for UNIX エンドポイント管理ガイド*」を参照してください。

password

SUDO クラスの場合に、sesudo コマンドを実行するには元のユーザのパスワードが必要であることを指定します。

password-

password パラメータを取り消します。その結果、元のユーザのパスワードを指定しなくても sesudo コマンドを実行できるようになります。このパラメータは chres コマンドまたは editres コマンドでのみ使用できます。これまでに password パラメータが使用されていない場合、このパラメータは必要ありません。

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

ノードのサブスクライバを伝達ツリーに追加し、ステータスを指定します。または、既存のポリシー バージョンを更新し、ポリシー偏差があるかどうかを指定します。ポリシー ステータスを更新するときは、updated_by プロパティを更新する必要があります。これは、ポリシー ステータスを変更したユーザの名前を表す文字列です。

ポリシー ステータスは、Transferred、Deployed、Undeployed、Failed、SigFailed、Queued、UndeployFailed、TransferFailed のいずれかです。

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

ノードから名前付きポリシー バージョンを削除します。ポリシーの指定がない場合は、このノードにデプロイされたすべてのポリシーが削除されます。

resourceName

変更または追加するリソース レコードの名前を指定します。複数のリソースを変更または追加する場合は、リソース名のリストを丸かっこで囲み、各リソース名をスペースまたはカンマで区切ります。リソース名は、少なくとも 1 つ指定する必要があります。

CA Access Control では、指定したパラメータに従って、各リソース レコードが個別に処理されます。リソースの処理中にエラーが発生すると、メッセージが発行され、リストの次のリソースから処理が続行されます。

Note: リソース名で変数を使用する場合、変数名を参照するには次の構文を使用します。<!*変数*>、例: <!AC_ROOT_PATH>¥bin。ポリシーの selang ルールでは、変数のみ使用できます。

restrictions([days] [time])

ユーザがファイルにアクセスできる曜日と時間帯を指定します。

days 引数を指定せずに time 引数を指定した場合、レコード内にすでに設定されている曜日制限に対して、指定した時間帯制限が適用されます。
time 引数を指定せずに days 引数を指定した場合、レコード内にすでに設定されている時間帯制限に対して、指定した曜日制限が適用されます。
days 引数と time 引数の両方を指定した場合、ユーザは、指定した曜日の指定した時間帯にのみシステムにアクセスできます。

- [days] には、ユーザがファイルにアクセスできる曜日を指定します。
  days 引数には次のサブ引数があります。
  - **anyday** - ユーザは曜日を問わずファイルにアクセスできます。
  - **weekdays** - ユーザは月曜から金曜までの平日に限りリソースにアクセスできます。
  - **Mon**、**Tue**、**Wed**、**Thu**、**Fri**、**Sat**、**Sun** - ユーザは指定した曜日にのみリソースにアクセスできます。曜日は任意の順で指定できます。複数の曜日を指定する場合は、各曜日をスペースまたはカンマで区切ります。
- [time] には、ユーザがリソースにアクセスできる時間帯を指定します。
  time 引数には次のサブ引数があります。
  - **anytime** - 特定の曜日の任意の時間帯にリソースにアクセスできます。
  - **startTime:endTime** - 指定した時間帯にのみリソースにアクセスできます。startTime および endTime は両方とも *hhmm* の形式で指定します。*hh* は 24 時間表記の時間（00 から 23）、*mm* は分（00 から 59）を表します。2400 は有効な time 値ではないことに注意してください。startTime が endTime より小さいこと、および両方が同じ日の時間であることが必要です。端末がホストと異なるタイム ゾーンにある場合は、端末の開始時間と終了時間をホストのローカル時間に相当する時間に変換し、時間の値を調整してください。たとえば、ホストがニューヨークにあり、端末がロサンゼルスにある場合、ロサンゼルスの端末からのアクセスを午前 8 時から午後 5 時まで許可するには、「time(1100:2000)」と指定します。

restrictions-([days] [time])

ユーザによるファイルへのアクセスを限定するすべての曜日および時間帯の制限を削除します。

ruleset+(*name*)

ポリシーに関連付けるルール セットを指定します。

ruleset-(*name*)

ポリシーからルール セットを削除します。ルール セットの指定がない場合は、すべてのルール セットがポリシーから削除されます。

signature(*hash_value*)

ハッシュ値を指定します。ポリシーの場合は、ポリシーに関連付けられている RULESET オブジェクトのシグネチャを基にします。ルール セットの場合は、ポリシー デプロイ コマンド リストとポリシー デプロイ解除(削除)コマンド リストを基にします。

subscriber(name(*sub_name*) status(*status*))

ノードのサブスクライバを伝達ツリーに追加し、ステータスを指定します。ステータスは、**unknown**、**available**、**unavailable**、**sync** のいずれかです。

subscriber-(name(*sub_name*)) | sub-

サブスクライバ データベースをノードから削除します。サブスクライバの指定がない場合は、すべてのサブスクライバが削除されます。

targuid(*userName*)

SUDO クラスに対して、コマンドの実行に権限を借用されるユーザの名前を指定します。デフォルトでは root ユーザです。

trust

リソースを trusted として指定します。trust パラメータは、PROGRAM クラスおよび SECFILE クラスのリソースにのみ適用されます。プログラムが trusted の場合のみ、そのプログラムを実行できます。詳細については、「*A Access Control for UNIX エンドポイント管理ガイド*」を参照してください。このパラメータは chres コマンドまたは editres コマンドにのみ使用できます。

trust-

リソースを untrusted として指定します。trust パラメータは、PROGRAM クラスおよび SECFILE クラスのリソースのみに適用されます。untrusted プログラムは実行できません。詳細については、「 *CA Access Control for UNIX エンドポイント管理ガイド*」を参照してください。このパラメータは chres コマンドまたは editres コマンドにのみ使用できます。

undocmd+(*selang_command_string*)

ポリシー デプロイ解除を定義する selang コマンドのリストを指定します。これが、デプロイ済みのポリシーの削除（デプロイ解除）に使用するコマンドです。以下に例を示します。

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

undocmd-

ポリシー削除コマンド リストを RULESET オブジェクトから削除します。

value+(*value*)

指定された値を指定された変数（ACVAR オブジェクト）に追加します。

value-(*value*)

指定された変数（ACVAR オブジェクト）から指定された値を削除します。

warning

アクセサがリソースにアクセスできる権限を持たない場合でもリソースにアクセスできるように指定します。ただし、監査ログに警告メッセージが書き込まれます。

注：警告モードの場合、CA Access Control では、リソース グループに対する警告メッセージは作成されません。

warning-

アクセサの権限ではリソースにアクセスできない場合、リソースへのユーザ アクセスを拒否して、警告メッセージを書き込まないように指定します。このパラメータは chres コマンドまたは editres コマンドにのみ使用できます。

例

- ユーザ *admin1* が、端末 *tty30* に対して所有者とデフォルト アクセス権を変更し、その端末の使用を平日の通常業務時間内（午前 8 時から午後 6 時）に制限します。
  - ユーザ admin1 には ADMIN 属性が割り当てられているとします。

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

- ADMIN 属性を持つユーザ Sally が、ファイル account.txt の FILE クラス レコードに格納されているグループと所有者のプロパティを削除する操作を実行します。
  - ユーザ Sally は Jared のユーザ レコードの所有者だとします。

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

  レコード プロパティが文字列で定義されている場合、レコード プロパティを削除するには、「-」符号または空のかっこ「()」のいずれかを付けてプロパティを入力します。

- ユーザ Bob が、端末 tty190 のコメント フィールドを削除し、その端末へのアクセスが許可されるたびに通知を受け取るように設定します。
  - ユーザ Bob は、CA Access Control ユーザで、端末 tty190 の所有者だとします。

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

- ユーザ Admin1 が、SURROGATE クラスにあるリソース USER.root のセキュリティ カテゴリのリストに OPERATOR カテゴリを追加します。
  - ユーザ Admin1 に ADMIN 属性が割り当てられているとします。
  - OPERATOR カテゴリがデータベースに定義されているとします。

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

- ユーザ admin1 が、/bin/su をグローバルな EXECUTE アクセス権が指定された trusted プログラムとして定義します。
  - ユーザ admin1 には ADMIN 属性が割り当てられているとします。
  - 以下のデフォルト値が適用されるとします。
    - restrictions(days(anyday) time(anytime))
    - owner(admin1)
    - audit(failure)

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

- ユーザ admin1 が、admin1 を含めすべてのユーザがアクセスできない保護されたリソースとしてグループ system にグループ ID の一時変更を定義します。
  - ユーザ admin1 には ADMIN 属性が割り当てられているとします。ユーザ nobody が CA Access Control に定義されているとします。
  - 以下のデフォルト値が適用されるとします。
    - restrictions(days(anyday) time(anytime))
    - audit(failure)

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

- ユーザ SecAdmin が、ProjATerms（端末 T1、T8、および T11 を含む端末のグループ）を定義します。この端末グループは、PROJECTA グループだけが、平日の通常業務時間内（午前 8 時から午後 6 時）にのみ使用します。
  - ユーザ SecAdmin に ADMIN 属性が割り当てられているとします。
  - 端末 T1、T8、および T11 は CA Access Control に定義されているとします。
  - グループ PROJECTA は CA Access Control に定義されているとします。
  - audit(failure)

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

詳細情報:

rmres コマンド - リソースの削除 (P. 152)
showres コマンド - リソース プロパティの表示 (P. 172)
authorize コマンド - リソースに対するアクセス権限の設定 (P. 51)
chres コマンド - Windows リソースの変更 (P. 215)
find コマンド - データベース レコードの一覧表示 (P. 132)
CONTAINER クラス (P. 288)
クラス別アクセス権限 (P. 32)

ch[x]usr コマンド - ユーザ プロパティの変更

AC 環境で有効

chusr、chxusr、editusr、editxusr、newusr、および newxusr の各コマンドは、CA Access Control データベース内でユーザのプロパティを変更するため、および必要に応じて、ユーザ レコードを定義するために使用します。

各コマンドには以下のような省略形があります。

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

たとえば、コマンド cu はコマンド chusr と同一です。

これらのコマンドはすべて構造は同じですが、対象のみが異なります。それぞれ、以下のように使い分けます。

- chusr、editusr、および newusr コマンドは、内部ユーザを対象に使用します。これらのコマンド間の相違点は以下のとおりです。
  - chusr コマンド*は、1 つ以上の* USER レコードを変更します。
  - editusr コマンド*は、1 つ以上の* USER レコードを作成または変更します。
  - newusr コマンド*は、1 つ以上の* USER レコードを作成します。

  注: このコマンドはネイティブ環境にもありますが、動作が異なります。
- chxusr、editxusr、および newxusr コマンドは、エンタープライズ ユーザを対象に使用します。これらのコマンド間の相違点は以下のとおりです。
  - chxusr コマンドは、1 つ以上の XUSER レコードを*変更*します。
  - editxusr コマンドは、1 つ以上の XUSER レコードを*作成または変更*します。
  - newxusr コマンドは、1 つ以上の XUSER レコードを*作成*します。

USER クラスと XUSER クラスのレコードはすべてのプロパティがまったく同じです。相違点は、エンタープライズ ユーザ ストアに定義されているプロパティが、XUSER レコードでは再定義されないことです。

これらのコマンドを実行すると、対象ユーザが現在システムにログイン中であっても、行われた変更によりユーザ レコードはただちに変更されます。

必要な権限

CA Access Control ユーザを作成するには、以下の条件が少なくとも 1 つ満たされている必要があります。

- ADMIN 属性が割り当てられていること
- ADMIN クラスの USER または XUSER レコードのアクセス制御リストに CREATE アクセス権が割り当てられていること

ユーザを追加または変更するには、以下の条件が少なくとも 1 つ満たされている必要があります。

- ADMIN 属性が割り当てられていること
- ユーザ レコードが、GROUP-ADMIN 属性が割り当てられているグループの有効範囲に含まれており、レコードの所有者と同じ権限が与えられていること
- ユーザ レコードが、GROUP-AUDITOR 属性が割り当てられているグループの有効範囲に含まれており、audit パラメータが指定されること
- グループの所有者であること
- ADMIN クラスの USER または XUSER レコードのアクセス制御リストに MODIFY アクセス権（ch[x]usr の場合）または CREATE アクセス権（edit[x]usr の場合）が割り当てられていること

```
{{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]|[interactive]}})] ¥
    [{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)] ¥
    [workstations(workstations-list)] )] ¥
[unix({ [gecos(string)] ¥
    [homedir(path)] ¥
    [pgroup(groupName)] ¥
    [shellprog(fileName)] ¥
    [userid(number)]}]
```

admin

ユーザに ADMIN 属性を割り当てます。ADMIN 属性を持つユーザは、audit パラメータ以外のすべてのパラメータを指定して selang のすべてのコマンドを発行できます。admin パラメータを使用するには ADMIN 属性が必要です。

admin-

ユーザから ADMIN 属性を削除します（CA Access Control は少なくとも 1 人のユーザが ADMIN 属性を持つことを確認します）。

このパラメータは、new[x]usr コマンドでは使用できません。

audit

CA Access Control で保護されたリソースに対するどのユーザ アクティビティを監査ログに記録するかを指定します。イベント タイプを複数指定するには、イベント タイプの名前をスペースまたはカンマで区切ります。audit の属性は以下のとおりです。

- **all** - すべてのユーザ アクティビティがログに記録されます。監視されるアクティビティは、failure、loginfail、loginsuccess、success、interactive、および trace です。
- **failure** - 失敗したアクセスの試みがログに記録されます。
- **loginfail** - 失敗したログインの試みがログに記録されます。
- **loginsuccess** - 成功したログインがログに記録されます。
- **none** - ユーザ アクティビティはログに一切記録されません。
- **success** - 成功したアクセスがログに記録されます。
- **interactive** - CA Access Control は対話式セッションをログに記録します。
- **trace** - このユーザのアクションに基づいて、トレース ファイルに表示されるすべてのメッセージがログに記録されます。

auditor

ユーザに AUDITOR 属性を割り当てます。AUDITOR 属性を持つユーザは、システム リソースの使用状況を監査できます。また、CA Access Control の権限チェックで検出された、CA Access Control の保護対象であるすべてのリソースへのアクセス、およびデータベースへのアクセスに対するログの記録を制御できます。AUDITOR 属性を持つユーザに与えられる権限の詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

auditor-

ユーザ レコードから AUDITOR 属性を削除します。

このパラメータは、new[x]usr コマンドでは使用できません。

auth_type

認証方法を指定します。

SSO でのみ使用されます。

このパラメータは、エンタープライズ ユーザに対しては使用できません。

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

1 つ以上のセキュリティ カテゴリをユーザに割り当てます。

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

ユーザ レコードから 1 つ以上のセキュリティ カテゴリを削除します。

このパラメータは、new[x]usr コマンドでは使用できません。

comment(*commentString*)

ユーザ レコードにコメントを追加します。

commentString

コメントを指定します。*commentString* は最大 255 文字の英数字の文字列です。*commentString* に空白文字が含まれる場合は、文字列全体を一重引用符で囲みます。

comment-

ユーザ レコードからコメントを削除します。

このパラメータは、new[x]usr コマンドでは使用できません。

country(*countryName*)

ユーザの国名を指定します。国は、認証プロセスでは使用されません。

countryName

国を定義します。このパラメータは最大 19 文字の英数字から成る文字列です。文字列に空白が含まれる場合は、文字列全体を一重引用符で囲みます。

email(*emailAddress*)

ユーザの電子メール アドレスを定義します。

emailAddress

ユーザの電子メール アドレスを定義します。

制限: 128 文字以下

enable

何らかの理由で無効になっているユーザのログインを有効にします。

このパラメータは、new[x]usr コマンドでは使用できません。

epwasown(*password*)

ユーザが自分のパスワードを変更するように、ユーザのパスワードを変更します。このパスワード変更は管理上の変更でなく、従って、パスワードが自動的に失効することはありません。

注: このコマンドは内部使用のみです。このコマンドは、/etc/shadow (パスワード ファイル) への引数として指定されるように平文テキストでパスワードを設定します。

expire(*dateTime*)

ユーザ アカウントが失効する日付を設定します。日付の指定がない場合、アカウントはただちに失効します。ユーザがログイン中の場合は、ユーザがログアウトした時点で失効します。

このプロパティの値がユーザ レコードに指定されている場合は、ユーザ レコードの値が GROUP クラスのレコードの値より優先されます。

注: expire- パラメータを使用して、失効したユーザ レコードを有効にします。これを行うのに、resume パラメータは使用しません。

dateTime

日付と、オプションで時刻を指定します。形式は以下のとおりです。
mm/*dd*/[*yy*]*yy*[@*HH*:*MM*]

年は、2 桁または 4 桁で指定できます。

expire-

new[x]usr コマンドの場合は、有効期限のないユーザ アカウントを定義します。

ch[x]usr コマンドおよび edit[x]usr コマンドの場合は、ユーザ アカウントから有効期限を削除します。

flags(*accountFlags*|*-accountFlags*)

ユーザ アカウントの特定の属性を指定します。有効なフラグ値の詳細については、付録「Windows の値」を参照してください。

ユーザ レコードからフラグを削除するには、*accountFlags* の前にマイナス記号 (-) を付けます。

fullname(*fullName*)

ユーザのフル ネームを指定します。

fullName

フル ネームを指定します。最大 255 文字の英数字から成る文字列です。*fullName* に空白が含まれる場合は、文字列全体を一重引用符で囲みます。

gecos(*string*)

ユーザのコメント文字列を指定します。文字列は一重引用符で囲みます。

gowner(*groupName*)

ユーザ レコードの所有者として CA Access Control グループを割り当てます。ユーザ レコードのグループ所有者には、ユーザ レコードに対する無制限のアクセス権が与えられます。ただし、前提として、グループ所有者のセキュリティ レベルとセキュリティ カテゴリに適切な権限が設定されている必要があります。ユーザ レコードのグループ所有者は、ユーザ レコードをいつでも更新および削除することができます。

grace(*nLogins*)

ユーザに許可する猶予ログイン回数を定義します。

猶予ログイン回数に達するとユーザはシステムにアクセスできなくなるため、システム管理者に連絡して新しいパスワードを設定する必要があります。猶予ログイン回数が 0 に設定されている場合、ユーザはログインできません。

このパラメータの値がユーザ レコードに指定されている場合は、ユーザ レコードの値が GROUP クラスのレコードの値より優先されます。

このパラメータを指定しない場合でも、ユーザのプロファイル グループにこのパラメータの値が含まれている場合は、GROUP クラスのレコードの値が使用されます。USER クラスのレコードにも GROUP クラスのレコードにも値が含まれていない場合は、CA Access Control のグローバル猶予ログイン設定が使用されます。

nLogins

猶予ログイン回数を指定します。0 ～ 255 の整数を入力してください。

注: 猶予ログイン回数の値が 0 に達する前に、パスワードを変更する必要があります。猶予ログイン回数の値に達ってしまった場合、システム管理者に連絡して新しいパスワードを選択してください。

grace-

ユーザの猶予ログイン設定を削除します。代わりに、CA Access Control のグローバル猶予ログイン設定が使用されます。

このパラメータは、newusr コマンドでは使用できません。

homedir(*path*)

ユーザのホーム ディレクトリの完全パスを指定します。*path* の最後にスラッシュを付けると、パスに *userName* が自動的に追加されます。

homedrive(*drive*)

ユーザのホーム ディレクトリのドライブを指定します。

ign_hol

ユーザに IGN_HOL 属性を割り当てます。IGN_HOL 属性を持つユーザは、holiday レコードに定義された期間中にログインできます。

ign_hol-

IGN_HOL 属性をユーザから削除します。

inactive(*nDays*)

ユーザのステータスが非アクティブに変更されるまでの経過日数を指定します。指定した日数に達すると、ユーザはログインできなくなります。

注: 非アクティブ ユーザはユーザ レコードにマークが設定されません。アクティブでないユーザを識別するには、Inactive Days 値と Last Accessed Time 値を比較する必要があります。

nDays

日数を指定します。*nDays* には、0 または正の整数を指定します。*nDays* を 0 に設定した結果は、inactive- パラメータを指定した場合と同じになります。

inactive-

ユーザのステータスを非アクティブからアクティブに変更します。

このパラメータは、newusr コマンドでは使用できません。

interval(*nDays*)

パスワードの設定または変更後、ユーザに対して新しいパスワードの入力を促すメッセージが表示されるまでの経過日数を定義します。0 または正の整数を指定します。*nDays* が 0 の場合、パスワード期間のチェックが無効になり、パスワードが失効しません。つまり、setoptions コマンドで設定したデフォルト値は使用されません。*nDays* を 0 に設定するのは、セキュリティ要件が厳しくないユーザに限定してください。

nDays が経過すると、現在のパスワードが期限切れであることがユーザに通知されます。通知を受けたユーザは、猶予ログイン回数に達するまでパスワードを引き続き使用することができます。猶予ログイン回数に達するとシステムへのアクセスを拒否されるため、ユーザはシステム管理者に連絡して新しいパスワードを取得する必要があります。

interval-

ユーザのパスワード期間の設定を取り消します。このパラメータの値がユーザのプロファイル グループに含まれている場合は、その値が使用されます。それ以外の場合は、setoptions コマンドで設定したデフォルト値が使用されます。

このパラメータは、new[x]usr コマンドでは使用できません。

label(*labelName*)

ユーザにセキュリティ ラベルを割り当てます。

label-

ユーザ レコードからセキュリティ ラベルを削除します。

このパラメータは、new[x]usr コマンドでは使用できません。

level(*levelNumber*)

ユーザ レコードにセキュリティ レベルを割り当てます。

levelNumber は、0 ～ 255 の整数です。

level-

ユーザ レコードからセキュリティ レベルを削除します。

このパラメータは、newusr コマンドでは使用できません。

localapps

CA SSO で使用されます。

location(*locationString*)

ユーザの所在地を指定します。所在地は、認証プロセスでは使用されません。

locationString

所在地を指定します。*locationString* は最大 47 文字の英数字から成る文字列です。*locationString* に空白文字が含まれる場合は、文字列を一重引用符で囲みます。

logical

ユーザに LOGICAL 属性を割り当てます。LOGICAL 属性が割り当てられたユーザはログインすることができず、CA Access Control 内部でのみ使用されます。

たとえば、リソースの所有者であってもリソースへのアクセスを妨げるために、リソースの所有者として使用するユーザ nobody は、デフォルトの論理ユーザです。これは、ユーザがこのアカウントを使用してログインすることができないことを意味します。

logical-

ユーザから LOGICAL 属性を削除します

logonserver(*server-name*)

ユーザのログイン情報を確認するサーバを指定します。ユーザがドメインワークステーションにログインすると、この引数で指定したサーバにログイン情報が送られ、ユーザがワークステーションを使用することが許可されます。

maxlogins(*nLogins*)

ユーザの最大同時ログイン数を設定します。値 0（ゼロ）は、同時に任意の数の端末からログインできることを意味します。このパラメータを指定しない場合は、グローバルな最大ログイン回数の設定が使用されます。

注: maxlogins を 1 に設定すると、selang を実行できません。この場合、CA Access Control を停止し、setpropadm ユーティリティなどを使用して maxlogins の設定を 2 以上の値に変更し、CA Access Control を再起動する必要があります。

maxlogins-

ユーザの最大ログイン数の設定を削除します。代わりに、グローバルな設定が使用されます。

このパラメータは、new[x]usr コマンドでは使用できません。

min_life(*nDays*)

ユーザがパスワードを再度変更できるまでの最短経過日数を指定します。正の整数を入力します。

min_life-

ユーザの min_life 設定を削除します。このパラメータの値がユーザのプロファイル グループに含まれている場合は、その値が使用されます。それ以外の場合は、setoptions コマンドで設定したデフォルト値が使用されます。

このパラメータは、new[x]usr コマンドでは使用できません。

nochngpass

ユーザが別のユーザのパスワードを変更できないように指定します。

notify(*notifyAddress*)

ユーザがログインするたびに、*notifyAddress* 宛に電子メールを送信します。通知メッセージを受け取るユーザは、頻繁にログインして、各メッセージに示された不正なアクセスの試みに対処する必要があります。

通知メッセージが送信されるたびに、監査ログに監査レコードが書き込まれます。

notifyAddress

ユーザ名または電子メール アドレスを指定します。

制限: 30 文字。

notify-

ユーザがログインしたときに誰にも通知を行わないように指定します。

このパラメータは、new[x]usr コマンドでは使用できません。

nt

chusr コマンドおよび editusr コマンドの場合、このパラメータは、ローカル Windows システムのユーザ定義を変更します。

newusr コマンドの場合、このパラメータはユーザをローカル Windows システムに追加します。

複数の引数を指定する場合は、各引数をスペースで区切ります。

ローカル Windows システムを CA Access Control 内で操作する方法の詳細については、environment コマンドの説明を参照してください。

nt オプションと nt オプションのサブオプションは、エンタープライズ ユーザに対しては無効です。

operator

ユーザに OPERATOR 属性を割り当てます。OPERATOR 属性を持つユーザは、データベースのすべてのリソース レコードを一覧表示できます。また、このユーザには CA Access Control で定義されたすべてのファイルに対する読み取り権限が与えられます。

この属性を持つユーザは、secons コマンドのすべてのオプションも使用できます。secons ユーティリティの詳細については、「*リファレンス ガイド*」を参照してください。

operator-

ユーザ レコードから OPERATOR 属性を削除します。

このパラメータは、newusr コマンドでは使用できません。

organization(*organizationString*)

ユーザの組織を指定します。組織は、認証プロセスでは使用されません。

organizationString

組織を指定します。*organizationString* は最大 255 文字の英数字から成る文字列です。 *organizationString* に空白文字が含まれる場合は、文字列を一重引用符で囲みます。

org_unit(*org_unitString*)

ユーザの組織単位を指定します。組織単位は、認証プロセスでは使用されません。

org_unitString

組織単位を指定します。*org_unitString* は最大 255 文字の英数字から成る文字列です。 *organizationString* に空白文字が含まれる場合は、文字列を一重引用符で囲みます。

owner(*Name*)

ユーザ レコードの所有者として CA Access Control ユーザまたはグループを割り当てます。詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

password(*string*)

ユーザにパスワードを割り当てます。スペースまたはカンマ以外の任意の文字を指定します。パスワード チェックが有効になっている場合、指定したパスワードでログインできるのは 1 回のみです。次回システムにログインする際に、ユーザは新しいパスワードを設定する必要があります。

自分のパスワードを変更するには、*setoptions cng_ownpwd* を使用して selang オプションを設定するか、sepass を使用する必要があります。

pgroup(*groupName*)

ユーザのプライマリ グループ ID を設定します。*groupName* には UNIX グループの名前を指定します。

phone(*phoneString*)

ユーザの電話番号を指定します。電話番号は、認証プロセスでは使用されません。

phoneString

電話番号を指定します。*phoneString* は最大 19 文字の英数字から成る文字列です。*phoneString* に空白文字が含まれる場合は、文字列を一重引用符で囲みます。

pmdb(*pmdbName*)

ユーザが sepass ユーティリティを使用してパスワードを変更した場合、指定された PMDB に新しいパスワードを伝達するように指定します。PMDB の完全修飾名を入力します。このパスワードは、seos.ini の [seos] セクションにある parent_pmd トークンまたは passwd_pmd トークンに定義されている Policy Model には送信されません。

このオプションは、エンタープライズ ユーザに対しては使用できません。

pmdb-

ユーザ レコードから PMDB 属性を削除します。

このパラメータは、new[x]usr コマンドでは使用できません。

privileges(*privilege-list*)

Windows のユーザ レコードに特定の権限を追加します。privList の前にマイナス記号 (-) を付けた場合は、指定した権限を削除します。

このパラメータは、newusr コマンドでは使用できません。

profile(*groupName*)

ユーザをプロファイル グループに割り当てます。次の値をプロファイル グループから取得できます。

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

profile-

ユーザをプロファイル グループから削除します。

このパラメータは、new[x]usr コマンドでは使用できません。

pwmanager

ユーザに PWMANAGER 属性を割り当てます。この属性を持つユーザは、データベースにあるユーザのパスワードを変更できます。 詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

pwmanager-

ユーザ レコードから PWMANAGER 属性を削除します。

このパラメータは、new[x]usr コマンドでは使用できません。

pwasown(*string*)

ユーザが変更した場合と同じようにパスワードを置き換えます。このパラメータを指定すると、データベースを最後に変更した日時が更新され、猶予ログインが終了します。

regular

レコードの OBJ_TYPE プロパティをリセットし、ユーザの権限属性を削除します。

restrictions([*Days*] [*Time*])

ユーザがログインできる曜日と時間帯を指定します。この制限は、[X]USER レコードの DAYTIME プロパティに格納されます。

Days 引数を指定せずに *Time* 引数を指定した場合、レコードですでに定義されている曜日制限に時間制限が適用されます。

Time 引数を指定せずに *Days* 引数を指定した場合、レコード内にすでに設定されている *Days* 制限に対して、指定した曜日制限が適用されます。

Days 引数と *Time* 引数の両方を指定した場合、ユーザは、指定した曜日の指定した時間帯にのみシステムにアクセスできます。

Days

ユーザがログインできる曜日を指定します。*Days* の指定には以下のキーワードを使用できます。

- **anyday** - ユーザは曜日を問わずファイルにアクセスできます。
- **weekdays** - ユーザは月曜から金曜までの平日に限りリソースにアクセスできます。
- **Mon**、**Tue**、**Wed**、**Thu**、**Fri**、**Sat**、**Sun** - ユーザは指定した曜日にのみリソースにアクセスできます。曜日は任意の順で指定できます。複数の曜日を指定する場合は、各曜日をスペースまたはカンマで区切ります。

Time

ユーザがログインできる時間帯を指定します。time 引数には次のサブ引数があります。

- **anytime** - 特定の曜日の任意の時間帯にリソースにアクセスできます。
- *startTime*:*endTime* - 指定した時間帯に限りリソースにアクセスできます。

  startTime と *endTime* はどちらも *hhmm* の形式で指定します。*hh* は時間(00 ～ 23)、*mm* は分(00 ～ 59)を表します。2400 は有効な Time 値ではないことに注意してください。代わりに 0000 を使用してください。

  startTime は *endTime* より小さい必要があります。

  注: CA Access Control では、プロセッサのタイム ゾーンを使用します。プロセッサと異なるタイム ゾーンの端末にログインする際には注意が必要です。

restrictions-([days] [time])

ユーザによるログインを限定するすべての曜日および時間帯の制限を削除します。

resume([*dateTime*])

suspend パラメータを指定して無効にしたユーザ レコードを有効にします。suspend パラメータと resume パラメータの両方を指定する場合、再開日を一時停止日より後に設定する必要があります。*dateTime* を省略すると、chusr コマンドの実行直後にユーザ レコードが再開されます。詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

dateTime は、*[m]m/[d]d/yy[@HH:MM]* の形式で指定します。

resume-

再開日および再開時間(指定されている場合)をユーザ レコードから消去します。これにより、ユーザのステータスがアクティブ(有効)から一時停止に変更されます。

このパラメータは、new[x]usr コマンドでは使用できません。

script(*logon-script-path*)

ユーザがログインしたときに自動的に実行されるファイルの場所を指定します。このパラメータは任意です。通常は、このログイン スクリプトによって作業環境が設定されます。ユーザの作業環境の設定には profile パラメータも使用できます。

server

SERVER 属性を設定します。現在のユーザに代わり実行しているプロセスから、他のユーザの権限をクエリできるようになります。詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

server-

SERVER 属性の設定を解除します。

このパラメータは、new[x]usr コマンドでは使用できません。

shellprog(*fileName*)

ユーザが login コマンドまたは su コマンドを起動した後に実行される初期プログラムまたはシェルの完全パスを指定します。*fileName* には文字列を指定します。

このオプションは、エンタープライズ ユーザに対しては使用できません。

suspend([*dateTime*])

ユーザ レコードを無効にします。ただし、データベースには定義を残します。ユーザは、無効にされたユーザ アカウントを使用してシステムにログインすることはできません。

dateTime を指定すると、指定された日にユーザ レコードが無効になります。*dateTime* を省略すると、ch[x]usr コマンドの実行直後にユーザ レコードが無効になります。

dateTime は *mm/dd/yy[@HH:MM]* の形式で指定します。

suspend-

一時停止日をユーザ レコードから消去し、ユーザのステータスを無効から有効(アクティブ)に変更します。

このパラメータは、new[x]usr コマンドでは使用できません。

unix

chusr コマンドおよび editusr コマンドの場合、このパラメータは、ローカル UNIX システムのユーザ定義を変更します。

newusr コマンドの場合、このパラメータはユーザをローカル UNIX システムに追加します。

複数の引数を指定する場合は、各引数をスペースで区切ります。

ローカル UNIX システムを CA Access Control 内で操作する方法の詳細については、この章の environment コマンドの説明を参照してください。

unix オプションと unix オプションのサブオプションは、エンタープライズ ユーザに対しては無効です。

userid(*number*)

一意の随意アクセス制御に使用するユーザの一意の ID 番号(UID)を設定します。*Number* には 10 進数を指定します。デフォルトでは、100 より小さい数値は使用できません。除外される数値の詳細については、「*リファレンス ガイド*」の AllowedGidRange トークンの説明を参照してください。

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

ユーザ名(複数可)を指定します。各ユーザ名は一意である必要があります。

newusr コマンドを使用すると、CA Access Control は *userName* を新しいユーザとして認識します。newusr コマンドで指定したユーザが、ネイティブ環境にすでに定義されている場合、そのユーザ名は該当ユーザの USER レコードとして使用されます。ただし、一般的には、newusr コマンドを使用してネイティブ環境に既存のユーザ名に対する USER レコードを作成するより、CA Access Control でエンタープライズ ユーザを使用できることを活用する方が得策です。目的のユーザの CA Access Control プロパティを変更するには、代わりに chgxusr コマンドを使用します。

ネイティブ ログイン名ではない CA Access Control ユーザ名を使用する場合が考えられます。その場合、login コマンドではそのユーザを使用できませんが、sesu などの他のコマンドで使用できます。

注: UNIX でユーザ名に円記号が含まれる場合は、*userName* を指定する際に円記号を 2 つ重ねます。

例

- ユーザ Bob が、Jim のレコードに FINANCIAL カテゴリを追加し、Jim のセキュリティレベルを 155 に変更し、さらに Jim によるシステムへのアクセスを平日の午前 8 時から午後 8 時までに制限します。
  - ユーザ Bob に ADMIN 属性が割り当てられているとします。
  - CA Access Control にユーザ Jim が定義されているとします。
  - CA Access Control に FINANCIAL カテゴリが定義されているとします。

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

- ユーザ admin が、1995 年 8 月 5 日から 3 週間の休暇に入る予定のユーザ Joel を一時停止します。
  - ユーザ admin に ADMIN 属性が割り当てられているとします。
  - CA Access Control にユーザ Joel が定義されているとします。
  - 現在の日付は 1994 年 8 月 3 日だとします。

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

- ユーザ Security2 が、ユーザ Bill から AUDITOR 属性を削除し、Bill のすべてのアクティビティを監査します。
  - ユーザ Security2 に ADMIN 属性および AUDITOR 属性が割り当てられているとします。
  - CA Access Control にユーザ Bill が定義されているとします。

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

- ユーザ Rob が、ユーザ Mary のレコードに格納されているコメントを変更します。
  - ユーザ Rob が Mary のユーザ レコードの所有者だとします。

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

- ADMIN 属性を持つユーザ Sally が、ユーザ Jared のレコードに格納されている国名および所在地のプロパティを削除します。
  - ユーザ Sally は Jared のユーザ レコードの所有者だとします。

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

- ユーザ Bob が、ユーザ Peter およびユーザ Joe を CA Access Control に定義します。
  - ユーザ Bob に ADMIN 属性が割り当てられているとします。
  - ユーザ Peter およびユーザ Joe が CA Access Control に定義されていないとします。
  - 以下のデフォルト値が適用されるとします。
    - owner(Bob)
    - audit(failure,loginfailure)

```
newusr (Peter Joe)
```

- ユーザ Bob がユーザ Jane を CA Access Control に定義し、Jane を所有するグループとして payroll を割り当てます。
  - ユーザ Bob に ADMIN 属性が割り当てられているとします。
  - CA Access Control にユーザ Jane が定義されていないとします。
  - ユーザ Jane のフル ネームは JG Harris だとします。
  - audit(failure,loginfailure)

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

- ユーザ Bob がユーザ *JohnD* を CA Access Control に定義し、セキュリティ カテゴリ NewEmployee およびセキュリティ レベル 3 を設定します。JohnD がシステムを使用できる時間帯を、平日の午前 8 時から午後 6 時までのみに設定します。
  - ユーザ Bob に ADMIN 属性が割り当てられているとします。
  - CA Access Control に NewEmployee カテゴリが定義されているとします。
  - 新しいユーザのフル ネームは John Doe だとします。
  - 以下のデフォルト値が適用されるとします。
    - owner(Bob)
    - audit(failure)

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

deploy コマンド - ポリシーのデプロイの開始

AC 環境で有効

deploy コマンドは、ポリシーのデプロイを開始するのに使用します。このコマンドは、デプロイする POLICY オブジェクトに関連付けられている RULESET オブジェクトに格納されている selang コマンドを実行します。これがポリシー デプロイ コマンドです。

重要: 格納されているポリシーのデプロイには policydeploy ユーティリティを使用することを強くお勧めします。deploy コマンドはポリシーのデプロイの一部しか実行せず、ポリシーをエンドポイントにデプロイする際に DMS を更新しません。

deploy コマンドを実行するには、以下の権限が必要です。

- ポリシーのデプロイ先データベースの下の階層にある各データベースの POLICY、HNODE、および RULESET クラスに対するサブ管理権限。
- ポリシーのデプロイ先データベースの下の階層にある各データベースに対する適切なサブ管理権限。

  これらは、各コンピュータのポリシーを構成する selang コマンドを実行するために必要な権限です。

  たとえば、新しいファイル リソースを作成する場合、FILE クラスに対するサブ管理権限が必要になります。

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

注: ポリシー デプロイの詳細については、「エンタープライズ管理ガイド」を参照してください。

このコマンドの形式は以下のようになります。

```
deploy POLICY name#xx
```

name#xx

デプロイするポリシーの POLICY オブジェクトの名前（ポリシー名とバージョン番号）。

deploy- コマンド - ポリシーの削除の開始

AC 環境で有効

ポリシーのデプロイ解除を開始するには、deploy-（または undeploy）コマンドを使用します。このコマンドは、デプロイする POLICY オブジェクトに関連付けられている RULESET オブジェクトに格納されている selang コマンドを実行します。これがポリシー デプロイ解除コマンドです。

重要：ポリシーのデプロイ解除には policydeploy ユーティリティを使用することを強くお勧めします。deploy- コマンドはポリシーのデプロイ解除の一部しか実行せず、ポリシーをエンドポイントからデプロイ解除する際に DMS を更新しません。

このコマンドを実行するには、以下の権限が必要です。

- ポリシーのデプロイ解除先データベースの下の階層にある各データベースの POLICY、HNODE、および RULESET クラスに対するサブ管理権限。
- ポリシーのデプロイ解除先データベースの下の階層にある各データベースに対する適切なサブ管理権限。

  これらは、各コンピュータのポリシーデプロイ解除スクリプトを構成する selang コマンドを実行するために必要な権限です。

注：ポリシーのデプロイ解除の詳細については、「*エンタープライズ管理ガイド*」を参照してください。

このコマンドの形式は以下のようになります。

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

name#xx

デプロイ解除するポリシーの POLICY オブジェクトの名前（ポリシー名とバージョン番号）。

editfile コマンド - ファイル レコードの作成と変更

AC 環境で有効

このコマンドについては、chfile コマンドの項で説明しています。

詳細情報:

chfile コマンド - ファイル レコードの変更 (P. 65)

edit[x]grp コマンド - グループ レコードの作成と変更

AC 環境で有効

このコマンドについては、ch[x]grp コマンドの項で説明しています。

詳細情報:

ch[x]grp コマンド - グループ プロパティの変更 (P. 72)

editres コマンド - リソース レコードの変更

AC 環境で有効

このコマンドについては、chres コマンドの項で説明しています。

詳細情報:

chres コマンド - リソース レコードの変更 (P. 89)

edit[x]usr コマンド - ユーザ レコードの変更

AC 環境で有効

このコマンドについては、chxusr コマンドの項で説明しています。

詳細情報:

ch[x]usr コマンド - ユーザ プロパティの変更 (P. 107)

end_transaction コマンド - デュアル コントロール トランザクションの記録の完了

AC 環境内の UNIX ホストで有効

end_transaction コマンドは、デュアル コントロール PMDB プロセスの start_transaction コマンドを完了します。

environment コマンド - セキュリティ環境の設定

すべての環境で有効

environment コマンドはセキュリティ環境を設定します。CA Access Control では、CA Access Control セキュリティ環境と UNIX セキュリティ環境をサポートしています。selang コマンド シェルを起動すると、デフォルトでは AC 環境が選択されます。

このコマンドの形式は以下のようになります。

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

ac

CA Access Control セキュリティ環境を指定します。selang コマンドは、ローカル CA Access Control データベースに対して実行されます。一部のコマンドでは、接続先ホストのネイティブ OS のセキュリティ設定を同時に更新できます。CA Access Control 環境の selang プロンプトは以下のとおりです。

```
AC>
```

config

リモート設定環境を指定します。リモート設定環境ではエンドポイントの設定を変更できます。

etrust

CA Access Control セキュリティ環境を指定します。

注: これは *ac* を指定したことと同じであり、旧バージョンとの互換性を維持するために用意されています。

native

ローカル、リモートを問わず、接続先ホストのネイティブ オペレーティング システムのセキュリティ環境（Windows または UNIX）を指定します。selang コマンドは、ネイティブ OS データベースに対して実行されます。ネイティブ環境の selang プロンプトは次のとおりです。

```
AC(native)>
```

nt

Windows のセキュリティ環境を指定します。selang コマンドは、Windows データベースに対して実行されます。一部のコマンドでは、CA Access Control のセキュリティ設定を同時に更新できます。Windows 環境の selang プロンプトは次のとおりです。

```
AC(nt)>
```

pmd

リモート管理環境で selang コマンドを指定します。selang コマンド シェルを pmd 環境に設定すると、コマンドは選択されたホストの PMDB に対して実行されます。pmd 環境の selang プロンプトは以下のとおりです。

```
AC(pmd)>
```

seos

CA Access Control セキュリティ環境を指定します。

注: これは *ac* を指定したことと同じであり、旧バージョンとの互換性を維持するために用意されています。

unix

UNIX のセキュリティ環境を指定します。selang コマンドは、UNIX のセキュリティ システムに対して実行されます。UNIX 環境の selang プロンプトは以下のとおりです。

```
AC(unix)>
```

find コマンド - データベース レコードの一覧表示

AC 環境とネイティブ環境で有効

find コマンドは、指定したクラスのレコードの名前を表示します。パラメータの指定がない場合は、全クラスの名前を表示します。

注: find コマンドは、*list* コマンドおよび *search* コマンドと同じです。

このコマンドを使用するには、適切な権限が必要です。以下に条件を示します。

- ADMIN 属性、AUDITOR 属性、または OPERATOR 属性が割り当てられている場合は、find コマンドにすべてのパラメータを指定できます。
- ADMIN クラスのレコードの READ 権限が割り当てられている場合は、レコードが示すクラスに class パラメータを指定できます。

このコマンドの形式は以下のようになります。

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

className

find でレコードを検索するクラスを指定します。*className* の指定がない場合、*find* はすべてのクラスを一覧表示します。

objName

CA Access Control が検索するレコードを指定します。*objName* にはワイルドカード文字を使用できます。

例: TERMINAL クラスのすべてのレコードの表示

TERMINAL クラスのすべてのメンバを表示するには、以下のコマンドを入力します。

```
find terminal
```

get dbexport コマンド - エクスポートされたデータベース ルールの取得

AC 環境で有効

get dbexport コマンドを使用すると、接続しているホストの CA Access Control データベースまたは PMD データベースからエクスポートされたルールを取得します。エクスポートされたデータベースが存在する場合、get dbexport コマンドを発行する前に start dbexport コマンドを発行する必要があります。

このコマンドの形式は以下のようになります。

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

pmdname(*name*)

(オプション) エクスポートした PMD データベースの名前を定義します。

params(OFFSET=*number*)

(オプション) データベース出力から多数の行を取得するときに使用するオフセットを定義します。get dbexport コマンドでは、1 つのリクエストにつき、エクスポートされたデータベースから 200 行のみを返すことができます。出力された情報がこれより多い場合、このコマンドは、返された最終行を示すオフセット データを返します。

例: エクスポートされたデータベースからルールを取得

以下の例では、get dbexport コマンドを使用して、接続しているホストにあるエクスポートされた CA Access Control データベースから情報を取得する方法を示します。最初のコマンドでは先頭の 200 行、2 度目のコマンドではその次の 200 行が出力から取得されます。

```
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
```

詳細情報:

start dbexport コマンド - データベース エクスポートの開始 (P. 178)

get devcalc コマンド - ポリシー偏差データの取得

AC 環境で有効

get devcalc コマンドは、ポリシー偏差の計算結果が格納されたポリシー偏差データ ファイル（deviation.dat）から情報を取得し、1 つ以上の DMS データベースに送信します。データ ファイルが存在するように、start devcalc コマンドを先に発行しておく必要があります。

ポリシー レポートまたはホスト レポートを作成するときに、偏差計算結果を含めるように指定することができます。指定すると、レポート ユーティリティがこのコマンドを発行します。

重要：偏差計算では、ネイティブ ルールが適用されるかどうかはチェックされません。データベースからオブジェクト（ユーザまたはオブジェクト属性、ユーザまたはリソース権限、あるいは実際のユーザまたはリソース）を削除するルールも無視されます。たとえば、偏差計算では、以下のルールが適用されるかどうかは確認できません。
rr SUDO admCommand

注：ポリシー偏差データ ファイルと拡張ポリシー レポートの詳細については、「*エンタープライズ管理ガイド*」を参照してください。

get devcalc コマンドを実行するには、使用コンピュータに対する端末アクセス権限と DEVCALC サブ管理クラスに対する読み取りアクセス権限が必要です。

このコマンドの形式は以下のようになります。

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

offset=*number*

（オプション）ポリシー偏差データ ファイルから多数の行を取得するときに使用するオフセットを定義します。get devcalc コマンドでは、一度の要求に対して最大行数（max_lines_request 環境設定で設定される）しか返すことができません。ファイルの行数がこれより多い場合、このコマンドは、返す最終行を示すオフセット データを返します。

例: ポリシー偏差データの取得

次の例では、max_lines_request が 10 に設定されている場合に、get devcalc コマンドを使用してポリシー偏差データ ファイルから情報を取得する方法を示しています。最初のコマンドでは先頭の 10 行、2 度目のコマンドではその次の 10 行が出力から取得されます。

```
AC> get devcalc
(localhost)
DEVCALC 'deviation' のデータ
-------------------------------------------
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)
DEVCALC 'deviation' のデータ
-------------------------------------------
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
```

詳細情報:

start devcalc コマンド - ポリシー偏差計算の開始 (P. 180)
setoptions コマンド - CA Access Control オプションの設定 (P. 158)

help コマンド - selang ヘルプの表示

すべての環境で有効

help コマンドは、selang の構文をいくつかの方法で表示します。

- パラメータの指定がない場合、selang コマンドのリストが各コマンドの簡単な説明と共に一覧表示されます。
- selang のコマンド名を指定した場合は、指定したコマンドの構文が表示されます
- access パラメータを指定した場合は、authorize コマンドの access パラメータの値と、new* コマンド、ch* コマンド、および edit* コマンドの defaccess パラメータの値が一覧表示されます。
- lineedit パラメータを指定した場合は、selang のコマンド ライン操作で使用する特殊文字が一覧表示されます。

注: コマンド ラインのテキストを削除せずに、コマンド ラインに入力したコマンドのヘルプ テキストを表示するには、Ctrl キーを押しながら 2 を押します。

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

形式

access パラメータと defaccess パラメータで指定できる、アクセス タイプのクラス別リストを要求します。

className

指定したクラスの短い説明を要求します。

command-name

指定したコマンドの構文を要求します。

lineedit

selang のコマンド ライン操作に使用する特殊文字のリストを要求します。

properties

(AC 環境)ユーザ定義プロパティの更新方法に関する情報を要求します。

privilege

(Windows 環境)ch[x]grp、ch[x]usr、edit[x]grp、および edit[x]usr の各コマンドで可能な Windows 権限の一覧を要求します。

詳細情報:

selang コマンド リファレンス (P. 43)
selang 環境 (P. 37)
selang ヘルプの表示 (P. 41)

history コマンド - 以前発行したコマンドの表示

すべての環境で有効

history コマンドは、selang コマンド シェルの現在のセッション中に入力されたすべてのコマンドを一覧表示します。コマンドは入力した順に表示されます。各コマンドの先頭にはコマンド番号が表示されます。たとえば、3 番目に入力されたコマンドの先頭には番号 3 が表示されます。

history コマンドでは、ch[x]usr コマンド、new[x]usr コマンド、または edit[x]usr コマンドの一部としてパスワードを入力した場合でも、パスワードは表示されません。パスワードは、通常のテキストではなく複数のアスタリスク(***)で表示されます。

このコマンドの形式は以下のようになります。

```
history
```

詳細情報:

コマンド履歴 (P. 25)

hosts コマンド - リモート CA Access Control 端末への接続

すべての環境で有効

hosts コマンドは、selang コマンドを受け取るホストまたは Policy Model を指定します。このコマンドを使用すると、名前が異なるリモート CA Access Control コンピュータにも接続することができます。したがって、ローカル CA Access Control サービスが実行されていなくてもコンピュータのリモート管理が可能です。デフォルトでは、すべての selang コマンドがローカル ホスト上のデータベースに送信されます。

ホストに送信するコマンドを実行する場合は、その前に hosts コマンドを実行する必要があります。

ローカル ホストからリモート ホスト データベースを管理(更新)するユーザは、以下の条件の*いずれか*を満たしている必要があります。

- ローカル データベースからリモート ホスト データベースを更新する権限が明示的に与えられていること
- ローカル データベースからリモート ホスト データベースを更新する許可が与えられているグループのメンバであること
- リモート ホストに定義された、ローカル ホストの所有者であること

現在使用可能なすべてのホストおよび PMDB を一覧表示するには、パラメータを指定せずに hosts コマンドを指定します。

注: CA Access Control では、別名ではなく正規のホスト名を使用してホストを保護します。別名を使用することで起こる混乱を回避するために、別名に対して HOST ルールを定義すると警告が発行されます。同様に、CA Access Control では、完全修飾名を使用せずに HOST を定義すると、警告が発行されます。これは、CA Access Control では、完全修飾名 (コンピュータ名.会社名.com など) でホストを識別するためです。

このコマンドの形式は以下のようになります。

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

systemIds

selang コマンドの実行対象であるホストのシステム ID を指定します。複数のホストを指定する場合は、システム ID のリストを丸かっこで囲み、各システム ID をスペースまたはカンマで区切ります。

policyModel@[*hostname*]

selang コマンドの実行対象である Policy Model のアドレスを指定します。複数の Policy Model を指定する場合は、Policy Model のアドレスのリストを丸かっこで囲み、Policy Model の各アドレスをスペースまたはカンマで区切ります。

hostname の指定がない場合、CA Access Control はローカル ホスト上の PMDB に接続しようとします。

注: ホストを明示的に指定するより Policy Model を使用する方が優れている点は、Policy Model が格納されているシステムが、Policy Model に定義されているすべてのシステムを、現在使用できないシステムも含めて、継続的に更新しようとすることです。Policy Model の詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

例: ユーザまたはグループに対するリモート ホスト更新の許可

ローカル データベースからリモート ホスト データベースを更新する権限をユーザに与えるには、リモート ホスト上で以下のコマンドを入力します。

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

ローカル データベースからリモート ホスト データベースを更新する権限をグループに与えるには、リモート ホスト上で以下のコマンドを入力します。

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

例: リモート Policy Model への selang コマンドの適用

後続のすべてのコマンドを端末 h1 上の Policy Model に適用するには、以下のコマンドを入力します。

```
hosts Policy@h1
```

Policy@h1 への接続が確立されると、次のメッセージが表示されます。

```
接続に成功しました。
```

これ以降に入力するすべてのコマンドは、ローカル ホストではなく Policy@h1 に送信されます。selang プロンプトが次のように変わります。

```
Remote_AC>
```

例: リモート ホストへの selang コマンドの適用

以降のコマンドをすべて端末 athena に適用するには、以下のコマンドを入力します。

```
hosts athena
```

athena への接続が確立されると、以下のメッセージが画面に表示されます。

```
(athena)
Successfully connected
情報: ターゲット ホストのバージョンは 2.50 です。
```

入力するすべてのコマンドは athena に適用され、ローカル ホストには送信されません。次の例のように、新しいユーザを追加すると、ユーザは athena のみに追加されます。

```
Remote_AC>newusr steve
(athena) USER steve の作成に成功しました。
```

join[x] コマンド - ユーザの内部グループへの追加

AC 環境で有効

join[x] コマンドは、ユーザを 1 つ以上の内部グループに追加するか、グループに関連するユーザのプロパティを変更します。指定するユーザまたはグループは、CA Access Control にすでに定義されている必要があります。

内部ユーザをグループに追加する場合は join を使用します。

エンタープライズ ユーザをグループに追加する場合は joinx を使用します。

注: このコマンドはネイティブ環境にもありますが、動作が異なります。

指定したグループ内の指定したユーザの以前のプロパティ セットはすべて、join コマンドのプロパティ セットで*完全に置き換えられます*。以前に定義した古いプロパティは、join コマンドで再度指定しない限り維持されません。

注: グループ プロパティの詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。

join コマンドを実行するには、以下の条件を少なくとも 1 つ満たしている必要があります。

- ADMIN 属性が割り当てられていること

  注: CA Access Control GROUP レコードとエンタープライズ グループをどちらも変更するには、MODIFY アクセス権限と JOIN アクセス権限が両方必要です。

- GROUP-ADMIN 属性で管理者権限を与えられたグループの適用範囲にグループ レコードが含まれていること
- グループの所有者であること
- ADMIN クラスの GROUP レコードのアクセス制御リストに CONNECT 権限が割り当てられていること

このコマンドの形式は以下のようになります。

```
{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 で指定されたユーザに GROUP-ADMIN 属性を割り当てます。

admin-

ユーザから GROUP-ADMIN 属性を削除します。

auditor

userName で指定されたユーザに GROUP-AUDIT 属性を割り当てます。

auditor-

ユーザから GROUP-AUDIT 属性を削除します。

gowner(*groupName*)

ユーザをグループ *groupName* に追加するように指定します。

group(*groupName* [,*groupName...*])

ユーザをメンバとして追加するグループ（複数可）を指定します。

nt

userName を Windows データベースのグループに関連付けます。

operator

userName で指定されたユーザに GROUP-OPERATOR 属性を割り当てます。

operator-

ユーザから GROUP-OPERATOR 属性を削除します。

owner(*Name*)

join レコードの所有者として CA Access Control ユーザまたはグループを指定します。接続を確立するときに所有者を指定しなかった場合は、接続を確立したユーザに所有者権限が割り当てられます。

pwmanager

userName で指定されたユーザに GROUP-PWMANAGER 属性を割り当てます。

regular

ユーザの管理フラグをリセットします。

unix

userName を UNIX セキュリティ システムのグループに関連付けます。

userName

グループ パラメータによって指定された 1 つまたは複数のグループに関連付ける(または、新しいプロパティ セットを使用して関連付け直す)ユーザを指定します。

join コマンドの場合、*userName* には USER レコードの名前を指定します。joinx コマンドの場合、*userName* にはエンタープライズ ユーザの名前を指定します。

例

- ユーザ Rorri が、ユーザ Bob を内部グループ staff に追加します。
  - Rorri に ADMIN 属性が割り当てられているとします。
  - 以下のデフォルト値が適用されるとします。
    - admin
    - auditor
    - owner(Rorri)
    - pwmanager

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

- ユーザ Rorri が、グループ staff の Sue の定義を変更します。Sue には現在 GROUP-AUDITOR 属性が割り当てられていて、Rorri は GROUP-PWMANAGER 属性を追加します。
  - Rorri に ADMIN 属性が割り当てられているとします。
  - 以下のデフォルト値が適用されるとします。
    - admin
    - owner(Rorri)

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

  このコマンドを実行すると、以前のレコードは削除されます。Sue の以前の属性に関するレコードは保存されません。したがって、Rorri は、Sue に現在必要な 2 つの属性を指定する必要があります。

詳細情報:

join[x]- コマンド - ユーザのグループからの削除 (P. 146)
show[x]grp コマンド - グループ プロパティの表示 (P. 170)
show[x]usr コマンド - ユーザ プロパティの表示 (P. 175)

join[x]- コマンド - ユーザのグループからの削除

AC 環境で有効

join[x]- は、内部グループからユーザを削除するコマンドです。

join- は、内部ユーザを内部グループから削除します。

joinx- は、エンタープライズ ユーザを内部グループから削除します。

注: join[-] コマンドはネイティブ環境にもありますが、動作が異なります。

join[x]- コマンドを使用するには、以下の条件のいずれかを満たす必要があります。

- ADMIN 属性が割り当てられていること

  注: CA Access Control GROUP レコードとネイティブ グループをどちらも変更するには、MODIFY アクセス権限と JOIN アクセス権限が両方必要です。
- GROUP-ADMIN 属性で管理者権限を与えられたグループの適用範囲にグループ レコードが含まれていること
- グループの所有者であること
- ADMIN クラスの GROUP レコードのアクセス制御リストに CONNECT 権限が割り当てられていること

このコマンドの形式は以下のようになります。

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

group(*groupName* [,*groupName...*])

ユーザを削除するグループ(複数可)を指定します。

userName

グループから削除するユーザを指定します。

join コマンドの場合、*userName* には USER レコードの名前を指定します。

joinx コマンドの場合、*userName* にはエンタープライズ ユーザの名前を指定します。

例

ユーザ Bill が、グループ PAYROLL からユーザ sales25 および sales43 を削除します。

ユーザ Bill に ADMIN 属性が割り当てられているとします。

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

詳細情報:

join[x] コマンド - ユーザの内部グループへの追加 (P. 142)
show[x]grp コマンド - グループ プロパティの表示 (P. 170)
show[x]usr コマンド - ユーザ プロパティの表示 (P. 175)

list コマンド - データベース レコードの一覧表示

AC 環境とネイティブ環境で有効

これは find コマンドと同じです。

詳細情報:

find コマンド - データベース レコードの一覧表示 (P. 132)

newfile コマンド - ファイル レコードの作成

AC 環境で有効

このコマンドについては、chfile コマンドの項で説明しています。

詳細情報:

chfile コマンド - ファイル レコードの変更 (P. 65)

new[x]grp コマンド - グループ レコードの作成

AC 環境で有効

このコマンドについては、chgrp コマンドの項で説明しています。

詳細情報:

ch[x]grp コマンド - グループ プロパティの変更 (P. 72)

newres コマンド - リソース レコードの作成

AC 環境で有効

このコマンドについては、chres コマンドの項で説明しています。

詳細情報:

chres コマンド - リソース レコードの変更 (P. 89)

new[x]usr コマンド - ユーザ レコードの作成

AC 環境で有効

このコマンドについては、ch[x]usr コマンドの項で説明しています。

詳細情報:

ch[x]usr コマンド - ユーザ プロパティの変更 (P. 107)

rename コマンド - データベース レコード名の変更

AC 環境で有効

データベース内のレコード名を変更します。変更すると、レコードは新しい名前でのみ認識されます。

注: SEOS クラス、UACC クラス、および ADMIN クラスのレコードの名前は変更できません。

rename コマンドを使用するには、レコードに対する適切な権限が必要です。CA Access Control では、ユーザに対し以下の条件がチェックされます。いずれかの条件が満たされるとチェックは終了します。

- ADMIN 属性が割り当てられていること
- GROUPADMIN 属性で管理者権限を与えられたグルーの有効範囲内に、目的のリソース レコードが含まれていること
- レコードの所有者であること
- ADMIN クラスにあるリソース クラスのレコードのアクセス制御リストに CREATE アクセス権限（editres の場合）が割り当てられていること

このコマンドの形式は以下のようになります。

```
rename className oldresourceName newresourceName
```

className

名前を変更するレコードが属するクラスを指定します。

oldresourceName

CA Access Control のレコードの現在の名前を指定します。

newresourceName

レコードに割り当てる新しい名前を指定します。

例

ユーザ ADMIN 1 が、Host クラスのレコード名 *spree3* を *spree4* に変更します。

- このセキュリティ管理者に ADMIN 属性が割り当てられているとします。

```
rename host spree3  spree4
```

rmfile コマンド - ファイル レコードの削除

AC 環境で有効

rmfile コマンドは、FILE クラスに属するレコードをデータベースから削除します。

ファイル レコードを削除するには、以下の条件のいずれかを満たしている必要があります。

- ADMIN 属性が割り当てられていること
- GROUP-ADMIN 属性で管理者権限を与えられたグループの有効範囲内にレコードが含まれていること
- ファイルの所有者であること
- ADMIN クラスの FILE レコードの ACL に DELETE アクセス権限が割り当てられていること

このコマンドの形式は以下のようになります。

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

fileName

削除するファイルを指定します。

各ファイル レコードは個別に処理されます。ファイルの処理中にエラーが発生すると、メッセージが発行され、リストの次のファイルから処理が続行されます。

例: ファイル保護の削除

セキュリティ管理者(ADMIN 属性が与えられている)が、ファイルの CA Access Control 保護を削除しようとしています。UNIX の場合は、以下のようなコマンドを実行します。

```
rmfile /etc/passwd
```

Windows では、同じことを行うコマンドは以下のようになります。

```
rmfile C:¥temp¥passwords.txt
```

詳細情報:

chfile コマンド - ファイル レコードの変更 (P. 65)
showfile コマンド - ファイルのプロパティの表示 (P. 168)

rm[x]grp コマンド - グループ レコードの削除

AC 環境で有効

rmgrp コマンドと rmxgrp コマンドは、1 つ以上のグループを CA Access Control から削除し、オプションでネイティブ環境から削除します。

注: rmgrp コマンドでは削除されないグループのグループ ID がデータベースに存在する可能性があります。たとえば、グループが、他のグループの所有者である場合、他のレコードの所有者である場合、またはリソースのアクセス制御リストに指定されている場合です。chgrp、chusr、chres、および authorize の各コマンドを必要に応じて実行して、手動により、所有者権限を変更し、削除するグループ レコードに関連するアクセス権限を削除します。また、sepurgedb ユーティリティを使用してデータベース内の不整合を自動的に解決することもできます。

注: rmgrp コマンドはネイティブ環境にもありますが、動作が異なります。

rmgrp コマンドを実行するには、ユーザは以下の条件を少なくとも 1 つ満たしている必要があります。

- ADMIN 属性が割り当てられていること
- GROUP-ADMIN 属性で管理者権限を与えられたグループの有効範囲内に、削除するグループが含まれていること
- 削除するグループの所有者であること
- AUDIT クラスの GROUP レコードに DELETE 権限が割り当てられていること

このコマンドの形式は以下のようになります。

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

groupName

削除する CA Access Control グループを指定します。

nt

(オプション) CA Access Control データベースからだけではなく、ローカル Windows システムからもグループを削除します。

unix

(オプション) CA Access Control データベースからだけではなく、ローカル UNIX システムからもグループを削除します。

例

ユーザ Joe が、データベースからグループ DEPT1 および DEPT2 を削除します。

- ユーザ Joe に SALES グループに対する GROUP-ADMIN 権限が割り当てられています。
- グループ DEPT1 および DEPT2 は SALES グループが所有しているとします。

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

詳細情報:

ch[x]grp コマンド - グループ プロパティの変更 (P. 72)
join[x] コマンド - ユーザの内部グループへの追加 (P. 142)
join[x]- コマンド - ユーザのグループからの削除 (P. 146)
show[x]grp コマンド - グループ プロパティの表示 (P. 170)
rmgrp コマンド - UNIX グループの削除 (P. 202)
rmgrp コマンド - Windows グループの削除 (P. 232)

rmres コマンド - リソースの削除

AC 環境で有効

rmres コマンドは、データベースからリソースを削除します。rmres コマンドを実行してレコードを削除できるレコードは、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 の各クラス、および任意のユーザ定義クラスに属しています。

注: このコマンドはネイティブ Windows 環境にもありますが、動作が異なります。

データベースからレコードを削除するには、以下のいずれかの条件を満たしている必要があります。

- ADMIN 属性が割り当てられていること
- GROUPADMIN 属性で管理者権限を与えられたグループの有効範囲内に、目的のリソース レコードが含まれていること
- リソース レコードの所有者であること
- ADMIN クラスにあるリソース クラスのレコードのアクセス制御リストに DELETE アクセス権限が割り当てられていること

このコマンドの形式は以下のようになります。

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

className

リソースが属するクラスの名前を指定します。CA Access Control に定義されているリソース クラスを一覧表示するには、find コマンドを実行します。詳細については、この章の find コマンドの説明を参照してください。

resourceName

削除するリソース レコードの名前を指定します。複数のリソースを削除する場合は、リソース名のリストを丸かっこで囲み、各リソース名をスペースまたはカンマで区切ります。

各リソース レコードは個別に処理されます。リソースの処理中にエラーが発生すると、メッセージが発行され、リストの次のリソースから処理が続行されます。

例

ユーザ Admin1 が、データベースの TERMINAL クラスからレコード TERMS を削除します。

- ユーザ Admin1 に ADMIN 属性が割り当てられているとします。

```
rmres TERMINAL TERMS
```

詳細情報:

chres コマンド - リソース レコードの変更 (P. 89)
showres コマンド - リソース プロパティの表示 (P. 172)
rmres コマンド - Windows リソースの削除 (P. 232)
find コマンド - データベース レコードの一覧表示 (P. 132)

rm[x]usr コマンド - ユーザ レコードの削除

AC 環境で有効

rmusr コマンドと rmxusr コマンドは、ユーザを CA Access Control データベースから削除し、また CA Access Control グループ レコードに存在するユーザ レコードの参照も削除します。

rmxusr は、エンタープライズ ユーザを CA Access Control データベースから削除します。rmusr は、内部ユーザをデータベースから削除します。rmusr コマンドは、オプションで、ユーザをネイティブ環境からも削除します。

注: rm[x]usr では削除されないユーザがデータベースに存在する可能性があります。たとえば、ユーザがグループまたは他のレコードの所有者である場合、またはユーザがリソースのアクセス制御リストに指定されている場合です。必要に応じて、ch[x]grp、ch[x]usr、ch[x]res、および authorize の各コマンドを実行して、所有者権限を手動で変更し、削除するユーザ レコードに関連するアクセス権限を削除します。また、sepurgedb ユーティリティを使用してデータベース内の不整合を自動的に解決することもできます。

注: rmusr コマンドはネイティブ環境にもありますが、動作が異なります。

rm[x]usr コマンドを実行するには、少なくとも以下の条件のいずれかを満たしている必要があります。

- ADMIN 属性が割り当てられていること
- GROUP-ADMIN 属性で管理者権限を与えられているグループの有効範囲内に削除するユーザ レコードが含まれていること
- ADMIN クラスの USER レコードのアクセス制御リストに DELETE 権限が割り当てられていること
- ユーザ レコードの所有者であること

ru は rmusr の省略形です。

rxu は rmxusr の省略形です。

このコマンドの形式は以下のようになります。

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

appl(homedir=yes)

（UNIX のみ）。ユーザのホーム ディレクトリを削除します。

この引数は、/home、/tmp、または /users にユーザのホーム ディレクトリがあるかどうかをチェックします。ホーム ディレクトリが別のディレクトリにある場合は、S99DELETE_postrmusrdir.sh スクリプトを編集して、そのホーム ディレクトリを組み込みます。

注: このオプションを指定する前に unix オプションを指定する必要があります。

nt

CA Access Control からだけでなく、Windows 環境からもユーザを削除します。

rmusr でのみ有効です。

userName

ユーザ レコードを定義します。

unix

CA Access Control からだけでなく、UNIX 環境からもユーザを削除します。

rmusr でのみ有効です。

例

以下のコマンドは、エンタープライズ ユーザ Terry および Jane を CA Access Control から削除します。

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

詳細情報:

ch[x]usr コマンド - ユーザ プロパティの変更 (P. 107)
show[x]usr コマンド - ユーザ プロパティの表示 (P. 175)
rmusr コマンド - UNIX ユーザの削除 (P. 203)
rmusr コマンド - Windows ユーザの削除 (P. 233)

ruler コマンド - 表示するプロパティの選択

AC 環境とネイティブ環境で有効

ruler コマンドはクラスのルーラを定義し、CA Access Control が表示するクラスのプロパティ セットを定義できるようにします。

ruler コマンドは、現在のセッションのホストにのみ適用されます。各ホストのプロパティは、個別のリストに表示されます。ホストを変更した場合、ruler コマンドで新しいホストのプロパティの表示は変更されません。

このコマンドを発行できるのは、以下のユーザです。

- ADMIN 属性、AUDITOR 属性、または OPERATOR 属性を持つユーザ。
- ルーラを設定する対象のクラスに対する読み取りアクセス権が ADMIN クラスに定義されているユーザ。たとえば、TERMINAL クラスを表すレコードに対する読み取りアクセス権が ADMIN クラスに定義されているユーザは、TERMINAL クラスのルーラを設定できます。

このコマンドの形式は以下のようになります。

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

className

表示を変更するクラスの名前です。

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

表示するプロパティを指定します。

props パラメータを省略すると、現在のルーラに含まれているプロパティの名前が表示されます。

all

クラスのすべてのプロパティを表示するように指定します。

propName

表示する CA Access Control プロパティを指定します。最高 40 プロパティを、スペースまたはカンマで区切って指定できます。

例

- ユーザ admin が、所有者と変更が通知されるユーザという 2 つのプロパティのみを各ユーザに表示するよう設定します。

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

- ユーザ admin が、クラス USER に対する現在のルーラのプロパティを表示するとします。

  ```
  ruler USER
  ```

- ユーザ admin が、CA Access Control のルーラの設定をデフォルトに戻し、USER クラスのすべてのプロパティを表示します。

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

詳細情報:

showfile コマンド - ファイルのプロパティの表示 (P. 168)
show[x]grp コマンド - グループ プロパティの表示 (P. 170)
showres コマンド - リソース プロパティの表示 (P. 172)
show[x]usr コマンド - ユーザ プロパティの表示 (P. 175)

setoptions コマンド - CA Access Control オプションの設定

AC 環境で有効

setoptions コマンドを使用すると、実行中のシステムでシステム全体の CA Access Control オプションを設定します。たとえば、setoptions を使用して、個別のクラスまたはすべてのクラスのセキュリティ チェックの有効化と無効化、パスワード ポリシーの設定、および CA Access Control オプションの現在の設定の一覧表示を行うことができます。

注: このコマンドは Windows 環境にもありますが、動作が異なります。

setoptions コマンドを使用するには ADMIN 属性が必要です。ただし、setoptions list コマンドは AUDITOR 属性または OPERATOR 属性があれば使用できます。

このコマンドの形式は以下のようになります。

```
{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

累積グループ権限（ACCGRR）オプションを有効にします。

デフォルト値は enabled です。

accgrr-

累積グループ権限（ACCGRR）オプションを無効にします。

accpacl

すべてのリソースでの PACL の使用を有効にします。

accpacl-

PACL の使用を無効にします。

class (*className*)

CA Access Control クラスを設定またはクリアします。

class+(*className*)

1 つ以上の CA Access Control クラスを有効にします。CA Access Control でそのクラスのリソースを保護するためには、クラスが有効である必要があります。クラスの有効化は、クラスに属するリソースへのアクセスを許可するために必要なレコードを定義した後に行う必要があります。CA Access Control で提供されるリソース クラスの詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

以下のいずれかの値を使用します。

- CA Access Control クラスの名前
- SECLEVEL。これにより、セキュリティ レベル チェックが有効になります。
- PASSWORD。パスワード ルールが有効になります。Windows では、任意の長さのパスワードを使用できるようになります。

class-(*className*)

1 つ以上の CA Access Control クラスを無効にします。無効なクラスに属するリソースは保護されません。以下のいずれかの値を使用します。

- CA Access Control クラスの名前
- SECLEVEL。セキュリティ レベル チェックを無効にします。
- PASSWORD。パスワード ルールが無効になります。Windows では、長いパスワードも無効になります。

GROUP、SECFILE、SEOS、UACC、および USER の各クラスを無効にすることはできません。

cng_adminpwd

PWMANAGER 属性を持つユーザが ADMIN ユーザのパスワードを変更できるようにします。

cng_adminpwd-

PWMANAGER 属性を持つユーザが ADMIN ユーザのパスワードを変更できないようにします。これがデフォルトの設定です。

cng_ownpwd

ユーザが selang を使用してパスワードを変更できるようにします。

cng_ownpwd-

ユーザが selang を使用してパスワードを変更できないようにします。これがデフォルトの設定です。

cwarnlist

警告モードのクラスに関するデータのテーブルを表示します。

dms{+|-}(*dms@hostname*)

このデータベースの DMS データベースリストに対する DMS データベースを追加または削除します。

flags{+|-} (I|W)

クラスに対して関連する機能を設定またはクリアします。有効な値は以下のとおりです。

I

指定したクラスで、オブジェクトの大文字と小文字を区別するかどうか。

W

指定したクラスの警告モード。

注: フラグは大文字と小文字を区別します。大文字を使用してください。

history(*NStoredPasswords*)

履歴リストに保存するパスワード履歴の数を指定します。パスワードが変更されると、前回のパスワードがリストに追加され、必要に応じて最も古いパスワードがリストから削除されます。CA Access Control では、ユーザがリストに含まれているパスワードを変更できないようにします。

1 から 24 までの整数を入力します。0 を指定すると、パスワードは保存されません。

Windows の場合、history オプションを使用すると、8 文字より長いパスワードを使用できるようになります。 パスワード格納時に使用される暗号方式は、setoptions bidirectional または bidirectional- オプションで決まります。

UNIX の場合、長いパスワードが有効かどうかに history オプションは*影響しません*。長いパスワードを有効にするかどうかには、passwd_local_encryption_method 環境設定を使用します。

history-

パスワード履歴のチェックを無効にします。

Windows では、このオプションにより長いパスワードが使用できなくなります。

inactive(*nDays*)

ユーザのログインを一時停止するまでの非アクティブ状態の日数を指定します。非アクティブ状態の日とは、ユーザがログインできない日を指します。正の整数を入力します。inactive を 0 に設定すると、inactive- パラメータを使用した場合と同じ結果になります。

inactive-

非アクティブ ログイン チェックを無効にします。

interval(*nDays*)

パスワードの設定または変更後、ユーザに対して新しいパスワードの入力を促すメッセージを表示するまでの経過日数を設定します。正の整数または 0 を入力します。interval を 0 に設定すると、ユーザに対するパスワード期間のチェックは無効になります。パスワードに有効期限を設定しない場合は、interval を 0 に設定します。

ユーザのログイン スクリプトに segrace ユーティリティが含まれている場合は、指定された日数が経過すると、現在のパスワードが期限切れになったことがユーザに通知されます。通知を受けたユーザは、ただちにパスワードを更新するか、猶予ログイン回数に達するまで古いパスワードを引き続き使用することができます。猶予ログイン回数に達すると、ユーザはシステムへのアクセスを拒否されるため、システム管理者に連絡して新しいパスワードを設定する必要があります。

interval-

パスワード期間の設定を取り消します。

is_dms+

現在のデータベースを DMS に指定します。

is_dms-

現在のデータベースの DMS としての指定を解除します。

list

CA Access Control の現在の設定を画面に表示します。

maxlogins(*nLogins*)

ユーザが同時にログインできる端末台数の最大値を設定します。値 0(ゼロ)は、同時に任意の数の端末からログインできることを意味します。ユーザのユーザ レコードに値を指定すると、この値より優先されます。

注: maxlogins を 1 に設定すると、selang を実行できません。この場合、CA Access Control を停止し、maxlogins の設定を 2 以上の値に変更し、CA Access Control を再起動する必要があります。

注: Unix と Linux のオペレーティング システム上でのみ有効です。

maxlogins-

グローバルな最大ログイン回数のチェックを無効にします。ユーザ レコードでログインが制限されていない限り、ユーザがログインできる端末台数は無制限となります。

min_life(*NDays*)

変更したパスワードを再度変更できるようになるまでの最短日数を設定します。正の整数を入力します。

password

パスワード オプションを設定します。

rules

新しいパスワードの品質をチェックする際に使用される 1 つ以上のパスワード ルールを設定します。ルールは以下のとおりです。

alpha(*nCharacters*)

新しいパスワードで使用する必要がある英字の最小文字数を設定します。整数を入力します。

alphanum(*nCharacters*)

新しいパスワードで使用する必要がある英数字の最小文字数を設定します。整数を入力します。

bidirectional

パスワードが他のシステムに PMDB の一部として送信されるときに、クリア テキスト形式で（暗号化されたメッセージ内で）配信するように指定します。

UNIX の場合、このオプションは passwd セクションに以下の値を設定することに相当します。

```
Passwd_distribution_encryption_mode=bidirectional
```

注: setoptions コマンドを使用するのではなく、環境設定を行うことをお勧めします。

Windows の場合、パスワードは以下のレジストリ値で指定された暗号方式を使用して履歴リストに格納されます。

```
HKEY_LOCAL_MACHINE¥SOFTWARE¥ComputerAssociates¥AccessControl¥Encryption Package
```

bidirectional-

パスワードがハッシュ暗号化形式で送信されるように指定します。

Windows の場合、使用されるハッシュ関数は SHA-1 です。

UNIX の場合、このオプションは passwd セクションに以下の値を設定することに相当します。

```
Passwd_distribution_encryption_mode=compatibility
```

注: setoptions コマンドを使用するのではなく、環境設定を行うことをお勧めします。

このオプションを指定すると、長いパスワードを異種オペレーティング システム間で送信できなくなります。

grace(*nLogins*)

ユーザのアカウントが一時停止になるまでに猶予ログインできる最大回数を設定します。猶予ログイン回数には、0 ～ 255 の値を指定する必要があります。

lowercase(*nCharacters*)

新しいパスワードで使用する必要がある文字の小文字の最小数を指定します。整数を入力します。

min_len(*nCharacters*)

パスワードの最小文字数を設定します。新しいパスワードで使用する必要がある文字の合計最小数を指定します。

max_len(*nCharacters*)

パスワードの最大文字数を設定します。新しいパスワードで使用する必要がある文字の合計最大数を指定します。

max_rep(*nCharacters*)

新しいパスワードで使用する必要がある同じ文字の最大繰り返し回数を設定します。整数を入力します。

namechk

パスワードにユーザ名の一部または全部が含まれているかどうかをチェックします。デフォルトでは、このチェックが実行されます。

namechk-

namechk チェックをオフにします。

numeric(*nCharacters*)

新しいパスワードで使用する必要がある数字の合計最小数を指定します。整数を入力します。

oldpwchk

新しいパスワードに古いパスワードの一部または全部が含まれているかどうかをチェックします。デフォルトでは、このチェックが実行されます。

注: Unix と Linux のオペレーティング システム上でのみ有効です。

oldpwchk-

oldpwchk をオフにします。

prohibited(*prohibitedCharacters*)

ユーザがパスワードで使用できない文字を指定します。使用を禁止する文字を入力してください。

注: Tab キーの使用をブロックするために、「¥」および「t」両方の制御文字が禁止文字リストに指定されていることを確認するようにお勧めします。

special(*nCharacters*)

新しいパスワードで使用する必要がある特殊文字の最小数を指定します。整数を入力します。

sub_str_len(*nCharacters*)

新しいパスワードと古いパスワードとで共通する文字の最大数を指定します。整数を入力します。

uppercase(*nCharacters*)

新しいパスワードで使用する必要がある英字の大文字の最小数を設定します。整数を入力します。

use_dbdict | use_dbdict-

パスワード辞書を設定します。use_dbdict はトークンを **db** に設定し、パスワードを CA Access Control データベースの単語と照合して比較します。use_dbdict- トークンを **file** に設定し、UNIX の場合は seos.ini ファイル、Windows の場合は Windows レジストリに指定されたファイルとパスワードを照合して比較します。

rules-

パスワード品質のチェックを無効にします。 rules 引数で指定したルールは、パスワード品質のチェックに使用されません。

例: CA Access Control オプションの設定

- ユーザ John が、オペレータ アクションの保護に使用される導入先定義のクラスである OpsAct クラスを有効にするとします。

  ユーザ John に ADMIN 属性が割り当てられているとします。

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

- ユーザ Mike が、6 文字以上のパスワードをユーザに選択させるパスワード ポリシーを設定します。さらに、パスワード ポリシーの適用を有効にします。

  ユーザ Mike に ADMIN 属性が割り当てられているとします。

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

- ユーザ SecAdmin がセキュリティ レベル チェックを有効にします。

  ユーザ SecAdmin に ADMIN 属性が割り当てられているとします。

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

- ユーザ Janani が、このデータベースの通知の送信先 DMS を設定します。

  ユーザ Janani に ADMIN 属性が割り当てられているとします。

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

例: クラスを警告モードに設定する

クラスを警告モードに設定するには、そのクラスの Warning プロパティを設定します。このためには、以下のように setoptions の selang コマンドを実行します。

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

classname

警告モードに設定するクラスの名前を定義します。

注: W フラグは大文字と小文字の区別があるので、大文字で指定する必要があります。

クラスの警告モードをオフにするには、以下のように setoptions コマンドを使用します。

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

詳細情報:

setoptions コマンド - CA Access Control Windows オプションの設定 (P. 234)

search コマンド - データベース レコードの一覧表示

AC 環境とネイティブ環境で有効

これは find コマンドと同じです。

詳細情報:

find コマンド - データベース レコードの一覧表示 (P. 132)

showfile コマンド - ファイルのプロパティの表示

AC 環境で有効

showfile コマンドは、ファイル レコードのプロパティを一覧表示します。プロパティは、アルファベット順に一覧表示されます。CA Access Control では、各レコードを個別に処理し、十分な権限を持つリソースに対してのみ情報を表示します。

注: このコマンドはネイティブ環境にもありますが、動作が異なります。

showfile コマンドを実行するには、以下の条件を少なくとも 1 つ満たしている必要があります。

- 少なくとも、ADMIN 属性、AUDITOR 属性、および OPERATOR 属性のいずれかが割り当てられていること
- ファイルの所有者であること
- ADMIN クラスの FILE クラス レコードを表すオブジェクトのアクセス制御リストに読み取り権限が割り当てられていること
- ファイルを所有するグループまたはファイルを所有するグループの親グループで、GROUP-ADMIN 属性または GROUP-AUDITOR 属性が割り当てられていること

このコマンドの形式は以下のようになります。

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

addprops(*propName* [,*propName* ...])

このクエリでのみ使用するクラス ルーラに追加するプロパティを定義します。

fileName

一覧表示するプロパティを含むファイル レコードの名前を指定します。

各ファイル レコードは個別に処理されます。ファイルの処理中にエラーが発生すると、メッセージが発行され、リストの次のファイルから処理が続行されます。

fileName にワイルドカード文字を含めて、複数のファイル名に一致するようにできます。

UNIX の場合、名前に特殊文字またはスペースが使用されているファイルのプロパティを表示するには、ファイル名の前にスラッシュ (/) を追加します。

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトの query_size 設定は 100 です。

nt

Windows ファイル属性と CA Access Control のプロパティを表示します。

props(all|*propName* [,*propName* ...])

このクエリと今後のクエリで使用する、このクラス用の新しいルーラを定義します。

unix

UNIX ファイル属性と CA Access Control のプロパティを表示します。

useprops(*propName* [,*propName* ...])

このクエリでのみ使用するルーラを定義します。クラス ルーラへの影響はありません。

例

ユーザ root がファイル レコード /etc/passwd のプロパティを一覧表示するとします。

- ユーザ root に ADMIN 属性が割り当てられているとします。

```
showfile /etc/passwd
```

詳細情報:

checklogin コマンド - ログイン情報の取得 (P. 62)
chfile コマンド - ファイル レコードの変更 (P. 65)
rmfile コマンド - ファイル レコードの削除 (P. 150)
showfile コマンド - ネイティブ ファイルのプロパティの表示 (P. 203)

show[x]grp コマンド - グループ プロパティの表示

AC 環境で有効

show[x]grp コマンドは、グループ レコードのすべての CA Access Control プロパティの設定を表示します。オプションで、ネイティブ環境プロパティも表示されます。

注: showgrp コマンドはネイティブ環境にもありますが、動作が異なります。

show[x]grp コマンドを実行するには、以下の条件の最低 1 つを満たしている必要があります。

- 少なくとも、ADMIN 属性、AUDITOR 属性、および OPERATOR 属性のいずれかが割り当てられていること
- 一覧表示する各グループに GROUP-ADMIN 属性または GROUP-AUDITOR 属性が割り当てられている、あるいは一覧表示する各グループが GROUP-ADMIN 属性が割り当てられているグループの有効範囲内にあること
- グループの所有者であること
- ADMIN クラスの GROUP レコードのアクセス制御リストに読み取り権限が割り当てられていること

このコマンドの形式は以下のようになります。

```
{showgrp|sg} {groupName |groupName [,groupName...] )} ¥
    [addprops(propName[,propName ...])] ¥
    [next] ¥
    [props(all | propName[,propName ...])] ¥
    [useprops(propName[,propName ...])] ¥
    [nt|unix]
```

addprops(*propName* [,*propName* ...])

このクエリでのみ使用するルーラに追加するプロパティを定義します。

groupName

プロパティを一覧表示するグループの名前を指定します。

groupName には、ワイルドカード文字を使用できます。

UNIX の場合、名前に特殊文字またはスペースが使用されているグループのプロパティを表示するには、グループ名の前にスラッシュ (/) を追加します。

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトの query_size は 100 です。

nt

データベースのプロパティおよびローカル Windows システムのグループの詳細情報を表示します。

props(all|*propName* [,*propName* ...])

このクエリと今後のクエリで使用する、このクラス用のルーラを定義します。

useprops(*propName* [,*propName* ...])

このクエリでのみ使用するルーラを定義します。クラス ルーラへの影響はありません。

unix

データベースのプロパティおよびローカル UNIX システムのグループの詳細情報を表示します。

例

- ユーザ root が、security グループのプロパティを表示します。
  - ユーザ root にセキュリティ グループの GROUP-ADMIN 属性が割り当てられているとします。

  ```
  showgrp security
  ```

- ユーザ admin がすべてのエンタープライズ グループのプロパティを表示します。
  - ユーザ admin に ADMIN 属性および AUDITOR 属性が割り当てられているとします。

  ```
  showxgrp *
  ```

  CA Access Control に定義されているすべてのエンタープライズ グループのプロパティが一覧表示されます。

詳細情報:

ch[x]grp コマンド - グループ プロパティの変更 (P. 72)
rm[x]grp コマンド - グループ レコードの削除 (P. 151)
showgrp コマンド - ネイティブ グループのプロパティの表示 (P. 205)

showres コマンド - リソース プロパティの表示

AC 環境で有効

showres コマンドは、データベースのクラスに属するリソースのプロパティを表示します。プロパティは、アルファベット順に一覧表示されます。showres コマンドを実行して一覧表示できるクラスは、ACVAR、ADMIN、CATEGORY、CONNECT、FILE、GHOST、GSUDO、GTERMINAL、HOST、HOSTNET、HOSTNP、SECFILE、SECLABEL、SUDO、SURROGATE、TERMINAL、PROGRAM、PROCESS、TCP、UACC の各クラスおよび任意のユーザ定義クラスです。CA Access Control では、各リソースを個別に処理し、十分な権限を持つリソースに対してのみ情報を表示します。

注: このコマンドはネイティブ Windows 環境にもありますが、動作が異なります。

また、showres コマンドは、untrusted になったすべてのプログラムに関する情報も表示します。次の情報が表示されます。

- プログラムが untrusted になった理由
- そのプログラムに最後にアクセスしたユーザの UID(ただし、このユーザが原因でプログラムが untrusted になったとは限りません)。
- このユーザがそのプログラムにアクセスした日時

showres コマンドを実行するには、以下の条件を少なくとも 1 つ満たしている必要があります。

- 少なくとも、ADMIN 属性、AUDITOR 属性、および OPERATOR 属性のいずれかが割り当てられていること
- リソースの所有者であること
- ADMIN クラスで、目的のリソース クラス レコードを表すオブジェクトのアクセス制御リストに読み取り権限が割り当てられていること

このコマンドの形式は以下のようになります。

```
{showres|sr} className resourceName ¥
    [addprops(propName [,propName...])] ¥
    [next] ¥
    [props(all | propName [,propName...])] ¥
    [useprops(propName [,propName...])]
```

addprops(*propName* [,*propName*...])

このクエリでのみ使用する現在のルーラに追加するプロパティを定義します。

className

リソースが属するクラスの名前を指定します。CA Access Control に定義されているリソース クラスを一覧表示するには、find コマンドを実行します。

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトのクエリ サイズは 100 に設定されています。

props(all|*propName* [,*propName* ...])

このクエリと今後のクエリで使用する、このクラス用の新しいルーラを定義します。

resourceName

一覧表示するプロパティを含むリソース レコードの名前を指定します。複数のリソースのプロパティを一覧表示する場合は、リソース名のリストを丸かっこで囲み、各リソース名をスペースまたはカンマで区切ります。

各リソース レコードは個別に処理されます。リソースの処理中にエラーが発生すると、メッセージが発行され、リストの次のリソースから処理が続行されます。

resourceName には、ワイルドカード文字を使用できます。

UNIX の場合、名前に特殊文字またはスペースが使用されている単一リソース レコードのプロパティを表示するには、リソース名の前にスラッシュ (/) を追加します。

useprops(*propName* [,*propName* ...])

このクエリでのみ使用するルーラを定義します。クラス ルーラへの影響はありません。

例: リスト レコード プロパティ

この例では、ユーザ Admin1 は、TERMINAL クラスのレコードのうち、マスク ath* に名前が一致するレコードのプロパティを一覧表示します。

ユーザ Admin1 に ADMIN 属性および AUDITOR 属性が割り当てられている。

```
showres TERMINAL ath*
```

例: ホスト属性のリスト表示

この例では、ユーザ Admin1 は、HNODE クラスのローカル ホストの属性をリスト表示します。

```
AC> showres HNODE '__local__'
(localhost)
Data for HNODE '__local__'
------------------------------------------------------------
所有者              : LOCALHOST¥Administrator (USER)
作成日時            : 13-Oct-2010 12:34
更新日時            : 13-Oct-2010 02:34
更新者              : LOCALHOST¥Administrator (USER)
属性                :
    REGISTERED_NAME=localhost.domain.com
    MAC_ADDRESS=00-50-56-B5-6B-XD
```

この例では、コマンドは以下の属性を返します。

- REGISTERED_NAME=localhost.domain.com
- MAC_ADDRESS=00-50-56-B5-6B-XD

詳細情報:

chres コマンド - リソース レコードの変更 (P. 89)
rmres コマンド - リソースの削除 (P. 152)
showres コマンド - ネイティブ リソース プロパティの表示 (P. 239)
find コマンド - データベース レコードの一覧表示 (P. 132)

show[x]usr コマンド - ユーザ プロパティの表示

AC 環境で有効

show[x]usr コマンドは、CA Access Control に定義されている 1 人以上のユーザのすべてのプロパティの値を表示します。

内部ユーザのプロパティを表示するには、showusr を使用します。エンタープライズ ユーザのプロパティを表示するには、showxusr を使用します。

注: showusr コマンドはネイティブ環境にもありますが、動作が異なります。

自分のユーザ レコードのプロパティはいつでも一覧表示できます。他のユーザのレコードのプロパティを一覧表示するには、以下の条件のいずれかを満たしている必要があります。

- ユーザ レコードの所有者であること
- 少なくとも、ADMIN 属性、AUDITOR 属性、および OPERATOR 属性のいずれかが割り当てられていること
- ADMIN、AUDITOR、および OPERATOR の各グループ属性の少なくとも 1 つで管理者権限を与えられたグループの有効範囲内にユーザ レコードが含まれていること
- ADMIN クラスの USER レコードのアクセス制御リストに読み取り権限が割り当てられていること

このコマンドの形式は以下のようになります。

```
{showusr|su |showxusr |sxu } [ {userName |(userName [,userName...]) } ] ¥
    [addprops(propName [,propName...])] ¥
    [next] ¥
    [props( all | propName [,propName...])] ¥
    [useprops(propName[,propName...])] ¥
    [nt|unix]
```

addprops(*propName* [,*propName*...])

このクエリでのみ使用する現在のルーラに追加するプロパティを定義します。

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトのクエリ サイズは 100 に設定されています。

nt

データベースのプロパティおよびユーザの Windows プロパティを表示します。

props(all|*propName* [,*propName* ...])

このクエリと今後のクエリで使用する、このクラス用の新しいルーラを定義します。

unix

データベースのプロパティおよびユーザの UNIX プロパティを表示します。

userName

ユーザの名前を指定します。ワイルドカード文字を使用できます。

UNIX の場合、名前に特殊文字またはスペースが使用されている単一ユーザ レコードのプロパティを表示するには、グループ名の前にスラッシュ(/)を追加します。

userName の指定がない場合は、自分のユーザ レコードのプロパティが表示されます。

useprops(*propName* [,*propName* ...])

このクエリでのみ使用するルーラを定義します。クラス ルーラへの影響はありません。

例

- ユーザ root が、エンタープライズ ユーザ Robin のプロパティを一覧表示します。ユーザ root に ADMIN 属性および AUDITOR 属性が割り当てられているとします。

  ```
  showxusr Robin
  ```

- ユーザ root が、エンタープライズ ユーザ Robin および Leslie のユーザ プロパティを一覧表示します。ユーザ root に ADMIN 属性および AUDITOR 属性が割り当てられているとします。

  ```
  showxusr (Robin,Leslie)
  ```

詳細情報:

rm[x]usr コマンド - ユーザ レコードの削除 (P. 154)
ch[x]usr コマンド - ユーザ プロパティの変更 (P. 107)
showusr コマンド - ネイティブ ユーザ プロパティの表示 (P. 206)

source コマンド - ファイルからのコマンドの実行

すべての環境で有効

source コマンドを使用すると、ファイルに保存されている 1 つ以上の selang コマンドを実行することができます。CA Access Control は、指定されたファイルを読み取り、コマンドを実行して、selang プロンプトを返します。データベースに定義されているすべてのユーザがこのコマンドを実行できます。

このコマンドは、UNIX の csh や tcsh の source コマンドと同様のコマンドです。

このコマンドの形式は以下のようになります。

```
source fileName
```

fileName

selang コマンドが保存されているファイルの名前を指定します。

例

ユーザ admin が、initf1 というファイル内のコマンドを実行します。この場合は、以下のコマンドを入力します。

```
source initf1
```

start dbexport コマンド - データベース エクスポートの開始

AC 環境で有効

start dbexport コマンドを使用すると、接続しているホストの CA Access Control データベースをエクスポートし、出力をバッファにコピーします。PMDB に接続している場合、PMD データベースをエクスポートする場合にもこのコマンドを使用できます。

Note: 出力を表示するには、get dbexport コマンドを使用します。

このコマンドの形式は以下のようになります。

```
start dbexport [pmdname(name)] [filter("CLASS, CLASS...")] [param("depend=yes")]
[param("edit=yes")]
```

filter("*CLASS*, *CLASS*...")

(オプション) データベースからエクスポートするクラスを定義します。クラスを指定しない場合、データベース内のすべてのルールがエクスポートされます。

param("depend=yes")

(オプション) フィルタ パラメータで指定したクラスおよび依存するクラスをエクスポートするように指定します。このパラメータを指定すると、CA Access Control では指定されたクラスおよび以下の依存するクラスをエクスポートします。

- 特定のクラスのリソースを変更するルールをエクスポートし、そのクラスに対応するリソース グループが含まれる場合、CA Access Control はそのリソース グループに存在するリソースを変更するルールもエクスポートします。
- 特定のリソース グループのリソースを変更するルールをエクスポートする場合、CA Access Control はそのリソース グループのメンバ リソースを変更するルールもエクスポートします。
- 特定のクラスのリソースを変更するルールをエクスポートし、そのクラスに PACL が含まれる場合、CA Access Control は PROGRAM クラスに存在するリソースを変更するルールもエクスポートします。
- 特定のクラスのリソースを変更するルールをエクスポートし、そのクラスに CALACL が含まれる場合、CA Access Control は CALENDAR クラスに存在するリソースを変更するルールもエクスポートします。
- 特定のクラスのリソースを変更するルールをエクスポートし、そのクラスのリソースの 1 つが CONTAINER リソースグループのメンバである場合、CA Access Control は CONTAINER クラスのリソースを変更するルール、および各 CONTAINER リソース ループのメンバとなっているリソースを変更するルールをエクスポートします。

param("edit=yes")

(オプション) CA Access Control は、新しいリソースまたはアクセサを作成する各ルールをリソースまたはアクセサを変更するルールに変更します。

例: このパラメータを指定すると、CA Access Control ではすべての newres ルールを editres ルールに変更します。

pmdname(*name*)

(オプション) エクスポートする PMD データベースの名前を指定します。

例: データベース エクスポートの開始

以下には、FILE クラスおよび GFILE クラスのリソースを変更するルールのエクスポートを開始する例を示します。ルールは、seosdb (接続しているホストの CA Access Control データベース)からエクスポートされます。

```
start dbexport filter("FILE, GFILE")
```

例: 依存するクラスを含むデータベース エクスポートの開始

以下には、FILE クラス リソース、および FILE クラス リソースに依存するクラスを変更するルールのエクスポートを開始し、新しいリソースまたはアクセサを作成する各ルールをリソースまたはアクセサを変更するルールに変更する例を示します。

```
start dbexport filter("FILE") param("depend=yes edit=yes")
```

詳細情報:

get dbexport コマンド - エクスポートされたデータベース ルールの取得 (P. 134)

start devcalc コマンド - ポリシー偏差計算の開始

AC 環境で有効

start devcalc コマンドは、ポリシー偏差計算を開始し、偏差ステータスを送信します。偏差データはローカルのポリシー偏差データ ファイル(deviation.dat)に格納され、ポリシー偏差ステータスは設定された 1 つ以上の DH を通じて DMS に送信されます。計算された偏差データを取得するには、get devcalc コマンドを実行する必要があります。

注: 偏差計算を手動で実行する必要はありません。ユーザが拡張ポリシー管理を使用すれば、policyfetcher はこれを定期的に実行します。エンタープライズ レポートが有効になっていれば、レポート エージェントもこれを定期的に実行します。ポリシー偏差計算の詳細については、「*エンタープライズ管理ガイド*」を参照してください。

start devcalc コマンドを実行するには、使用コンピュータに対する端末アクセス権限と DEVCALC サブ管理クラスに対する実行アクセス権限が必要です。

このコマンドの形式は以下のようになります。

```
start devcalc [params("-pn name#xx -strict -nonotify -precise")]
```

-nonotify

(オプション) devcalc が DH を通じて DMS に偏差ステータスを*送信しない*ように設定します。

注: policyfetcher が実行する偏差計算コマンドは devcalc_command 環境設定で定義されており、デフォルトでこの設定が使用され、偏差ステータスを 2 度送信してしまわないようになっています。

-pn *name#xx*

(オプション) 偏差計算機能による偏差の計算対象となる POLICY オブジェクト(ポリシー バージョン)のカンマで区切られたリストを指定します。ポリシーの指定がない場合、偏差計算機能はローカル ホストにデプロイされたすべてのポリシーに対する偏差を計算します。

-strict

(オプション) ローカル HNODE オブジェクトに関連付けられているポリシーと、最初に使用できる DMS 上の HNODE に関連付けられているポリシーとを比較します。

通常、偏差計算機能はローカル ホスト上でのみ偏差をチェックします。このオプションを指定すると、偏差計算機能はローカルのポリシーとリストの最初の DMS にあるポリシーも比較します。比較される内容は以下のとおりです。

1. ローカル ホストを表す HNODE オブジェクトに関連付けられたポリシーのリスト。
2. HNODE オブジェクトに関連付けられた各 POLICY オブジェクトのポリシーのステータス。
3. HNODE オブジェクトに関連付けられた各 POLICY オブジェクトのポリシーのシグネチャ。

このオプションは、偏差計算の結果を検証する必要がある場合に使用します。

注: 偏差計算を同時に実行するエンドポイント数が多いと、DMS に対する負荷が重くなります。DMS リストを使用するようにエンドポイントを構成すること、または、階層を小さい階層に分けて、このオプションをその小さい階層に対して使用することをお勧めします。

-precise

(オプション)エンドポイント データベースには存在するがポリシー内では検出されない追加オブジェクト、プロパティ、および値も偏差レポートに表示されるように指定します。デフォルトでは、存在しない項目および一致しない項目のみがレポートに表示されます。このオプションは、エンドポイント データベースの内容を表示させてデプロイ済みポリシーと比較する場合に使用します。

例: 特定のポリシーに対するポリシー偏差計算の開始

以下の例は、start devcalc コマンドを使用して、myPolicy というポリシーの 2 番目のバージョンに対するポリシー偏差を計算し、偏差ステータスをローカル CA Access Control データベースに指定されている DMS リストに送信しています。

```
AC> start devcalc params("-pn myPolicy#02")
```

start_transaction コマンド - デュアル コントロール トランザクションの記録の開始

AC 環境内の UNIX ホストで有効

start_transaction および end_transaction は、デュアル コントロール PMDB プロセスの未処理のトランザクションを保存するファイルを作成するコマンドです。このプロセスは 1 つ以上のコマンドで構成されています。トランザクションにコマンドを入力する管理者(ADMIN 属性を持つ任意のユーザ)を Maker(作成者)といいます。このコマンドは、Checker(チェッカ)によって許可されてから、PMDB で実行する必要があります。Checker とは、Maker *ではない* 任意の管理者です。

Checker は、処理前のトランザクションをロックする必要があります。Checker がトランザクションをロックするまでの間、Maker は、トランザクションの取得、コマンドの変更、およびトランザクションの削除を行うことができます (詳細については、「*リファレンス ガイド*」の sepmd ユーティリティの説明を参照)。Maker が end_transaction コマンドを入力すると、トランザクションに一意の識別番号が表示されます。Maker がトランザクションを後で編集または取得する場合は、この識別番号を start_transaction コマンドのトランザクション名の後に追加する必要があります。Maker がトランザクションを取得すると、Maker の名前、トランザクションの識別番号、および簡単な説明(*transactionName* パラメータに説明が入力されている場合)が表示されます。

Maker は他の Maker のトランザクションを変更できません。トランザクションで使用されているオブジェクトは、そのコマンドの処理が終了するまで、別のトランザクションで他の Maker が使用することはできません。

未処理の各トランザクションは、Checker が処理するまで個別のファイルに保持されます。Checker はトランザクションを許可または拒否できます。トランザクションが許可されると、そのコマンドが実行され、PMDB が変更されます。Checker がトランザクションを拒否すると、そのコマンドは削除され、PMDB は変更されません。

Maker が最後に end_transaction コマンドを入力すると、そのトランザクションの ID 番号が表示されます。コマンドは以下の場合に失敗します。

- まだ処理の完了していない別のトランザクションで使用されているオブジェクトをコマンドが参照する場合
- Maker に関係するコマンドの場合。自分自身を変更することはできません。
- コマンドに無効な構文が含まれている場合
- コマンドが、存在しないオブジェクトを参照する場合。この場合は警告メッセージが表示されます。
- start_transaction コマンドおよび end_transaction コマンドを実行するには、ADMIN 属性が必要です。
- hosts コマンドは、start_transaction コマンドおよび end_transaction コマンドの起動前に実行する必要があるため、hosts コマンドの実行許可が与えられている必要があります。

注: デュアル コントロールの詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

使用上の注意

- hosts コマンドは start_transaction コマンドおよび end_transaction コマンドの起動前に実行する必要があるため、PMDB の名前を「maker」と指定する必要があります。
- start_transaction コマンドおよび end_transaction コマンドが機能するためには、pmd.ini ファイルにある is_maker_checker トークン、および seos.ini ファイルの［pmd］セクションにある is_maker_checker トークンの値を yes に設定する必要があります。

このコマンドの形式は以下のようになります。

```
start_transaction transactionName [transactionId]
.
.
.
end_transaction
```

transactionName

トランザクションの名前または説明を指定します。最大 256 文字の英数字から成る文字列を入力できます。

transactionId

作成時にトランザクションに指定された一意の番号を指定します。この識別番号は、トランザクションの作成時に自動的に表示されます。同じトランザクションの更新時には、この ID 番号を指定する必要があります。

例

- Maker Sally は、PMDB へユーザ Anne を追加し、そのシステムへのアクセスを平日の午前 8 時から午後 8 時までに制限したいとします。また、Sally は tty30 端末へのデフォルト アクセス権を読み取り専用に変更したいとします。Sally は、このトランザクションに「general」という名前を付けます。

  – Maker に ADMIN 属性が割り当てられているとします。

  ```
  hosts maker@
  start_transaction general
  newusr anne
  (days(weekdays)time(0800:2000))
  chres TERMINAL tty30
  defaccess(read)
  end_transaction
  ```

  Sally が end_transaction コマンドを入力すると、このトランザクションには、7 などの識別番号が割り当てられます。

- Maker である Sally が、ユーザ Anne に FINANCIAL カテゴリを追加します。Sally は同じ日にユーザ Anne のレコードを追加したばかりで、そのコマンドはまだ PMDB 上で処理または実行されていません。

  – Maker に ADMIN 属性が割り当てられているとします。

  ```
  hosts maker@
  start_transaction general 7
  chusr anne category(FINANCIAL)
  end_transaction
  ```

unalias コマンド - selang の別名の削除

UNIX ホストで有効

unalias コマンドは、alias コマンドで定義された別名を削除します。

注: 定義されているすべての別名とその値を一覧表示するには、alias コマンドを使用します。

このコマンドの形式は以下のようになります。

```
unalias aliasName
```

aliasName

データベースから削除する別名の名前を指定します。

詳細情報:

alias コマンド - selang 別名の定義 (P. 49)

undeploy コマンド - ポリシーの削除の開始

AC 環境で有効

このコマンドは deploy- コマンドと同じです。

詳細情報:

deploy- コマンド - ポリシーの削除の開始 (P. 129)

リモート設定環境の selang コマンド

このセクションでは、CA Access Control 設定リソースに対して実行される selang コマンド（config 環境のコマンド）のすべてをアルファベット順に説明します。

editres config - 環境設定の変更

config 環境で有効

editres config コマンドは、CA Access Control 環境設定の変更に使用します。

editres config コマンドは、クラスのグループによって形式が異なります。クラスは以下のグループに分類されます。

- 監査設定ファイル（audit.cfg および auditrouteflt.cfg）および PMDB フィルタ ファイル
- その他すべてのファイル

監査設定ファイルおよび PMDB フィルタ ファイルに関するこのコマンドの形式は以下のとおりです。

```
editres config name [line+|-(value)] [clear]
```

その他すべてのファイルに関するこのコマンドの形式は以下のとおりです。

```
editres config name section(path) token[-](name) value[+|-](value) data_type(type)
```

name

変更する設定リソースを指定します。PMDB フィルタ ファイルを更新するには、*pmdname@filter* の形式でファイル名を指定します（例：master_pmdb@filter.flt）。

注：管理対象ホストの設定リソースの一覧を表示するには、*find config* コマンドを使用します。

クリア

監査設定ファイルまたは PMDB フィルタ ファイルからすべての値を削除します。

注：このオプションでは、ファイル内のコメントは削除されません。

data_type(*type*)

設定エントリのデータ型を指定します。

値： str、numeric、multi_str

デフォルト： str

注：UNIX の場合、指定できる data_type は *str* のみです。UNIX では環境設定をファイル（テキスト文字列）の形で格納するため、その他のデータ型は使用できません。

line+(*value*)

監査設定ファイルまたは PMDB フィルタ ファイルに追加する値を定義します。

注: *value* は、値またはコメントです。

line-(*value*)

監査設定ファイルまたは PMDB フィルタ ファイルから削除する値を定義します。

注: *value* は、値またはコメントです。

section(*path*)

変更する設定リソースのセクションを指定します。

注: Windows レジストリ設定を対象とするときにこのオプションの指定がない場合は、レジストリ キーの*名前*の定義が変更されます。

token(*name*)

変更する設定エントリの名前を指定します。

token-(*name*)

削除する設定エントリの名前を指定します。

value(*value*)

設定エントリに指定する値を指定します。設定エントリの値がすでにある場合、CA Access Control はその値を *value* で置き換えます。

value の指定がない場合は、設定エントリ値がリセットされます。

value+(*value*)

(Windows REG_MULTI_SZ レジストリ エントリのみ) 設定エントリに追加する値を定義します。

(その他すべての設定値) 設定エントリに指定する値を指定します。設定エントリの値がすでにある場合、CA Access Control はその値を *value* で置き換えます。

注: selang が正確に割り当てられた値を変換できるように、値を引用符(" ")で囲みます。

value-(*value*)

(Windows REG_MULTI_SZ レジストリ エントリのみ) 設定エントリから削除する値を定義します。

(その他すべての設定値) 設定値から削除する任意の値を指定します。

例: Windows での ACROOT 環境設定の変更

以下の例では、CA Access Control for Windows の環境設定を変更する方法を示します。

- この例では、Audit Only モードを使用するように CA Access Control を設定します。

  ```
  er CONFIG ACROOT section(SeOSD) token(GeneralInterceptionMode) value(1)
  ```

- この例では、ホスト名解決用に CA Access Control が管理しているドメイン名リストにドメイン名を追加します。domain_names レジストリ エントリは REG_MULTI_SZ 型のレジストリ エントリです。

  ```
  er CONFIG ACROOT section(SeOSD) token(domain_names) value+(company.com)
  ```

- この例では、ホスト名解決用に CA Access Control が管理しているドメイン名リストからドメイン名を削除します。domain_names レジストリ エントリは REG_MULTI_SZ 型のレジストリ エントリです。

  ```
  er CONFIG ACROOT section(SeOSD) token(domain_names) value-(company.com)
  ```

- この例では、環境設定を削除します。

  ```
  er CONFIG ACROOT section(AccessControl) token-(Emulate)
  ```

- この例では、管理対象ホスト上の Policy Model の親 Policy Model を設定します。

  ```
  er config myPMDB@PMDROOT token(Parent_Pmd) value(topPMDB@host1.comp.ca)
  ```

例: UNIX での seos.ini 環境設定の変更

以下の例では、CA Access Control for UNIX の環境設定を変更する方法を示します。

- この例では、PAM 認証を有効にするように CA Access Control を設定します。

  ```
  er CONFIG seos.ini section(seos) token(pam_enabled) value(yes)
  ```

- この例では、ホスト名解決用に CA Access Control が管理するドメイン名を設定します。

  ```
  er CONFIG seos.ini section(seosd) token(domain_names) value+(company.com)
  ```

- この例では、ホスト名解決用に CA Access Control が管理しているドメイン名を削除します。

  ```
  er CONFIG seos.ini section(seosd) token(domain_names) value-(company.com)
  ```

- この例では、環境設定を削除します。

  ```
  er CONFIG seos.ini section(serevu) token-(admin_user)
  ```

例: 監査設定ファイルの変更

以下の例では、監査設定ファイルに 1 行追加します。

```
er CONFIG audit.cfg line+("FILE;*;Administrator;*;R;P")
```

例: PMD フィルタ ファイルの更新

以下の例では、PMD フィルタ ファイルに 1 行追加します。

```
er config pmdb@filter line+("*;*;USER;*;OLD_PASSWD;PASS")
```

find config - 設定リソースの一覧表示

config 環境で有効

find config コマンドは、管理対象ホストの CA Access Control 設定リソースを一覧表示します。対象には、レジストリ キーや環境設定ファイルなどが含まれます。

表示されるリソースはホストの種類によって異なります。

UNIX	Windows
seos.ini	ACROOT
pmd.ini@*pmd_name*	pmd_name@PMDROOT
	SEOSDRV

このコマンドの形式は以下のようになります。

```
find config
```

注: このコマンドでは、audit.cfg または auditrouteflt.cfg 設定ファイルのリストを返しません。

例: Windows ホストの設定リソースの一覧表示

次の例は、pmdb という Policy Model を持つ Windows ホストに対する find config コマンドの出力を示しています。

```
AC(config)> find config
(localhost)
pmdb@PMDROOT
ACROOT
SEOSDRV
```

showres config - 設定情報の表示

config 環境で有効

showres config コマンドは、CA Access Control の設定情報を表示します。

showres config コマンドは、クラスのグループによって形式が異なります。クラスは以下のグループに分類されます。

- 監査設定ファイル(audit.cfg および auditrouteflt.cfg)および PMDB フィルタ ファイル
- その他すべてのファイル

監査設定ファイルおよび PMDB フィルタ ファイルに関するこのコマンドの形式は以下のとおりです。

```
showres config name
```

その他すべてのファイルに関するこのコマンドの形式は以下のとおりです。

```
showres config name [section(path)] [token(name)] [recursive] [section_only]
```

name

情報を表示する設定リソースを指定します。PMDB フィルタ ファイルに関する情報を表示するには、ファイル名を「*pmdname@filter*」フォーマットで指定します(例: master_pmdb@filter.flt)。

注: 管理対象ホストの設定リソースの一覧を表示するには、*find config* コマンドを使用します。

section(*path*)

（オプション）情報を表示する設定リソース セクションを定義します。

このオプションの指定がない場合は、*name* 設定リソースのすべての設定エントリおよびセクションが一覧表示されます。

token(*name*)

（オプション）情報を表示する設定エントリ名を指定します。

このオプションの指定がない場合は、section(*path*) 内のすべての設定エントリおよびセクションが一覧表示されます。

recursive

すべてのサブ セクション内のすべての設定エントリおよびセクションに関する情報を表示します。

section_only

セクションに関する情報のみ表示するように指定します（設定エントリは表示されない）。

ネイティブ UNIX 環境の selang コマンド

このセクションでは、UNIX システム ファイルに対して実行される selang コマンド（ネイティブ UNIX 環境のコマンド）のすべてをアルファベット順に説明します。

chfile コマンド - UNIX ファイル設定の変更

ネイティブ UNIX 環境で有効

chfile コマンドと editfile コマンドは、1 つ以上の UNIX ファイルの設定を変更します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

```
{{chfile|cf}|{editfile|ef}} fileName ¥
    [owner(userName)] ¥
    [group(groupName)] ¥
    [mode( ¥
        [fowner(string)] ¥
        [fgroup(string)] ¥
        [fother(string)] ¥
    )]
```

fileName

設定を変更するファイルの名前を指定します。UNIX ファイル名を 1 つ以上入力します。複数のファイルを変更する場合は、ファイル名のリストを丸かっこで囲み、各ファイル名をスペースまたはカンマで区切ります。

group(*groupName*)

ファイルが属するグループを変更します。有効なグループ名を指定します。

mode

ファイルのアクセス モードを更新します。

fowner(*string*)

ファイルの所有者に対してアクセス モードを指定します。読み取り、書き込み、および実行の各アクセス許可を割り当てるには、*string* に文字 r、w、または x をそれぞれ指定します。ファイルを setuid に設定するには、文字 s を指定します。

既存のアクセス許可に別の許可を追加するには、*string* の先頭にプラス記号(+)を指定します。アクセス許可を削除するには、string の先頭にマイナス記号(-)を指定します。プレフィクスの指定がない場合、既存のアクセス許可は *string* にリセットされます。

fgroup(*string*)

ファイルのグループに対してアクセス モードを指定します。読み取り、書き込み、および実行の各アクセス許可を割り当てるには、*string* に文字 r、w、または x をそれぞれ指定します。ファイルを setgid に設定するには、文字 s を使用します。

既存のアクセス許可に別の許可を追加するには、*string* の先頭にプラス記号(+)を指定します。アクセス許可を削除するには、string の先頭にマイナス記号(-)を指定します。プレフィクスの指定がない場合、既存のアクセス許可は *string* にリセットされます。

fother(*string*)

他のアクセサに適用するアクセス モードを指定します。 読み取り、書き込み、および実行のアクセス許可を割り当てるには、*string* に文字 r、w、または x をそれぞれ指定します。既存のアクセス許可に別の許可を追加するには、*string* の先頭にプラス記号(+)を指定します。 アクセス許可を削除するには、string の先頭にマイナス記号(-)を指定します。プレフィクスの指定がない場合、既存のアクセス許可は *string* にリセットされます。

owner(*userName*)

ファイルの所有者を変更します。有効な UNIX ユーザのユーザ名を指定します。

詳細情報:

find file コマンド - ネイティブ ファイルの一覧表示 (P. 197)
showfile コマンド - ネイティブ ファイルのプロパティの表示 (P. 203)
chres コマンド - リソース レコードの変更 (P. 89)

chgrp コマンド - UNIX グループの変更

ネイティブ UNIX 環境で有効

UNIX グループに対する作業には、chgrp コマンド、editgrp コマンド、および newgrp コマンドを使用します。これらのコマンドは構造が同じですが、以下の点のみ異なっています。

- chgrp コマンドは、*1 つ以上の UNIX グループを変更します。*
- editgrp コマンドは、*1 つ以上の UNIX グループを作成または変更します。*
- newgrp コマンドは、*1 つ以上の UNIX グループを作成します。*

注: 環境設定(seos.ini)に指定されているファイルを対象にして、グループの読み込み、追加、更新、および削除が行われます。デフォルト設定では、このファイルは /etc/group です。詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

注: このコマンドは AC 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

```
{{chgrp|cg}|{editgrp|eg}|{newgrp|ng}} groupName ¥
    [groupid(integer)] ¥
    [userlist(userNames)]
```

groupid(*integer*)

グループのグループ ID を設定します。グループの一意の ID 番号を表す正の整数を指定します。CA Access Control では、グループ ID に 0 は使用できません。

groupName

変更するグループの名前を指定します。既存の UNIX グループの名前を指定します。複数のグループを変更する場合は、グループ名のリストをかっこで囲み、各グループ名をスペースまたはカンマで区切ります。

userlist(*userNames*)

新しいメンバリストを指定します。各ユーザ名は、あらかじめ UNIX に定義しておく必要があります。複数のユーザをリストに指定する場合は、各ユーザ名をスペースまたはカンマで区切ります。グループに定義されている既存のユーザリストはすべて、ここで指定したユーザリストに置き換えられます。

詳細情報:

rmusr コマンド - UNIX ユーザの削除 (P. 203)
showusr コマンド - ネイティブ ユーザ プロパティの表示 (P. 206)
ch[x]grp コマンド - グループ プロパティの変更 (P. 72)

chusr コマンド - UNIX ユーザの変更

ネイティブ UNIX 環境で有効

UNIX ユーザに対する作業には、chusr コマンド、editusr コマンド、および newusr コマンドを使用します。これらのコマンドの構造は同じで、以下の点が異なっています。

- chusr コマンド*は、1 つ以上の UNIX ユーザを変更します。*
- editusr コマンド*は、1 つ以上の UNIX ユーザを作成または変更します。*
- newusr コマンド*は、1 つ以上の UNIX ユーザを作成します。*

注: 環境設定(seos.ini)に指定されているファイルを対象にして、ユーザの読み込み、追加、更新、および削除が行われます。デフォルト設定では、このファイルは /etc/passwd です。詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

注: このコマンドは CA Access Control 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

```
{{chusr|cu}|{editusr|eu}|{newusr|nu}} userName ¥
    [enable] ¥
    [gecos(string)] ¥
    [homedir({path|nohomedir})] ¥
    [password(string)] ¥
    [pgroup(groupName)] ¥
    [shellprog(path)] ¥
    [userid(number)]
```

enable

何らかの理由で使用不可になっているユーザ アカウントのログインを有効にします。このパラメータは、chusr コマンドおよび editusr コマンドにのみ使用します。

gecos(*string*)

ユーザのフル ネームなど、ユーザに関する一般的なコメントを含む文字列を指定します。文字列は一重引用符で囲みます。

homedir(*path*|nohomedir)

ユーザのホーム ディレクトリの完全パスを指定します。CA Access Control はディレクトリを作成しようとします。ホーム ディレクトリが正しく作成されたかどうかに関係なく、UNIX ファイルが更新されます。

nohomedir を指定した場合、UNIX はそのユーザの homedir を作成しません。

password(*string*)

ユーザにパスワードを割り当てます。スペース以外の任意の文字を指定します。指定したパスワードでログインできるのは 1 回のみです。次回システムにログインする際に、ユーザは新しいパスワードを設定する必要があります。

pgroup(*groupName*)

ユーザのプライマリ グループ名を指定します。

shellprog(*path*)

ユーザが login コマンドまたは su コマンドを起動した後に実行される初期プログラムまたはシェルの完全パスを指定します。

userid(*number*)

一意の任意アクセス制御に使用する、ユーザの一意の ID 番号を指定します。100 以上の 10 進数を入力します。100 より小さい値は使用できません。

userName

既存の UNIX ユーザの名前です。複数のユーザを変更する場合は、ユーザ名のリストを丸かっこで囲み、各ユーザ名をスペースまたはカンマで区切ります。

詳細情報:

rmusr コマンド - UNIX ユーザの削除 (P. 203)
showusr コマンド - ネイティブ ユーザ プロパティの表示 (P. 206)
ch[x]usr コマンド - ユーザ プロパティの変更 (P. 107)

editfile コマンド - UNIX ファイル設定の変更

ネイティブ UNIX 環境で有効

このコマンドについては、chfile コマンドの項で説明しています。

詳細情報:

chfile コマンド - UNIX ファイル設定の変更 (P. 192)

editgrp コマンド - UNIX グループの作成と変更

ネイティブ UNIX 環境で有効

このコマンドについては、chgrp コマンドの項で説明しています。

詳細情報:

chgrp コマンド - UNIX グループの変更 (P. 194)

editusr コマンド - UNIX ユーザの作成と変更

ネイティブ UNIX 環境で有効

このコマンドについては、chusr コマンドの項で説明しています。

詳細情報:

chusr コマンド - UNIX ユーザの変更 (P. 195)

find file コマンド - ネイティブ ファイルの一覧表示

ネイティブ環境で有効

find file コマンドは、マスクに一致するすべてのシステム ファイルを一覧表示します。マスクは文字列で指定します。ファイルは、古いものから順番に 1 つの列に表示されます。

このコマンドの形式は以下のようになります。

```
find file [directory][/mask]
```

directory

directory で指定したディレクトリ内のすべてのファイルを一覧表示します。

マスク

directory で指定したディレクトリ内のファイルのうち、*mask* 変数に一致するすべてのファイルを一覧表示します。*mask* にはワイルドカード文字を使用できます。

例: Windows での特定のパスにある実行可能プログラムのファイルの検索

以下のコマンドは、CA Access Control bin ディレクトリにあるすべての実行可能ファイルを一覧表示します。

```
find file C:¥Program¥Files¥CA¥AccessControl¥bin¥*.exe
```

例: UNIX でのパターンに一致するファイルの検索

以下のコマンドは、CA Access Control bin ディレクトリにあって文字列 *se* で始まるすべてのファイルを一覧表示します。

```
find file /opt/CA/AccessControl//bin/se*
```

join コマンド - ユーザのネイティブ グループへの追加

ネイティブ環境で有効

join コマンドは、ユーザをグループに追加します。ネイティブ OS にすでに定義されているユーザまたはグループを指定する必要があります。

注: このコマンドは AC 環境にもありますが、動作が異なります。

join コマンドを実行するには、ユーザは以下の条件の少なくとも 1 つを満たしている必要があります。

- CA Access Control ユーザ レコードに ADMIN 属性が割り当てられていること
- GROUP-ADMIN 属性で管理者権限を与えられたグループの適用範囲にグループ レコードが含まれていること
- データベースのグループ レコードの所有者であること
- ADMIN クラスの GROUP レコードのアクセス制御リストに JOIN アクセス権または MODIFY アクセス権限が設定されていること

注: ADMIN 属性を持つユーザに、CA Access Control の GROUP レコードおよびネイティブ グループを変更する権限を与える場合は、MODIFY プロパティおよび JOIN プロパティの両方を設定する必要があります。

このコマンドの形式は以下のようになります。

```
{join|j} userName group(groupName)
```

group(*groupName*)

ユーザを追加するネイティブ グループを指定します。

userName

group パラメータで指定されたグループに追加するネイティブ ユーザのユーザ名を指定します。複数のユーザを指定する場合は、ユーザ名のリストをかっこで囲み、各ユーザ名をスペースまたはカンマで区切ります。

例

ユーザ Eli が、ユーザ Bob をグループ staff に追加します。

- ユーザ Eli に ADMIN 属性が割り当てられており、現在の環境が *native* であるとします。

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

詳細情報:

join- コマンド - ネイティブ グループからのユーザの削除 (P. 200)
showgrp コマンド - ネイティブ グループのプロパティの表示 (P. 205)
showusr コマンド - ネイティブ ユーザ プロパティの表示 (P. 206)
join[x] コマンド - ユーザの内部グループへの追加 (P. 142)

join- コマンド - ネイティブ グループからのユーザの削除

ネイティブ環境で有効

join- は、グループからユーザを削除するコマンドです。

注: このコマンドは AC 環境にもありますが、動作が異なります。

join- コマンドを使用するには、以下の条件のいずれか 1 つが満たされている必要があります。

- ADMIN 属性が割り当てられていること
- GROUP-ADMIN 属性で管理者権限を与えられたグループの適用範囲にグループ レコードが含まれていること
- データベースのグループ レコードの所有者であること
- ADMIN クラスの GROUP レコードのアクセス制御リストに JOIN アクセス権または MODIFY アクセス権限が設定されていること

ユーザのプロファイルの所有者権限のみが与えられている場合は、グループからユーザを削除できません。ADMIN 属性を持つユーザに CA Access Control レコードおよびネイティブ グループを変更する権限を与える場合は、MODIFY プロパティおよび JOIN プロパティの両方を設定する必要があります。

このコマンドの形式は以下のようになります。

```
{join-|j-} userName group(groupName)
```

group(*groupName*)

ユーザを削除する対象ネイティブ グループを指定します。

userName

グループから削除するユーザのユーザ名を指定します。グループから複数のユーザを削除する場合は、ユーザ名のリストを丸かっこで囲み、各ユーザ名をスペースまたはカンマで区切ります。

例

ユーザ Bill が、PAYROLL グループからユーザ sales25 と sales43 を削除します。

- ユーザ Bill に ADMIN 属性が割り当てられており、現在の環境が *native* であるとします。

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

詳細情報:

join コマンド - ユーザのネイティブ グループへの追加 (P. 199)
showgrp コマンド - ネイティブ グループのプロパティの表示 (P. 205)
showusr コマンド - ネイティブ ユーザ プロパティの表示 (P. 206)
join[x]- コマンド - ユーザのグループからの削除 (P. 146)

newgrp コマンド - UNIX グループの作成

ネイティブ UNIX 環境で有効

このコマンドについては、chgrp コマンドの項で説明しています。

詳細情報:

chgrp コマンド - UNIX グループの変更 (P. 194)

newusr コマンド - UNIX ユーザの作成

ネイティブ UNIX 環境で有効

このコマンドについては、chusr コマンドの項で説明しています。

詳細情報:

chusr コマンド - UNIX ユーザの変更 (P. 195)

rmgrp コマンド - UNIX グループの削除

ネイティブ UNIX 環境で有効

rmgrp コマンドは、UNIX システムから 1 つ以上のグループを削除します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

注: 環境設定(seos.ini)に指定されているファイルを対象にして、グループの読み込み、追加、更新、および削除が行われます。デフォルト設定では、このファイルは /etc/group です。詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

このコマンドの形式は以下のようになります。

`{rmgrp|rg}` *groupName*

groupName

削除するグループの名前を指定します。既存の UNIX グループ名を指定する必要があります。1 つまたは複数のグループ名を指定します。複数のグループを削除する場合は、グループ名のリストを丸かっこで囲み、各グループ名をスペースまたはカンマで区切ります。

詳細情報:

chgrp コマンド - UNIX グループの変更 (P. 194)
showgrp コマンド - ネイティブ グループのプロパティの表示 (P. 205)
rm[x]grp コマンド - グループ レコードの削除 (P. 151)

rmusr コマンド - UNIX ユーザの削除

ネイティブ UNIX 環境で有効

rmusr コマンドは、UNIX システムから 1 人以上のユーザを削除します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

注: 環境設定(seos.ini)に指定されているファイルを対象にして、ユーザの読み込み、追加、更新、および削除が行われます。デフォルト設定では、このファイルは /etc/passwd です。詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

このコマンドの形式は以下のようになります。

```
{rmusr|ru} userName
```

userName

既存の UNIX ユーザのユーザ名です。複数のユーザを削除する場合は、ユーザ名のリストを丸かっこで囲み、各ユーザ名をスペースまたはカンマで区切ります。

詳細情報:

chusr コマンド - UNIX ユーザの変更 (P. 195)
showusr コマンド - ネイティブ ユーザ プロパティの表示 (P. 206)
rm[x]usr コマンド - ユーザ レコードの削除 (P. 154)

showfile コマンド - ネイティブ ファイルのプロパティの表示

ネイティブ環境で有効

showfile コマンドは、1 つ以上のシステム ファイルのネイティブ詳細を一覧表示します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

```
{showfile|sf} fileName [next] ¥
    [{props|addprops}(propNames)]
```

addprops(*propName*)

表示するプロパティ(ルーラ)を設定します。プロパティのリストは現在のルーラに追加されます。ルーラは、現在のクエリに対してのみ設定され、現在のクエリが終了すると前のルーラ設定に戻ります。

fileName

詳細を一覧表示するファイルの名前を指定します。UNIX ファイル名を 1 つ以上入力します。複数のファイルを指定する場合は、ファイル名のリストを丸かっこで囲み、各ファイル名をスペースまたはカンマで区切ります。1>

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトのクエリ サイズは 100 に設定されています。

props(all|*propName*)

表示するプロパティ(ルーラ)を設定します。

設定したルーラは将来のクエリにも有効です。

例: UNIX ファイルの詳細の表示

UNIX の /tmp/foo ファイルの詳細を一覧表示します。

```
showfile /tmp/foo
```

例: Windows ファイルの所有者の表示

Windows ファイル C:¥tmp¥foo.exe の所有者が誰かを確認します。

```
showfile C:¥tmp¥foo.exe props(Owner)
```

詳細情報:

chfile コマンド - UNIX ファイル設定の変更 (P. 192)
chfile コマンド - Windows ファイル設定の変更 (P. 212)
showfile コマンド - ファイルのプロパティの表示 (P. 168)

showgrp コマンド - ネイティブ グループのプロパティの表示

ネイティブ環境で有効

showgrp コマンドは、ネイティブ オペレーティング システムの 1 つ以上のグループの詳細を表示します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

注: UNIX の場合、環境設定(seos.ini)に指定されているファイルを対象にして、グループの読み込み、追加、更新、および削除が行われます。デフォルト設定では、このファイルは /etc/group です。詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

このコマンドの形式は以下のようになります。

```
{showgrp|sg} groupName [next] ¥
    [{props|addprops}(propNames)]
```

addprops(*propName*)

表示するプロパティ(ルーラ)を設定します。プロパティのリストは現在のルーラに追加されます。ルーラは、現在のクエリに対してのみ設定され、現在のクエリが終了すると前のルーラ設定に戻ります。

groupName

詳細を表示するグループの名前を指定します。既存のネイティブ グループ名を指定する必要があります。 1 つまたは複数のグループ名を指定します。複数のグループを表示する場合は、グループ名のリストを丸かっこで囲み、各グループ名をスペースまたはカンマで区切ります。

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトのクエリ サイズは 100 に設定されています。

props(all|*propName*)

表示するプロパティ(ルーラ)を設定します。

設定したルーラは将来のクエリにも有効です。

例

UNIX グループ *security* の詳細を *unix* 環境にいるときに一覧表示するには、以下のコマンドを入力します。

```
showgrp security
```

詳細情報:

chgrp コマンド - UNIX グループの変更 (P. 194)
chgrp コマンド - Windows グループの変更 (P. 213)
show[x]grp コマンド - グループ プロパティの表示 (P. 170)

showusr コマンド - ネイティブ ユーザ プロパティの表示

ネイティブ UNIX 環境で有効

showusr コマンドは、ネイティブ オペレーティング システムに定義されている 1 人以上のユーザのプロパティを表示します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

注: UNIX の場合、環境設定(seos.ini)に指定されているファイルを対象にして、ユーザの読み込み、追加、更新、および削除が行われます。デフォルト設定では、このファイルは /etc/passwd です。詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

このコマンドの形式は以下のようになります。

```
{showusr|su} userName [next] ¥
    [{props|addprops}(propNames)]
```

addprops(*propName*)

表示するプロパティ(ルーラ)を設定します。プロパティのリストは現在のルーラに追加されます。ルーラは、現在のクエリに対してのみ設定され、現在のクエリが終了すると前のルーラ設定に戻ります。

userName

ネイティブ プロパティを表示するユーザの名前を指定します。既存のネイティブ ユーザ名を指定します。複数のユーザのプロパティを表示する場合は、ユーザ名のリストを丸かっこで囲み、各ユーザ名をスペースまたはカンマで区切ります。

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトのクエリ サイズは 100 に設定されています。

props(all|*propName*)

表示するプロパティ(ルーラ)を設定します。

設定したルーラは将来のクエリにも有効です。

例

UNIX ユーザ *leslie* の詳細を *unix* 環境にいるときに一覧表示するには、以下のコマンドを入力します。

```
showusr leslie
```

詳細情報:

chusr コマンド - UNIX ユーザの変更 (P. 195)
chusr コマンド - Windows ユーザの変更 (P. 219)
show[x]usr コマンド - ユーザ プロパティの表示 (P. 175)

ネイティブ Windows 環境の selang コマンド

このセクションでは、ネイティブ Windows 環境上で実行される selang コマンドのすべてをアルファベット順に説明します。

authorize コマンド - Windows リソースに対するアクセサのアクセス権限の設定

ネイティブ Windows 環境で有効

authorize コマンドは、特定のリソースへのアクセスを許可されているユーザおよびグループのリストを管理します。authorize コマンドを使用すると、ユーザまたはグループのリストを以下のように変更できます。

- 特定の CA Access Control ユーザまたはグループに対してリソースへのアクセスを許可します。
- 特定の CA Access Control ユーザまたはグループに対してリソースへのアクセスを禁止します。
- 特定のユーザまたはグループの、リソースへのアクセス権限レベルを変更します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

アクセス制御リストに対応している Windows 環境のクラスは次のとおりです。これらのクラスは、authorize コマンドを使用して制御できます。

- COM
- DISK
- FILE
- PRINTER
- REGKEY
- SHARE

上記リストにないクラスは、アクセス制御リストがないため authorize コマンドで制御できません。

このコマンドの形式は以下のようになります。

```
{authorize|auth} className resourceName ¥
    [access(accessValue)|deniedaccess(accessvalue)] ¥
    [gid(groupName, ...)] ¥
    [uid(userName, ...)]
```

access(*accessValue*)

uid パラメータまたは gid パラメータに指定したアクセサに対して設定する、リソースへのアクセス権限を指定します。

className

resourceName が属するクラスの名前を指定します。

deniedaccess(*accessvalue*)

uid パラメータまたは gid パラメータで識別するアクセサに対して、リソースへのアクセス拒否を指定します。

拒否できる *accessvalue* は、all、create、delete、join、modify、none、password、および read です。

注: *accessValue* は authorize コマンドでのみ使用できます。authorize- コマンドでは使用できません。

gid(*groupName*)

リソースへのアクセス権限を設定する対象の Windows グループを 1 つまたは複数指定します。*groupName* の値は、1 つ以上の Windows グループの名前を表します。複数のグループを指定する場合は、各グループ名をスペースまたはカンマで区切ります。

resourceName

変更または追加するリソース レコードの名前を指定します。複数のリソースを変更または追加する場合は、リソース名のリストを丸かっこで囲み、各リソース名をスペースまたはカンマで区切ります。リソース名は、少なくとも 1 つ指定する必要があります。

CA Access Control では、指定したパラメータに従って、各リソース レコードが個別に処理されます。リソースの処理中にエラーが発生すると、メッセージが発行され、リストの次のリソースから処理が続行されます。

uid(*userName*)

リソースへのアクセス権限を設定する対象の Windows ユーザを指定します。*userName* は、1 人以上の Windows ユーザのユーザ名を表します。複数のユーザを指定する場合は、各ユーザ名をスペースまたはカンマで区切ります。Windows に定義されているすべてのユーザを指定する場合は、*userName* にアスタリスク(*)を指定します。

詳細情報:

authorize- コマンド - Windows リソースに対するアクセサのアクセス権限の削除 (P. 210)
chfile コマンド - Windows ファイル設定の変更 (P. 212)
chgrp コマンド - Windows グループの変更 (P. 213)
chres コマンド - Windows リソースの変更 (P. 215)
chusr コマンド - Windows ユーザの変更 (P. 219)
authorize コマンド - リソースに対するアクセス権限の設定 (P. 51)
Windows でのクラス別アクセス権限 (P. 35)

authorize- コマンド - Windows リソースに対するアクセサのアクセス権限の削除

ネイティブ Windows 環境で有効

authorize- は、標準のアクセス制御リストからアクセサを削除することによって、リソースへのアクセス権を削除するコマンドです。このコマンドを実行すると、特定のリソースに対するアクセサのアクセス権限はデフォルトのアクセス権限のみになります。

注: このコマンドは AC 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

```
{authorize-|auth-} className resourceName ¥
    [gid(groupName, ...)] ¥
    [uid(userName, ...)]
```

className

resourceName が属するクラスの名前を指定します。

gid(*groupName*)

リソースへのアクセス権限を設定する対象の Windows グループを 1 つまたは複数指定します。*groupName* の値は、1 つ以上の Windows グループの名前を表します。複数のグループを指定する場合は、各グループ名をスペースまたはカンマで区切ります。

resourceName

変更または追加するリソース レコードの名前を指定します。複数のリソースを変更または追加する場合は、リソース名のリストを丸かっこで囲み、各リソース名をスペースまたはカンマで区切ります。リソース名は、少なくとも 1 つ指定する必要があります。

CA Access Control では、指定したパラメータに従って、各リソース レコードが個別に処理されます。リソースの処理中にエラーが発生すると、メッセージが発行され、リストの次のリソースから処理が続行されます。

uid(*userName*)

リソースへのアクセス権限を設定する対象の Windows ユーザを指定します。*userName* は、1 人以上の Windows ユーザのユーザ名を表します。複数のユーザを指定する場合は、各ユーザ名をスペースまたはカンマで区切ります。Windows に定義されているすべてのユーザを指定する場合は、*userName* にアスタリスク(*)を指定します。

詳細情報:

authorize コマンド - Windows リソースに対するアクセサのアクセス権限の設定 (P. 208)
chfile コマンド - Windows ファイル設定の変更 (P. 212)
chgrp コマンド - Windows グループの変更 (P. 213)
chres コマンド - Windows リソースの変更 (P. 215)
chusr コマンド - Windows ユーザの変更 (P. 219)
authorize- コマンド - リソースからのアクセス権限の削除 (P. 57)

chfile コマンド - Windows ファイル設定の変更

ネイティブ Windows 環境で有効

chfile と editfile は同じコマンドです。どちらも 1 つ以上の Windows ファイルを変更します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

NTFS ファイル システムが対象の場合のコマンド形式は以下のとおりです。

```
{{chfile|cf}|{editfile|ef}} fileName ¥
    [attrib(attributeValue)] ¥
    [attrib(-attributeValue)] ¥
    [defaccess(accessValue)] ¥
    [owner(userName|groupName)]
```

FAT ファイル システムが対象の場合のコマンド形式は以下のとおりです。

```
{{chfile|cf}|{editfile|ef}} fileName ¥
    [attrib([-]attributeValue)]
```

attrib([-]*attributeValue*)

ファイルの特性を決定する一連の属性を指定します。*value* 引数の前にマイナス記号(-)を付けた場合は、属性が削除されます。

defaccess(*accessValue*)

ネイティブ セキュリティが組み込まれているグループ Everyone に対するアクセス権限を指定します。システム ユーザはすべて Everyone グループのメンバです。Everyone グループにアクセス権を与えると、認証されたすべてのユーザだけではなく、すべての潜在的な匿名ユーザもアクセスできるようになります。

注: CA Access Control 環境で定義されたオブジェクトの defaccess には、別の意味があります。この場合、デフォルトのアクセス権限とは、リソースの CA Access Control リストに含まれていないアクセサがリソースへのアクセスを要求した場合に与えられる権限のことです。また、デフォルトのアクセス権限は、CA Access Control で定義されていないユーザにも適用されます。

defaccess パラメータは NTFS ファイル システムにのみ適用されます。

owner(*userName*|*groupName*)

ファイル レコードの所有者としてユーザまたはグループを割り当てます。ファイル レコードの所有者には、ファイルに対する無制限のアクセス権が与えられます。ファイルの所有者は、ファイル レコードを常時更新または削除することができます。

詳細情報:

showfile コマンド - ネイティブ ファイルのプロパティの表示 (P. 203)
chfile コマンド - ファイル レコードの変更 (P. 65)
Windows のファイル属性 (P. 565)

chgrp コマンド - Windows グループの変更

ネイティブ Windows 環境で有効

Windows グループに対する作業には、chgrp コマンド、editgrp コマンド、および newgrp コマンドを使用します。これらのコマンドは構造が同じですが、以下の点のみ異なっています。

- chgrp コマンドは、*1 つ以上の Windows グループを変更します。*
- editgrp コマンドは、*1 つ以上の Windows グループを作成または変更します。*
- newgrp コマンドは、*1 つ以上の Windows グループを作成します。*

注: このコマンドは AC 環境にもありますが、動作が異なります。

複数のグループを定義する場合、または複数グループのプロパティを変更する場合は、グループ名のリストを丸かっこで囲み、各グループ名をスペースまたはカンマで区切ります。

注: グループにメンバを追加するには join コマンドを使用し、グループからメンバを削除するには join- コマンドを使用します。

このコマンドの形式は以下のようになります。

```
{{chgrp|cg}|{editgrp|eg}|{newgrp|ng}} groupName ¥
    [global] ¥
    [comment(string)|comment-] ¥
    [privileges(privList)] ¥
    [privileges(-privList)] ¥
    [rename_group]
```

comment(*string*)

グループ レコードに最大 255 文字の英数字から成るコメント文字列を追加します。グループ レコードにすでにコメント文字列が追加されている場合、既存の文字列はここで指定した新しい文字列に置き換えられます。文字列に空白が含まれる場合は、文字列全体を一重引用符で囲みます。

標準の Windows グループには、システムのインストール時に説明のコメントが追加されています。Windows 環境と CA Access Control 環境の両方に新しいグループを作成すると、CA Access Control によって「CA Access Control Group」というコメントが追加されます。

global

グローバル グループを示します。Windows データベースに存在しない一意なグループ名を指定する必要があります。Windows では、グループとユーザに同じ名前を指定することはできません。

注: グローバル グループを作成し、CA Access Control バージョン 4.1 を使用する場合は、*~groupName* を使用します。バージョン 4.1 以上では、後方互換性を保つために、この形式がサポートされています。

groupName

newgrp コマンドの場合は、データベースに追加されるグループ レコードの名前を指定します。Windows データベースに存在しない一意なグループ名を指定する必要があります。CA Access Control データベースとは異なり、Windows ではグループとユーザに同じ名前を指定することはできません。

chgrp コマンドの場合、変更するプロパティを含むグループの名前を指定します。

複数のグループを定義する場合、または複数グループのプロパティを変更する場合は、グループ名のリストを丸かっこで囲み、各グループ名をスペースまたはカンマで区切ります。

privileges(*privList*|*-privList*)

Windows のグループ レコードに特定の権限を追加します。privList の前にマイナス符号 (-) を付けた場合は、指定した権限を削除します。 有効な値は、ネイティブ Windows で指定できるすべての権限です。

このパラメータは、chgrp コマンドまたは editgrp コマンドで既存のグループ レコードを変更する場合にのみ指定できます。新しいグループ レコードを作成するときに、このパラメータを使用して権限を割り当てることはできません。

rename_group

Windows データベースのグループ アカウント名を変更します。古いグループ名のすべてのプロパティは、名前を変更したグループ アカウントに適用されます。Windows データベースに存在する一意なグループ名を指定する必要があります。CA Access Control データベースとは異なり、Windows ではグループとユーザに同じ名前を指定することはできません。

注: Active Directory がインストールされている Windows 2000 に CA Access Control をインストールすると、CA Access Control によって Windows 2000 以前のグループ名が変更されます。

chres コマンド - Windows リソースの変更

ネイティブ Windows 環境で有効

chres、editres、および newres コマンドを使用して、Windows 環境内の CA Access Control クラスに属するリソース レコードに対する操作を実行します。これらのコマンドは構造が同じですが、以下の点のみ異なっています。

- chres コマンド*は、1 つ以上のリソースを*変更します。
- editres コマンド*は、1 つ以上のリソースを*作成または変更します。
- newres コマンド*は、1 つ以上のリソースを*作成します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

```
{{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-]
```

または

```
{{chres|cr}|{editres|er}|{newres|nr}} ¥
    DOMAIN resourceName ¥
    [computer(workstationName)|computer-(workstationName)] ¥
    [domainpwd(connectPassword)] ¥
    [trusted(domainName)|trusted-(domainName)]
```

binary(*hexastring*)

レジストリ キーが 16 進数の場合に、レジストリ キーの値を指定します。

className

resourceName が属するクラスの名前を指定します。

newres コマンドの場合、有効な値は REGKEY、REGVAL、OU、および SHARE です。chres コマンドおよび editres コマンドの場合、有効な値は COM、DISK、DOMAIN、FILE、PRINTER、REGKEY、REGVAL、SERVICE、DEVICE、SESSION、OU、および SHARE です。

comment(*string*)

リソース レコードにコメント文字列を追加します。リソース レコードにすでにコメント文字列が追加されている場合、既存の文字列はここで指定した新しい文字列に置き換えられます。このパラメータは SHARE リソースおよび PRINTER リソースに対してのみ有効です。

computer(*workstationName*)|computer-(*workstationName*)

ドメインに追加するワークステーションの名前を指定します。引数の前にマイナス記号を付けた場合は、ドメインから削除するワークステーションを指定します。このパラメータは、DOMAIN リソースに対してのみ使用でき、chres コマンドまたは editres コマンドにのみ指定可能です。

defaccess(*accessValue*)

ネイティブ セキュリティが組み込まれているグループ Everyone に対するアクセス権限を指定します。システム ユーザはすべて Everyone グループのメンバです。Everyone グループにアクセス権を与えると、認証されたすべてのユーザだけではなく、すべての潜在的な匿名ユーザもアクセスできるようになります。

注: CA Access Control 環境で定義されたオブジェクトの defaccess には、別の意味があります。この場合、デフォルトのアクセス権限とは、リソースの CA Access Control リストに含まれていないアクセサがリソースへのアクセスを要求した場合に与えられる権限のことです。また、デフォルトのアクセス権限は、CA Access Control で定義されていないユーザにも適用されます。

defaccess パラメータは NTFS ファイル システムにのみ適用されます。

domainpwd(*connectPassword*)

管理者が信頼関係を変更するときに入力する必要があるパスワードを指定します。

このパラメータは、DOMAIN リソースに対してのみ使用でき、chres コマンドまたは editres コマンドにのみ指定可能です。

dword(*integer*)

レジストリ キーが整数の場合に、レジストリ キーの値を指定します。

gen_prop(*propertyName*)

OU クラスのプロパティを指定します。

このパラメータは OU クラスに対してのみ有効です。

gen_value(*valueName*)

OU クラスのプロパティ値を指定します。

このパラメータは OU クラスに対してのみ有効です。

location(*string*)

プリンタの場所を指定します。このプロパティを削除するには、( ) に何も指定しません。

このパラメータは PRINTER リソースに対してのみ有効です。

maxusers(*integer*)

共有ディレクトリに同時に接続できるユーザの最大数（*integer*）を指定します。

このパラメータは SHARE リソースに対してのみ有効です。

multistring(*string*)

レジストリ キーが複数文字列の場合に、レジストリ キーの値を指定します。

owner(*userName*|*groupName*)

リソース レコードの所有者としてユーザまたはグループを割り当てます。リソース レコードの所有者には、リソースに対する無制限のアクセス権が与えられます。リソースの所有者には、リソース レコードを更新および削除する権限が常に与えられます。詳細については、「*Windows エンドポイント管理ガイド*」を参照してください。

FAT ファイル システムの FILE レコードまたは SHARE レコードには owner パラメータを指定できません。このパラメータは、DEVICE、DOMAIN、OU、PROCESS、REGVAL、SERVICE、および SESSION の各リソースに対しても指定できません。

resourceName

変更または追加するリソース レコードの名前を指定します。複数のリソースを変更または追加する場合は、リソース名のリストを丸かっこで囲み、各リソース名をスペースまたはカンマで区切ります。リソース名は、少なくとも 1 つ指定する必要があります。

CA Access Control では、指定したパラメータに従って、各リソース レコードが個別に処理されます。リソースの処理中にエラーが発生すると、メッセージが発行され、リストの次のリソースから処理が続行されます。

share_name(*shareName*)|share_name-

プリンタの共有ポイントを指定します。

このパラメータは PRINTER リソースに対してのみ有効です。

string(*string*)

レジストリ キーが文字列の場合に、レジストリ キーの値を指定します。

trusted(*domainName*) | trusted-(*domainName*)

信頼される側のドメインに追加するドメインの名前を指定します。ドメインを untrusted にする場合は、引数の前にマイナス記号を付けてドメイン名を指定します。このパラメータは、DOMAIN リソースに対してのみ使用でき、chres コマンドまたは editres コマンドにのみ指定可能です。

詳細情報:

rmres コマンド - Windows リソースの削除 (P. 232)
showres コマンド - ネイティブ リソース プロパティの表示 (P. 239)
chres コマンド - リソース レコードの変更 (P. 89)

chusr コマンド - Windows ユーザの変更

ネイティブ Windows 環境で有効

Windows ユーザに対する作業には、chusr コマンド、editusr コマンド、および newusr コマンドを使用します。これらのコマンドは構造が同じですが、以下の点のみ異なっています。

- chusr コマンドは、1 つ以上の Windows ユーザを*変更*します。
- editusr コマンドは、1 つ以上の Windows ユーザを*作成または変更*します。
- newusr コマンドは、*1 つ以上の Windows ユーザを作成します*。

注: このコマンドは AC 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

```
{{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)] ¥
    [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-

ユーザ レコードにコメント文字列を追加します。

引数は最大 255 文字の英数字から成る文字列です。文字列に空白が含まれる場合は、文字列全体を一重引用符で囲みます。

country(*string*)

ユーザの国名を指定します。この文字列は認証プロセスでは使用されません。

引数は最大 19 文字の英数字から成る文字列です。文字列に空白が含まれる場合は、文字列全体を一重引用符で囲みます。

expire|expire(*mm/dd/yy[@hh:mm*) | expire-

ユーザ アカウントが失効する日付を設定します。日付の指定がない場合、現在ログインしていないユーザのアカウントはただちに失効します。ユーザがログイン中だった場合は、ログアウトしたときに失効します。

newusr コマンドで expire- パラメータを指定して、有効期限のないユーザ アカウントを定義します。chusr コマンドおよび editusr コマンドの場合は、指定されたユーザ アカウントから有効期限を削除する場合にこのパラメータを指定します。

日付の引数は *mm*/*dd*/*yy* [@*hh*:*mm*] の形式で指定します。

flags(*accountFlags*|- *accountFlags*)

ユーザ アカウントの特定の属性を指定します。有効なフラグ値の詳細については、付録「Windows の値」を参照してください。

ユーザ レコードからフラグを削除するには、*accountFlags* の前にマイナス記号 (-) を付けます。

full_name(*fullName*)

ユーザ レコードに関連付けられたユーザのフル ネームを指定します。

引数は最大 256 文字の英数字から成る文字列です。文字列に空白が含まれる場合は、文字列全体を一重引用符で囲みます。

gecos(*string*)

ユーザのフル ネームなど、ユーザに関するコメント文字列を指定します。文字列は一重引用符で囲みます。

homedir(*homeDir*)

ユーザのホーム ディレクトリを指定します。ユーザは、自分のホームドライブおよびホーム ディレクトリに自動的にログインできます。

homedrive(*homeDrive*)

ユーザのホーム ディレクトリのドライブを指定します。ユーザは、自分のホームドライブおよびホーム ディレクトリに自動的にログインできます。

location(*string*)

ユーザの所在地を指定します。この文字列は認証プロセスでは使用されません。

引数は最大 19 文字の英数字から成る文字列です。文字列に空白が含まれる場合は、文字列全体を一重引用符で囲みます。

logonserver(*serverName*)

ユーザのログイン情報を確認するサーバを指定します。ユーザがドメイン ワークステーションにログインすると、この引数で指定したサーバにログイン情報が送られ、ユーザがワークステーションを使用することが許可されます。

organization(*name*)

ユーザが所属する組織を指定します。この情報は認証プロセスでは使用されません。

引数は最大 256 文字の英数字から成る文字列です。文字列に空白が含まれる場合は、文字列全体を一重引用符で囲みます。

org_unit(*name*)

ユーザが所属する組織単位を指定します。この情報は認証プロセスでは使用されません。

引数は最大 256 文字の英数字から成る文字列です。文字列に空白が含まれる場合は、文字列全体を一重引用符で囲みます。

password(*password*)

ユーザにパスワードを割り当てます。パスワード チェックが有効になっている場合、指定したパスワードでログインできるのは 1 回のみです。次回システムにログインする際に、ユーザは新しいパスワードを設定する必要があります。

引数はスペースやカンマを含まない最大 14 文字の文字列です。パスワード チェックが有効になっている場合、指定したパスワードでログインできるのは 1 回のみです。「パスワードを無期限にする」のフラグが設定されている場合を除いて、次回システムにログインする際に、ユーザは新しいパスワードを設定する必要があります。

自分のパスワードを変更するには、*setoptions cng_ownpwd* を使用して selang オプションを設定するか、sepass を使用する必要があります。

Windows システム上でユーザのパスワードを設定している場合、以下のメッセージが表示されることがあります。

パスワードが必要な長さよりも短い。

このエラーは、パスワードがポリシー要件を満たしていないことを意味します。このエラーの原因は、以下のいずれかです。

- パスワードが必要な長さよりも短いか、または長い。
- パスワードが最近使用されており、Windows NT Change History フィールドに存在する。
- パスワードに完全に一意の文字が含まれていない。
- パスワードが他のパスワード ポリシー要件(CA Access Control パスワード ポリシーで設定された要件など)を満たしていない。

このエラーを回避するには、該当するすべての要件を満たすパスワードを設定するようにしてください。

pgroup(*primaryGroup*)

ユーザのプライマリ グループ ID を設定します。プライマリ グループはユーザが定義されているグループの 1 つで、グローバル グループである必要があります。

引数はスペースやカンマを含まない最大 14 文字の文字列です。

phone(*string*)

ユーザの電話番号を指定します。この情報は認証プロセスでは使用されません。

privileges(*privList*)

Windows のユーザ レコードに特定の権限を追加します。*privList* の前にマイナス記号(-)を付けた場合は、指定した権限を削除します。このパラメータは、chusr コマンドまたは editusr コマンドで既存のユーザ レコードを変更する場合にのみ指定可能です。新しいユーザ レコードを作成するときに、このパラメータを使用して権限を割り当てることはできません。

profile(*path*)

デスクトップ環境(プログラム グループ、ネットワーク接続)のユーザのプロファイルを含むファイルの完全パスを指定します。ユーザがワークステーションにログインすると、毎回同じ環境が画面に表示されます。

restrictions([*days*] [*time*])|restrictions-([*days*] [*time*])

ユーザがファイルにアクセスできる曜日と時間帯を指定します。

days 引数を指定せずに time 引数を指定した場合、レコード内にすでに設定されている曜日制限に対して、指定した時間帯制限が適用されます。time 引数を指定せずに days 引数を指定した場合、レコード内にすでに設定されている時間帯制限に対して、指定した曜日制限が適用されます。days 引数と time 引数の両方を指定した場合、ユーザは、指定した曜日の指定した時間帯にのみシステムにアクセスできます。

- [days] には、ユーザがファイルにアクセスできる曜日を指定します。days 引数には次のサブ引数があります。
  - **anyday** - ユーザは曜日を問わずファイルにアクセスできます。
- **weekdays** - ユーザは月曜から金曜までの平日に限りリソースにアクセスできます。
  - **Mon**、**Tue**、**Wed**、**Thu**、**Fri**、**Sat**、**Sun** - ユーザは指定した曜日にのみリソースにアクセスできます。曜日は任意の順で指定できます。複数の曜日を指定する場合は、各曜日をスペースまたはカンマで区切ります。
- [time] には、ユーザがリソースにアクセスできる時間帯を指定します。time 引数には次のサブ引数があります。
  - **anytime** - 特定の曜日の任意の時間帯にリソースにアクセスできます。
  - **startTime:endTime** - 指定した時間帯にのみリソースにアクセスできます。startTime および endTime は両方とも *hhmm* の形式で指定します。*hh* は 24 時間表記の時間（00 から 23）、*mm* は分（00 から 59）を表します。2400 は有効な time 値ではないことに注意してください。startTime が endTime より小さいこと、および両方が同じ日の時間であることが必要です。端末がホストと異なるタイム ゾーンにある場合は、端末の開始時間と終了時間をホストのローカル時間に相当する時間に変換し、時間の値を調整してください。たとえば、ホストがニューヨークにあり、端末がロサンゼルスにある場合、ロサンゼルスの端末からのアクセスを午前 8 時から午後 5 時まで許可するには、「time(1100:2000)」と指定します。

resume(*date*)|resume-

ユーザ アカウントの再開日および再開時間(オプション)です。suspend パラメータと resume パラメータの両方を指定する場合、再開日を一時停止日より後に設定する必要があります。そうしないと、ユーザは永久に一時停止されたままになります。

失効の日付と時刻(オプション)は、以下の形式で指定します。時刻は省略可能です。

```
mm/dd/yy[@HH:MM]
```

resume- パラメータを使用して、ユーザ アカウントのステータスをアクティブ(有効)から一時停止に変更します。このパラメータは chusr コマンドまたは editusr コマンドにのみ使用できます。

script(*loginScriptPath)*

ユーザがログインしたときに自動的に実行されるファイルの場所を指定します。このログイン スクリプトによって作業環境が設定されます。ユーザの作業環境は profile パラメータでも設定されるため、このパラメータの指定は省略可能です。

suspend(*date*)|suspend-

ユーザ アカウントを無効にします。ユーザは一時停止されたユーザ アカウントを使用してシステムにログインすることはできません。 date を指定すると、指定した日にユーザ アカウントが一時停止されます。date を省略すると、chusr コマンドの実行後ただちにユーザ アカウントが一時停止されます。

日付と時刻は、*mm/dd/yy[@HH:MM]* 形式で指定します。時刻は省略可能です。

suspend- パラメータを使用して、ユーザ アカウントのステータスを無効からアクティブ(有効)に変更します。このパラメータは chusr コマンドまたは editusr コマンドにのみ使用できます。

terminals(*terminalList*)|terminals-(*terminalList*)

ユーザがログインできる端末を最高 8 つまで指定します。リストは二重引用符で囲み、名前はカンマで区切ります。例:

```
"terminal1,terminal2"
```

workstations(*workstationList*)|workstations-(*workstationList*)|workstations-

ユーザがログインできるワークステーションを最高 8 つまで指定します。リストは二重引用符で囲み、名前はカンマで区切ります。例:

```
"workstation1,workstation2"
```

editfile コマンド - Windows ファイル設定の変更

ネイティブ Windows 環境で有効

このコマンドについては、chfile コマンドの項で説明しています。

詳細情報:

chfile コマンド - Windows ファイル設定の変更 (P. 212)

editgrp コマンド - Windows グループの作成と変更

ネイティブ Windows 環境で有効

このコマンドについては、chgrp コマンドの項で説明しています。

詳細情報:

chgrp コマンド - Windows グループの変更 (P. 213)

editusr コマンド - Windows ユーザの作成と変更

ネイティブ Windows 環境で有効

このコマンドについては、chusr コマンドの項で説明しています。

詳細情報:

chusr コマンド - Windows ユーザの変更 (P. 219)

editres コマンド - Windows リソースの作成と変更

ネイティブ Windows 環境で有効

このコマンドについては、chres コマンドの項で説明しています。

詳細情報:

chres コマンド - Windows リソースの変更 (P. 215)

find file コマンド - ネイティブ ファイルの一覧表示

ネイティブ環境で有効

find file コマンドは、マスクに一致するすべてのシステム ファイルを一覧表示します。マスクは文字列で指定します。ファイルは、古いものから順番に 1 つの列に表示されます。

このコマンドの形式は以下のようになります。

```
find file [directory][/mask]
```

directory

directory で指定したディレクトリ内のすべてのファイルを一覧表示します。

マスク

directory で指定したディレクトリ内のファイルのうち、*mask* 変数に一致するすべてのファイルを一覧表示します。*mask* にはワイルドカード文字を使用できます。

例: Windows での特定のパスにある実行可能プログラムのファイルの検索

以下のコマンドは、CA Access Control bin ディレクトリにあるすべての実行可能ファイルを一覧表示します。

```
find file C:¥Program¥Files¥CA¥AccessControl¥bin¥*.exe
```

例: UNIX でのパターンに一致するファイルの検索

以下のコマンドは、CA Access Control bin ディレクトリにあって文字列 *se* で始まるすべてのファイルを一覧表示します。

```
find file /opt/CA/AccessControl//bin/se*
```

find {xuser|xgroup} コマンド - エンタープライズ ユーザまたはグループの一覧表示

ネイティブ Windows 環境で有効

find {xuser|xgroup} コマンドは、現在のドメインまたは *信頼されている*ドメインのエンタープライズ ユーザまたはグループの名前を一覧表示します。

注: このコマンドは、Directory Services を使用しているサポート対象 Windows 2000 オペレーティング システムでのみサポートされます。

このコマンドの形式は以下のようになります。

```
find {xuser|xgroup} mask [domain(domainName)] [next]
```

xgroup

コマンドに対してエンタープライズ グループを返すように指定します。

xuser

コマンドに対してエンタープライズ ユーザを返すように指定します。

domain(*domainName*)

検索対象として限定する信頼されているドメインを指定します。

このオプションの指定がなかった場合は、現在のドメインのユーザが返されます。

マスク

エンタープライズ ユーザのマスクを指定します。

next

以前実行された find xuser コマンドまたは find xgroup コマンドによって開始されたエンタープライズ ユーザまたはグループの一覧表示処理を selang 出力が継続するように指定します。

このオプションは一覧の項目数が 100 を超える場合に指定します。

例: エンタープライズ ユーザの表示

以下のコマンドは、abc で始まる現在のドメインの最初の 100 エンタープライズ ユーザを一覧表示します。

```
find xuser abc*
```

join コマンド - ユーザのネイティブ グループへの追加

ネイティブ環境で有効

join コマンドは、ユーザをグループに追加します。ネイティブ OS にすでに定義されているユーザまたはグループを指定する必要があります。

注: このコマンドは AC 環境にもありますが、動作が異なります。

join コマンドを実行するには、ユーザは以下の条件の少なくとも 1 つを満たしている必要があります。

- CA Access Control ユーザ レコードに ADMIN 属性が割り当てられていること
- GROUP-ADMIN 属性で管理者権限を与えられたグループの適用範囲にグループ レコードが含まれていること
- データベースのグループ レコードの所有者であること
- ADMIN クラスの GROUP レコードのアクセス制御リストに JOIN アクセス権または MODIFY アクセス権限が設定されていること

注: ADMIN 属性を持つユーザに、CA Access Control の GROUP レコードおよびネイティブ グループを変更する権限を与える場合は、MODIFY プロパティおよび JOIN プロパティの両方を設定する必要があります。

このコマンドの形式は以下のようになります。

```
{join|j} userName group(groupName)
```

group(*groupName*)

ユーザを追加するネイティブ グループを指定します。

userName

group パラメータで指定されたグループに追加するネイティブ ユーザのユーザ名を指定します。複数のユーザを指定する場合は、ユーザ名のリストをかっこで囲み、各ユーザ名をスペースまたはカンマで区切ります。

例

ユーザ Eli が、ユーザ Bob をグループ staff に追加します。

- ユーザ Eli に ADMIN 属性が割り当てられており、現在の環境が *native* であるとします。

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

詳細情報:

join- コマンド - ネイティブ グループからのユーザの削除 (P. 200)
showgrp コマンド - ネイティブ グループのプロパティの表示 (P. 205)
showusr コマンド - ネイティブ ユーザ プロパティの表示 (P. 206)
join[x] コマンド - ユーザの内部グループへの追加 (P. 142)

join- コマンド - ネイティブ グループからのユーザの削除

ネイティブ環境で有効

join- は、グループからユーザを削除するコマンドです。

注: このコマンドは AC 環境にもありますが、動作が異なります。

join- コマンドを使用するには、以下の条件のいずれか 1 つが満たされている必要があります。

- ADMIN 属性が割り当てられていること
- GROUP-ADMIN 属性で管理者権限を与えられたグループの適用範囲にグループ レコードが含まれていること
- データベースのグループ レコードの所有者であること
- ADMIN クラスの GROUP レコードのアクセス制御リストに JOIN アクセス権または MODIFY アクセス権限が設定されていること

ユーザのプロファイルの所有者権限のみが与えられている場合は、グループからユーザを削除できません。ADMIN 属性を持つユーザに CA Access Control レコードおよびネイティブ グループを変更する権限を与える場合は、MODIFY プロパティおよび JOIN プロパティの両方を設定する必要があります。

このコマンドの形式は以下のようになります。

```
{join-|j-} userName group(groupName)
```

group(*groupName*)

ユーザを削除する対象ネイティブ グループを指定します。

userName

グループから削除するユーザのユーザ名を指定します。グループから複数のユーザを削除する場合は、ユーザ名のリストを丸かっこで囲み、各ユーザ名をスペースまたはカンマで区切ります。

例

ユーザ Bill が、PAYROLL グループからユーザ sales25 と sales43 を削除します。

- ユーザ Bill に ADMIN 属性が割り当てられており、現在の環境が *native* であるとします。

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

詳細情報:

join コマンド - ユーザのネイティブ グループへの追加 (P. 199)
showgrp コマンド - ネイティブ グループのプロパティの表示 (P. 205)
showusr コマンド - ネイティブ ユーザ プロパティの表示 (P. 206)
join[x]- コマンド - ユーザのグループからの削除 (P. 146)

newgrp コマンド - Windows グループの作成

ネイティブ Windows 環境で有効

このコマンドについては、chgrp コマンドの項で説明しています。

詳細情報:

chgrp コマンド - Windows グループの変更 (P. 213)

newres コマンド - Windows リソースの作成

ネイティブ Windows 環境で有効

このコマンドについては、chres コマンドの項で説明しています。

詳細情報:

chres コマンド - Windows リソースの変更 (P. 215)

newusr コマンド - Windows ユーザの作成

ネイティブ Windows 環境で有効

このコマンドについては、chusr コマンドの項で説明しています。

詳細情報:

chusr コマンド - Windows ユーザの変更 (P. 219)

rmgrp コマンド - Windows グループの削除

ネイティブ Windows 環境で有効

rmgrp コマンドは、Windows データベースから 1 つ以上のグループを削除します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

`{rmgrp|rg} groupName`

groupName

削除するグループの名前を指定します。既存の Windows グループ名を指定する必要があります。1 つまたは複数のグループ名を指定します。複数のグループを削除する場合は、グループ名のリストを丸かっこで囲み、各グループ名をスペースまたはカンマで区切ります。

rmres コマンド - Windows リソースの削除

rmres コマンドは、Windows システム データベースから 1 つ以上のリソースを削除します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

`{rmres|rr} className resourceName`

className

リソースが属するクラスの名前を指定します。

resourceName

className で指定したクラスの既存の Windows リソース名を指定します。複数のリソースを削除する場合は、リソース名のリストを丸かっこで囲み、各リソース名をスペースまたはカンマで区切ります。

詳細情報:

chres コマンド - Windows リソースの変更 (P. 215)
showres コマンド - ネイティブ リソース プロパティの表示 (P. 239)
rm[x]usr コマンド - ユーザ レコードの削除 (P. 154)

rmusr コマンド - Windows ユーザの削除

ネイティブ Windows 環境で有効

rmusr コマンドは、Windows システム データベースから 1 人以上のユーザを削除します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

`{rmusr|ru} userName`

userName

既存の Windows ユーザのユーザ名を指定します。複数のユーザを削除する場合は、ユーザ名のリストを丸かっこで囲み、各ユーザ名をスペースまたはカンマで区切ります。

詳細情報:

chusr コマンド - Windows ユーザの変更 (P. 219)
showusr コマンド - ネイティブ ユーザ プロパティの表示 (P. 206)
rm[x]usr コマンド - ユーザ レコードの削除 (P. 154)

setoptions コマンド - CA Access Control Windows オプションの設定

setoptions コマンドは、Windows オペレーティング システムに関連する、システム全体に適用される CA Access Control オプションを動的に設定します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

setoptions コマンドを使用するには ADMIN 属性が必要です。ただし、setoptions list コマンドは AUDITOR 属性または OPERATOR 属性があれば使用できます。

このコマンドの形式は以下のようになります。

```
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{+|-}

監査を有効(+)または無効(-)に指定します。

audit_policy(success(system|logon|access|rights|process|security|manage))

ログに記録する認証されたアクセス イベントの検出を指定します。アクセス タイプは以下のとおりです。

- **system** - コンピュータのシャットダウンまたは再起動を試行します。
- **logon** - システムへのログオンまたはシステムからのログオフを試行します。
- **access** - ファイルなどのセキュリティ保護可能なオブジェクトへのアクセスを試行します。
- **rights** - Windows Server 権限の使用を試行します。
- **process** - プログラムのアクティブ化、何らかの形式でのハンドル複製、オブジェクトへの間接的なアクセス、プロセスの終了などのイベント。
- **security** - ポリシー オブジェクト ルールの変更を試行します。
- **manage** - ユーザまたはグループ アカウントの作成、削除または変更を試行します。パスワード変更も含みます。

audit_policy(failure(system|logon|access|rights|process|security|manage))

ログに記録する不正なアクセス イベントの検出を指定します。アクセス タイプは以下のとおりです。

- **system** - コンピュータのシャットダウンまたは再起動を試行します。
- **logon** - システムへのログオンまたはシステムからのログオフを試行します。
- **access** - ファイルなどのセキュリティ保護可能なオブジェクトへのアクセスを試行します。
- **rights** - Windows Server 権限の使用を試行します。
- **process** - プログラムのアクティブ化、何らかの形式でのハンドル複製、オブジェクトへの間接的なアクセス、プロセスの終了などのイベント。
- **security** - ポリシー オブジェクト ルールの変更を試行します。
- **manage** - ユーザまたはグループ アカウントの作成、削除または変更を試行します。パスワード変更も含みます。

history(*number-stored-passwords*)

データベースに保存するパスワード履歴の数を指定します。新しいパスワードの作成時、ユーザは履歴リストに保存されているパスワードを指定できません。*NStoredPasswords* は、1 ～ 24 の整数です。0 を指定すると、パスワードは保存されません。

interval(*nDays*)

パスワードの設定または変更後、ユーザに対して新しいパスワードの入力を促すメッセージを表示するまでの経過日数を設定します。

nDays の値としては、正の整数または 0 を指定します。期間を 0 に設定すると、ユーザに対するパスワード期間のチェックは無効になります。パスワードに有効期限を設定しない場合は、期間を 0 に設定します。

min_life(*nDays*)

変更したパスワードを再度変更できるようになるまでの最短日数を設定します。*NDays* には、正の整数を指定します。

詳細情報:

showfile コマンド - ファイルのプロパティの表示 (P. 168)

showfile コマンド - ネイティブ ファイルのプロパティの表示

ネイティブ環境で有効

showfile コマンドは、1 つ以上のシステム ファイルのネイティブ詳細を一覧表示します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

このコマンドの形式は以下のようになります。

```
{showfile|sf} fileName [next] ¥
    [{props|addprops}(propNames)]
```

addprops(*propName*)

表示するプロパティ(ルーラ)を設定します。プロパティのリストは現在のルーラに追加されます。ルーラは、現在のクエリに対してのみ設定され、現在のクエリが終了すると前のルーラ設定に戻ります。

fileName

詳細を一覧表示するファイルの名前を指定します。UNIX ファイル名を 1 つ以上入力します。複数のファイルを指定する場合は、ファイル名のリストを丸かっこで囲み、各ファイル名をスペースまたはカンマで区切ります。1>

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトのクエリ サイズは 100 に設定されています。

props(all|*propName*)

表示するプロパティ(ルーラ)を設定します。

設定したルーラは将来のクエリにも有効です。

例: UNIX ファイルの詳細の表示

UNIX の /tmp/foo ファイルの詳細を一覧表示します。

```
showfile /tmp/foo
```

例: Windows ファイルの所有者の表示

Windows ファイル C:¥tmp¥foo.exe の所有者が誰かを確認します。

```
showfile C:¥tmp¥foo.exe props(Owner)
```

詳細情報:

chfile コマンド - UNIX ファイル設定の変更 (P. 192)
chfile コマンド - Windows ファイル設定の変更 (P. 212)
showfile コマンド - ファイルのプロパティの表示 (P. 168)

showgrp コマンド - ネイティブ グループのプロパティの表示

ネイティブ環境で有効

showgrp コマンドは、ネイティブ オペレーティング システムの 1 つ以上のグループの詳細を表示します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

注: UNIX の場合、環境設定(seos.ini)に指定されているファイルを対象にして、グループの読み込み、追加、更新、および削除が行われます。デフォルト設定では、このファイルは /etc/group です。詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

このコマンドの形式は以下のようになります。

```
{showgrp|sg} groupName [next] ¥
    [{props|addprops}(propNames)]
```

addprops(*propName*)

表示するプロパティ(ルーラ)を設定します。プロパティのリストは現在のルーラに追加されます。ルーラは、現在のクエリに対してのみ設定され、現在のクエリが終了すると前のルーラ設定に戻ります。

groupName

詳細を表示するグループの名前を指定します。既存のネイティブ グループ名を指定する必要があります。 1 つまたは複数のグループ名を指定します。複数のグループを表示する場合は、グループ名のリストを丸かっこで囲み、各グループ名をスペースまたはカンマで区切ります。

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトのクエリ サイズは 100 に設定されています。

props(all|*propName*)

表示するプロパティ(ルーラ)を設定します。

設定したルーラは将来のクエリにも有効です。

例

UNIX グループ *security* の詳細を *unix* 環境にいるときに一覧表示するには、以下のコマンドを入力します。

```
showgrp security
```

詳細情報:

chgrp コマンド - UNIX グループの変更 (P. 194)
chgrp コマンド - Windows グループの変更 (P. 213)
show[x]grp コマンド - グループ プロパティの表示 (P. 170)

showres コマンド - ネイティブ リソース プロパティの表示

Windows リソースのプロパティを表示します。

このコマンドの形式は以下のようになります。

```
showres|sr className resourceName [next] ¥
    [{props|addprops}(propNames)]
```

addprops(*propName*)

表示するプロパティ(ルーラ)を設定します。プロパティのリストは現在のルーラに追加されます。ルーラは、現在のクエリに対してのみ設定され、現在のクエリが終了すると前のルーラ設定に戻ります。

className

リソースが属するクラスの名前を指定します。

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトのクエリ サイズは 100 に設定されています。

props(all|*propName*)

表示するプロパティ(ルーラ)を設定します。

設定したルーラは将来のクエリにも有効です。

resourceName

className で指定したクラスの既存の Windows リソース名を指定します。

showusr コマンド - ネイティブ ユーザ プロパティの表示

ネイティブ UNIX 環境で有効

showusr コマンドは、ネイティブ オペレーティング システムに定義されている 1 人以上のユーザのプロパティを表示します。

注: このコマンドは AC 環境にもありますが、動作が異なります。

注: UNIX の場合、環境設定(seos.ini)に指定されているファイルを対象にして、ユーザの読み込み、追加、更新、および削除が行われます。デフォルト設定では、このファイルは /etc/passwd です。詳細については、「*UNIX エンドポイント管理ガイド*」を参照してください。

このコマンドの形式は以下のようになります。

```
{showusr|su} userName [next] ¥
    [{props|addprops}(propNames)]
```

addprops(*propName*)

表示するプロパティ(ルーラ)を設定します。プロパティのリストは現在のルーラに追加されます。ルーラは、現在のクエリに対してのみ設定され、現在のクエリが終了すると前のルーラ設定に戻ります。

userName

ネイティブ プロパティを表示するユーザの名前を指定します。既存のネイティブ ユーザ名を指定します。複数のユーザのプロパティを表示する場合は、ユーザ名のリストを丸かっこで囲み、各ユーザ名をスペースまたはカンマで区切ります。

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトのクエリ サイズは 100 に設定されています。

props(all|*propName*)

表示するプロパティ(ルーラ)を設定します。

設定したルーラは将来のクエリにも有効です。

例

UNIX ユーザ *leslie* の詳細を *unix* 環境にいるときに一覧表示するには、以下のコマンドを入力します。

```
showusr leslie
```

詳細情報:

chusr コマンド - UNIX ユーザの変更 (P. 195)
chusr コマンド - Windows ユーザの変更 (P. 219)
show[x]usr コマンド - ユーザ プロパティの表示 (P. 175)

xaudit コマンド - システム アクセス制御リストの変更

xaudit コマンドは、システム アクセス制御リスト(SACL)にエントリを追加します。このリスト内の各エントリには、指定したユーザまたはグループがリソースへのアクセス権を取得しようとしたときに、監査メッセージが記録されます。xaudit- は、SACL からエントリを削除するコマンドです。このコマンドは、FILE、PRINTER、REGKEY、DISK、COM、SHARE タイプのリソースに対して有効です。

このコマンドの形式は以下のようになります。

```
xaudit className resourceName ¥
    [failure(auditMode)]¥
    [gid(groupName)] ¥
    [success(auditMode)] ¥
    [uid(userName)]
```

className

リソースが属するリソース タイプの名前を指定します。

failure(*auditMode*)

リソースに対して試みられた不正なアクセスを記録します。

auditmode の有効な値は、リソースが属するリソース タイプによって次のように異なります。

注: 監査モードを設定できるのは NTFS ファイルのみです。

- **DISK** および **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

その他すべてのリソース タイプ: *none* および *all*

gid(*groupName*)

リソースへのアクセスが監査対象になるグループを指定します。複数のグループを指定する場合は、各グループ名をスペースまたはカンマで区切ります。

resourceName

システム アクセス制御リスト(SACL)を変更するリソース レコードの名前を指定します。

success(*auditMode*)

リソースに対して許可されたアクセスを記録します。

auditmode の有効な値は、リソースが属するリソース タイプによって次のように異なります。

注: 監査モードを設定できるのは NTFS ファイルのみです。

- **DISK** および **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

その他すべてのリソース タイプ: *none* および *all*

uid(*userName*)

リソースへのアクセスが監査対象になるユーザを指定します。複数のユーザを指定する場合は、各ユーザ名をスペースまたはカンマで区切ります。Windows データベースで定義されているすべてのユーザを指定する場合は、*userName* にアスタリスク(*)を指定します。

詳細情報:

xaudit- コマンド - システム アクセス制御リストの削除 (P. 243)

xaudit- コマンド - システム アクセス制御リストの削除

xaudit- は、SACL からエントリを削除するコマンドです。このコマンドは、FILE、PRINTER、REGKEY、DISK、COM、SHARE タイプのリソースに対して有効です。

このコマンドの形式は以下のようになります。

```
xaudit- className, resourceName ¥
     [gid(groupName)] ¥
     [uid(userName)]
```

className

リソースが属するリソース タイプの名前を指定します。

gid(*groupName*)

リソースへのアクセスが監査対象になる 1 つ以上のグループを指定します。複数のグループを指定する場合は、各グループ名をスペースまたはカンマで区切ります。

resourceName

システム アクセス制御リスト(SACL)を削除するリソース レコードの名前を指定します。

uid(*userName*)

リソースへのアクセスが監査対象になるユーザを指定します。複数のユーザを指定する場合は、各ユーザ名をスペースまたはカンマで区切ります。Windows データベースで定義されているすべてのユーザを指定する場合は、*userName* にアスタリスク(*)を指定します。

詳細情報:

xaudit コマンド - システム アクセス制御リストの変更 (P. 241)

Policy Model 環境の selang コマンド

このセクションでは、Policy Model 環境上で実行される selang コマンドのすべてをアルファベット順に説明します。

backuppmd コマンド - PMDB のバックアップ

pmd 環境で有効

backuppmd コマンドは、PMDB データベース内のデータを指定されたディレクトリへバックアップします。ポリシー、デプロイメント情報、環境設定ファイルなど、PMDB データベース内のすべてのデータがバックアップされます。

DMS では、コマンド形式は以下のようになります。

```
backup pmdName destination(path)
```

PMDB では、コマンド形式は以下のようになります。

```
backup pmdName [destination(path)|hir_host(name)]
```

destination(*path*)

バックアップ ファイルが格納されるディレクトリを定義します。

注: パスを指定しない場合、ファイルは *_pmd_backup_directory_* トークンで指定されたデフォルトの場所にバックアップされます。

デフォルト: (UNIX) *ACInstallDir*/data/policies_backup/pmdName

デフォルト: (Windows) *ACInstallDir*/data/policies_backup/pmdName

pmdName

バックアップする PMDB または DMS の名前を定義します。

hir_host(*name*)

階層内のすべての PMDB を指定するホスト *name* にバックアップし、PMDB サブスクライバを変更して、バックアップが *name* ホストから削除されても、サブスクリプションが機能するようにします。

注: このコマンドがサポートされるのは、マスタ PMDB と子 PMDB が同じホストにデプロイされる場合のみです。

createpmd コマンド - PMDB のホスト上への作成

pmd 環境で有効

createpmd コマンドは、リモート ホスト上に PMDB を定義します。1 人以上のユーザを PMDB の管理者、監査者、およびパスワード管理者に指定できます。特定の PMDB に対する親 PMDB および 1 つ以上のサブスクライバ PMDB を定義することもできます。createpmd コマンドは、リモート ホストから実行できます。

このコマンドの形式は以下のようになります。

```
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 ...]*)

1 人以上の内部ユーザを PMDB 管理者に指定します。複数のユーザはスペースで区切ります。

auditors(*user [user ...]*)

PMDB の監査ファイルを表示できる内部ユーザを 1 人以上指定します。複数のユーザはスペースで区切ります。

pwmans(*user [user ...]*)

1 人以上のユーザを PMDB パスワード管理者に指定します。複数のユーザはスペースで区切ります。

parentpmd(*pmdname@host)*

作成している PMDB の親 PMDB の名前を指定します。

注: selang の remote コマンドで複数の親 Policy Model を定義する場合は、二重引用符を使用する必要があります。たとえば、Policy Model を作成し、その親を定義する場合、以下のコマンドを使用します。

```
createpmd subs2 admins(abc123 root) auditors(abc123 root) desktop(pcp36949) ¥
parentpmd("aa@pcp36949,bb@pcp36949")
```

desktop(*host [host ...]*)

管理者が PMDB の管理に使用する 1 つ以上のホストを指定します。複数のホストはスペースで区切ります。デフォルトでは、新しい PMDB のホストが設定されます。

subscribers(*host | pmd [host | pmd ...]*)

新しい PMDB のサブスクライバになるホストまたは PMDB を指定します。複数のホストまたは pmd はスペースで区切ります。

pwdfile(*filename*)

PMDB パスワード ファイルを指定します。

grpfile(*filename*)

PMDB グループ ファイルを指定します。

nis

新しい PMDB のホスト上で NIS 設定を実行し、UNIX のすべての更新内容をフィルタ処理するフィルタ ファイルを作成します。

xadmins(*user [user ...]*)

1 人以上のエンタープライズ ユーザを PMDB 管理者に指定します。複数のユーザはスペースで区切ります。

xauditors(*user [user ...]*)

PMDB の監査ファイルを表示できるエンタープライズ ユーザを 1 人以上指定します。複数のユーザはスペースで区切ります。

pwmans(*user [user ...]*)

1 人以上のエンタープライズ ユーザを PMDB パスワード管理者に指定します。複数のユーザはスペースで区切ります。

deletepmd コマンド - PMDB のホストからの削除

pmd 環境で有効

deletepmd コマンドは、リモート ホストから以下の項目を削除します。

- PMDB の selang 保護ファイル
  - データベース ファイル
  - レジストリ エントリ
- PMDB ディレクトリの内容
- PMDB ディレクトリ

重要: PMDB を削除する場合、PMDB の各ファイルを手動で削除しないでください(処理に重大な問題が生じるのを防ぐため)。PMDB に対しては deletepmd コマンドを必ず使用してください。

このコマンドの形式は以下のようになります。

`deletepmd` *`pmdname`*

findpmd コマンド - ホスト上の PMDB の一覧表示

pmd 環境で有効

findpmd コマンドは、接続先のホストの PMDB とデーモンの読み込み状況を一覧表示します。

このコマンドの形式は以下のようになります。

```
findpmd
```

listpmd コマンド - PMDB に関する情報の一覧表示

pmd 環境で有効

listpmd コマンドは、PMDB とそのサブスクライバ、更新ファイル、およびエラー ログに関する情報を一覧表示します。オプションの指定がない場合は、*pmdName* で指定した Policy Model のすべてのサブスクライバが一覧表示されます。

このコマンドの形式は以下のようになります。

```
listpmd pmdName ¥
    [{info|subscriber(subNames)|cmd(offset) ¥
    |errors|all_errors|log}] ¥
    [next]
```

cmd*(offset)*

更新ファイル内のすべてのコマンドおよび各コマンドのオフセットを表示します。

オフセットは、ファイル内での更新の位置を示します。オフセットを指定すると、リストはオフセット位置から開始されます。offset が指定されていない場合は、更新ファイルの先頭から表示が開始されます。

注: 更新ファイルには、PMDB によって伝達する必要がある更新情報、またはすでに伝達済みの更新情報が保存されます。オフセットは、サブスクライバに送信する必要がある次の更新情報の位置を示します。更新ファイルの初期オフセットと最新のオフセットが表示されます。

errors|all_errors

Policy Model のエラー ログを表示します。*errors* パラメータを指定すると、接続失敗以外のエラーを除く、すべてのエラーのタイプが表示されます。*all_errors* を指定すると、すべてのエラーが表示されます。

info

pmdName に指定した Policy Model に関する一般情報を表示します。Policy Model に親が存在するかどうかなどの情報が表示されます。

next

要求されたデータの一部を表示します。このオプションは、設定されているクエリ サイズよりクエリ データが大きい場合に便利です。

最大クエリ サイズは、query_size 環境設定に基づいて決まります。デフォルトのクエリ サイズは 100 に設定されています。

pmdname

情報の一覧表示対象 PMDB の名前を指定します。

subscriber(*subNames*)

Policy Model のサブスクライバおよび各サブスクライバのステータスを一覧表示します。エラーの数、可用性、オフセット、次に伝達するコマンドなどの情報が表示されます。*subNames* パラメータを指定すると、サブスクライバのサブセットを選択できます。

log

Policy Model の一般ログ ファイルを表示します。

例: 選択したサブスクライバの PMDB サブスクライバ情報の表示

名前が *compInt* で始まる myPMDB Policy Model のサブスクライバの一覧を表示するには、以下のコマンドを入力します。

```
listpmd myPMDB subscriber(compInt*)
```

pmd コマンド - PMDB の管理

pmd 環境で有効

pmd コマンドは、Policy Model エラー ログの消去、サブスクライバ リストの更新、Policy Model サービスの開始または停止、および更新ファイルの切り捨てを行います。

このコマンドの形式は以下のようになります。

```
pmd pmdName ¥
    {[release(subname)|start|stop|truncate(offset)|lock|unlock ¥
    |reloadini|startlog|killog|clrerror|backup|operation]}
```

backup

Policy Model をバックアップ ステータスに移行します。

clrerror|clrerr

Policy Model のエラー ログを消去します。

killog

Policy Model の一般ログ ファイルを無効にします。このオプションを指定すると、メッセージがログに記録されなくなります。

重要: PMDB サービスを停止するのに kill コマンドは使用しないでください。

lock

Policy Model をロック ステータスに移行し、Policy Model がそのサブスクライバへ更新を送信するのを停止します。

operation

Policy Model をバックアップ ステータスから運用ステータスに移行します。

pmdname

選択したオプションの実行対象 PMDB の名前を指定します。

release(*subName*)

利用できないサブスクライバのリストから、*subName* で指定されたサブスクライバを削除します。その結果、サブスクライバはただちに更新情報を受信できます。*subName* には、更新情報を受信できるようにするサブスクライバを指定します。

reloadini

(UNIX のみ) Policy Model の pmd.ini ファイルおよび seos.ini ファイルを読み込み直し、Policy Model デーモンを再ロードする必要なく環境設定を変更できるようにします。

startlog

Policy Model の一般ログ ファイルへの書き込みを有効にします。このオプションは、ログ ファイルが無効の場合に使用します。

start

CA Access Control Policy Model サービスを開始します。このオプションは、ほかに実行するコマンドがない場合に使用します。

stop

CA Access Control Policy Model デーモンおよびサービスを停止します。

truncate|trunc[(*offset*)]

更新ファイルからエントリを削除します。オフセットを指定していない場合、ファイルは可能な最大オフセットで切り捨てられます。可能な最大オフセットは、正常にサブスクライバを更新した最後のコマンドの位置になります。*offset* を指定すると、指定したオフセットまでのすべてのエントリが削除されます。

注: 更新ファイルを切り捨てるには、開始オフセットから減算した結果のオフセットではなく、*listpmd* コマンドによって取得した正確なオフセットを使用する必要があります。

unlock

Policy Model をロックからロック解除ステータスへ移行し、Policy Model がそのサブスクライバへ更新を送信できるようにします。

restorepmd コマンド - PMDB のリストア

pmd 環境で有効

restorepmd コマンドを使用すると、ローカル ホスト上の PMDB をリストアします。PMDB のリストアに使用するバックアップ ファイルは、CA Access Control のリストア ホストと同じプラットフォーム、オペレーティング システムおよびバージョンを実行するホストから作成されている必要があります。また、リストア ホスト上で CA Access Control が実行中である必要があります。

注: PMDB を別の端末にバックアップおよびリストアする場合、PMDB はリストアされた PMDB データベースにあるターミナル リソースの更新を、自動的には行いません。新しいターミナル リソースはリストアされた PMDB に追加する必要があります。新しい端末リソースを追加するには、リストアされた PMDB を停止して「*selang -p pmdb*」をコマンド実行します。その後、リストアされた PMDB を起動します。

このコマンドの形式は以下のようになります。

`restorepmd` *`pmdName`* `[source(`*`path`*`)] [admin(`*`user`*`)] [xadmin(`*`user`*`)] [parentpmd(`*`name`*`)]`

admin(*user*)

（UNIX）リストアされた PMDB の管理者として内部ユーザを定義します。

pmdName

リストアする PMDB の名前を定義します。

parentpmd(*name*)

（オプション）リストアされた PMDB の親の名前を定義します。名前は「*pmd@host*」の形式で指定します。

source(*path*)

（オプション）バックアップ ファイルが配置されているディレクトリを定義します。ソース ディレクトリを指定しない場合、PMDB はデフォルトの場所にあるファイルからリストアされます。デフォルトの場所は「*_pmd_backup_directory_*」トークンで定義されます。

デフォルト:（UNIX）*ACInstallDir*/data/policies_backup/pmdName

デフォルト:（Windows）*ACInstallDir*/data/policies_backup/pmdName

xadmin(*user*)

（UNIX）リストアされた PMDB の管理者としてエンタープライズ ユーザを定義します。

subs コマンド - サブスクライバまたはサブスクライブ データベースの追加

pmd 環境で有効

subs コマンドは、親 PMDB にサブスクライバを追加するか、親 PMDB に対してデータベースをサブスクライブします。

ホストを PMDB にサブスクライブする場合は、以下の条件が満たされている必要があります。

- ホストが起動していること
- そのホスト上で CA Access Control が実行中であること
- PMDB が、サブスクライブされるホストの親 PMDB であること

PMDB を別の PMDB にサブスクライブする場合は、以下の条件が満たされている必要があります。

- サブスクライブされる PMDB の parent_pmd 環境設定に、サブスクライブ先となる PMDB（親 PMDB）の名前が設定されていること
- サブスクライブされる PMDB が格納されているホスト上で CA Access Control が実行中であること

このコマンドの形式は以下のようになります。

```
subs pmdname ¥
    [subs(subsname)] ¥
    [host_type(mfHost) sysid(sysID) mf_admin(mfAdmin) port(port)] ¥
    {offset(offset) }
```

または

```
subs pmdname [newsubs(subsname)]
```

または

```
subs pmdname [parentpmd(pmdname2@host)]
```

host_type(*mfhost*)

サブスクライバのメインフレーム ホスト タイプを指定します。

mf_admin(*mfAdmin*)

サブスクライバのメインフレーム管理者を指定します。

newsubs(*subsname*)

subsname を *pmdname* という Policy Model にサブスクライブし、新しいサブスクライバに PMDB 全体、パスワード、およびグループ ファイルの内容を送信します。

parentpmd(*pmdName2@host*)

pmdName2@host で指定された PMDB を *pmdName* の親 Policy Model にします。

pmdname

選択したオプションの実行対象 PMDB の名前を指定します。

port(*port*)

サブスクライバのポート番号を指定します。

subs(*subsname*)

サブスクライバを PMDB に割り当てます。

sysid(*sysId*)

サブスクライバのシステム ID を指定します。

subspmd コマンド - 親 PMDB の変更

pmd 環境で有効

subspmd コマンドは、接続先ホストの CA Access Control データベースの親を変更します。

このコマンドの形式は以下のようになります。

```
subspmd parentpmd(pmdname@host)
```

parentpmd(*pmdname@host*)

pmdname@host を現在のホストの親 Policy Model にします。

unsubs コマンド - サブスクライバの削除

pmd 環境で有効

unsubs コマンドは、Policy Model のサブスクライバ リストからサブスクライバを削除します。

このコマンドの形式は以下のようになります。

unsubs *pmdName* subs(*subName*)

pmdname

選択したオプションの実行対象 PMDB の名前を指定します。

subs(*subName*)

pmdname で指定されるサブスクライバ リストから削除するサブスクライバの名前を指定します。

第 4 章: クラスとプロパティ

ここでは、**CA Access Control データベースおよびネイティブ オペレーティング システムに定義されているすべてのクラスの各プロパティについて説明します。変更可能なプロパティ、それらのプロパティを更新する際に使用する selang パラメータ、およびそれらのパラメータを指定するコマンドに関する情報を、クラス別にまとめて環境ごとにアルファベット順に示します。**

このセクションには、以下のトピックが含まれています。

クラスとプロパティの情報

クラスとプロパティの情報には以下の表記規則が適用されています。

- 各プロパティの最初に記載される説明部分では、クラスのレコードのキーを定義します。

  キーは、新しいレコードの作成時に指定するレコード識別子です。レコードの作成が完了すると、キーは変更できないプロパティになります。

- パラメータにマイナス記号 (-) を付けて入力すると、データベースからそのパラメータが削除されます。

  たとえば、*comment* で適切なテキストを指定すると、データベース レコードにコメントが追加されますが、*comment-* を指定すると、データベースからコメントが削除されます。レコードを作成する場合は、パラメータにマイナス記号 (-) を付けることはできません。

- データベースには、アクセサ クラスとリソース クラスという 2 種類のクラスがあります。

  アクセサ クラス(USER および GROUP)のレコードを操作する場合は、リソース クラスに対して使用する selang のコマンド セットとは異なるコマンド セットを使用します

  - USER クラスのレコードを操作するには、chusr、editusr、および newusr を使用します。
  - GROUP クラスのレコードを操作するには、chgrp、editgrp、および newgrp を使用します。
  - リソース クラスのレコードを操作するには、chres、editres、および newres を使用します。リソースがファイルの場合は、chfile コマンドまたは editfile コマンドを使用することもできます。
  - レコードのプロパティを一覧表示するには、showgrp、showres、showfile、または showusr を使用します。
  - リソース レコードの ACL を追加、変更、または削除するには、authorize および authorize- を使用します。

詳細情報:

selang コマンド リファレンス (P. 43)

AC 環境のクラス

このセクションでは、CA Access Control データベースに存在するすべてのクラスとプロパティ(AC 環境のクラス)をアルファベット順に説明します。

ACVAR クラス

ACVAR クラスの各レコードは、エンドポイント上にデプロイされた変数を定義します。このクラスを無効にすることはできません。

ACVAR クラスのキーは、変数の名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

OWNER

レコードを所有するユーザまたはグループを定義します。

POLICIES

(情報のみ)この変数を使用するポリシー(POLICY オブジェクト)のリストです。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

VARIABLE_TYPE

変数のタイプを定義します。有効な値は以下のとおりです。

built-in

CA Access Control によってインストール時に作成された変数を使用するように指定します。スタティック変数はエンドポイントのシステム設定に基づいて解決されます。

注: built-in 変数を変更または削除することはできません。

osvar

オペレーティング システムの値に基づいて変数を解決するように指定します。

regval

(Windows)レジストリの値に基づいて変数を解決するように指定します。

注: REG_SZ または REG_EXPAND_SZ レジストリ タイプを指すレジストリ タイプの値のみを定義することができます。

static

変数を定義した文字列値に解決するように指定します。

注: 既存の変数の変数タイプを変更することはできません。

VARIABLE_VALUE

変数の値を定義します。

注: このプロパティでは、変数の値にある入れ子の変数は展開されません。

VARIABLE_EXPANDED_VALUE

(情報のみ)変数の値を定義し、その変数の値にある入れ子の変数を展開します。

ADMIN クラス

ADMIN クラスの各レコードには、ADMIN 以外のユーザに対して特定のクラスの管理を許可するための定義が含まれます。委任されたユーザが管理する CA Access Control の各クラスを表すには、ADMIN クラスのレコードを作成する必要があります。ADMIN レコードには、各クラスのアクセス権限を持つアクセサのリストが格納されます。条件付きアクセス制御リスト(CACL)もサポートされます。

ADMIN クラス レコードのキーは、保護されるクラスの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

AAUDIT

（情報のみ）。CA Access Control が監査するアクティビティの種類を表示します。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

AGENT クラス

AGENT クラスの各レコードは、CA SSO でエージェントとして使用されるオブジェクトを定義します。

AGENT クラスのレコードのキーは、エージェントの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

AGENT_TYPE

エージェントのタイプです。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

OWNER

レコードを所有するユーザまたはグループを定義します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

AGENT_TYPE クラス

AGENT_TYPE クラスの各レコードは、CA SSO で使用されるエージェント タイプを定義します。

AGENT_TYPE クラスのレコードのキーは、エージェントのタイプです。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

AGENT_FLAG

属性に関する情報が含まれます。フラグには、以下の値を指定できます。

- **aznchk** - この属性を権限付与に使用するかどうかを指定します。
- **predef**（事前定義済み）、**freetext** 自由形式のテキスト）、または **userdir**（ユーザ ディレクトリ）- これらの値を使用して、ユーザ属性のソースを指定します。
- **user** または **group** - これらの値を使用して、属性（アクセサ）がユーザであるかグループであるかを指定します。

AGENT_LIST

agent_type パラメータの値として AGENT_TYPE オブジェクトを指定して作成された AGENT クラスのオブジェクトのリストです。たとえば、このプロパティは、AGENT クラスのオブジェクトの作成時に暗黙的に更新されます。

CLASSES

このエージェントに関連するクラスまたはリソースの複数文字列リストです。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

OWNER

レコードを所有するユーザまたはグループを定義します。

UPDATE_TIME

（情報のみ）レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

APPL クラス

APPL クラスの各レコードは、CA SSO で使用されるアプリケーションを定義します。

APPL クラスのレコードのキーは、アプリケーションの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

APPLTYPE

CA SSO で使用されます。

AZNACL

権限 ACL を定義します。これは、リソースの説明に基づいてリソースへのアクセスを許可する ACL です。説明は、オブジェクトではなく認証エンジンに送信されます。一般に、AZNACL が使用される場合、オブジェクトはデータベースにありません。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

CAPTION

デスクトップのアプリケーション アイコンの下に表示されるテキストです。デフォルトは APPL クラスのレコードの名前です。

制限: 47 文字の英数字。

CMDLINE

アプリケーション実行可能ファイルのファイル名です。CA SSO で使用されます。

制限: 255 文字。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CONTAINED_ITEMS

レコードがコンテナである場合に、コンテナに含まれるアプリケーションのレコード名です。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドの item[-](*applName*) パラメータを使用します。

CONTAINERS

レコードが他のアプリケーションに含まれている場合は、コンテナ アプリケーションのレコード名です。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

DIALOG_FILE

アプリケーションのログイン シーケンスを含むディレクトリ内の CA SSO スクリプトの名前です。デフォルトのディレクトリの場所は、/usr/sso/scripts です。デフォルト値は「no script」です。

このプロパティを変更するには、chres、editres、newres の各コマンドで、script[-](*fileName*) パラメータを使用します。

GROUPS

アプリケーションの使用を許可されているユーザ グループのリストです。

HOST

アプリケーションが存在するホストの名前です。

このプロパティを変更するには、chres、editres、newres の各コマンドで、host[-](*hostName*) パラメータを使用します。

ICONFILE

デスクトップに表示するアプリケーションのアイコンが保存されているファイルのファイル名または完全パスです。CA Access Control では、エンド ユーザのワークステーションにアイコン ファイルが存在することを前提としています。ファイル名のみを入力した場合は、次の順序でファイルが検索されます。

1. 現在のディレクトリ
2. 環境変数 PATH に指定されているディレクトリ

デフォルトは、ワークステーションのデフォルト アイコンです。

ICONID

アイコン ファイル内のアイコンの（必要に応じた）ID 番号です。ICONID が指定されていない場合は、デフォルト アイコンが使用されます。

IS_CONTAINER

アプリケーションがコンテナかどうかを指定します。デフォルトは「no」です。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドの container[-] パラメータを使用します。

IS_DISABLED

アプリケーションが無効化された状態かどうかを指定します。アプリケーションが無効化された状態である場合、ユーザはアプリケーションにログインできません。この機能は、ユーザがアプリケーションを変更しているときに、他のユーザがアプリケーションにログインできないようにする場合に便利です。無効化された状態のアプリケーションはアプリケーション メニュー リストに表示されますが、ユーザがそのアプリケーションを選択すると、メッセージが表示され、ログインは中止されます。デフォルトは「not disabled」です。

IS_HIDDEN

アプリケーションを起動できるユーザのデスクトップにもアプリケーション アイコンを表示するかどうかを指定します。たとえば、他のアプリケーションにパスワードを提供する目的のみを果たすアプリケーションなどの マスタ アプリケーションを非表示にすることができます。デフォルトは「not hidden」です。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドの hidden[-] パラメータを使用します。

IS_SENSITIVE

事前設定された時間が経過した後にユーザがアプリケーションを開いた場合に、再認証が必要かどうかを指定します。デフォルトは「not sensitive」です。

このプロパティを変更するには、chres、editres、newres の各コマンドで、sensitive[-] パラメータを使用します。

LOGIN_TYPE

ユーザ パスワードの指定方法です。値は、*pwd*（平文パスワード）、*otp*（ワンタイム パスワード）、*appticket*（**メインフレーム アプリケーション認証専用チケット**）、**none**（**パスワード不要**）、または **passticket**（**IBM が開発したワンタイム パスワード置換フォーマット。メインフレームのセキュリティ パッケージで使用される**）です。**デフォルトは pwd です。**

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドの login_type(*value*) パラメータを使用します。

MASTER_APPL

他のアプリケーションにパスワードを提供するアプリケーションのレコード名です。デフォルトは「no master」です。

このプロパティを変更するには、chres、editres、newres の各コマンドで、master[-](*applName*) パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PGMDIR

アプリケーションの実行可能ファイルが格納されているディレクトリまたはディレクトリのリストです。CA SSO で使用されます。

PWD_AUTOGEN

アプリケーション パスワードを CA SSO で自動的に生成するかどうかを指定します。デフォルトは no です。

PWD_SYNC

アプリケーション パスワードを自動的に他のアプリケーションのパスワードと同一にするかどうかを指定します。デフォルトは no です。

PWPOLICY

アプリケーションに適用するパスワード ポリシーのレコード名です。パスワード ポリシーは、新しいパスワードの妥当性をチェックし、パスワードの有効期限を定義する一連のルールです。デフォルトでは、妥当性チェックは行われません。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SCRIPT_POSTCMD

ログイン スクリプトの後に 1 つ以上のコマンドを実行するかどうかを指定します。

SCRIPT_PRECMD

ログイン スクリプトの前に 1 つ以上のコマンドを実行するかどうかを指定します。

SCRIPT_VARS

CA SSO で使用されます。アプリケーションごとに保存されるアプリケーション スクリプトの変数値を含む変数リストです。

TKTKEY

CA SSO でのみ使用されます。

TKTPROFILE

CA SSO でのみ使用されます。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

AUTHHOST クラス

AUTHHOST クラスの各レコードは、CA SSO の認証ホストを定義します。

AUTHHOST クラスのレコードのキーは、認証ホストの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ (ユーザおよびグループ)、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト (ACL) の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

AZNACL

権限 ACL を定義します。これは、リソースの説明に基づいてリソースへのアクセスを許可する ACL です。説明は、オブジェクトではなく認証エンジンに送信されます。一般に、AZNACL が使用される場合、オブジェクトはデータベースにありません。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

ETHINFO

ホストのイーサネット情報です。

GROUPS

リソース レコードが属する GAUTHHOST クラスまたは CONTAINER クラスのレコードのリストです。

AUTHHOST クラスのレコードのこのプロパティを変更するには、適切な CONTAINER クラスまたは GAUTHHOST クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

KEY

CA SSO でのみ使用されます。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PATH

CA SSO でのみ使用されます。

PROPERTIES

UNIX の dbdump でのみ使用されます。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティレベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

SEED

CA SSO でのみ使用されます。

SERNUM

認証ホストのシリアル番号です。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UNTRUST

リソースが信頼されているかどうかを定義します。UNTRUST プロパティが設定されている場合、アクセサはこのリソースを使用できません。UNTRUST プロパティが設定されていない場合、アクセサのアクセス権限の決定には、このリソースについてデータベースにリストされている他のプロパティが使用されます。trusted リソースに何らかの変更が加えられると、CA Access Control によって UNTRUST プロパティが自動的に設定されます。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの trust[-] パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

USER_DIR_PROP

（情報のみ）。ユーザのディレクトリの名前です。

USER_FORMAT

CA SSO でのみ使用されます。

USERALIAS

特定の認証ホストに定義されているユーザのすべての別名を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

CALENDAR クラス

CALENDAR クラスの各レコードは、CA Access Control で時間帯制限が適用されるユーザ、グループ、およびリソースの Unicenter TNG カレンダ オブジェクトを定義します。CA Access Control により、適用された特定の時間帯に Unicenter TNG のアクティブなカレンダが取得されます。カレンダをリソースに割り当てるには、chgrp、chres、chusr、editgrp、editres、editusr、newgrp、newres、および newusr の各コマンドの calendar(*calendarName*) プロパティを使用します。

以下のクラスには、そのクラスのレコード内に CALENDAR プロパティがあります。これらのリソース クラスの各オブジェクトには、CALENDAR クラス オブジェクトを *1* つのみ割り当てることができます。

- ADMIN
- APPL
- AUTHHOST
- CONNECT
- CONTAINER
- DOMAIN（Windows のみ）
- FILE
- GFILE
- GHOST

- GROUP
- GSUDO
- GTERMINAL
- HOST
- HOSTNET
- HOSTNP
- LOGINAPPL（UNIX のみ）
- MFTERMINAL
- PROCESS
- PROGRAM
- REGKEY（Windows のみ）
- SUDO
- SURROGATE
- TCP
- TERMINAL
- USER

CALENDAR クラスのキーは、Unicenter TNG カレンダの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

OWNER

レコードを所有するユーザまたはグループを定義します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

CATEGORY クラス

CATEGORY クラスの各レコードは、データベース内のセキュリティ カテゴリを定義します。

CATEGORY クラス レコードのキーは、セキュリティ カテゴリの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

OWNER

レコードを所有するユーザまたはグループを定義します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

CONNECT クラス

CONNECT クラスの各レコードは、ローカル ホストからの接続に TCP over IPv4 を使用できるリモート ホストを定義します。

注: IP 通信用の CA Access Control アクセス ルールは IPv4 にのみ適用されます。CA Access Control は IPv6 によるアクセスを管理しません。

注: CONNECT クラスがアクセスの基準として使用されている場合、TCP クラスは事実上アクティブにできません。接続を保護するには、TCP クラスと CONECT クラスのどちらかを使用します。両方は使用しません。

CONNECT クラスのレコードのキーは、リモート ホストの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

CONTAINER クラス

CONTAINER クラスの各レコードは、他のリソース クラスにあるオブジェクトのグループを定義します。これにより、複数の異なるオブジェクトのクラスに 1 つのルールを適用する場合に、アクセス ルールを定義する作業が簡略化されます。CONTAINER クラス レコードのメンバは、以下のいずれかのクラスのオブジェクトになることができます。

- APPL
- AUTHHOST
- CONNECT
- CONTAINER
- DICTIONARY
- DOMAIN（Windows のみ）
- FILE
- GAPPL
- GAUTHHOST
- GFILE
- GHOST
- GSUDO
- GTERMINAL
- HNODE
- HOLIDAY
- HOST
- HOSTNET
- HOSTNP
- MFTERMINAL
- PARAM_DESC
- POLICY
- PROCESS
- PROGRAM
- REGKEY（Windows のみ）

- RULESET
- SUDO
- SURROGATE
- TCP
- TERMINAL
- WEBSERVICE

注: CONTAINER レコードは、他の CONTAINER レコードにネストすることができます。

オブジェクトを CONTAINER レコードのメンバとして指定する前に、適切なクラスにそのオブジェクトのレコードを作成する必要があります。

コンテナ内のオブジェクトが、その適切なクラス レコード内に ACL を持たない場合、そのオブジェクトは、所属している CONTAINER レコードの ACL を継承します。

CONTAINER クラスのキーは、CONTAINER レコードの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

MEMBERS

グループのメンバである任意のクラスのオブジェクトのリストです。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドで mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

DEPLOYMENT クラス

DEPLOYMENT クラスの各レコードは、エンドポイントのデプロイ タスクまたはデプロイ解除タスクを定義します。デプロイ タスクには、必要に応じてポリシーをデプロイまたはデプロイ解除するために必要なエンドポイントに関する情報が含まれます。

DEPLOYMENT クラスのキーは、デプロイ タスクの名前で、通常は自動生成されます。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

DMS_NAME

デプロイ タスクが作成された DMS の名前を指定します。

GPOLICY

デプロイ タスクの作成対象であるポリシーの名前を指定します。

GROUPS

デプロイ タスクが属しているデプロイ パッケージ(GDEPLOYMENT)を指定します。

HNODE

デプロイ タスクの作成対象であるホストの名前を指定します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OPERATION

このデプロイ タスクの結果としてエンドポイントが実行する操作の種類を指定します。Deploy と Undeploy のどちらかです。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

POLICY_VERSION

デプロイ タスクの作成対象であるポリシー バージョンの名前を指定します。

RESULT_MESSAGE

デプロイまたはデプロイ解除 selang スクリプトからの出力を定義します。これは、ポリシーのデプロイまたはデプロイ解除スクリプトが実行されたときに selang が出力するメッセージです。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

STATUS

デプロイ タスクのステータスを定義します。以下のいずれかです。

- **Success** - ポリシーはエラーなくデプロイされました。
- **Warning** - エラーが発生しましたがデプロイ スクリプトは実行されました。
- **Fail** - デプロイ タスクの実行中にエラーが発生しました。
- **No Action** - デプロイ パッケージは実質的に空であり、何も実行することがありません。

  注: これは、ポリシーがこのホストに別のデプロイ パス経由ですでに割り当てられているためである可能性があります。
- **Not Executed** - ポリシーの検証によって、ポリシーに 1 つまたは複数のエラーが見つかりました。
- **Out of Sync** - ポリシーには、エンドポイントで変更された変数および変数の値が含まれています。
- **Pending Deployment** - ポリシーには未定義または未解決の変数が含まれています。
- **Pending Prerequisite** - デプロイ タスクは、前提となるポリシーがすべてデプロイされている場合のみ実行されます。
- **Pending Dependents** - デプロイ タスクは、前提となるポリシーもすべてデプロイ解除されている場合のみ実行されます(デプロイ解除ポリシー)。
- **Fix** - デプロイ タスクは再デプロイされるのを待機しています。

TARGETYPE

ホスト(ターゲット)のタイプを定義し、policyfetcher が CA Access Control デプロイ パッケージのみを処理するように制限します。値は UNAB、AC、None のいずれかになります。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

DICTIONARY クラス

DICTIONARY クラスの各レコードは、CA Access Control データベースに格納されている共通辞書内の、パスワードと比較する単語を定義します。ユーザがパスワードを変更すると、変更されたパスワードは、DICTIONARY クラスの各レコードと照合してチェックされます。

DICTIONARY クラスへのレコード(単語)の追加に加えて、ユーティリティまたはプログラムを実行することにより、外部ファイルからディクショナリへ単語をインポートすることができます。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

DOMAIN クラス

Windows で該当

DOMAIN クラスの各レコードは、Windows ネットワークのドメインを定義します。

DOMAIN レコードのキーは、ドメイン名です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

FILE クラス

FILE クラスの各レコードは、特定のファイル、特定のディレクトリ、または*ファイル名パターン*が一致しているファイルに対するアクセス権を定義します。まだ作成していないファイルについてもルールを定義できます。

デバイス ファイルおよびシンボリック リンクも他のファイルと同様に保護できます。ただし、リンクを保護しても、リンク先のファイルは自動的に保護されません

注: NTFS ファイル システムの場合、FILE クラスのレコードはファイルのストリームへのアクセスも定義します。ファイル ストリームの保護の詳細については、「*CA Access Control for Windows エンドポイント管理ガイド*」を参照してください。

スクリプトをファイルとして定義する場合は、ファイルに対する ***read*** アクセス権および ***execute*** アクセス権の両方を許可します。バイナリを定義する場合は、***execute*** アクセス権のみで十分です。

特別な _restricted グループに属していないユーザの場合、FILE クラスの _default レコード(_default レコードがない場合は UACC クラスの FILE のレコード)では、seos.ini ファイル、seosd.trace ファイル、seos.audit ファイル、および seos.error ファイルなど、*CA Access Control の一部であるファイルのみが保護されます*。これらのファイルは CA Access Control に明示的に定義されていませんが、CA Access Control によって自動的に保護されます。

FILE クラス レコードのキーは、レコードが保護するファイルまたはディレクトリの名前です。完全パスを指定する必要があります。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

グループ

リソース レコードが属する GFILE クラスまたは CONTAINER クラスのレコードのリストです。

DB プロパティ: GROUPS

FILE クラスのレコードのこのプロパティを変更するには、適切な CONTAINER クラスまたは GFILE クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UNTRUST

リソースが信頼されているかどうかを定義します。UNTRUST プロパティが設定されている場合、アクセサはこのリソースを使用できません。UNTRUST プロパティが設定されていない場合、アクセサのアクセス権限の決定には、このリソースについてデータベースにリストされている他のプロパティが使用されます。trusted リソースに何らかの変更が加えられると、CA Access Control によって UNTRUST プロパティが自動的に設定されます。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの trust[-] パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

GAPPL クラス

このクラスの各レコードは、CA SSO で使用するアプリケーションのグループを定義します。各アプリケーションの APPL クラスのレコードを作成した後に、そのレコードを GAPPL クラスのレコードに追加する必要があります。次に、APPL クラスのレコードを GAPPL クラスのレコードに明示的に関連付けてグループ化します。

GAPPL クラス レコードのキーは、GAPPL レコードの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

AZNACL

権限 ACL を定義します。これは、リソースの説明に基づいてリソースへのアクセスを許可する ACL です。説明は、オブジェクトではなく認証エンジンに送信されます。一般に、AZNACL が使用される場合、オブジェクトはデータベースにありません。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストです。

GAPPL クラスのレコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

MEMBERS

グループのメンバとなる、APPL クラスのオブジェクトのリストです。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドで mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

GAUTHHOST クラス

GAUTHHOST クラスの各レコードは、CA SSO で使用する認証ホストのグループを定義します。 各アプリケーションの AUTHHOST クラスのレコードを作成した後に、そのレコードを GAUTHHOST クラスのレコードに追加する必要があります。 次に、AUTHHOST クラスのレコードを GAUTHHOST クラスのレコードに明示的に関連付けてグループ化します。

GAUTHHOST クラス レコードのキーは、GAUTHHOST レコードの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ(ユーザおよびグループ)、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト(ACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

AZNACL

権限 ACL を定義します。これは、リソースの説明に基づいてリソースへのアクセスを許可する ACL です。説明は、オブジェクトではなく認証エンジンに送信されます。一般に、AZNACL が使用される場合、オブジェクトはデータベースにありません。

CALACL

リソースへのアクセスが許可されるアクセサ(ユーザおよびグループ)およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト(CALACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストです。

GAUTHHOST クラスのレコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

MEMBERS

グループのメンバとなる、AUTHHOST クラスのオブジェクトのリストです。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドで mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

GFILE クラス

GFILE クラスの各レコードは、特定のファイルまたはディレクトリのグループ、または名前パターンと一致するファイルに対して許可するアクセス権限を定義します。各アプリケーションの FILE クラス レコードを作成した後に、作成したレコードを GFILE レコードに追加する必要があります。次に、FILE クラスのレコードを GFILE クラスのレコードに明示的に関連付けてグループ化します。まだ作成していないファイルについても、FILE クラスのレコードを定義できます。

GFILE クラス レコードのキーは、GFILE レコードの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

MEMBERS

グループのメンバとなる、FILE クラスのオブジェクトのリストです。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドで mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

GDEPLOYMENT クラス

GDEPLOYMENT クラスの各レコードは、デプロイ パッケージを定義します。デプロイメント パッケージは DMS 上で自動的に作成され、特定のホスト向けに同じトランザクション(ポリシー割り当て、アップグレードなど)の結果として作成されるすべてのデプロイメント タスクをひとまとめにします。つまり、作成する各トランザクションが、必要な数のデプロイ タスク(DEPLOYMENT オブジェクト)を作成し、それをホスト(GDEPLOYMENT オブジェクト)ごとにグループ化します。

GDEPLOYMENT クラスのキーは、デプロイ パッケージの名前で、通常は自動生成されます。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GHNODE

このデプロイ パッケージの作成対象であるホスト グループの名前を指定します。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

HNODE

このデプロイ パッケージの作成対象であるホストを指定します。

MEMBERS

グループのメンバとなる、DEPLOYMENT クラスのオブジェクトのリストです。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドで mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

POLICY

このデプロイ パッケージの作成対象であるポリシーを指定します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

TRIGGER

このデプロイ パッケージを作成した理由を指定します。以下のいずれかです。

- Assign - ポリシーをホストに、またはホストをホスト グループに割り当てた結果
- AutoAssign -- ホストをホスト グループに自動的に割り当てる DMS の結果。
- UnAssign - ポリシーをホストから、またはホストをホスト グループから割り当て解除した結果
- Direct Deploy - 直接デプロイ アクションの結果
- Direct Undeploy - 直接デプロイ解除アクションの結果
- Upgrade - アップグレード アクションの結果
- Restore - ホスト(HNODE) 上での復元アクションの結果
- Hnode Deletion - ホスト(HNODE)の削除の結果
- Ghnode Deletion - ホスト グループ(GHNODE)の削除の結果
- Reset - ホストのリセットの結果
- Downgrade - ホスト上のポリシーのダウングレードの結果

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

GHNODE クラス

GHNODE クラスの各レコードは、ホスト グループ、またはホスト(HNODE オブジェクト)によるグループを定義します。各ホストの HNODE クラス レコードを作成した後に、作成したレコードを GHOST レコードに追加する必要があります。

このクラスは、ポリシーのデプロイと割り当ての管理に使用します。

GHNODE クラス レコードのキーは、ホスト グループの論理名です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

CRITERIA

自動的にホストをこのホスト グループに追加するために DMS が使用する基準を定義します。以下の HNODE プロパティと一致するか、これらを除外する基準を指定できます ATTRIBUTES、COMMENT、HNODE_INFO、HNODE_IP、HNODE_VERSION、NODE_TYPE

たとえば、Windows エンドポイントの HNODE レコードにはプロパティ HNODE_INFO=Windows があります。GHNODE レコードの CRITERIA プロパティが HNODE_INFO=Windows の値を持っている場合、DMS は自動的にすべての新しい Windows HNODE を GHNODE に追加します。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

MEMBERS

グループのメンバとなる、HNODE クラスのオブジェクトのリストです。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドで mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ (write など) と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

OWNER

レコードを所有するユーザまたはグループを定義します。

POLICIES

このオブジェクトにデプロイする必要があるポリシーのリストです。

POLICYASSIGN

このオブジェクトに割り当てられるポリシーのリストを定義します。

表示名 : 割り当てられたポリシー

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求 (デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

GHOST クラス

GHOST クラスの各レコードは、ホストのグループを定義します。各ホストの HOST クラス レコードを作成した後に、作成したレコードを GHOST レコードに追加する必要があります。サービスは、/etc/services ファイル（UNIX の場合）、¥system32¥drivers¥etc¥services ファイル（Windows の場合）、または他のサービス名解決方法を使用して、システムに定義する必要があります。サービスに許可を与える場合は、サービスの名前ではなく TCP/IP プロトコルのポート番号で指定できます。サービスを追加する場合は、サービスの名前ではなく TCP/IP プロトコルのポート番号で指定できます。次に、HOST クラスのレコードを GHOST クラスのレコードに明示的に関連付けてグループ化します。

GHOST クラスのレコードはアクセス ルールを定義します。このアクセス ルールは、インターネットで通信する際に、ホストのグループに属する他の端末（ホスト）がローカル ホストに対して持つアクセス権限を管理します。各クライアント グループ（GHOST レコード）について、INETACL プロパティに、ローカル ホストがホストに提供するサービスを制御するサービス ルールのリストが表示されます。

GHOST クラス レコードのキーは、GHOST レコードの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

INETACL

ローカル ホストからクライアント ホストのグループに提供可能なサービスおよび各サービスのアクセス タイプを定義します。アクセス制御リストの各要素には、以下の情報が含まれます。

サービス参照

サービス(ポート番号または名前)への参照です。すべてのサービスを指定する場合は、サービス参照としてアスタリスク(*)を入力します。

また、CA Access Control では、/etc/rpc ファイル(UNIX の場合)または ¥etc¥rpc ファイル(Windows の場合)に指定された動的なポート名もサポートしています。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

INETACL プロパティでアクセサおよびそのアクセス タイプを変更するには、authorize[-] コマンドで、access(*type-of-access*)、service、および stationName パラメータを使用します。

INSERVRNGE

ローカル ホストがクライアント ホストのグループに提供するサービスの範囲を指定します。

INETACL プロパティと同じような機能を実行します。

INSERVRANGE プロパティでアクセサおよびアクセス タイプを変更するには、authorize[-] コマンドの service(*serviceRange*) パラメータを使用します 。

MEMBERS

グループのメンバとなる、HOST クラスのオブジェクトのリストです。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドで mem+ または mem- パラメータを使用します。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

（情報のみ）レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

GPOLICY クラス

GPOLICY クラスの各レコードは、論理ポリシーを定義します。各レコードには、このポリシーに属するポリシー バージョン（POLICY オブジェクト）と割り当て先となるホストとホスト グループに関する情報が含まれます。

GDEPLOYMENT クラスのキーは、論理ポリシーの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GHNODEASSIGN

このポリシーの割り当て先となるホスト グループを定義します。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

HNODEASSIGN

このポリシーの割り当て先となるホストを定義します。

LATEST_FINALIZED_VERSION

ファイナライズされた最新のポリシー バージョン（POLICY オブジェクト）の名前を指定します。

LATEST_VERSION

このポリシーに関連付けられる最新のポリシー バージョン（POLICY オブジェクト）の名前を指定します。

MEMBERS

グループのメンバとなる、POLICY クラス（ポリシー バージョン）のオブジェクトのリストです。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドで mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

POLICY TYPE

グループ ポリシー タイプを表わす値です。有効な値は以下のとおりです。

- なし
- Login - ポリシーを UNAB ログイン ポリシーに指定します。
- Configuration - ポリシーを UNAB 環境設定ポリシーに指定します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティレベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

GROUP クラス

GROUP クラスの各レコードは、データベースのユーザのグループを定義します。

各 GROUP クラス レコードのキーは、グループの名前です。

注: プロファイル グループのプロパティは、プロファイル グループに関連付けられた各ユーザに適用されます。ただし、ユーザ（USER または XUSER）レコードで同じプロパティが指定されている場合、ユーザ レコードがプロファイル グループ レコードのプロパティより優先されます。

ほとんどのプロパティは、CA Access Control エンドポイント管理 か selang の chgrp コマンドを使用して変更できます。

注: ほとんどの場合、特に記載がなければ、ch[x]grp を使用してプロパティを変更するには、コマンド パラメータとしてプロパティ名を使用します。

CA Access Control エンドポイント管理 または selang の showgrp コマンドを使用すると、すべてのプロパティを表示できます。

APPLS

（情報）アクセサがアクセスを許可されるアプリケーションのリストを表示します。CA SSO で使用されます。

AUDIT_MODE

CA Access Control が監査ログに記録するアクティビティを定義します。以下のアクティビティの任意の組み合わせを指定できます。

- ログへの記録を行わない
- トレース ファイルに記録されたすべてのアクティビティ
- 失敗したログインの試み
- 成功したログイン
- CA Access Control によって保護されているリソースに対する失敗したアクセスの試み
- CA Access Control によって保護されているリソースに対する成功したアクセス
- 対話式ログイン

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの audit パラメータに相当します。GROUP または XGROUP に AUDIT_MODE を使用してグループのすべてのメンバに監査モードを設定することができます。ただし、ユーザの監査モードが USER レコード、XUSER レコード、またはプロファイル グループに定義されている場合は、AUDIT_MODE を使用してグループ メンバに監査モードを設定することはできません。

AUTHNMTHD

(情報のみ)グループ レコードに対して使用する 1 つ以上の認証方法 (method 1 ～ method 32、または none)を表示します。CA SSO で使用されます。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

EXPIRE_DATE

アクセサが無効になる日付を指定します。ユーザ レコードの EXPIRE_DATE プロパティの値は、グループ レコードの値より優先されます。

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの expire[-] パラメータに相当します。

FULLNAME

アクセサに関連付けられるフル ネームを定義します。フル ネームは、監査ログ メッセージでアクセサを識別するために使用されますが、権限付与に使用されることはありません。

FULLNAME は英数字の文字列です。グループの場合、最大長は 255 文字です。ユーザの場合、最大長は 47 文字です。

GAPPLS

グループがアクセスを許可されているアプリケーション グループのリストを定義します。CA SSO で使用されます。

GROUP_MEMBER

このグループに属するグループを指定します。

GROUP_TYPE

グループ権限属性を指定します。各属性は、ch[x]grp コマンドの同じ名前のパラメータに相当します。グループは以下の 1 つ以上の権限属性を持つことができます。

ADMIN

グループに属するユーザが管理機能を実行できるかどうかを指定します（UNIX 環境内での root に相当）。

AUDITOR

グループに属するユーザが、システムの監視、データベース情報の一覧表示、および既存レコードに対する監査モードの設定ができるかどうかを指定します。

OPERATOR

グループに属するユーザがデータベース内のすべてを一覧表示し、secons ユーティリティを使用できるかどうかを指定します。

PWMANAGER

グループに属するユーザが他のユーザのパスワード設定を変更し、serevu ユーティリティによって無効化されたユーザ アカウントを有効化できるかどうかを指定します。

SERVER

プロセスにおいて、グループに属するユーザに対する権限の確認と、SEOSROUTE_VerifyCreate API コールの発行が可能かどうかを指定します。

HOMEDIR

新しいグループ メンバに割り当てられるホーム ディレクトリのパスを指定します。

このプロパティを変更するには、chgrp コマンド、editgrp コマンド、または newgrp コマンドの homedir パラメータを使用します。

制限: 255 文字の英数字。

INACTIVE

ユーザのステータスが非アクティブに変更されるまでの、ユーザのアクティビティがない状態の経過日数を指定します。アカウント ステータスが非アクティブの場合、ユーザはログインできません。

USER クラスのレコードの INACTIVE プロパティの値は、GROUP クラスのレコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの INACT プロパティより優先されます。

注: CA Access Control はステータスを格納しません。動的に計算します。非アクティブ ユーザを特定するためには、INACTIVE 値をユーザの LAST_ACC_TIME 値と比較します。

INACTIVE はプロファイル機能の一部です。

MAXLOGINS

ユーザに許可される同時ログインの最大数を示します。値 0 は、同時ログイン数の制限がないことを示します。

ユーザ レコードの MAXLOGINS プロパティの値は、グループ レコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの MAXLOGINS プロパティの値より優先されます。

MAXLOGINS はプロファイル機能の一部です。

MEMBER_OF

このグループが属するグループを指定します。

OWNER

レコードを所有するユーザまたはグループを定義します。

PASSWDRULES

パスワード ルールを指定します。このプロパティには、CA Access Control でのパスワード保護の処理方法を決定する多くのフィールドが含まれています。ルールの一覧については、USER クラスの変更可能なプロパティである PROFILE を参照してください。

このプロパティを変更するには、setoptions コマンドの password パラメータおよび rules オプションまたは rules- オプションを使用します。

PASSWDRULES はプロファイル機能の一部です。

POLICYMODEL

sepass ユーティリティを使用してユーザ パスワードを変更したときに新しいパスワードを受け取る PMDB を指定します。このプロパティの値を入力した場合、parent_pmd または passwd_pmd 環境設定で定義されている Policy Model にパスワードは送信*されません* v

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの pmdb[-] パラメータに相当します。

POLICYMODEL はプロファイル機能の一部です。

PROFUSR

このプロファイル グループに関連付けられているユーザのリストを表示します。

PWD_AUTOGEN

グループ パスワードを自動的に生成するかどうかを指定します。デフォルトは no です。CA SSO で使用されます。

PWD_SYNC

すべてのグループ アプリケーションでグループ パスワードを自動的に同一にするかどうかを指定します。デフォルトは no です。CA SSO で使用されます。

PWPOLICY

グループに適用するパスワード ポリシーのレコード名を指定します。パスワード ポリシーは、新しいパスワードの妥当性をチェックし、パスワードの有効期限を定義する一連のルールです。デフォルトでは、妥当性チェックは行われません。CA SSO で使用されます。

RESUME_DATE

一時停止された USER アカウントが有効になる日付を指定します。

RESUME_DATE と SUSPEND_DATE は連携して動作します。

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの resume パラメータに相当します。

RESUME_DATE はプロファイル機能の一部です。

REVACL

アクセサのアクセス制御リストを表示します。

SHELL

(UNIX のみ)このグループのメンバである新しい UNIX ユーザに割り当てられるシェル プログラムです。

このプロパティを変更するには、chxgrp コマンドで shellprog パラメータを使用します。

SUBGROUP

このグループが親に指定されているグループのリストを表示します。

SUPGROUP

親グループ(上位グループ)の名前を定義します。

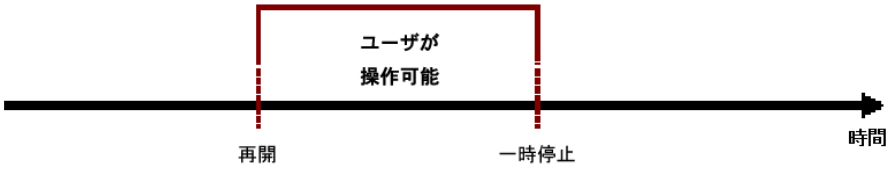
このプロパティを変更するには、ch[x]grp コマンドで parent[-] パラメータを使用します。

SUSPEND_DATE

ユーザ アカウントが一時停止されて無効になる日付を指定します。

レコードの一時停止日が再開日より前の日付である場合、ユーザは一時停止日より前および再開日より後に操作を実行できます。

ユーザの再開日が一時停止日より前の日付である場合は、再開日の*前*でもユーザ レコードは無効です。この場合、ユーザは再開日と一時停止日の間のみ操作を実行できます。



ユーザ レコードの SUSPEND_DATE プロパティの値は、グループ レコードの値より優先されます。

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの suspend[-] パラメータに相当します。

SUSPEND_WHO

一時停止日をアクティブにした管理者を表示します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

USERLIST

グループに属するユーザのリストを定義します。

このプロパティで設定するユーザ リストは、ネイティブ環境の USERS プロパティで設定するユーザ リストとは異なる場合があります。

このプロパティを変更するには、join[x][-] コマンドを使用します。

GSUDO クラス

GSUDO クラスの各レコードは、タスク委任、つまり DO（sesudo）によってユーザに実行が許可または禁止されるアクションのグループを定義します。各アクションの SUDO クラス レコードを作成した後に、作成したレコードを GSUDO レコードに追加する必要があります。

SUDO リソースのグループに対してアクセス ルールを定義するには、各リソースに対して同じアクセス ルールを指定するのではなく、GSUDO を使用します。次に、SUDO クラスのレコードを GSUDO クラスのレコードに明示的に関連付けてグループ化します。

GSUDO クラス レコードのキーは、グループの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

MEMBERS

グループのメンバとなる、SUDO クラスのオブジェクトのリストです。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドで mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

GTERMINAL クラス

GTERMINAL クラスの各レコードは、端末のグループを定義します。各端末の TERMINAL クラスのレコードを作成した後に、作成したレコードを GTERMINAL クラスのレコードに追加する必要があります。次に、TERMINAL クラスのレコードを GTERMINAL クラスのレコードに明示的に関連付けてグループ化します。

端末グループは、アクセス ルールを定義する場合に便利です。端末ごとに同じアクセス ルールを指定する代わりに、コマンド 1 つで端末グループにアクセス ルールを指定することができます。同様に、端末グループのルールをユーザ グループにコマンド 1 つで適用することもできます。

GTERMINAL クラス レコードのキーは、端末グループの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

MEMBERS

グループのメンバとなる、TERMINAL クラスのオブジェクトのリストです。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドで mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ(write など)と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

GWINSERVICE クラス

GWINSERVICE クラスの各レコードは、Windows サービスのグループを定義します。Windows サービスのグループに対してアクセス ルールを定義するには、GWINSERVICE クラスを使用します。

GWINSERVICE クラスのレコードのキーは、GWINSERVICE クラスのレコードの名前です。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ(ユーザおよびグループ)およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト(CALACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

HNODE クラス

HNODE クラスには、組織の CA Access Control ホストに関する情報が含まれます。クラスの各レコードは、組織内のノードを表します。

このクラスは、さまざまな PMDB やエンドポイントからアップロードされて DMS に格納される情報を管理するために使用されます。

HNODE クラスのレコードのキーは、エンドポイントの具体的なホスト名(myHost.ca.com など)または Policy Model ノードの PMDB 名(myPMD@myHost.ca.com)です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

ATTRIBUTES

ホストをホスト グループに自動的に追加するかどうか評価するために DMS 使用するカスタム基準を定義します。

注: DMS はまた、以下の HNODE プロパティを確認して、任意のホストがホスト グループに自動的に追加されるべきかどうか評価します。COMMENT、HNODE_INFO、HNODE_IP、HNODE_VERSION、NODE_TYPE

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

COMPLIANT

自動的に計算された HNODE の準拠ステータスを表示します。値は以下のとおりです。

- はい - CA Access Control がインストールされ、有効なポリシーがすべて正常にデプロイされています。
- いいえ - CA Access Control がインストールされているが、有効なポリシーが全くデプロイされていません。
- 偏差 - CA Access Control はインストールされているが、有効なポリシーの一部は正常にデプロイされていません。
- 不明 - CA Access Control がインストールされておらず、デプロイできる有効なポリシーがありません。

注: UNAB ポリシー(ログイン ポリシーおよび環境設定ポリシー)は準拠ステータスの値に割り当てられません。

COMPLIANT_UPDATE_TIME

(情報のみ)ステータスが最後に変更された日時を表示します。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

EFFECTIVE_POLICIES

このオブジェクトにデプロイする必要があるポリシー バージョンのリストを指定します。

表示名 : 有効なポリシー

GHNODES

このオブジェクトが属するホスト グループのリストを指定します。

表示名 : ノード グループ

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

HNODE_IP

ホストの IP アドレスです。

表示名 : IP

HNODE_KEEP_ALIVE

前回 HNODE がハートビートを分散ホストに送信した時刻を指定します。

表示名 : 最後のハートビート

LOGIN

ホストに対するデフォルト アクセス タイプを定義します。

表示名 : LOGIN

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NODE_INFO

（情報のみ）ノード OS の詳細を指定します。

NODE_TYPE

(情報のみ)ホスト上の CA Access Control インストールのタイプを定義します。有効な値は以下のとおりです。

- ACU - CA Access Control for UNIX
- ACW - CA Access Control for Windows
- UNAB - UNIX 認証ブローカ(UNAB)

注: HNODE レコードは、NODE_TYPE プロパティとして ACU および UNAB の両方の値を持つことができます。

NODE_VERSION

(情報のみ)ホストにインストールされる CA Access Control のバージョンを定義します。NODE_TYPE はバージョン番号に先行します。

例 ACU{12.50><00.647}

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

PARENTS

（情報のみ）。伝達ツリー内でそのノードの親である PMDB のリストです（parent_pmd 環境設定によっても定義される）。

POLICYASSIGN

このオブジェクトに割り当てられるポリシーのリストを定義します。

表示名 : 割り当てられたポリシー

POLICY_STATUS

POLICIES プロパティにリストされた各ポリシーのステータスです。このプロパティの値は、以下のフィールドを持つ構造体です。

oidPolicy

POLICY オブジェクトのオブジェクト ID です。POLICIES プロパティの値と同じです。

policy_status

以下のいずれかを表す整数です。

- デプロイされました - ポリシーはエンドポイントに正常にデプロイされました。
- デプロイされましたがエラーがあります - ポリシーはデプロイされましたが、エンドポイントでデプロイ スクリプトに含まれている 1 つ以上のルールの実行が失敗しました。
- デプロイ解除されました - ポリシーはエンドポイントから正常にデプロイ解除されました。

  注: ポリシーがデプロイ解除されると、ホストのステータスが表示されなくなります（ステータスなし）。
- デプロイ解除されましたがエラーがあります - ポリシーはデプロイ解除されましたが、エンドポイントでデプロイ解除スクリプトに含まれている 1 つ以上のルールの実行に失敗しました。
- デプロイに失敗しました - デプロイ スクリプトでエラーが発生したため、ポリシーのデプロイが失敗しました。

  注: ポリシー検証が有効な場合にのみ、このステータスが現れます。それ以外の場合、policyfetcher はポリシーにエラーが含まれていてもポリシーをデプロイします（「デプロイされましたがエラーがあります」ステータス）。
- 不明 - ポリシー ステータスは不明です。
- 展開する必須のポリシー用の Pending-Waiting を展開します。そうしないと、ポリシーは不確定か未決着の変数を含んでいます。
- デプロイ解除の一時停止中 - 依存しているポリシーがデプロイ解除されるのを待機しています。
- Out of Sync - ポリシーには、エンドポイントで変更された変数および変数の値が含まれています。

- 実行されていません - ポリシーの検証によって、ポリシーに 1 つまたは複数のエラーが見つかりました。
- キューに入っています - 使用されなくなりました（後方互換性維持のためにのみ残されています）
- 送信されました - 使用されなくなりました（後方互換性維持のためにのみ残されています）
- 送信が失敗しました - 使用されなくなりました（後方互換性維持のためにのみ残されています）
- シグネチャが失敗しました - 使用されなくなりました（後方互換性維持のためにのみ残されています）

deviation

このノードにポリシー偏差があるかどうかを表す値です。有効な値は以下のとおりです。

- はい
- いいえ
- Unset

dev_time

偏差ステータスの最終更新時刻です。

ptime

ポリシー ステータスの最終更新時刻です。

updator

ポリシーをデプロイまたは削除したユーザの名前です。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティレベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

SUBSCRIBER_STATUS

親ごとのノードのステータスです。このプロパティの値は、以下のフィールドを持つ構造体です。

oidSubs

HNODE オブジェクトのオブジェクト ID です。SUBSCRIBERS プロパティの値と同じです。

status

以下のいずれかのステータスを表す値です。

- 利用可能
- 利用不可
- 同期(同期中)
- 不明

stime

ステータスの最終更新時刻です。

SUBSCRIBERS

伝達ツリー内のそのノードのサブスクライバのリストです。このプロパティを更新すると、PARENTS プロパティが HNODE オブジェクト名の値で暗黙に更新されます。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UNAB_ID

（情報のみ）UNAB ホスト ID をレポート用に表示します。

UPDATE_TIME

（情報のみ）レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

HOLIDAY クラス

HOLIDAY クラスの各レコードは、ログイン時に特別な許可が必要となる 1 つ以上の期間を定義します。

各ユーザには、レコード内のすべての期間について同じアクセス権限が設定されます。これは、複数の休日期間を 1 つの HOLIDAY レコードに格納した場合、ある期間中にユーザにログインを許可し、別の期間中にはログインを禁止するという処理はできないことを意味します。たとえば、特定のユーザが元日にはログインでき、クリスマスにはログインできないようにする場合、この 2 つの休日は別々のレコードに定義する必要があります。

特定の年を指定しない場合、休日は 毎年適用されると見なされます。

newusr コマンド、chusr コマンド、または editusr コマンドで IGN_HOL 属性を指定することによって、個々のユーザに対する HOLIDAY クラス制限を無効にできます。

HOLIDAY クラス レコードのキーは、HOLIDAY レコードの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

HOL_DATE

ユーザがログインできない期間を指定します。

HOL_DATE プロパティには、以下のルールが適用されます。

- 特定の年を指定しない場合、その期間または休日は毎年適用されると見なされます。年は、99 または 1999 のように、2 桁または 4 桁で指定できます。
- 開始時刻を指定しない場合、その日の開始時刻(午前 0 時)が使用され、終了時刻を指定しない場合、その日の終了時刻(午前 0 時)が使用されます。
- 時間帯を指定せずに日付のみを指定した場合、その日 1 日が休日と見なされます。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドの dates パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ(write など)と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

（情報のみ）レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

HOST クラス

HOST クラスの各レコードは、IPv4 で接続されたときにホストがローカル コンピュータに対して持つアクセス権限を定義します。

注: IP 通信用の CA Access Control アクセス ルールは IPv4 にのみ適用されます。CA Access Control は IPv6 によるアクセスを管理しません。

CA Access Control は、HOST クラスに追加したホスト名のアドレスを解決する必要があります。つまり、これらの名前はオペレーティング システムの hosts ファイルに指定されているか、NIS または DNS に定義されている必要があります。

各 HOST レコードの INETACL プロパティは、ローカル ホストがそのホストに提供できるサービスを定義します。

CA Access Control では、ホスト名に別名を使用できます。ただし、別名を表すレコードが権限チェックに使用されることはありません。CA Access Control でホストとの接続を保護するには、ホストの正規名を把握している必要があります。

HOST クラス レコードのキーは、ホストの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する GHOST クラスまたは CONTAINER クラスのレコードのリストです。

HOST クラスのレコードのこのプロパティを変更するには、適切な CONTAINER クラスまたは GHOST クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

INETACL

ローカル ホストからクライアント ホストのグループに提供可能なサービスおよび各サービスのアクセス タイプを定義します。アクセス制御リストの各要素には、以下の情報が含まれます。

サービス参照

サービス(ポート番号または名前)への参照です。すべてのサービスを指定する場合は、サービス参照としてアスタリスク(*)を入力します。

また、CA Access Control では、/etc/rpc ファイル(UNIX の場合)または ¥etc¥rpc ファイル(Windows の場合)に指定された動的なポート名もサポートしています。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

INETACL プロパティでアクセサおよびそのアクセス タイプを変更するには、authorize[-] コマンドで、access(*type-of-access*)、service、および stationName パラメータを使用します。

INSERVRNGE

ローカル ホストがクライアント ホストのグループに提供するサービスの範囲を指定します。

INETACL プロパティと同じような機能を実行します。

INSERVRANGE プロパティでアクセサおよびアクセス タイプを変更するには、authorize[-] コマンドの service(*serviceRange*) パラメータを使用します 。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

HOSTNET クラス

HOSTNET クラスの各レコードは、特定のネットワーク上のホストによるグループを定義します。HOSTNET クラスのレコードはルールを定義します。このルールは、IPv4 で通信する場合に、グループの他のホストがローカル ホストに対して持つアクセス権を管理します。

注: IP 通信用の CA Access Control アクセス ルールは IPv4 にのみ適用されます。CA Access Control は IPv6 によるアクセスを管理しません。

INMASKMATCH プロパティは、HOSTNET クラスのレコードの対象になる他のホストを規定します。INETACL プロパティは、ローカル ホストが他のホストに提供できるサービスを定義します。

HOSTNET クラス レコードのキーは、HOSTNET レコードの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

INETACL

ローカル ホストからクライアント ホストのグループに提供可能なサービスおよび各サービスのアクセス タイプを定義します。アクセス制御リストの各要素には、以下の情報が含まれます。

サービス参照

サービス(ポート番号または名前)への参照です。すべてのサービスを指定する場合は、サービス参照としてアスタリスク(*)を入力します。

また、CA Access Control では、/etc/rpc ファイル(UNIX の場合)または ¥etc¥rpc ファイル(Windows の場合)に指定された動的なポート名もサポートしています。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

INETACL プロパティでアクセサおよびそのアクセス タイプを変更するには、authorize[-] コマンドで、access(*type-of-access*)、service、および stationName パラメータを使用します。

INSERVRNGE

ローカル ホストがクライアント ホストのグループに提供するサービスの範囲を指定します。

INETACL プロパティと同じような機能を実行します。

INSERVRANGE プロパティでアクセサおよびアクセス タイプを変更するには、authorize[-] コマンドの service(*serviceRange*) パラメータを使用します 。

INMASKMATCH

この HOSTNET レコードが適用されるホストのグループを定義します。このプロパティには mask 値と match 値があり、要求元ホストがグループに属しているかどうかを判断するために、要求元ホストの IP アドレスに適用されます。

INMASKMATCH プロパティは、IPv4 形式のアドレスのみサポートします。

注: このプロパティは、chres コマンドの mask パラメータと match パラメータに相当します。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

HOSTNP クラス

HOSTNP クラスの各レコードは、類似した名前を持つホストのグループを定義します。HOSTNP レコードはアクセス ルールを定義します。このルールは、IPv4 で通信する場合に、レコードの名前パターンに一致する他の端末(ホスト)のローカル ホストに対するアクセス権を管理します。各マスク(HOSTNP レコード)について、INETACL プロパティに、ローカル ホストがホスト グループに提供するサービスを制御するサービス ルールのリストが表示されます。

HOSTNP クラス レコードのキーは、HOSTNP レコードによって保護されるホストのホスト名のフィルタ処理に使用される名前パターンです。

注: IP 通信用の CA Access Control アクセス ルールは IPv4 にのみ適用されます。CA Access Control は IPv6 によるアクセスを管理しません。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

INETACL

ローカル ホストからクライアント ホストのグループに提供可能なサービスおよび各サービスのアクセス タイプを定義します。アクセス制御リストの各要素には、以下の情報が含まれます。

サービス参照

サービス(ポート番号または名前)への参照です。すべてのサービスを指定する場合は、サービス参照としてアスタリスク(*)を入力します。

また、CA Access Control では、/etc/rpc ファイル(UNIX の場合)または ¥etc¥rpc ファイル(Windows の場合)に指定された動的なポート名もサポートしています。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

INETACL プロパティでアクセサおよびそのアクセス タイプを変更するには、authorize[-] コマンドで、access(*type-of-access*)、service、および stationName パラメータを使用します。

INSERVRNGE

ローカル ホストがクライアント ホストのグループに提供するサービスの範囲を指定します。

INETACL プロパティと同じような機能を実行します。

INSERVRANGE プロパティでアクセサおよびアクセス タイプを変更するには、authorize[-] コマンドの service(*serviceRange*) パラメータを使用します 。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

KMODULE クラス

KMODULE クラスの各レコードは、オペレーティング システムのカーネル モジュールを定義します。

モジュールが KMODULE クラスに定義されていると、そのモジュールをロードまたはアンロードするためにオペレーティング システムを呼び出すたび、CA Access Control がそのモジュールに定義されている権限をチェックします。

KMODULE クラスのレコードのキーは、保護されているカーネル モジュールの名前です。

KMODULE クラスの各レコードには、以下のプロパティがあります。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。KMODULE レコードの有効なアクセス権限は load と unload です。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

FILEPATH

ファイルへの絶対パスのリストを定義します。各ファイルにはカーネル モジュールが含まれています。各ファイル パスはコロン(:)で区切ります。

同じモジュールに複数のバージョンがある場合は、複数のファイル パスを使用します。

ファイル パスの指定がない場合、CA Access Control はカーネル モジュール ロード時のファイル パス チェックを行いません。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ(write など)と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

SIGNATURE

FILEPATH プロパティに定義されているカーネル モジュール ファイルの一意の値を表示します。

CA Access Control は、起動されたとき、および KMODULE レコードが selang コマンドを使用して変更されたときに、カーネル モジュールのシグネチャを計算します。シグネチャは、seretrust -m コマンドを使用して明示的に設定することができます。

注: CA Access Control では、SIGNATURE プロパティを Linux システムでのみ使用します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

LOGINAPPL クラス

UNIX で該当

LOGINAPPL クラスの各レコードは、ログイン アプリケーションの定義、ログイン プログラムを使用してログインできるユーザの指定、およびログイン プログラムの使用方法の制御を行います。

LOGINAPPL クラスのレコードのキーは、アプリケーションの名前です。この名前は、ログイン アプリケーションを表す論理名です。この論理名は、LOGINPATH プロパティで、実行可能ファイルのフル パス名に関連付けられます。

CA Access Control では、包括的なログイン アプリケーションを制御および保護することもできます。つまり、特定のルールを汎用パターンに一致させるログイン アプリケーションのグループを保護できます。包括的なログイン アプリケーションを selang で定義するには、LOGINPATH パラメータを除く、通常のログイン制限を設定するときと同じコマンドを使用します。LOGINPATH パラメータには、[、]、*、? のうち 1 つ以上の文字を使用した正規表現で構成された包括的なパスを含める必要があります。

標準のログイン プログラムについては、LOGINAPPL クラスのレコードのプロパティ値があらかじめ設定されています。変更を行う前に、既存の設定を一覧表示して確認してください。

重要: LOGINAPPL は_default エントリを使用しません。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

LOGINFLAGS

デバイス番号の変更や猶予ログイン回数の差し引きなど、ログイン アプリケーションの特別な機能を制御します。有効な値は以下のとおりです。

- **execlogin** - ログイン トリガはプロセスが実行する最初の EXEC アクションであることを指定します。
- loginprefix - ログインしているユーザ名へのプレフィックスとして、CA Access Control が LOGINAPPL リソース名を追加することを指定します。たとえば、このプロパティを設定している場合、user1 という名前のユーザが CRON タスクをスケジュールしていると、CA Access Control は CRON タスク ログインを検出したとき、ユーザ名を USR_SBIN_CRON_user1 に設定します。

  注: CA Access Control はルートへのプレフィックスとして LOGINAPPL リソース名を追加しません。
- **nograce** - ユーザがこのアプリケーションを使用してログインした場合は、猶予ログイン回数を差し引かないことを示します。
- **nograceroot** - ユーザがこのアプリケーションを使用してログインした場合は、猶予ログイン回数を差し引かないことを示します。

- **nologin** - ユーザのみに対してログインが入力されるようにします。ログインは、親プログラムのログに記録されません。

  いくつかのプラットフォーム上にある rlogin のようなプログラムはログインをトリガし、ログイン シーケンス自体を終了します。この結果、実際のログインは root ユーザのログに記録されます。ログインの実行後、rlogin は、実際のログインを行うために別のプログラムに対して fork 要求を発行します。

  この問題は、rlogin や telnet などのログイン プログラムを使用して seaudit -a を実行した場合に明らかになります。uid のログイン レコードだけでなく、root ユーザによるログインのログイン レコードも記録されていることがわかります。

- **pamlogin** - ユーザがこのアプリケーションを使用してログインする際に、CA Access Control PAM ログイン インターセプトが使用されることを示します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの loginflags パラメータを使用します。

LOGINMETHOD

ログイン アプリケーションが CA Access Control の保護を目的とする擬似ログイン プログラムかどうかを指定します。有効な値は以下のとおりです。

- **normal** - このログイン アプリケーションで setuid と setgid の呼び出しを実行するように指定します。seosd は、指定したプログラムのルールをチェックします。
- **pseudo** - このログイン アプリケーションが別のプログラムを呼び出して、setuid および setgid 呼び出しを実行することを指定します。seosd は、他のプログラムでルールをチェックします。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの loginmethod パラメータを使用します。

重要: このプロパティはすでに設定されているため、変更しないことをお勧めします。

LOGINPATH

ログイン アプリケーションの完全パス(または包括的なパス)です。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの loginpath パラメータを使用します。

LOGINSEQUENCE

seosd が処理する一連の seteuid イベント、setuid イベント、setgid イベント、および setgroups イベントを定義します。これらのイベントは、ログイン プロセスを開始するデーモン（通常は root の inetd）からのユーザを、実際にログオンしたユーザに設定します。最大 8 つのシステム イベントを定義できます。

ログイン インターセプト シーケンスは、常に setgid イベントまたは setgroups イベントで始まります。これらのイベントを「*トリガ*」と言います。このシーケンスは、ユーザの ID を実際にログインしたユーザに変更する setuid イベントで終わります。

ログインを正しく行うために、プログラムは、setgroups イベントまたは setgid イベントで始まり setuid イベントまたは seteuid イベントで終わる、指定されたすべてのプロセスを順番に実行する必要があります。

プログラムの LoginSequence を適切に設定するのは困難な作業です。大部分のログイン プログラムは、デフォルトの SGRP, SUID 設定で適切に機能します。この設定では、プログラムはまず setgroups システム コールを発行し、次に setuid コマンドを実行して、ユーザの ID をターゲット ユーザに変更します。

ただし、SGRP, SUID 設定が機能しない場合は、以下のフラグを使用して、正しい順序を指定する必要があります。

- **SEID** - 最初の seteuid イベント
- **SUID** - 最初の setuid イベント
- **SGID** - 最初の setgid イベント
- **SGRP** - 最初の setgroup イベント
- **FEID** - 2 番目の seteuid イベント
- **FUID** - 2 番目の setuid イベント
- **FGID** - 2 番目の setgid イベント
- **FGRP** - 2 番目の setgroup イベント
- **N3EID** - 3 番目の seteuid イベント
- **N3UID** - 3 番目の setuid イベント
- **N3GID** - 3 番目の setgid イベント
- **N3GRP** - 3 番目の setgroup イベント

重要: 正確なログイン シーケンスを指定するために、フラグを使用する必要があります。ただし、フラグの順序は LOGINSEQUENCE パラメータ内で任意に指定できます。たとえば、「SGRP, SEID, FEID, N3EID」は「N3EID, FEID, SGRP, SEID」と同じものです。

注: ログイン プログラムが実行するシステム コールのシーケンスがわからない場合は、トレースを表示し、ユーザをターゲット UID に変更した setuid イベントを検索できます。次に、そのトレースで、最初の setgid イベントまたは setgroups イベントで始まる以前のイベントを調べます。

たとえば、1 つの setgroups イベントが存在し、3 番目の setuid 呼び出しのみがターゲット ユーザに設定されている場合は、LOGINSEQUENCE を SGRP,SUID,FUID,N3UID に設定する必要があります。これらのフラグは任意の順で指定できます。

```
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 プロセスは、トリガを示します。
- 最初の SUID コマンドは、root がトリガ ユーザではなく root に戻っているため、無視してください (これは、シーケンス内では SUID になります)。
- 2 番目の SUID コマンドも、root がトリガ ユーザではなく root に戻っているため、同様に無視する必要があります (これは、シーケンス内では FUID になります)。
- LOGIN イベントは、ログインを行う実際の SETUID イベントです (これは 3 番目のイベントなので、シーケンス内では N3UID フラグになります)。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの loginsequence パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ (write など) と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

MFTERMINAL クラス

MFTERMINAL クラスの各レコードは、CA Access Control の管理に使用されるメインフレーム コンピュータを定義します。MFTERMINAL クラスは、TERMINAL クラスと特性は同じですが、CA Access Control によってインターセプトされません。

MFTERMINAL クラスのレコードのキーは、メインフレーム コンピュータの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ(ユーザおよびグループ)およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト(CALACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

POLICY クラス

POLICY クラスの各レコードは、ポリシー バージョンのデプロイおよびデプロイ解除に必要な情報を定義します。これらのレコードには、ポリシーをデプロイおよびデプロイ解除するための selang コマンドのリストを含む RULESET オブジェクトへのリンクが含まれます。ポリシーがデプロイされる場合、selang の deploy コマンドが実行され、それにより、ポリシーを定義するすべてのコマンドが実行され、リンクされた RULESET オブジェクトに格納されます。ポリシーがデプロイ解除される場合、selang の deploy- コマンドが実行され、それにより、ポリシーのデプロイ解除を定義するすべてのコマンドが実行され、リンクされた RULESET オブジェクトに格納されます。

POLICY クラスのキーは、ポリシー名とそれに続くシャープ記号(#)および 2 桁のバージョン番号です。例: mypolicy#13。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ(ユーザおよびグループ)、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト(ACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

EFFECTS_ON

このポリシーが有効な(デプロイする必要がある)ホスト(HNODE オブジェクト)のリストを定義します。

FINALIZE

このポリシー バージョンがファイナライズされているかどうか(デプロイできるかどうか)を指定します。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストまたはこのポリシー バージョンが属する GPOLICY オブジェクトのリストを定義します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

HNODES

(情報のみ)。このポリシーをデプロイする必要がある CA Access Control ノードのリストです。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ(write など)と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

POLICY_BASE_NAME

このポリシー バージョンが属する GPOLICY オブジェクトの名前を指定します。

POLICY_VERSION

このポリシー バージョンのバージョン番号を指定します。

policy_type

ポリシー タイプを選択します。有効な値は以下のとおりです。

- なし
- Login - ポリシーを UNAB ログイン ポリシーに指定します。
- Configuration - ポリシーを UNAB 環境設定ポリシーに指定します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

RULESETS

ポリシーを定義する RULESET オブジェクトのリストです。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティレベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

SIGNATURE

ポリシーに関連付けられている RULESET オブジェクトのシグネチャに基づくハッシュ値です。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

VARIABLES

(情報のみ)ポリシーに含まれる変数のバージョンをすべて表示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

PROCESS クラス

PROCESS クラスの各レコードは、プログラム（実行可能ファイル）を定義します。それぞれのアドレス空間で実行するプログラムは、（kill コマンドによって）強制終了されないように保護する必要があります。特に、主要なユーティリティやデータベース サーバは、そのプロセスがサービス妨害（DoS）攻撃の主な標的になりやすいため、

注: PROCESS クラスにプログラムを定義する場合、FILE クラスにもプログラムを定義することをお勧めします。それにより、承認なく実行可能ファイルを変更（置き換えまたは破損）できなくなり、実行可能ファイルが保護されます。

CA Access Control では、通常の終了シグナル（SIGTERM）と、アプリケーションがマスクできない 2 つのシグナル（SIGKILL および SIGSTOP）の 3 つの終了シグナル（kill）からプロセスを保護することができます。

環境	シグナル	数値
Windows	KILL	Win32 API
UNIX	Terminate Process	9
UNIX および Windows	STOP	マシンによって異なります。
UNIX および Windows	TERM	15

SIGHUP や SIGUSR1 などのその他のシグナルは、ターゲットとなるプロセスに渡されます。そのプロセスでは、終了シグナルを無視するかどうか、あるいは何らかの方法でそのシグナルに対処するかどうかを決定します。

PROCESS クラス レコードのキーは、レコードが保護するプログラムの名前です。完全パスを指定します。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。「*情報のみ*」と記載されているプロパティは変更できません。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

PROGRAM クラス

PROGRAM クラスの各レコードは、trusted computing base の一部と見なされるプログラムを定義します。このクラスに属するプログラムは、変更されたかどうかが Watchdog 機能によって監視されるため、セキュリティ違反がないものとして信頼できます。trusted プログラムが変更されると、変更されたプログラムによって自動的に untrusted のマークが付けられ、実行できなくなります。オプションで、BLOCKRUN プロパティを使用して untrusted プログラムを許可または拒否することもできます。

各 PROGRAM レコードには、trusted プログラム ファイルに関する情報を定義するいくつかのプロパティが含まれています。

使用上の注意

- UNIX の場合、PROGRAM クラスには、setuid または setgid としてマークされていないプログラムが含まれる可能性もあります。
- CA Access Control では、どんなプログラムでも trusted プログラムとして定義できます。

  プログラムは、PROGRAM クラスに定義されていない限り、プログラム アクセス制御リスト（PACL）で使用できません（ただし、プログラムを PACL に追加すると、プログラムは自動的に PROGRAM クラスに追加されます）。
- ディレクトリは PROGRAM クラスに定義できません。

PROGRAM クラスのレコードのキーは、レコードが保護するプログラムのファイル名です。オブジェクト**名として、ファイルの完全パスを指定する必要があります。**

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。「*情報のみ*」と記載されているプロパティは変更できません。

ACCSTIME

（情報のみ）。レコードが最後にアクセスされた日時です。

ACCSWHO

（情報のみ）。レコードに最後にアクセスした管理者です。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

BLOCKRUN

プログラムが trusted であるかどうか、および untrusted プログラムの実行をブロックするかどうかを指定します。プログラムが setuid か通常のプログラムかどうかに関わらず、実行のブロックが行われます。

リソースのこのプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドの blockrun[-] パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

MD5

(情報のみ)。ファイルの RSA-MD5 シグネチャです。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

注: PROGRAM クラスのリソースに対し、PACL は UNIX では setuid/setgid プログラムにのみ、Windows では*ファイル*リソースがあるプログラムにのみ適用されます。CA Access Control はまずファイル リソース レコードをチェックし、アクセスが許可されている場合に、プログラム リソース レコードをチェックします。

PGMINFO

CA Access Control によって自動生成されるプログラム情報を定義します。

Watchdog 機能は、このプロパティに格納されている情報を自動的に検証します。情報が変更されている場合、プログラムは CA Access Control により untrusted として定義されます。

以下のフラグを選択すると、この検証プロセスから関連情報を*除外*できます。

crc

CRC（Cyclic Redundant Check）および MD5 シグネチャ。

ctime

（UNIX のみ）ファイル ステータスが最後に変更された時間。

device

UNIX の場合は、ファイルが存在する論理ディスク。 Windows の場合は、ファイルが存在するディスクのドライブ番号。

group

プログラム ファイルを所有するグループ。

inode

UNIX の場合は、プログラム ファイルのファイル システム アドレス。 Windows の場合は、意味はありません。

mode

プログラム ファイルに関連付けられているセキュリティ保護モード。

mtime

プログラムが最後に変更された時間。

owner

プログラム ファイルを所有するユーザ。

sha1

SHA1 シグネチャ。SHA は Secure Hash Algorithm の略で、プログラム ファイルや機密ファイルに適用できるデジタル署名方式です。

size

プログラム ファイルのサイズ。

このプロパティのフラグを変更するには、chres コマンド、editres コマンド、または newres コマンドの flags パラメータ、flags+ パラメータ、または flags- パラメータを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティレベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UNTRUST

リソースが信頼されているかどうかを定義します。UNTRUST プロパティが設定されている場合、アクセサはこのリソースを使用できません。UNTRUST プロパティが設定されていない場合、アクセサのアクセス権限の決定には、このリソースについてデータベースにリストされている他のプロパティが使用されます。trusted リソースに何らかの変更が加えられると、CA Access Control によって UNTRUST プロパティが自動的に設定されます。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの trust[-] パラメータを使用します。

UNTRUSTREASON

(情報のみ)。プログラムが untrusted になった理由です。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

PWPOLICY クラス

PWPOLICY クラスの各レコードは、パスワード ポリシーを定義します。パスワード ポリシーは、新しいパスワードの妥当性とパスワードの有効期間の両方に関する一連のルールです。

PWPOLICY クラスのキーは、パスワード ポリシーの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

APPLS

(情報のみ)。パスワード ポリシーにリンクされている CA SSO アプリケーションのリストです。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

OWNER

レコードを所有するユーザまたはグループを定義します。

PASSWDRULES

パスワード ルールを指定します。このプロパティには、CA Access Control でのパスワード保護の処理方法を決定する多くのフィールドが含まれています。ルールの一覧については、USER クラスの変更可能なプロパティである PROFILE を参照してください。

このプロパティを変更するには、setoptions コマンドの password パラメータおよび rules オプションまたは rules- オプションを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

REGKEY クラス

Windows で該当

REGKEY クラスの各レコードは、Windows レジストリのキーを定義します。

REGKEY レコードのキーは、レジストリ キーの完全パスです。

注: パスの指定にはワイルドカード文字を使用できます。

デフォルトでは、CA Access Control により CA Access Control レジストリ エントリが保護されます。このレジストリ エントリのルートは以下のとおりです。

```
HKEY_LOCAL_MACHINE¥SOFTWARE¥ComputerAssociates¥AccessControl
```

CA Access Control は以下のキーも保護します。

```
HKEY_LOCAL_MACHINE¥SYSTEM¥CurrentControlSet¥Services
```

REGKEY クラスと REGVAL クラスはプロパティが同じです。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

REGVAL クラス

Windows で該当

REGVAL クラスの各レコードは、Windows レジストリの値を定義します。

REGVAL レコードのキーは、レジストリ値の完全パスです。

注: パスの指定にはワイルドカード文字を使用できます。

REGVAL クラスでは NONE、READ、WRITE、および DELETE の各アクセス タイプを使用できます。

REGVAL クラスと REGKEY クラスはプロパティが同じです。プロパティを以下に示します (変更できないプロパティには、「*情報のみ*」と記載されています)。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ(write など)と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

（情報のみ）レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

RESOURCE_DESC クラス

RESOURCE_DESC クラスの各レコードは、CA SSO で新規ユーザ定義クラスのオブジェクトがアクセスを許可された、すべての名前を定義します。RESOURCE_DESC クラスに新しいオブジェクトを作成することはできません。既存のオブジェクトの変更のみ可能です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

CLASS_RIGHT

32 種類のオプションのアクセス権は、すべて変更可能です。最初の 4 種類のアクセス権のデフォルト値は、次のとおりです。

- CLASS_RIGHT1 - read
- CLASS_RIGHT2 - write
- CLASS_RIGHT3 - execute
- CLASS_RIGHT4 - rename

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

OWNER

レコードを所有するユーザまたはグループを定義します。

RESPONSE_LIST

このオブジェクトの名前が含まれる RESPONSE_TAB クラスのオブジェクトの名前です。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

RESPONSE_TAB クラス

このクラスの各レコードは、さまざまな権限付与の決定に応じた CA SSO の応答テーブルを定義します。

応答は、権限要求が許可または拒否された後にアプリケーションに返されるパーソナライズされた答えです。応答はキーと値のペアで構成され、特定のアプリケーションによって認識されます。応答を定義すると、ユーザの特定のニーズおよび権限付与の許可に従って、ポータル サイトをパーソナライズすることができます。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

CLASS_RIGHT

キーと値のペア（たとえば、button1=yes、picture2=no など）を含む文字列を一覧表示する 32 種類のオプションの応答プロパティです。各アクセス値に対して 1 つのプロパティを指定します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

OF_RESOURCE

同一のユーザ定義クラスを参照する RESOURCE_DESC クラスのオブジェクトの名前です。

OWNER

レコードを所有するユーザまたはグループを定義します。

UPDATE_TIME

（情報のみ）レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

RULESET クラス

RULESET クラスの各レコードは、ポリシーを定義するルールのセットを表します。

RULESET クラス レコードのキーは、レコードがリンクされているポリシーの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

EXPANDED COMMANDS

(情報のみ)デプロイされたポリシーでコマンドの変数の値を表示します。

EXPANDED UNDO COMMANDS

(情報のみ)デプロイされたポリシーで undo コマンドの変数の値を表示します。

FINALIZE

selang スクリプトがファイナライズされているかどうか(つまりそのポリシーバージョンをデプロイしてよいかどうか)を指定します。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ(write など)と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

RULESET_DOCMD_IDX

(情報のみ)。コマンド インデックスです。これは、RULESET_DOCMDS リストのコマンド数のカウンタです。

RULESET_DOCMDS

それ全体でポリシーを定義する selang コマンドのリストです。リストされるコマンドは、ポリシーをデプロイするために実行されるコマンドです。

重要: ポリシーのデプロイでは、ユーザ パスワードを設定するコマンドはサポートされていません。そのようなコマンドをデプロイ スクリプト ファイルに含めないでください。UNIX(ネイティブ)selang コマンドはサポートされていますが、偏差レポートには示されません。

RULESET_POLICIES

(情報のみ)。このルール セットを使用するポリシー(POLICY オブジェクト)のリストです。

RULESET_UNDOCMD_IDX

(情報のみ)。コマンド インデックスです。これは、RULESET_UNDOCMDS リストのコマンド数のカウンタです。

RULESET_UNDOCMDS

それ全体でポリシー デプロイ解除スクリプトを定義する selang コマンドのリストです。リストされるコマンドは、ポリシーをデプロイ解除するために実行されるコマンドです。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

SIGNATURE

RULESET_DOCMDS プロパティと RULESET_UNDOCMDS プロパティに基づくハッシュ値です。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

SECFILE クラス

SECFILE クラスの各レコードは、監視対象ファイルを定義します。SECFILE クラスのレコードによって、システムの重要なファイルを検証できます。ただし、このレコードは条件付きアクセス制御リストには表示できません。

頻繁に更新されない機密システム ファイルをこのクラスに追加し、権限のないユーザがこれらのファイルを変更していないことを確認します。監視対象として SECFILE クラスに指定するファイルの例を以下に示します。

UNIX の場合	Windows の場合
/.rhosts	¥system32¥drivers¥etc¥hosts
/etc/services	¥system32¥drivers¥etc¥services
/etc/protocols	¥system32¥drivers¥etc¥protocols
/etc/hosts	

UNIX の場合	Windows の場合
/etc/hosts.equiv	

Watchdog はこれらのファイルをスキャンし、これらのファイルに関する既知の情報が変更されていないことを確認します。

注: SECFILE クラスにディレクトリを定義することはできません。

SECFILE クラス レコードのキーは、SECFILE レコードが保護するファイルの名前です。完全パスを指定します。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

AIXACL

AIX システム ACL です。

AICEXTI

AIX システム拡張情報です。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

HPUXACL

HP-UX システム ACL です。

MD5

（情報のみ）。ファイルの RSA-MD5 シグネチャです。

OWNER

レコードを所有するユーザまたはグループを定義します。

PGMINFO

CA Access Control によって自動生成されるプログラム情報を定義します。

Watchdog 機能は、このプロパティに格納されている情報を自動的に検証します。情報が変更されている場合、プログラムは CA Access Control により untrusted として定義されます。

以下のフラグを選択すると、この検証プロセスから関連情報を*除外*できます。

crc

CRC（Cyclic Redundant Check）および MD5 シグネチャ。

ctime

（UNIX のみ）ファイル ステータスが最後に変更された時間。

device

UNIX の場合は、ファイルが存在する論理ディスク。 Windows の場合は、ファイルが存在するディスクのドライブ番号。

group

プログラム ファイルを所有するグループ。

inode

UNIX の場合は、プログラム ファイルのファイル システム アドレス。Windows の場合は、意味はありません。

mode

プログラム ファイルに関連付けられているセキュリティ保護モード。

mtime

プログラムが最後に変更された時間。

owner

プログラム ファイルを所有するユーザ。

sha1

SHA1 シグネチャ。SHA は Secure Hash Algorithm の略で、プログラム ファイルや機密ファイルに適用できるデジタル署名方式です。

size

プログラム ファイルのサイズ。

このプロパティのフラグを変更するには、chres コマンド、editres コマンド、または newres コマンドの flags パラメータ、flags+ パラメータ、または flags- パラメータを使用します。

UNTRUST

リソースが信頼されているかどうかを定義します。UNTRUST プロパティが設定されている場合、アクセサはこのリソースを使用できません。UNTRUST プロパティが設定されていない場合、アクセサのアクセス権限の決定には、このリソースについてデータベースにリストされている他のプロパティが使用されます。trusted リソースに何らかの変更が加えられると、CA Access Control によって UNTRUST プロパティが自動的に設定されます。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの trust[-] パラメータを使用します。

UNTRUSTREASON

(情報のみ)。プログラムが untrusted になった理由です。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

SECLABEL クラス

SECLABEL クラスの各レコードは、セキュリティ レベルをセキュリティ カテゴリに関連付けます。SECLABEL クラスがアクティブな場合、セキュリティ ラベルは、USER レコードの特定のセキュリティ レベルおよびセキュリティ カテゴリの割り当てより優先されます。セキュリティ ラベルの割り当ては、セキュリティ ラベルのセキュリティ レベルおよびセキュリティ カテゴリをユーザに明示的に割り当てることと同じです。

ユーザ レコードにセキュリティ ラベルが設定されている場合は、次の条件が満たされている場合にのみ、リソースに対するアクセス権限がユーザに与えられます。

- セキュリティ ラベルに指定されたユーザのセキュリティ レベルが、リソースのセキュリティ レベル以上の場合。
- リソース レコードに指定されたすべてのセキュリティ カテゴリが、ユーザのセキュリティ ラベルのセキュリティ カテゴリ リストにある場合。

注: Windows の場合、CA Access Control に定義されている各セキュリティ ラベルは、SECLABEL クラスのレコードを持っている必要があります。

SECLABEL クラス レコードのキーは、セキュリティ ラベルの名前です。この名前は、ユーザまたはリソースに割り当てる場合、セキュリティ ラベルの識別に使用されます。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

OWNER

レコードを所有するユーザまたはグループを定義します。

SECLEVEL

アクセサまたはリソースのセキュリティレベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

SEOS クラス

SEOS クラスは、CA Access Control 権限付与システムの動作を制御します。

クラスには、SEOS というレコードが 1 つだけ含まれます。このレコードは、一般的なセキュリティと権限のオプションを指定します。SEOS クラス プロパティのステータスを表示または変更するには、setoptions コマンドを使用します。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACCPACL

認証プロセスで UACC（defaccess）および PACL のリストをスキャンする順序を指定します。

ACCPACL がアクティブであり、ユーザのアクセス権が ACL で明示的に指定されている場合は、そのアクセサが許可されたアクセス権となります。アクセス権が ACL ではなく PACL で明示的に指定されている場合は、PACL アクセス権が許可されたアクセス権となります。ACL と PACL のいずれにも明示的なアクセス権が指定されていない場合は、defaccess のアクセス定義がチェックされます。

ACCPACL がアクティブでない場合は、最初に ACL の明示的なアクセス権がチェックされます。ACL にチェック対象リソースに関する明示的なアクセス権が定義されていない場合は、次に defaccess 定義がチェックされます。defaccess に明示的なアクセス権が定義されていない場合は、次に PACL アクセス権の定義がチェックされます。

CA Access Control のインストール時に、このプロパティの値は yes に設定されます。

このプロパティを変更するには、setoptions コマンドの accpacl パラメータまたは accpacl- パラメータを使用します。

ADMIN

ADMIN クラスをアクティブにするかどうかを指定します。通常、ADMIN クラスはアクティブで、セキュリティ管理タスクの実行許可を制御します。ADMIN クラスがアクティブでない場合は、すべてのユーザが CA Access Control 管理者と同様の作業を行うことができます。

APPL

APPL クラスをアクティブにするかどうかを指定します。

AUTHHOST

AUTHHOST クラスをアクティブにするかどうかを指定します。

CALENDAR

CALENDAR クラスをアクティブにするかどうかを指定します。

カテゴリ

CATEGORY クラスをアクティブにするかどうかを指定します。

CNG_ADMIN_PWD

PWMANAGER 属性を持つユーザが selang を使用して ADMIN ユーザのパスワードを変更できるかどうかを指定します。デフォルトは yes です。

このプロパティをアクティブまたは非アクティブにするには、setoptions コマンドの class+ パラメータまたは class- パラメータおよび *cng_adminpwd* オプションを使用します。

CNG_OWN_PWD

ユーザが selang を使用して自分のパスワードを変更できるかどうかを指定します。

このプロパティをアクティブまたは非アクティブにするには、setoptions コマンドの class+ パラメータまたは class- パラメータおよび *cng_ownpwd* オプションを使用します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CONNECT

CONNECT クラスをアクティブにするかどうかを指定します。CONNECT クラスがアクティブな場合、このクラスのレコードは外部への接続を保護します。

HOST クラスがアクティブな場合、CONNECT クラスは、アクティブであってもアクティブなクラスとして使用されません。

TCP クラスがアクティブな場合、CONNECT クラスはアクティブなクラスとして使用されません。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIMERES

(UNIX のみ)CA Access Control でリソースの日時制限をチェックするかどうかを指定します。

DMS

このデータベースによる通知の送信先 DMS サーバのリストです。

DOMAIN

(Windows のみ)DOMAIN クラスをアクティブにするかどうかを指定します。

ENDTIME

（情報のみ）。データベース ファイルが通常の方法で最後に閉じられた日時です。

FILE

FILE クラスをアクティブにするかどうかを指定します。FILE クラスがアクティブな場合、このクラスのレコードはファイルおよびディレクトリを保護します。

ACCGRR

累積グループ権限オプション（ACCGRR）では、CA Access Control がリソースの ACL をチェックする方法を制御します。ACCGRR が有効な場合、CA Access Control は、ACL で、ユーザが属するすべてのグループで許可されている権限をチェックします。ACCGRR が無効な場合、CA Access Control は、ACL で適用可能なエントリのいずれかに値 none が含まれているかどうかをチェックします。none が含まれている場合、アクセスは拒否されます。none が含まれていない場合、CA Access Control は、ACL 内の最初の適用可能なグループ エントリを除くすべてのグループ エントリを無視します。

このプロパティを有効または無効にするには、setoptions ACCGRR コマンドを使用します。

HOLIDAY

HOLIDAY クラスをアクティブにするかどうかを指定します。HOLIDAY クラスがアクティブな場合、定義された休日期間中にユーザがログインするには特別な許可が必要となります。

HOST

HOST クラスをアクティブにするかどうかを指定します。HOST クラスがアクティブな場合、CA Access Control は、リモート ホストから受信する TCP/IP サービス要求を保護します。

HOST クラスがアクティブな場合、TCP クラスおよび CONNECT クラスは、アクティブであってもアクティブなクラスとして使用されません。

HOST クラスは、デフォルトではアクティブです。

INACT

ユーザ ログインを一時停止するまでの非アクティブ状態の日数を指定します。非アクティブ状態の日とは、ユーザがログインしていない日を指します。

USER クラスのレコードの INACTIVE プロパティの値は、GROUP クラスのレコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの INACT プロパティより優先されます。

このプロパティを更新するには、setoptions コマンドの inactive パラメータまたは inactive- パラメータを使用します。

ISDMS

PMDB が DMS として機能している場合に true です。

LOGINAPPL

(UNIX のみ) LOGINAPPL クラスをアクティブにするかどうかを指定します。

MAXLOGINS

ユーザに許可される同時ログインの最大数(端末セッション数)です。この値を超えると、ユーザのアクセスは拒否されます。値 0 は最大数を設定しないことを意味します。ユーザは任意の数の端末セッションに同時にログインできます。CA Access Control では、ログイン、selang、GUI などの個々のタスクが 1 つの端末セッションと見なされます。そのため、ユーザがログインして selang を実行するか、またはデータベースを管理する場合は、0 を指定するか、1 より大きい値を指定する必要があります。

USER クラスのレコードの MAXLOGINS プロパティの値は、GROUP クラスのレコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの MAXLOGINS プロパティより優先されます。SEOS クラスのレコードの値は、アクセサ レコードに明示値の指定がない場合に使用されるデフォルト値です。

SEOS クラスのこのプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドの maxlogins パラメータを使用します。

MFTERMINAL

MFTERMINAL クラスをアクティブにするかどうかを指定します。

PASSWDRULES

パスワード ルールを指定します。このプロパティには、CA Access Control でのパスワード保護の処理方法を決定する多くのフィールドが含まれています。ルールの一覧については、USER クラスの変更可能なプロパティである PROFILE を参照してください。

このプロパティを変更するには、setoptions コマンドの password パラメータおよび rules オプションまたは rules- オプションを使用します。

PASSWORD

パスワード チェックをアクティブにするかどうかを指定します。

このプロパティをアクティブまたは非アクティブにするには、setoptions コマンドの class+ パラメータまたは class- パラメータおよび PASSWORD オプションを使用します。

PROCESS

PROCESS クラスをアクティブにするかどうかを指定します。PROCESS クラスがアクティブな場合、このクラスのレコードは、定義されているプロセスが（kill コマンドによって）強制終了されないように保護します。

ファイルは、FILE クラスにも定義されている必要があります。

PROGRAM

PROGRAM クラスをアクティブにするかどうかを指定します。PROGRAM クラスがアクティブな場合、このクラスのレコードは、trusted のマークを付加して定義されたプログラムを保護します。

PWPOLICY

PWPOLICY クラスをアクティブにするかどうかを指定します。

REGKEY

（Windows のみ）REGKEY クラスをアクティブにするかどうかを指定します。

REGVAL

（Windows のみ）REGVAL クラスをアクティブにするかどうかを指定します。

RESOURCE_DESC

RESOURCE_DESC クラスをアクティブにするかどうかを指定します。

RESPONSE_TAB

RESPONSE_TAB クラスをアクティブにするかどうかを指定します。

SECLABEL

SECLABEL クラスをアクティブにするかどうかを指定します。

SECLEVEL

SECLEVEL クラスをアクティブにするかどうかを指定します。

STARTTIME

（情報のみ）。データベース ファイルが最後に開かれた日時です。

SUDO

sesudo で使用する SUDO クラスをアクティブにするかどうかを指定します。

SYSTEM_AAUDIT_MODE

ユーザおよびエンタープライズ ユーザのデフォルト監査モード（システム全体の監査モード）を指定します。

デフォルト： Failure LoginSuccess LoginFailure

SURROGATE

SURROGATE クラスをアクティブにするかどうかを指定します。SURROGATE クラスがアクティブな場合、CA Access Control は代理要求を保護します。

TCP

TCP クラスをアクティブにするかどうかを指定します。TCP クラスがアクティブな場合、CA Access Control は、メール、ftp、http などの TCP サービスの送受信を保護します。

HOST クラスがアクティブな場合、TCP クラスは、アクティブであってもアクティブなクラスとして使用されません。

TCP クラスがアクティブな場合、CONNECT クラスはアクティブなクラスとして使用されません。

TERMINAL

TERMINAL クラスをアクティブにするかどうかを指定します。TERMINAL クラスがアクティブな場合、CA Access Control では、サインオン時に端末アクセス チェックを行い、X Window セッションを保護します。

USER_ATTR

USER_ATTR クラスをアクティブにするかどうかを指定します。

USER_DIR

USER_DIR クラスをアクティブにするかどうかを指定します。

UPDATE_TIME

（情報のみ）レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

SPECIALPGM クラス

SPECIALPGM クラスは、特定のプログラムに特別なセキュリティ権限を指定します。

SPECIALPGM クラスの各レコードには、次のいずれかの機能があります。

- Windows の場合は、backup、DCM、PBF、PBN、STOP、SURROGATE、REGISTRY、KILL の各プログラムを登録します。UNIX の場合は、xdm、backup、mail、DCM、PBF、PBN、stop、および surrogate の各プログラムを登録します。
- CA Access Control の特別な権限付与によって保護する必要があるアプリケーションを論理ユーザ ID に関連付けます。これにより、誰が実行しているかではなく*何*が実行されているかによって、*アクセス許可を効率的に設定できます*。

注: SPECIALPGM クラスにプログラムを定義する場合、FILE クラスにもプログラムを定義することをお勧めします。FILE リソースは実行可能ファイルが許可なく変更（置換または破損）されないようにすることで実行可能ファイルを保護し、PROGRAM リソースは CA Access Control が実行されていないときに変更されていた場合にプログラムが実行されないようにします。

注: 受信ネットワーク インターセプト イベントに対しレコードを SPECIALPGM クラスに定義できません。これは、受信ネットワーク インターセプト イベントがこのコンテキストにプロセス名を持ってないために起こります。インターセプト イベントに対する監査レコードの作成をバイパスするには、TCP クラスの対応するレコードの AUDIT プロパティを [NONE] に設定します。

PGMTYPE プロパティを使用して、システム サービス、デーモン、またはその他の特別なプログラムを登録します。

SEOSUID プロパティおよび NATIVEUID プロパティを使用して、論理ユーザをプログラムに割り当てます。

SPECIALPGM クラス レコードのキーは、特殊プログラムへのパス、または特殊プログラムの範囲またはパターンへのパスです。

注: SPECIALPGM クラスのテーブルに配置できるルールの最大数は 512 です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

NATIVEUID

プログラムまたはプロセスを起動するユーザを指定します。すべての CA Access Control ユーザを指定するには、* を使用します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの nativeuid パラメータを使用します。

注: CA Access Control の旧バージョンとの後方互換性を維持するために、NATIVEUID プロパティの代わりに UNIXUID プロパティを使用できます。

OWNER

レコードを所有するユーザまたはグループを定義します。

PGMTYPE

アクセスを許可する際に、CA Access Control が無視するアクセス チェックのタイプを決定します。

backup

READ アクセス、CHDIR アクセス、および UTIME アクセスを省略します。

注: バックアップを実行する方法は 2 つあります。バックアップ プログラムを実行したユーザが root 以外のユーザである場合は、このユーザを OPERATOR として定義する必要があります。バックアップ プログラムを実行したのが root である場合、バックアップ プログラムを SPECIALPGM クラスに pgmtype(backup) として登録するだけで済みます。

changeid

（UNIX のみ）su のような PAM を有効にしたサロゲート ID 変更ツールを無視します。

例: *er specialpgm /bin/su pgmtype(changeid)*

dcm

STOP イベントを除くすべてのイベントに対するセキュリティ チェックを省略します。

fullbypass

CA Access Control 認可およびデータベース チェックをすべてバイパスします。CA Access Control は、このプロパティがあるプロセスを無視します。また、プロセス イベントのレコードはいずれも CA Access Control 監査、トレースまたはデバッグ ログ内に表示されません。

kill

(Windows のみ)プロセスに対するプログラム終了を省略します。

たとえば、次のルールでは、プロセスがアクセス マスク KILL で CA Access Control サービス(プロセス)のハンドルを開こうとする場合、services.exe に省略されます。

```
nr specialpgm c:¥Windows¥system32¥services.exe pgmtype(kill)
```

Windows Server 2008 の場合、サービスの停止および開始を管理する services.exe プロセスは、アクセス タイプ KILL で CA Access Control サービス(プロセス)のハンドルを開いて、プロセス終了および開始を管理します。Windows Server 2008 でのインストール時に、CA Access Control は services.exe を見つける検出プロセスを実行し、services.exe に対する省略ルールを作成します。この省略がない場合、services.exe が CA Access Control サービスのハンドルを開こうとするとき、CA Access Control 監査イベント拒否になります。

mail

(UNIX のみ)setuid イベントおよび setgid イベントに対するデータベース チェックを無視します。mail によるこのデータベース チェックの省略により、アクセスを試みるメールをトレースできます。

なし

以前に設定された PGMTYPE を削除します。

pbf

ファイル処理イベントに対するデータベース チェックを省略します。

pbn

ネットワーク関連のイベントに対するデータベース チェックを省略します。

propagate

(UNIX のみ) PGMTYPE でプログラムから呼び出されるプログラムに独自のセキュリティ権限を伝達します。これを設定しない場合、SPECIALPGM 権限が親プログラムに影響するのみです。

注: セキュリティ権限の伝達は、PBF、PBN、DCM、FULLBYPASS、および SURROGATE 権限の場合にのみ有効です。

registry

(Windows のみ)Windows レジストリを操作するプログラムに対するデータベース チェックを省略します。

stop

STOP 機能に対するデータベース チェックを省略します。

surrogate

カーネル内の ID 変更イベントに対するデータベース チェックを省略します。surrogate を使用してデータベース チェックを省略した場合は、トレースを行うことができません。

xdm

(UNIX のみ)制限されたネットワーク範囲(6000 ～ 6010)に対してネットワーク イベント(TCP クラス、HOST クラス、および CONNECT クラスなど)を省略します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの pgmtype パラメータを使用します。

SEOSUID

この特別なプログラムを実行する権限がある、代理論理ユーザを定義します。この論理ユーザは、データベースの USER クラスのレコードに定義されている必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの seosuid パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

例: UNIX ファイルの保護

/DATABASE/data/* にあるファイルを保護するために、データベースの管理者は、ファイル サーバ デーモン firmdb_filemgr を使用します。このファイル サーバは、/opt/dbfirm/bin/firmdb_filemgr にあります。このデーモンは通常 root 権限で実行され、データはルート シェル ハックによってアクセスが可能な状態になっています。

以下の例では、これらのファイルの唯一のアクセサとして論理ユーザが定義されます。つまり、他のユーザはアクセスを制限されます。

1. 以下のコマンドを使用して、「機密」ファイルを CA Access Control に定義します。

   ```
   newres file /DATABASE/data/* defaccess(NONE)owner(nobody)
   ```

2. ファイルにアクセスする論理ユーザを定義します。

   ```
   newusr firmDB_mgr
   ```

3. 論理ユーザ firmDB_mgr のみにファイルへのアクセスを許可します。

   ```
   authorize file /DATABASE/data/* uid(firmDB_mgr) access(ALL)
   ```

4. 最後に、論理ユーザ firmDB_mgr が firmdb_filemgr を実行できるようにします。

   ```
   newres SPECIALPGM /opt/dbfirm/bin/firmdb_filemgr unixuid(root) ¥
   seosuid(firmDB_mgr)
   ```

この結果、デーモンがファイルにアクセスすると、CA Access Control は、root ユーザではなく論理ユーザをファイルのアクセサとして認識します。ハッカーが root ユーザとしてファイルにアクセスしようとしても、アクセスできません。

例: Windows ファイルの保護

C:¥DATABASE¥data にあるファイルを保護するために、データベースの管理者は、firmdb_filemgr.exe というファイル サーバ サービスを使用します。このファイル サーバは、C:¥Program Files¥dbfirm¥bin¥firmdb_filemgr.exe にあります。このサービスは通常システム アカウントで実行され、データはあらゆるシステム ハックが可能な状態になっています。

以下の例では、これらのファイルの唯一のアクセサとして論理ユーザが定義されます。つまり、他のユーザはアクセスを制限されます。

1. 以下のコマンドを使用して、「機密」ファイルを CA Access Control に定義します。

   ```
   newres file C:¥DATABASE¥data¥* defaccess(NONE)owner(nobody)
   ```

2. ファイルにアクセスする論理ユーザを定義します。

   ```
   newusr firmDB_mgr
   ```

3. 論理ユーザ firmDB_mgr のみにファイルへのアクセスを許可します。

   ```
   authorize file C:¥DATABASE¥data¥* uid(firmDB_mgr) access(ALL)
   ```

4. 最後に、論理ユーザ firmDB_mgr が firmdb_filemgr を実行できるようにします。

   ```
   newres SPECIALPGM ("C:¥Program Files¥dbfirm¥bin¥firmdb_filemgr.exe") ¥
   nativeuid(system) seosuid(firmDB_mgr)
   ```

この結果、サービスがファイルにアクセスすると、CA Access Control は、システム アカウントではなく論理ユーザをファイルのアクセサとして認識します。ハッカーがシステム アカウントでファイルにアクセスしようとしても、アクセスできません。

SUDO クラス

SUDO クラスの各レコードは、あるユーザが sesudo コマンドを使用して別のユーザの権限を借用できるようにするためのコマンドを識別します。

SUDO クラス レコードのキーは、SUDO レコードの名前です。この名前は、ユーザが SUDO レコードでコマンドを実行する際に、コマンド名の代わりに使用されます。

注: 対話型の Windows アプリケーション用の SUDO レコードを作成する場合、SUDO レコード用の対話型のフラグを設定する必要があります。対話型のフラグを設定しない場合、アプリケーションはバックグラウンドで実行されるため、ユーザは操作できません。詳細については、「*トラブルシューティング ガイド*」を参照してください。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ (ユーザおよびグループ)、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト (ACL) の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

sesudo が実行するコマンドです。

最大 255 文字の英数字から成る文字列です。この文字列には、コマンドが含まれます。さらに、許可されているパラメータおよび禁止されているパラメータも含まれます。

たとえば、以下のプロファイル定義では、COMMENT プロパティが正しく使用されています。

```
newres SUDO profile_name comment('command;;NAME')
```

注: このクラスでの COMMENT プロパティの使用法は、その他のクラスでの使用法とは異なります。SUDO レコードの定義の詳細については、お使いの OS に対応する「*エンドポイント管理ガイド*」を参照してください。このプロパティは、CA Access Control の旧バージョンで使用されていた DATA パラメータとして知られていたものです。

制限: 255 文字。

このプロパティを変更するには、chres コマンド、editres コマンド、および newres コマンドの comment[-] パラメータを使用します。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する GSUDO クラスまたは CONTAINER クラスのレコードのリストです。

SUDO クラスのレコードのこのプロパティを変更するには、適切な CONTAINER クラスまたは GSUDO クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

INTERACTIVE

(Windows のみ) このスイッチは、sesudo 使用して実行する予定のアプリケーションが、対話式 Windows アプリケーション(notepad.exe や cmd.exe)などであり、サービス アプリケーションではない場合にマークする必要があります。対話式アプリケーションの実行に *interactive* とマークされていない sesudo を使用すると、アプリケーションは対話する手段なしにバックグラウンドで実行されます。

注: 一部の Windows アプリケーションは、Windows の制約によりフォアグラウンドでは実行できません。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ(write など)と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

PASSWORDREQ

（UNIX のみ）sesudo コマンドが実行前に元のユーザのパスワードを要求するかどうかを指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの password パラメータを使用します。

POLICYMODEL

sepass ユーティリティを使用してユーザ パスワードを変更したときに新しいパスワードを受け取る PMDB を指定します。このプロパティの値を入力した場合、parent_pmd または passwd_pmd 環境設定で定義されている Policy Model にパスワードは送信*されません* v

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの pmdb[-] パラメータに相当します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティレベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

TARGUSR

(UNIX のみ)ターゲット UID を指定します。この UID は、コマンドを実行するためのアクセス許可の借用先ユーザを指定します。デフォルトは root です。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの targuid パラメータを使用します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

SURROGATE クラス

SURROGATE クラスの各レコードは、あるユーザの ID を他のユーザが変更しようとしたときに保護する制約を定義します。CA Access Control では、ID 変更要求を、権限を持つユーザのみがアクセスできる抽象オブジェクトとして処理します。

SURROGATE クラスのレコードは、代理保護が適用される各ユーザまたはグループを表します。特別な 2 つのレコード、USER._default および GROUP._default は、個別の SURROGATE レコードを持たないユーザおよびグループを表します。ユーザのデフォルトとグループのデフォルトを区別する必要がない場合は、代わりに SURROGATE クラスに _default レコードを使用できます。

注: Windows の多くのユーティリティおよびサービス([名前を指定して実行] など)では、それを実行している元のユーザとしてではなく、ユーザ「*NT AUTHORITY¥SYSTEM*」として識別されます。これらのユーティリティおよびサービスを使用するユーザが別のユーザとして実行できるようにするには、CA Access Control データベースにこの SYSTEM ユーザを作成し、ターゲット ユーザとして実行する権限を与える必要があります。

SURROGATE クラス レコードのキーは、SURROGATE レコードの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ(ユーザおよびグループ)、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト(ACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

TCP クラス

TCP クラスの各レコードは、メール、FTP、http などの TCP/IP サービスを定義します。TCP クラスが認証に使用されている場合、TCP リソースがアクセスを許可する場合のみ、ホストはローカル ホストからサービスを取得することができます。また、ローカル ホスト上のユーザまたはグループは、TCP リソースがアクセスを許可する場合のみ、TCP/IP サービスを使用してリモート ホストにアクセスすることができます。

TCP レコード内の ACL には、ホストのアクセス タイプ（HOST）、ホストのグループ（GHOST）、ネットワーク（HOSTNET）、およびホストのセット（HOSTNP）を指定することができます。

TCP レコード内の CACL には、ホストのアクセス タイプ（HOST）、ホストのグループ（GHOST）、ネットワーク（HOSTNET）、およびホストのセット（HOSTNP）を指定することができるほか、ユーザやグループのアクセス タイプも指定することができます。

ホスト名だけではなく、IPv4 アドレスにも基づいてルールを設定することができます。つまり、ドメイン名変更に対応することができます。

注: IP 通信用の CA Access Control アクセス ルールは IPv4 にのみ適用されます。CA Access Control は IPv6 によるアクセスを管理しません。

注: CONNECT クラスがアクセスの基準として使用されている場合、TCP クラスは事実上アクティブにできません。接続を保護するには、TCP クラスと CONECT クラスのどちらかを使用します。両方は使用しません。

TCP レコードのキーは、TCP/IP サービスの名前です。TCP クラスは、送信サービスおよび受信サービスの両方を制御します。

以下の定義では、TCP クラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

ローカル ホストによるサービスの提供先ホストと許可されるアクセス タイプを定義します。

アクセス制御リストの各要素には、以下の情報が含まれます。

ホスト参照

HOST レコード、GHOST レコード、HOSTNET レコード、または HOSTNP レコードを定義します。

許可されるアクセス

参照ホストに与えられる、リソースに対するアクセス権限です。有効なアクセス権限は以下のとおりです。

- **none** - どの操作の実行もホストに許可しません。
- **read** - ローカル ホストからの TCP サービスの取得をホストに許可します。

このプロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサがアクセスできるホストのリストを定義します。条件付きアクセス制御リスト（CACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

ホスト参照

HOST レコード、GHOST レコード、HOSTNET レコード、または HOSTNP レコードを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。有効なアクセス タイプは以下のとおりです。

- **write** - このサービスを使用したホストまたはホストのグループへのアクセスをアクセサに許可します。
- **none** - このサービスを使用したホストまたはホストのグループへのアクセスをアクセサに許可しません。

このプロパティを変更するには、authorize コマンドまたは authorize- コマンドを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

TERMINAL クラス

TERMINAL クラスの各レコードは、ローカル ホストの端末、ネットワーク上にある別のホストの端末、またはログイン セッションを実行できる X 端末を定義します。また、端末 名や IP アドレス パターンと(ワイルドカードを使用して)一致する端末も定義できます。端末のアクセス許可はユーザ ログイン手続きの過程でチェックされ、使用権限のない端末からユーザがログインすることはできません。

TERMINAL クラスは、管理アクセスも制御します。ADMIN ユーザは、適切なアクセス権限がある端末からのみ CA Access Control を管理できます。

新しい TERMINAL クラスのレコードを定義すると、CA Access Control は、ユーザが指定した名前を完全修飾名に変換しようとします。成功すると、完全修飾名がデータベースに格納されます。 失敗すると、指定された名前が格納されます。これ以降、このレコードを参照するコマンド(chres、showres、mres、authorize など)を発行する際に、データベースに表示されている名前を使用する必要があります。

TERMINAL レコードのキーは、端末の名前です。CA Access Control では、端末はこの名前によって識別されます。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ(ユーザおよびグループ)、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト(ACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ(ユーザおよびグループ)およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト(CALACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する GTERMINAL クラスまたは CONTAINER クラスのレコードのリストです。

TERMINAL クラスのレコードのこのプロパティを変更するには、適切な CONTAINER クラスまたは GTERMINAL クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム(または名前パターンに一致するプログラム)とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト(PACL)の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

UACC クラス

UACC クラスの各レコードは、リソース クラスに許可するデフォルト アクセスを定義します。UACC クラスのレコードは、CA Access Control で保護されないクラスのリソースに許可するアクセス レベルも決定します。

UACC は一部のクラスを除いたほとんどのクラスに適用できます。各クラスでの UACC クラスの使用法を次の表に示します。

UACC の使用法	クラス
標準	ADMIN、APPL、AUTHHOST、CALENDAR、CONNECT、CONTAINER、DOMAIN、GAPPL、GAUTHHOST、GHOST、GSUDO、GTERMINAL、HOLIDAY、HOST、HOSTNET、HOSTNP、MFTERMINAL、POLICY、PROCESS、PROGRAM、REGKEY、REGVAL、RULESET、SUDO、SURROGATE、TCP、TERMINAL、USER_DIR、ユーザ定義クラス
非標準	FILE、GFILE
なし	AGENT、AGENT_TYPE、CATEGORY、GROUP、PWPOLICY、RESOURCE_DESC、RESPONSE_TAB、SECFILE、SECLABEL、SEOS、SPECIALPGM、USER、USER_ATTR

特別な _restricted グループに属していないユーザの場合、UACC クラスの FILE のレコードでは、seos.ini ファイル、seosd.trace ファイル、seos.audit ファイル、および seos.error ファイルなど、CA Access Control の一部であるファイルのみが保護されます。これらのファイルは CA Access Control に明示的に定義されていませんが、CA Access Control によって自動的に保護されます。

UACC クラスのレコードのキーは、UACC プロパティを定義するクラスの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACL

リソースへのアクセスを許可されているアクセサ（ユーザおよびグループ）、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト（ACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

ALLOWACCS

このクラスに対して許可されるすべてのアクセス権のリストです。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

OWNER

レコードを所有するユーザまたはグループを定義します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

（情報のみ）レコードが最後に変更された日時を示します。

UPDATE_WHO

（情報のみ）更新を実行した管理者を示します。

USER クラス

USER クラスの各レコードは、CA Access Control データベース内でユーザを定義します。

USER クラスのレコードのキーは、ユーザがシステムへのログイン時に入力したユーザ名です。

USER プロパティのほとんどは、CA Access Control エンドポイント管理 か selang の chusr コマンドを使用して変更できます。chusr で変更できないプロパティには「*情報のみ*」と記載されます。

注: ほとんどの場合、および特に記載がなければ、chusr を使用してプロパティを変更するには、コマンド パラメータとしてプロパティ名を使用します。

CA Access Control エンドポイント管理 または selang の showusr コマンドを使用すると、すべてのプロパティを表示できます。

APPLIST

CA SSO で使用されます。

APPLIST_TIME

CA SSO で使用されます。

APPLS

(情報)アクセサがアクセスを許可されるアプリケーションのリストを表示します。CA SSO で使用されます。

AUDIT_MODE

CA Access Control が監査ログに記録するアクティビティを定義します。以下のアクティビティの任意の組み合わせを指定できます。

- ログへの記録を行わない
- トレース ファイルに記録されたすべてのアクティビティ
- 失敗したログインの試み
- 成功したログイン
- CA Access Control によって保護されているリソースに対する失敗したアクセスの試み
- CA Access Control によって保護されているリソースに対する成功したアクセス
- 対話式ログイン

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの audit パラメータに相当します。

AUTHNMTHD

(情報のみ)グループ レコードに対して使用する 1 つ以上の認証方法(method 1 ～ method 32、または none)を表示します。CA SSO で使用されます。

BADPASSWD

CA SSO で使用されます。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

COUNTRY

ユーザの国記述子を指定する文字列です。この文字列は、X.500 ネーミング スキーマの一部です。この情報が権限付与に使用されることはありません。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

EMAIL

最大 128 文字のユーザの電子メール アドレスを指定します。

EXPIRE_DATE

アクセサが無効になる日付を指定します。ユーザ レコードの EXPIRE_DATE プロパティの値は、グループ レコードの値より優先されます。

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの expire[-] パラメータに相当します。

FULLNAME

アクセサに関連付けられるフル ネームを定義します。フル ネームは、監査ログ メッセージでアクセサを識別するために使用されますが、権限付与に使用されることはありません。

FULLNAME は英数字の文字列です。グループの場合、最大長は 255 文字です。ユーザの場合、最大長は 47 文字です。

GAPPLS

(情報)ユーザがアクセスを許可されているアプリケーション グループのリストを示します。CA SSO で使用されます。

GRACELOGIN

パスワードの有効期限が切れた後の猶予ログイン回数を指定します。指定された猶予ログイン回数を超えるとユーザはシステムへのアクセスを拒否されるため、システム管理者に連絡して新しいパスワードを取得する必要があります。

猶予ログイン回数には、0 ～ 255 の値を指定する必要があります。この値が 0 の場合、ユーザはログインできません。

USER クラスのレコードの GRACELOGIN プロパティの値は、GROUP クラスのレコードの NGRACE の値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの PASSWDRULES プロパティより優先されます。

注: このプロパティは、ch[x]usr コマンドの grace パラメータに相当します。

GROUPS

(情報)ユーザが属するユーザ グループのリストを表示します。このプロパティには、グループ管理者権限(GROUP-ADMIN)など、ユーザが属するグループ単位でユーザに割り当てられるグループ権限も含まれます。

このプロパティで設定するグループ リストは、ネイティブ環境の GROUPS プロパティで設定するユーザ リストとは異なる場合があります。

注: このプロパティは、ch[x]usr コマンドでは変更されません。変更するには、join[-] コマンドまたは joinx[-] コマンドを使用します。

HOMEDIR

(UNIX のみ)ユーザのホーム ディレクトリを定義します。CA SSO で使用されます。

INACTIVE

ユーザのステータスが非アクティブに変更されるまでの、ユーザのアクティビティがない状態の経過日数を指定します。アカウント ステータスが非アクティブの場合、ユーザはログインできません。

USER クラスのレコードの INACTIVE プロパティの値は、GROUP クラスのレコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの INACT プロパティより優先されます。

注: CA Access Control はステータスを格納しません。動的に計算します。非アクティブ ユーザを特定するためには、INACTIVE 値をユーザの LAST_ACC_TIME 値と比較します。

LAST_ACC_TERM

最後にログインが実行された端末を示します。

LAST_ACC_TIME

前回のログインの日時を示します。

LOCALAPPS

CA SSO で使用されます。

LOCATION

ユーザの所在地を定義します。この情報が権限付与に使用されることはありません。

LOGININFO

レコードで、ユーザが特定のアプリケーションおよび監査データにログインするために必要な情報を定義します。LOGININFO には、ユーザがアクセスを許可されているアプリケーションごとに、個別にリストが保存されています。CA SSO で使用されます。

LOGSHIFT

シフト時間枠外にログインを許可するかどうかを示します。CA Access Control は、このイベントに関する監査レコードを監査ログに書き込みます。

MAXLOGINS

ユーザに許可される同時ログインの最大数を示します。値 0 は、同時ログイン数の制限がないことを示します。

ユーザ レコードの MAXLOGINS プロパティの値は、グループ レコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの MAXLOGINS プロパティの値より優先されます。

MIN_TIME

ユーザのパスワード変更間隔として許可する最短期間(日数)を定義します。

USER クラスのレコードの MIN_TIME プロパティの値は、GROUP クラスのレコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの PASSWDRULES プロパティより優先されます。

注: このプロパティは、ch[x]usr コマンドの min_life パラメータに相当します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OBJ_TYPE

ユーザ権限属性を指定します。各属性は、ch[x]usr コマンドの同じ名前のパラメータに相当します。ユーザは以下の 1 つ以上の権限属性を持つことができます。

ADMIN

UNIX 環境の root ユーザと同様に、ほとんどの管理機能の実行をユーザに許可するかどうかを指定します。

AUDITOR

システムの監視、データベース情報の一覧表示、および既存のレコードに対する監査モードの設定をユーザに許可するかどうかを指定します。

IGN_HOL

休日レコードによって定義された期間中にユーザがログインできるかどうかを指定します。

LOGICAL

ユーザが CA Access Control 内部でのみ使用され、実際のユーザのログインには使用できないことを示します。

たとえば、リソースの所有者であってもリソースへのアクセスを妨げるために、リソースの所有者として使用するユーザ nobody は、デフォルトの論理ユーザです。これは、ユーザがこのアカウントを使用してログインすることができないことを意味します。

OPERATOR

データベース内のすべての情報の一覧表示と secons ユーティリティの使用をユーザに許可するかどうかを指定します。

PWMANAGER

他のユーザのパスワード設定の変更、および serevu ユーティリティによって無効化されたユーザ アカウントの有効化を、ユーザに許可するかどうかを指定します。

SERVER

ユーザへの権限のクエリ、および SEOSROUTE_VerifyCreate API コールの発行を、プロセスに許可するかどうかを指定します。

OIDCRDDATA

CA SSO で使用されます。

OLD_PASSWD

ユーザの以前のパスワードの暗号化されたリストが格納されます。ユーザは、このリストから新しいパスワードを選択することはできません。OLD_PASSWD に保存されるパスワードの最大数は、setoptions コマンドで指定します。

ORG_UNIT

ユーザが所属する組織単位に関する情報を格納する文字列です。この文字列は、X.500 ネーミング スキーマの一部です。この情報が権限付与に使用されることはありません。

ORGANIZATION

ユーザが所属する組織を指定します。この文字列は、X.500 ネーミング スキーマの一部です。この情報が CA Access Control による権限付与に使用されることはありません。

OWNER

レコードを所有するユーザまたはグループを定義します。

PASSWD_A_C_W

このレコードのユーザ パスワードを最後に変更した ADMIN ユーザを示します。

PASSWD_INT

ユーザのパスワード変更間隔として許可する最長期間（日数）を指定します。

USER クラスのレコードの PASSWD_INT プロパティの値は、GROUP クラスのレコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの PASSWDRULES プロパティより優先されます。

注: このプロパティは、ch[x]usr コマンドの interval パラメータに相当します。

PASSWD_L_A_C

管理者が最後にパスワードを更新した日時を示します。

PASSWD_L_C

ユーザが最後にパスワードを更新した日時を示します。

PGMINFO

CA Access Control によって自動生成されるプログラム情報を定義します。

Watchdog 機能は、このプロパティに格納されている情報を自動的に検証します。情報が変更されている場合、プログラムは CA Access Control により untrusted として定義されます。

以下のフラグを選択すると、この検証プロセスから関連情報を*除外*できます。

crc

CRC（Cyclic Redundant Check）および MD5 シグネチャ。

ctime

（UNIX のみ）ファイル ステータスが最後に変更された時間。

device

UNIX の場合は、ファイルが存在する論理ディスク。 Windows の場合は、ファイルが存在するディスクのドライブ番号。

group

プログラム ファイルを所有するグループ。

inode

UNIX の場合は、プログラム ファイルのファイル システム アドレス。 Windows の場合は、意味はありません。

mode

プログラム ファイルに関連付けられているセキュリティ保護モード。

mtime

プログラムが最後に変更された時間。

owner

プログラム ファイルを所有するユーザ。

sha1

SHA1 シグネチャ。SHA は Secure Hash Algorithm の略で、プログラム ファイルや機密ファイルに適用できるデジタル署名方式です。

size

プログラム ファイルのサイズ。

このプロパティのフラグを変更するには、chres コマンド、editres コマンド、または newres コマンドの flags パラメータ、flags+ パラメータ、または flags- パラメータを使用します。

PHONE

ユーザの電話番号を入力します。この情報が権限付与に使用されることはありません。

POLICYMODEL

sepass ユーティリティを使用してユーザ パスワードを変更したときに新しいパスワードを受け取る PMDB を指定します。このプロパティの値を入力した場合、parent_pmd または passwd_pmd 環境設定で定義されている Policy Model にパスワードは送信*されません* v

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの pmdb[-] パラメータに相当します。

PROFILE

ユーザのプロファイルのパスを定義します。この文字列には、ローカルの絶対パスまたは UNC パスを含めることができます。

PUPM_FLAGS

端末の統合属性を指定します。PUPM で CA Access Control エンドポイント上の特権アカウントを統合するとき、端末統合を使用します。特権アカウントは以下の端末統合属性のうちどちらか、または両方を持つことができます。

use_original_identity

CA Access Control は許可に関する決定を行う際、特権アカウント名ではなくアカウントをチェック アウトしたユーザ名を使用することを指定します。セッションの監査レコードは、実ユーザ名フィールドの元のユーザおよび有効なユーザ名フィールドの特権アカウントを一覧表示します。

required_checkout

ユーザがアカウントを使用してエンドポイントにログインする前に、アカウントが PUPM でチェックアウトされる必要があることを指定します。

PWD_AUTOGEN

ユーザ パスワードを自動的に生成するかどうかを表示します。CA SSO で使用されます。

デフォルトは no です。

PWD_SYNC

すべてのユーザ アプリケーションでユーザ パスワードを自動的に同一にするかどうかを表示します。CA SSO で使用されます。

デフォルトは no です。

RESUME_DATE

一時停止された USER アカウントが有効になる日付を指定します。

RESUME_DATE と SUSPEND_DATE は連携して動作します。

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの resume パラメータに相当します。

REVACL

アクセサのアクセス制御リストを表示します。

REVOKE_COUNT

CA SSO で使用されます。

SCRIPT_VARS

CA SSO で使用されます。アプリケーションごとに保存されるアプリケーション スクリプトの変数値を含む変数リストを定義します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

SESSION_GROUP

ユーザの SSO セッション グループを定義します。SESSION_GROUP プロパティは、最大 16 文字の文字列です。

Windows では、適切な名前がドロップダウン リストに存在しない場合、管理者がセッション グループの新しい名前を入力できます。

CA SSO で使用されます。

SHIFT

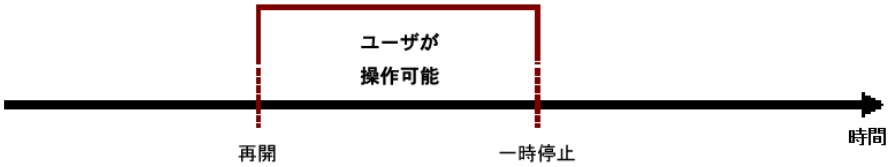
CA SSO で使用されます。

SUSPEND_DATE

ユーザ アカウントが一時停止されて無効になる日付を指定します。

レコードの一時停止日が再開日より前の日付である場合、ユーザは一時停止日より前および再開日より後に操作を実行できます。

ユーザの再開日が一時停止日より前の日付である場合は、再開日の*前*でもユーザ レコードは無効です。この場合、ユーザは再開日と一時停止日の間のみ操作を実行できます。



ユーザ レコードの SUSPEND_DATE プロパティの値は、グループ レコードの値より優先されます。

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの suspend[-] パラメータに相当します。

SUSPEND_WHO

一時停止日をアクティブにした管理者を表示します。

注: このプロパティは、ch[x]usr コマンドの suspend[-] パラメータに相当します。

UALIAS

1 つ以上の認証ホストに定義されている特定ユーザの別名を表示します。CA SSO で使用されます。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

USER_ATTR クラス

USER_ATTR クラスの各レコードは、CA SSO ユーザ ディレクトリの有効なユーザ属性を定義します。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ATTR_PREDEFS

特定の属性に対して許可される値のリストです。

ATTRNAME

(情報のみ)。属性の名前です。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DBFIELD

userdir データベースに登録されているフィールドの名前です。異なるデータベースには異なる属性を指定できるため、属性フィールドは同期させる必要があります。

FIELDID

(情報のみ)。DB フィールドの ID です。

OWNER

レコードを所有するユーザまたはグループを定義します。

PARAMETER_TYPE

ユーザ属性が文字列か数値かを示します。

PRIORITY

ユーザ属性の優先度です。権限ルールを PARAM_RULE オブジェクト(APPL、URL など)に設定すると、そのルールはユーザ属性が参照している優先度に定義されます。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

USER_DIR_PROP

(情報のみ)。ユーザのディレクトリの名前です。

USERATTR_FLAGS

属性に関する情報が含まれます。フラグには、以下の値を指定できます。

- **aznchk** - この属性を権限付与に使用するかどうかを指定します。
- **predef**(事前定義済み)、**freetext**(自由形式のテキスト)、または **userdir**(ユーザ ディレクトリ) - これら 3 つの値で、ユーザ属性のソースを指定します。
- **user** または **group** - これらの値を使用して、属性(アクセサ)がユーザであるかグループであるかを指定します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

USER_DIR クラス

USER_DIR クラスの各レコードは、CA SSO ユーザ ディレクトリを定義します。

USER_DIR クラスのレコードのキーは、ディレクトリの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ADMIN_NAME

ディレクトリ管理者のログイン名です。

ADMIN_PWD

ディレクトリ管理者のパスワードです。パスワードは、テキスト形式の平文で格納されます。selang では表示されませんが、seadmapi 関数を使用して表示できます。

AZNACL

権限 ACL を定義します。これは、リソースの説明に基づいてリソースへのアクセスを許可する ACL です。説明は、オブジェクトではなく認証エンジンに送信されます。一般に、AZNACL が使用される場合、オブジェクトはデータベースにありません。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CONTOBJ_CLS

コンテナ オブジェクトに継承されるクラスの名前です(LDAP で新規ログイン情報コンテナを作成するために必要)。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DIR_TYPE

ディレクトリのタイプです。有効な値は、ETRUST_AC、LDAP、ODBC、NT_Domain、または none です。

GRPOBJ_CLS

グループ オブジェクトに継承されるクラスの名前です(LDAP で新規グループを作成するために必要)。

LICONTOBJ_CLS

ログイン情報コンテナ オブジェクトに継承されるクラスの名前です(LDAP で新規ログイン情報コンテナを作成するために必要)。

LIOBJ_CLS

ログイン情報オブジェクトに継承されるクラスの名前です(LDAP で新規ログイン情報を作成するために必要)。

MAX_RET_ITEMS

取得される項目の最大数です。デフォルトは、ディレクトリ タイプによって異なります。

OWNER

レコードを所有するユーザまたはグループを定義します。

PATH

すべてのクエリを開始するための LDAP ツリー内の相対識別名です。

PORT_NUM

ディレクトリへのアクセスに使用するホスト コンピュータでのポート番号です。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求(デフォルト)

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

TIMEOUT_CON

タイムアウト エラー メッセージを発行するまでに、システムがディレクトリへの接続を待機する時間(秒単位)です。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

USERATTR_LIST

この USER_DIR オブジェクトで USER_DIR パラメータの値として作成された USER_ATTR クラスのオブジェクトのリストです。

USERDIR_HOST

ディレクトリのホスト コンピュータの名前です。このプロパティは、クラスのレコードに定義されている必要があります。

USROBJ_CLS

ユーザ オブジェクトに継承されるクラスの名前です（LDAP で新規ユーザを作成するために必要）。

VERSION

ディレクトリのバージョン番号です。

WEBSERVICE クラス

WEBSERVICE クラスは使用されなくなりました。CA Access Control は使用しません。

WINSERVICE クラス

WINSERVICE クラスの各レコードは、Windows サービスを定義します。Windows サービスのアクセス ルールを定義するには、WINSERVICE クラスを使用します。

WINSERVICE クラスのレコードのキーは、サービスの Windows 名です。

注: ほとんどの場合、および特に記載がなければ、selang の chres コマンドを使用してプロパティを変更するには、コマンド パラメータとしてプロパティ名を使用します。

CA Access Control エンドポイント管理 または selang の showres WINSERVICE コマンドを使用すると、すべてのプロパティを表示できます。

ACL

リソースへのアクセスを許可されているアクセサ (ユーザおよびグループ)、およびアクセサのアクセス タイプのリストを定義します。

アクセス制御リスト (ACL) の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

ACL プロパティを変更するには、authorize コマンドまたは authorize- コマンドの access パラメータを使用します。

CALACL

リソースへのアクセスが許可されるアクセサ（ユーザおよびグループ）およびそれぞれの Unicenter NSM カレンダ ステータスに基づくアクセス タイプのリストを定義します。

カレンダ アクセス制御リスト（CALACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Calendar

Unicenter TNG のカレンダへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

カレンダが有効な場合のみアクセスが許可されます。その他の場合はすべてのアクセスが拒否されます。

ACL プロパティに定義されているアクセスに基づいて、リソースへのアクセスをユーザまたはグループに許可するには、authorize コマンドで calendar パラメータを使用します。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

（情報のみ）レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

GROUPS

リソース レコードが属する CONTAINER クラスのレコードのリストを定義します。

クラス レコードのこのプロパティを変更するには、適切な CONTAINER クラスのレコードの MEMBERS プロパティを変更する必要があります。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの mem+ または mem- パラメータを使用します。

NACL

リソースの *NACL* プロパティは、リソースへのアクセス権限が拒否されるアクセサを、拒否されるアクセス タイプ（write など）と共に定義するアクセス制御リストです。ACL、CALACL、PACL も参照してください。NACL の各エントリには、以下の情報が含まれます。

アクセサ

アクセサを定義します。

アクセス

アクセサに対して拒否されるアクセス タイプを定義します。

このプロパティを変更するには、authorize deniedaccess コマンドまたは authorize- deniedaccess- コマンドを使用します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OWNER

レコードを所有するユーザまたはグループを定義します。

PACL

アクセス要求が特定のプログラム（または名前パターンに一致するプログラム）とそのアクセス タイプを使用して行われる場合に、リソースへのアクセスが許可されるアクセサのリストを定義します。プログラム アクセス制御リスト（PACL）の各要素には、以下の情報が含まれます。

アクセサ

アクセサを定義します。

Program

指定またはワイルドカード パターン一致によって、PROGRAM クラスのレコードへの参照を定義します。

アクセス

アクセサに与えられる、リソースに対するアクセス権限を定義します。

注: PACL のリソースの指定にはワイルドカード文字を使用できます。

プログラム、アクセサ、およびそのアクセス タイプを PACL に追加するには、selang の authorize コマンドで via(*pgm*) パラメータを使用します。アクセサを PACL から削除するには、authorize- コマンドを使用します。

RAUDIT

CA Access Control の監査ログに記録されるアクセス イベントのタイプを定義します。RAUDIT という名前は *R*esource *AUDIT* の短縮形です。有効な値は以下のとおりです。

all

すべてのアクセス要求

success

許可されたアクセス要求

failure

拒否されたアクセス要求（デフォルト）

none

アクセス要求を記録しない

CA Access Control では、リソースへのアクセス試行が発生するたびにイベントが記録されます。ただし、アクセス ルールがそのリソースに直接適用されたか、またはそのリソースをメンバとするグループまたはクラスに適用されたか、については記録されません。

監査モードを変更するには、chres コマンドおよび chfile コマンドの audit パラメータを使用します。

SECLABEL

ユーザまたはリソースのセキュリティ ラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティ レベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

UACC

リソースに対するデフォルトのアクセス権限を定義します。CA Access Control に定義されていないアクセサ、またはリソースの ACL に登録されていないアクセサに与えるアクセス権限を指定します。

このプロパティを変更するには、chres コマンド、editres コマンド、または newres コマンドの defaccess パラメータを使用します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

WARNING

警告モードを有効にするかどうかを指定します。リソースの警告モードを有効にすると、そのリソースに対するアクセス要求はすべて許可され、アクセス要求がアクセス ルールに違反した場合、監査ログにレコードが記録されます。

XGROUP クラス

XGROUP クラスの各レコードは、データベースのユーザのグループを定義します。

各 XGROUP クラス レコードのキーは、グループの名前です。

注: プロファイル グループのプロパティは、プロファイル グループに関連付けられた各ユーザに適用されます。ただし、ユーザ（USER または XUSER）レコードで同じプロパティが指定されている場合、ユーザ レコードがプロファイル グループ レコードのプロパティより優先されます。

ほとんどのプロパティは、CA Access Control エンドポイント管理 か selang の chxgrp コマンドを使用して変更できます。

注: ほとんどの場合、特に記載がなければ、chxgrp を使用してプロパティを変更するには、コマンド パラメータとしてプロパティ名を使用します。

CA Access Control エンドポイント管理 または selang の showxgrp コマンドを使用すると、すべてのプロパティを表示できます。

APPLS

（情報）アクセサがアクセスを許可されるアプリケーションのリストを表示します。CA SSO で使用されます。

AUDIT_MODE

CA Access Control が監査ログに記録するアクティビティを定義します。以下のアクティビティの任意の組み合わせを指定できます。

- ログへの記録を行わない
- トレース ファイルに記録されたすべてのアクティビティ
- 失敗したログインの試み
- 成功したログイン
- CA Access Control によって保護されているリソースに対する失敗したアクセスの試み
- CA Access Control によって保護されているリソースに対する成功したアクセス
- 対話式ログイン

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの audit パラメータに相当します。GROUP または XGROUP に AUDIT_MODE を使用してグループのすべてのメンバに監査モードを設定することができます。ただし、ユーザの監査モードが USER レコード、XUSER レコード、またはプロファイル グループに定義されている場合は、AUDIT_MODE を使用してグループ メンバに監査モードを設定することはできません。

AUTHNMTHD

(情報のみ)グループ レコードに対して使用する 1 つ以上の認証方法 (method 1 ～ method 32、または none)を表示します。CA SSO で使用されます。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

EXPIRE_DATE

アクセサが無効になる日付を指定します。ユーザ レコードの EXPIRE_DATE プロパティの値は、グループ レコードの値より優先されます。

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの expire[-] パラメータに相当します。

FULLNAME

アクセサに関連付けられるフル ネームを定義します。フル ネームは、監査ログ メッセージでアクセサを識別するために使用されますが、権限付与に使用されることはありません。

FULLNAME は英数字の文字列です。グループの場合、最大長は 255 文字です。ユーザの場合、最大長は 47 文字です。

GAPPLS

グループがアクセスを許可されているアプリケーション グループのリストを定義します。CA SSO で使用されます。

GROUP_MEMBER

このグループに属するグループを指定します。

GROUP_TYPE

グループ権限属性を指定します。各属性は、ch[x]grp コマンドの同じ名前のパラメータに相当します。グループは以下の 1 つ以上の権限属性を持つことができます。

ADMIN

グループに属するユーザが管理機能を実行できるかどうかを指定します（UNIX 環境内での root に相当）。

AUDITOR

グループに属するユーザが、システムの監視、データベース情報の一覧表示、および既存レコードに対する監査モードの設定ができるかどうかを指定します。

OPERATOR

グループに属するユーザがデータベース内のすべてを一覧表示し、secons ユーティリティを使用できるかどうかを指定します。

PWMANAGER

グループに属するユーザが他のユーザのパスワード設定を変更し、serevu ユーティリティによって無効化されたユーザ アカウントを有効化できるかどうかを指定します。

SERVER

プロセスにおいて、グループに属するユーザに対する権限の確認と、SEOSROUTE_VerifyCreate API コールの発行が可能かどうかを指定します。

MEMBER_OF

このグループが属するグループを指定します。

OWNER

レコードを所有するユーザまたはグループを定義します。

PROFUSR

このプロファイル グループに関連付けられているユーザのリストを表示します。

PWD_AUTOGEN

グループ パスワードを自動的に生成するかどうかを指定します。デフォルトは no です。CA SSO で使用されます。

PWD_SYNC

すべてのグループ アプリケーションでグループ パスワードを自動的に同一にするかどうかを指定します。デフォルトは no です。CA SSO で使用されます。

PWPOLICY

グループに適用するパスワード ポリシーのレコード名を指定します。パスワード ポリシーは、新しいパスワードの妥当性をチェックし、パスワードの有効期限を定義する一連のルールです。デフォルトでは、妥当性チェックは行われません。CA SSO で使用されます。

REVACL

アクセサのアクセス制御リストを表示します。

SHELL

(UNIX のみ)このグループのメンバである新しい UNIX ユーザに割り当てられるシェル プログラムです。

このプロパティを変更するには、chxgrp コマンドで shellprog パラメータを使用します。

SUBGROUP

このグループが親に指定されているグループのリストを表示します。

SUPGROUP

親グループ(上位グループ)の名前を定義します。

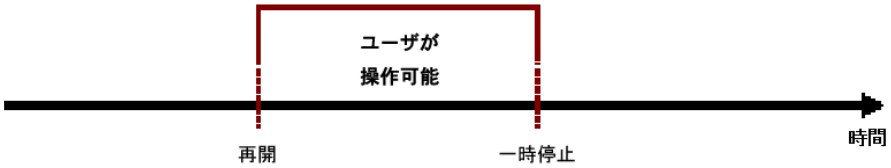
このプロパティを変更するには、ch[x]grp コマンドで parent[-] パラメータを使用します。

SUSPEND_DATE

ユーザ アカウントが一時停止されて無効になる日付を指定します。

レコードの一時停止日が再開日より前の日付である場合、ユーザは一時停止日より前および再開日より後に操作を実行できます。

ユーザの再開日が一時停止日より前の日付である場合は、再開日の*前*でもユーザ レコードは無効です。この場合、ユーザは再開日と一時停止日の間のみ操作を実行できます。



ユーザ レコードの SUSPEND_DATE プロパティの値は、グループ レコードの値より優先されます。

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの suspend[-] パラメータに相当します。

SUSPEND_WHO

一時停止日をアクティブにした管理者を表示します。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ)更新を実行した管理者を示します。

USERLIST

グループに属するユーザを示します。

このプロパティで設定するユーザ リストは、ネイティブ環境の USERS プロパティで設定するユーザ リストとは異なる場合があります。

XUSER クラス

XUSER クラスの各レコードは、データベース内のエンタープライズ ユーザを定義します。

XUSER クラスのレコードのキーは、ユーザがシステムへのログイン時に入力したユーザ名です。

ほとんどのプロパティは、CA Access Control エンドポイント管理 か selang の chxusr コマンドを使用して変更できます。

注: ほとんどの場合、特に記載がなければ、chxusr を使用してプロパティを変更するには、コマンド パラメータとしてプロパティ名を使用します。

CA Access Control エンドポイント管理 または selang コマンドの showxusr を使用すると、すべてのプロパティを表示できます。

APPLIST

CA SSO で使用されます。

APPLIST_TIME

CA SSO で使用されます。

APPLS

(情報)アクセサがアクセスを許可されるアプリケーションのリストを表示します。CA SSO で使用されます。

AUDIT_MODE

CA Access Control が監査ログに記録するアクティビティを定義します。以下のアクティビティの任意の組み合わせを指定できます。

- ログへの記録を行わない
- トレース ファイルに記録されたすべてのアクティビティ
- 失敗したログインの試み
- 成功したログイン
- CA Access Control によって保護されているリソースに対する失敗したアクセスの試み
- CA Access Control によって保護されているリソースに対する成功したアクセス
- 対話式ログイン

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの audit パラメータに相当します。

AUTHNMTHD

(情報のみ)グループ レコードに対して使用する 1 つ以上の認証方法(method 1 ～ method 32、または none)を表示します。CA SSO で使用されます。

BADPASSWD

CA SSO で使用されます。

CALENDAR

CA Access Control のユーザ、グループ、およびリソース制限事項の Unicenter TNG カレンダ オブジェクトを表します。CA Access Control により、指定された時間間隔で Unicenter TNG のアクティブなカレンダが取得されます。

カテゴリ

ユーザまたはリソースに割り当てる 1 つ以上のセキュリティ カテゴリを定義します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

COUNTRY

ユーザの国記述子を指定する文字列です。この文字列は、X.500 ネーミング スキーマの一部です。この情報が権限付与に使用されることはありません。

CREATE_TIME

(情報のみ)レコードが作成された日時が表示されます。

DAYTIME

アクセサがリソースにアクセスできる日時を規定する、曜日と時間帯の制限を定義します。

このプロパティを変更するには、chres コマンド、ch[x]usr コマンド、または ch[x]grp コマンドで restrictions パラメータを使用します。

日時の制約の単位は 1 分です。

EMAIL

最大 128 文字のユーザの電子メール アドレスを指定します。

FULLNAME

アクセサに関連付けられるフル ネームを定義します。フル ネームは、監査ログ メッセージでアクセサを識別するために使用されますが、権限付与に使用されることはありません。

FULLNAME は英数字の文字列です。グループの場合、最大長は 255 文字です。ユーザの場合、最大長は 47 文字です。

GAPPLS

(情報)ユーザがアクセスを許可されているアプリケーション グループのリストを示します。CA SSO で使用されます。

GRACELOGIN

パスワードの有効期限が切れた後の猶予ログイン回数を指定します。指定された猶予ログイン回数を超えるとユーザはシステムへのアクセスを拒否されるため、システム管理者に連絡して新しいパスワードを取得する必要があります。

猶予ログイン回数には、0 ～ 255 の値を指定する必要があります。この値が 0 の場合、ユーザはログインできません。

USER クラスのレコードの GRACELOGIN プロパティの値は、GROUP クラスのレコードの NGRACE の値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの PASSWDRULES プロパティより優先されます。

注: このプロパティは、ch[x]usr コマンドの grace パラメータに相当します。

GROUPS

(情報)ユーザが属するユーザ グループのリストを表示します。このプロパティには、グループ管理者権限(GROUP-ADMIN)など、ユーザが属するグループ単位でユーザに割り当てられるグループ権限も含まれます。

このプロパティで設定するグループ リストは、ネイティブ環境の GROUPS プロパティで設定するユーザ リストとは異なる場合があります。

注: このプロパティは、ch[x]usr コマンドでは変更されません。変更するには、join[-] コマンドまたは joinx[-] コマンドを使用します。

INACTIVE

ユーザのステータスが非アクティブに変更されるまでの、ユーザのアクティビティがない状態の経過日数を指定します。アカウント ステータスが非アクティブの場合、ユーザはログインできません。

USER クラスのレコードの INACTIVE プロパティの値は、GROUP クラスのレコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの INACT プロパティより優先されます。

注: CA Access Control はステータスを格納しません。動的に計算します。非アクティブ ユーザを特定するためには、INACTIVE 値をユーザの LAST_ACC_TIME 値と比較します。

LAST_ACC_TERM

最後にログインが実行された端末を示します。

LAST_ACC_TIME

前回のログインの日時を示します。

LOCALAPPS

CA SSO で使用されます。

LOCATION

ユーザの所在地を定義します。この情報が権限付与に使用されることはありません。

LOGININFO

レコードで、ユーザが特定のアプリケーションおよび監査データにログインするために必要な情報を定義します。LOGININFO には、ユーザがアクセスを許可されているアプリケーションごとに、個別にリストが保存されています。CA SSO で使用されます。

LOGSHIFT

シフト時間枠外にログインを許可するかどうかを示します。CA Access Control は、このイベントに関する監査レコードを監査ログに書き込みます。

MAXLOGINS

ユーザに許可される同時ログインの最大数を示します。値 0 は、同時ログイン数の制限がないことを示します。

ユーザ レコードの MAXLOGINS プロパティの値は、グループ レコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの MAXLOGINS プロパティの値より優先されます。

MIN_TIME

ユーザのパスワード変更間隔として許可する最短期間(日数)を定義します。

USER クラスのレコードの MIN_TIME プロパティの値は、GROUP クラスのレコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの PASSWDRULES プロパティより優先されます。

注: このプロパティは、ch[x]usr コマンドの min_life パラメータに相当します。

NOTIFY

リソースまたはユーザによって監査イベントが生成されたときに通知されるユーザを定義します。CA Access Control では、指定したユーザ宛に監査レコードを電子メールで送信できます。

制限: 30 文字。

OBJ_TYPE

ユーザ権限属性を指定します。各属性は、ch[x]usr コマンドの同じ名前のパラメータに相当します。ユーザは以下の 1 つ以上の権限属性を持つことができます。

ADMIN

UNIX 環境の root ユーザと同様に、ほとんどの管理機能の実行をユーザに許可するかどうかを指定します。

AUDITOR

システムの監視、データベース情報の一覧表示、および既存のレコードに対する監査モードの設定をユーザに許可するかどうかを指定します。

IGN_HOL

休日レコードによって定義された期間中にユーザがログインできるかどうかを指定します。

LOGICAL

ユーザが CA Access Control 内部でのみ使用され、実際のユーザのログインには使用できないことを示します。

たとえば、リソースの所有者であってもリソースへのアクセスを妨げるために、リソースの所有者として使用するユーザ nobody は、デフォルトの論理ユーザです。これは、ユーザがこのアカウントを使用してログインすることができないことを意味します。

OPERATOR

データベース内のすべての情報の一覧表示と secons ユーティリティの使用をユーザに許可するかどうかを指定します。

PWMANAGER

他のユーザのパスワード設定の変更、および serevu ユーティリティによって無効化されたユーザ アカウントの有効化を、ユーザに許可するかどうかを指定します。

SERVER

ユーザへの権限のクエリ、および SEOSROUTE_VerifyCreate API コールの発行を、プロセスに許可するかどうかを指定します。

OIDCRDDATA

CA SSO で使用されます。

OLD_PASSWD

ユーザの以前のパスワードの暗号化されたリストが格納されます。ユーザは、このリストから新しいパスワードを選択することはできません。OLD_PASSWD に保存されるパスワードの最大数は、setoptions コマンドで指定します。

ORG_UNIT

ユーザが所属する組織単位に関する情報を格納する文字列です。この文字列は、X.500 ネーミング スキーマの一部です。この情報が権限付与に使用されることはありません。

ORGANIZATION

ユーザが所属する組織を指定します。この文字列は、X.500 ネーミング スキーマの一部です。この情報が CA Access Control による権限付与に使用されることはありません。

PASSWD_A_C_W

このレコードのユーザ パスワードを最後に変更した ADMIN ユーザを示します。

PASSWD_INT

ユーザのパスワード変更間隔として許可する最長期間（日数）を指定します。

USER クラスのレコードの PASSWD_INT プロパティの値は、GROUP クラスのレコードの値より優先されます。このどちらのプロパティ値も、SEOS クラスのレコードの PASSWDRULES プロパティより優先されます。

注: このプロパティは、ch[x]usr コマンドの interval パラメータに相当します。

PASSWD_L_A_C

管理者が最後にパスワードを更新した日時を示します。

PASSWD_L_C

ユーザが最後にパスワードを更新した日時を示します。

PHONE

ユーザの電話番号を入力します。この情報が権限付与に使用されることはありません。

PUPM_FLAGS

端末の統合属性を指定します。PUPM で CA Access Control エンドポイント上の特権アカウントを統合するとき、端末統合を使用します。特権アカウントは以下の端末統合属性のうちどちらか、または両方を持つことができます。

use_original_identity

CA Access Control は許可に関する決定を行う際、特権アカウント名ではなくアカウントをチェック アウトしたユーザ名を使用することを指定します。セッションの監査レコードは、実ユーザ名フィールドの元のユーザおよび有効なユーザ名フィールドの特権アカウントを一覧表示します。

required_checkout

ユーザがアカウントを使用してエンドポイントにログインする前に、アカウントが PUPM でチェックアウトされる必要があることを指定します。

PWD_AUTOGEN

ユーザ パスワードを自動的に生成するかどうかを表示します。CA SSO で使用されます。

デフォルトは no です。

PWD_SYNC

すべてのユーザ アプリケーションでユーザ パスワードを自動的に同一にするかどうかを表示します。CA SSO で使用されます。

デフォルトは no です。

RESUME_DATE

一時停止された USER アカウントが有効になる日付を指定します。

RESUME_DATE と SUSPEND_DATE は連携して動作します。

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの resume パラメータに相当します。

REVACL

アクセサのアクセス制御リストを表示します。

REVOKE_COUNT

CA SSO で使用されます。

SCRIPT_VARS

CA SSO で使用されます。アプリケーションごとに保存されるアプリケーション スクリプトの変数値を含む変数リストを定義します。

SECLABEL

ユーザまたはリソースのセキュリティラベルを定義します。

注: SECLABEL プロパティは、chres コマンドと ch[x]usr コマンドの label[-] パラメータに相当します。

SECLEVEL

アクセサまたはリソースのセキュリティレベルを定義します。

注: このプロパティは、ch[x]usr コマンドと chres コマンドの level[-] パラメータに相当します。

SESSION_GROUP

ユーザの SSO セッション グループを定義します。SESSION_GROUP プロパティは、最大 16 文字の文字列です。

Windows では、適切な名前がドロップダウン リストに存在しない場合、管理者がセッション グループの新しい名前を入力できます。

CA SSO で使用されます。

SHIFT

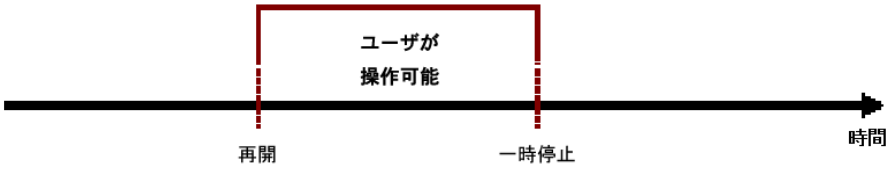
CA SSO で使用されます。

SUSPEND_DATE

ユーザ アカウントが一時停止されて無効になる日付を指定します。

レコードの一時停止日が再開日より前の日付である場合、ユーザは一時停止日より前および再開日より後に操作を実行できます。

ユーザの再開日が一時停止日より前の日付である場合は、再開日の*前*でもユーザ レコードは無効です。この場合、ユーザは再開日と一時停止日の間のみ操作を実行できます。



ユーザ レコードの SUSPEND_DATE プロパティの値は、グループ レコードの値より優先されます。

注: このプロパティは、ch[x]usr コマンドと ch[x]grp コマンドの suspend[-] パラメータに相当します。

SUSPEND_WHO

一時停止日をアクティブにした管理者を表示します。

UALIAS

1 つ以上の認証ホストに定義されている特定ユーザの別名を表示します。CA SSO で使用されます。

UPDATE_TIME

(情報のみ)レコードが最後に変更された日時を示します。

UPDATE_WHO

(情報のみ) 更新を実行した管理者を示します。

Windows 環境のクラス

このセクションでは、Windows データベースに存在するすべての Windows クラスおよび Windows プロパティ(nt 環境のクラス)をアルファベット順に説明します。

注: 用語「*nt 環境*」は、selang の env nt コマンドでアクセスされるデータベースのことです。これは、ユーザ、グループ、およびリソースを管理する Windows オペレーティング システムのデータベースと同じです。

COM クラス

COM クラスの各レコードでは、Windows の[コントロール パネル]-[ポート]で表示されるシリアル ポート(COM)またはパラレル ポート(LPT)を指定することによってデバイスを定義します。

注: CA Access Control を使用して COM クラスに新しいオブジェクトを作成することはできません。

COM クラスのキーは、制御されるポートの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

DEV

(情報のみ)。デバイスのシリアル番号を示す文字列。

DACL

標準アクセス制御リストを定義します。ここには、リソースへのアクセスを許可されたユーザとグループの名前、およびユーザまたはグループごとに与えられたアクセス権のレベルが登録されています。

このプロパティを変更するユーザは、リソースの所有者であるか、またはリソースへの特別なアクセス権（ACL を変更する権限）を持っている必要があります。

アクセス制御リストの各要素には、以下の情報が含まれます。

アクセス タイプ

リソースに以下のアクセス権を指定します。

- **Allowed** - リソースへの特別なアクセスを許可する。
- **Denied** - リソースへの特別なアクセスを拒否する。

アクセサ

アクセス権の許可または拒否の対象になるユーザまたはグループです。

アクセス

アクセサに与えられる、リソースに対するアクセス権限です。

注: ACL が空の場合、アクセス権が明示的に与えられません。したがって、アクセスは暗黙的に拒否されます。ACL を持たないリソースの場合、保護がオブジェクトに割り当てられません。したがって、すべてのアクセス要求が許可されます。

このプロパティを変更するには、auth コマンドまたは auth- コマンドを使用します。

GID

ファイルまたはデバイスのグループ情報を示します。

OWNER

レコードを所有するユーザまたはグループを定義します。

SACL

Windows システム アクセス制御リストです。監査ディレクティブを示します。

DEVICE クラス

DEVICE クラスの各レコードは、Windows の[コントロール パネル]-[デバイス]に表示される Windows のハードウェア デバイスを定義します。

DEVICE クラス レコードのキーは、制御されるデバイスの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

STARTUPTYPE

デバイスの起動方法(また、いつ起動するか)を定義します。以下のオプションがあります。

automatic

システムの起動中にデバイスを自動的に起動します。

boot

システムが起動するたびに、他のデバイスの起動前にデバイスを起動します。このオプションは、システムの動作に不可欠な、重要なデバイスに対して設定してください。

disabled

ユーザがデバイスを起動できないようにします。disabled でデバイスを無効にしても、システムによるデバイスの起動は可能です。

manual

ユーザまたは依存関係にあるデバイスによるデバイスの起動を許可します。

system

システムが起動するたびに、Boot デバイスの起動後にデバイスを起動します。このオプションは、システムの動作に不可欠な、重要なデバイスに対して設定してください。

このプロパティを変更するには、chres コマンドまたは editres コマンドで starttype パラメータを使用します。

STATUS

現在のサービスの状態を変更します。オプションには、started、stopped、および paused があります。

このプロパティを変更するには、chres コマンドまたは editres コマンドで status パラメータを使用します。

IMAGEPATH

指定したデバイスの完全修飾パスです。

PROFILE

ユーザのプロファイルへのパスを指定する文字列です。この文字列には、ローカルの絶対パスまたは UNC パスを含めることができます。

このプロパティを変更するには、chusr コマンド、editusr コマンド、または newusr コマンドで profile パラメータを使用します。

例: モデムのアクティブ化

モデムの状態を表示するには、以下の selang コマンドを入力します。

```
showres DEVICE modem
```

モデムをアクティブにするには、以下のコマンドを入力します。

```
chres device modem status(started)
```

DISK クラス

DISK クラスの各レコードは、システム ボリュームを定義します。ボリュームとは、プライマリ パーティション、拡張パーティションの論理ドライブ、ボリューム セット、ストライプ セット、ミラー セット、パリティ付きのストライプ セットなど、Windows オペレーティング システム(サーバ版)を実行しているコンピュータで作成および使用できるエンティティを示す一般的な用語です。ボリュームには、1 つのドライブ文字が割り当てられます。また、ボリュームはファイル システムで使用するためにフォーマットされます。

注: CA Access Control を使用して DISK クラスに新しいオブジェクトを作成することはできません。

DISK クラスのキーは、割り当てられたドライブ文字(C:、D: など)です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ATIME

（情報のみ）。レコードが最後にアクセスされた時刻。

CTIME

（情報のみ）。作成時刻です。

DACL

標準アクセス制御リストを定義します。ここには、リソースへのアクセスを許可されたユーザとグループの名前、およびユーザまたはグループごとに与えられたアクセス権のレベルが登録されています。

このプロパティを変更するユーザは、リソースの所有者であるか、またはリソースへの特別なアクセス権（ACL を変更する権限）を持っている必要があります。

アクセス制御リストの各要素には、以下の情報が含まれます。

アクセス タイプ

リソースに以下のアクセス権を指定します。

- **Allowed** - リソースへの特別なアクセスを許可する。
- **Denied** - リソースへの特別なアクセスを拒否する。

アクセサ

アクセス権の許可または拒否の対象になるユーザまたはグループです。

アクセス

アクセサに与えられる、リソースに対するアクセス権限です。

注: ACL が空の場合、アクセス権が明示的に与えられません。したがって、アクセスは暗黙的に拒否されます。ACL を持たないリソースの場合、保護がオブジェクトに割り当てられません。したがって、すべてのアクセス要求が許可されます。

このプロパティを変更するには、auth コマンドまたは auth- コマンドを使用します。

FILE_SYSTEM

（情報のみ）。ファイル システム（FAT または NTFS など）を指定する名前です。

FREE_SPACE

（情報のみ）。ディスクの空き領域の合計容量（KB 単位）です。

GID

ファイルまたはデバイスのグループ情報を示します。

LABEL

（情報のみ）。指定したボリュームの名前です。

LINK_NUMB

（情報のみ）。リンク数を指定します。 NTFS 以外のファイル システムの場合、このプロパティは常に 1 です。

MTIME

（情報のみ）。レコードが最後に変更された時刻です。

OWNER

レコードを所有するユーザまたはグループを定義します。

SACL

Windows システム アクセス制御リストです。監査ディレクティブを示します。

TYPE

（情報のみ）。リムーバブル、固定、CD-ROM、RAM ディスク、またはネットワークドライブからディスクのタイプを指定します。

USED_SPACE

（情報のみ）。ディスクの使用領域の合計容量（KB 単位）です。

DOMAIN クラス

DOMAIN クラスの各レコードは、共通のデータベースとセキュリティ ポリシー(ドメイン)を共有するコンピュータの集合を定義します。ドメインによって、ドメイン管理者が一元管理するユーザ アカウントとグループ アカウントへのアクセスが可能になります。各ドメインには一意の名前があります。

注: CA Access Control を使用して DOMAIN クラスに新しいオブジェクトを作成することはできません。

DOMAIN レコードのキーは、ドメイン名です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

BDC

(情報のみ)。ドメインのディレクトリ データベースのコピーを受け取り、ドメインのすべてのアカウント情報とセキュリティ ポリシー情報を含むコンピュータの名前。コピーは、プライマリドメイン コントローラ(PDC) 上のマスタ コピーと定期的に自動同期されます。バックアップ ドメイン コントローラ(BDC)も、ユーザ ログインを認証します。また、BDC は、必要に応じて PDC として機能することができます。1 つのドメインに複数の BDC を使用できます。

COMPUTERS

指定したドメインのメンバであるコンピュータを示します。

このプロパティを変更するには、chres コマンドまたは editres コマンドで computer パラメータまたは computer- パラメータを使用します。

DOMAIN_NAME

ドメイン名を定義します。

DOMAIN_USERS

(情報のみ)。指定したドメインのメンバであるユーザ アカウントおよびグループ アカウントを示します。

PDC

(情報のみ)。ドメイン内で最初に作成したコンピュータの名前。つまり、このコンピュータにはドメイン データのプライマリ格納域が含まれています。このコンピュータによって、ドメイン ログインが認証され、ドメインのディレクトリ データベースが保守されます。プライマリドメイン コントローラ(PDC)は、ドメイン上のすべてのコンピュータのアカウントに対して行われた変更を追跡します。これらの変更を直接受け取るのは、このコンピュータのみです。1 つのドメインには PDC が 1 つだけ存在します。

TRUSTED

信頼される側のドメインおよび信頼する側のドメインを示します。

信頼関係は、パス スルー認証を許可するドメイン間のリンクです。パス スルー認証では、信頼する側のドメインが信頼される側のドメインのログイン認証を認めます。信頼関係を結ぶと、1 つのドメイン内に 1 つのユーザ アカウントのみを持つユーザがネットワーク全体にアクセスできる場合があります。信頼される側のドメインの権限で定義されるユーザ アカウントとグローバル グループ、および信頼する側のドメイン内のリソース アクセス許可を提供できます。これは、これらのアカウントが、信頼する側のドメインのディレクトリ データベースに存在しない場合でも同様です。

このプロパティを変更するには、chres コマンドまたは editres コマンドで trusted パラメータまたは trusting- パラメータを使用します。このコマンドにはパスワードを指定する必要があります。

TRUSTING

ターゲットドメインを信頼する側のドメインです。

FILE クラス

Windows 環境で有効

FILE クラスの各レコードは、コンピュータの物理ドライブまたは論理ドライブ上のファイル システム(FAT、NTFS、CDFS など)上にあるファイルを定義します。

注: CA Access Control を使用してファイルを物理的にファイル上に作成することはできません。

FILE クラス レコードのキーは、レコードが保護するファイルまたはディレクトリの名前です。完全パスを指定する必要があります。

以下の定義では、FILE クラスのレコードに含まれるプロパティについて説明します。レコードの変更可能なプロパティを変更するには、selang または Web ベースの GUI を使用することができます。

ATIME

ファイルが最後にアクセスされた時刻を示します。

ATTRIB

ファイルまたはディレクトリの属性を示します。以下の 1 つまたは複数の属性を指定できます。

- ARCHIVE
- COMPRESSED
- DIRECTORY
- HIDDEN
- NORMAL
- OFFLINE
- READONLY
- SYSTEM
- TEMPORARY

CTIME

作成時刻を示します。

DACL

標準アクセス制御リストを定義します。ここには、リソースへのアクセスを許可されたユーザとグループの名前、およびユーザまたはグループごとに与えられたアクセス権のレベルが登録されています。

このプロパティを変更するユーザは、リソースの所有者であるか、またはリソースへの特別なアクセス権（ACL を変更する権限）を持っている必要があります。

アクセス制御リストの各要素には、以下の情報が含まれます。

アクセス タイプ

リソースに以下のアクセス権を指定します。

- **Allowed** - リソースへの特別なアクセスを許可する。
- **Denied** - リソースへの特別なアクセスを拒否する。

アクセサ

アクセス権の許可または拒否の対象になるユーザまたはグループです。

アクセス

アクセサに与えられる、リソースに対するアクセス権限です。

注: ACL が空の場合、アクセス権が明示的に与えられません。したがって、アクセスは暗黙的に拒否されます。ACL を持たないリソースの場合、保護がオブジェクトに割り当てられません。したがって、すべてのアクセス要求が許可されます。

このプロパティを変更するには、auth コマンドまたは auth- コマンドを使用します。

DEV

ファイルが存在するボリュームのシリアル番号を示します。

FILE_SYSTEM

ファイルが存在するファイル システムの名前を示します。

GID

ファイルまたはデバイスのグループ情報を示します。

INDEX

ファイルに関連付けられた一意の識別子を示します。

ISDIR

ファイルがディレクトリかどうかを示します。

LINKS_NUMB

ファイルへのリンク数を示します。FAT ファイル システムの場合、このプロパティは常に 1 です。NTFS ファイル システムの場合、このプロパティは 2 以上です。

MTIME

ファイルが最後に変更された時刻を示します。

NAME

ファイル名を示します。

OWNER

レコードを所有するユーザまたはグループを定義します。

SACL

Windows システム アクセス制御リストです。監査ディレクティブを示します。

SIZE

ファイルのサイズ(バイト単位)を示します。

詳細情報:

Windows のファイル属性 (P. 565)
chfile コマンド - Windows ファイル設定の変更 (P. 212)

GROUP クラス

GROUP クラスには、Windows オペレーティング システムに定義されているすべてのグループ レコードが含まれます。GROUP クラスのレコードは、ユーザのすべてのグループを表します。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

COMMENT

レコードに含める追加情報です。この情報が権限付与に使用されることはありません。

このプロパティを変更するには、chgrp コマンド、editgrp コマンド、および newgrp コマンドの comment[-] パラメータを使用します。

制限: 255 文字。

FULL_NAME

ユーザに関連付けられたフル ネームです。フル ネームは、CA Access Control の監査ログ メッセージでユーザを識別するために使用されますが、権限付与に使用されることはありません。

このプロパティを変更するには、chusr コマンド、editusr コマンド、または newusr コマンドの name パラメータを使用します。

GID

(情報のみ)。グループの相対識別子を含む値。相対識別子は、グループの作成時にアカウント データベースによって決定されます。相対識別子によって、ドメイン内のアカウント マネージャに対してグループを一意に識別できます。

GLOBAL

グローバル グループを示します。このプロパティは、Windows のグループにのみ適用できます。このプロパティは、CA Access Control の旧バージョンの ISGLOBAL プロパティに代わるものです。

このプロパティを追加するには、newgrp コマンド(専用)で globalGroup パラメータを使用します。

USERLIST

グループに所属するユーザおよびグローバル グループ（ローカル グループ専用）のリスト。このプロパティで設定するリストは、CA Access Control データベースで設定するリストとは異なる場合があります。

このプロパティを変更するには、join[-] コマンドで username（*groupname*）パラメータを使用します。

PRIVILEGES

グループに割り当てられた Windows 権限。

このプロパティを変更するには、chgrp コマンド、editgrp コマンド、および newgrp コマンドで privileges パラメータを使用します。

詳細情報:

chgrp コマンド - Windows グループの変更 (P. 213)
Windows の権限 (P. 569)

OU クラス

OU（組織単位）クラスには、ユーザ、グループ、コンピュータなどのオブジェクトが含まれます。OU クラスのオブジェクトは、プライマリドメイン コントローラ上で作成でき、子オブジェクトとして他のオブジェクト（グループなど）を持つことができます。したがって、OU クラスのオブジェクトはコンテナ オブジェクトです。

注: OU クラスは、Active Directory がインストールされている Windows 2000 Advanced Server でのみ利用できます。

OU クラスには、事前定義されたプロパティがありません（他のクラスには事前定義されたプロパティがあります）。ただし、以下の OU のプロパティを更新できます。

- Country/Region
- 説明
- Desktop
- City
- Display Name
- Folder（読み取り専用プロパティ）

- Fax number
- Managed objects（読み取り専用プロパティ）
- Member of（読み取り専用プロパティ）
- Name（読み取り専用プロパティ）
- Postal address
- Postal code
- P.O. box
- State/Province
- Street
- Telephone
- Object changed（読み取り専用プロパティ）
- Object created（読み取り専用プロパティ）
- Web page

PRINTER クラス

PRINTER クラスの各レコードは、メディア上にビジュアル イメージを再現できる、Windows コンピュータ システムに接続されているデバイス（[プリンタ]フォルダに表示される）を定義します。

注: CA Access Control を使用して、PRINTER クラスの新しいオブジェクトを作成することはできません。

PRINTER クラス レコードのキーは、ローカル プリンタの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

DACL

標準アクセス制御リストを定義します。ここには、リソースへのアクセスを許可されたユーザとグループの名前、およびユーザまたはグループごとに与えられたアクセス権のレベルが登録されています。

このプロパティを変更するユーザは、リソースの所有者であるか、またはリソースへの特別なアクセス権（ACL を変更する権限）を持っている必要があります。

アクセス制御リストの各要素には、以下の情報が含まれます。

アクセス タイプ

リソースに以下のアクセス権を指定します。

- **Allowed** - リソースへの特別なアクセスを許可する。
- **Denied** - リソースへの特別なアクセスを拒否する。

アクセサ

アクセス権の許可または拒否の対象になるユーザまたはグループです。

アクセス

アクセサに与えられる、リソースに対するアクセス権限です。

注: ACL が空の場合、アクセス権が明示的に与えられません。したがって、アクセスは暗黙的に拒否されます。ACL を持たないリソースの場合、保護がオブジェクトに割り当てられません。したがって、すべてのアクセス要求が許可されます。

このプロパティを変更するには、auth コマンドまたは auth- コマンドを使用します。

COMMENT

レコードに含める追加情報を定義します。この情報が権限付与に使用されることはありません。

制限: 255 文字。

LOCATION

プリンタの場所を示す文字列です。この情報が権限付与に使用されることはありません。

このプロパティを変更するには、chres コマンドまたは editres コマンドで location パラメータを使用します。このプロパティを削除するには、空白の () を使用します。

OWNER

レコードを所有するユーザまたはグループを定義します。

SHARE

プリンタの共有ポイントを識別する名前です。プリンタにアクセスするユーザまたはグループは、その共有名を使用できます。

このプロパティを変更するには、chres コマンドまたは editres コマンドで share_name パラメータまたは share_name- パラメータを使用します。

NAME

プリンタ名です。

SACL

Windows システム アクセス制御リストです。監査ディレクティブを示します。

SERVER

(情報のみ)。プリンタを制御するサーバを識別する文字列です。このプロパティが存在しない場合、プリンタはローカルで制御されます。

PROCESS クラス

PROCESS クラスの各レコードは、実行可能プログラム、一連の仮想メモリ アドレス、およびスレッドで構成されている(Windows のタスク マネージャに表示される)オブジェクトを定義します。

注: CA Access Control を使用して PROCESS クラスに新しいオブジェクトを作成することはできません。

PROCESS クラス レコードのキーは、実行中のプログラムの実行可能モジュールの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。このクラスに変更可能なプロパティはありません。 変更できないプロパティには、「*情報のみ*」と記載されます。

IMAGE_PATH

（情報のみ）。指定した実行可能モジュールの完全修飾パスです。

PROCESS_ID

（情報のみ）。プロセスの一意の識別子です。プロセス ID 番号は再利用されるため、そのプロセスの有効期間のみプロセスが識別されます。

PROCESS クラスを使用するときには、以下の制限を考慮してください。

- CA Access Control は、Windows での プロセス作成をトレースします。しかし、seosd が新規プロセス引数を取得し、取得した引数を全般トレースに書き込むのは、プロセスを開始したユーザがトレース対象としてマーキングされている場合のみです。
- 新規プロセスが作成されても、プロセスの初期設定が終了するまで、その引数は利用可能になりません。seosd は、プロセス引数の非同期トレースを試行します。しかし、プロセスが非常に短い場合は、seosd がプロセス引数を取得し、取得した引数をトレースに書き込む前に、プロセスが終了する場合があります。この場合、トレースに以下のメッセージが表示されます。

  EXECARGS: 利用不可 (87)

- プロセス ID は、Windows で再利用されます。プロセスが非常に短い場合、seosd が同じプロセス ID を取得した別のプロセスのプロセス引数を取得し、取得した引数をトレースに書き込むことは理論的には可能です。

REGKEY クラス

REGKEY クラスの各レコードは、Windows レジストリのキーを定義します。

REGKEY レコードのキーは、Windows レジストリ キーの完全パスです。

注: パスの指定にはワイルドカード文字を使用できます。

以下の定義では、REGKEY クラスのレコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。変更できないプロパティには、「*情報のみ*」と記載されます。

DACL

標準アクセス制御リスト。リソースへのアクセスを許可されたユーザとグループの名前、およびユーザまたはグループごとに与えられたアクセス権のレベルが登録されています。

このプロパティを変更するユーザは、リソースの所有者であるか、またはリソースへの特別なアクセス権（ACL を変更する権限）を持っている必要があります。

アクセス制御リストの各要素には、以下の情報が含まれます。

アクセス タイプ

リソースに以下のアクセス権を指定します。

- **Allowed** - リソースへの特別なアクセスを許可する。
- **Denied** - リソースへの特別なアクセスを拒否する。

アクセサ

アクセス権の許可または拒否の対象になるユーザまたはグループの名前です。

アクセス

アクセサに与えられる、リソースに対するアクセス権限です。REGKEY クラスの有効なアクセス権限は、以下のとおりです。

- **all** - クラスに許可できるすべての操作の実行をアクセサに対して許可または拒否する。
- **append/create/subkey** - レジストリ キーのサブキーの作成または変更をアクセサに対して許可または拒否する。
- **changeperm/sec/dac/writedac/perm** - リソースの ACL の変更（つまりアクセサの追加または削除）をアクセサに対して許可または拒否する。
- **chown/owner/takeownership** - リソースの所有者の変更をアクセサに対して許可または拒否する。
- **delete** - リソースの削除をアクセサに対して許可または拒否する。

- **enum** - レジストリ キーのサブキーの列挙をアクセサに対して許可または拒否する。
- **link** - レジストリ キーへのリンクの作成をアクセサに対して許可または拒否する。
- **notify** - レジストリ キーの変更通知またはレジストリ キーのサブキーの要求をアクセサに対して許可または拒否する。
- **query** - レジストリ キーの値のクエリをアクセサに対して許可または拒否する。
- **read** - キーの内容の読み取りをアクセサに対して許可または拒否する。ただし、変更は保存できなくなります。
- **readcontrol/manage** - レジストリ キーのセキュリティ記述子の情報（システム（監査）アクセス制御リストの情報は含まない）の読み取りをアクセサに対して許可または拒否する。
- **set** - レジストリ キーの値の作成または設定をアクセサに対して許可または拒否する。
- **write** - レジストリ キーとそのサブキーの変更をアクセサに対して許可または拒否する。

注: 空の ACL（エントリのない ACL）と ACL を持たないリソースとの違いに注意してください。空の ACL の場合、アクセス権が明示的に与えられません。したがって、アクセスは暗黙的に拒否されます。ACL を持たないリソースの場合、保護がオブジェクトに割り当てられません。したがって、すべてのアクセス要求が許可されます。

このプロパティを変更するには、auth コマンドまたは auth- コマンドを使用します。

OWNER

リソースの所有者として指定されているユーザまたはグループ。

このプロパティを変更するには、newres コマンド、chres コマンド、および editres コマンドで owner パラメータを使用します。

SACL

Windows システム アクセス制御リストは、監査ディレクティブを指定します。

SUBKEYS

（情報のみ）。キーの下に存在するレジストリ キー（サブキー）のリストです。

SUBVALUES

（情報のみ）。現在のレジストリ キーに記述されているレジストリ値のリストです。

REGVAL クラス

REGVAL クラスの各レコードは、レジストリ キーを記述するデータを定義します。このデータは、単一または複数のユーザ、アプリケーション、およびハードウェア デバイスに関するシステム構成に必要な情報を保存します。レジストリ値には、操作中に頻繁に参照される情報が含まれます。たとえば、以下のような情報が含まれます。

- 各ユーザのプロファイル
- コンピュータにインストールしたアプリケーションと、各アプリケーションで作成できるファイルのタイプ
- フォルダやアプリケーション アイコンのプロパティ シートの設定
- ハードウェア構成
- 使用されているポート

REGVAL レコードのキーは、レジストリ キーの完全パス名とその値です。

注: レジストリ キーやその値を間違って変更または削除すると、システム全体に影響する重大な問題を引き起こす可能性があり、問題を解決するためには Windows の再インストールが必要になる場合があります。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

TYPE

データを格納する形式。レジストリ値にデータを格納するときに、格納するデータの型を示すために以下の値のいずれかを指定できます。

注: レジストリ値を作成または変更するときに、以下のデータ型を指定します。

DWORD

4 バイト超の数で表されるデータ。デバイスドライバやサービスの多くのパラメータがこのデータ型で、バイナリ、16 進数、および 10 進数の形式で表示できます。

STRING

読み取り可能なテキストを表す一連の文字。

MULTISTRING

複数の文字列。読み取り可能なテキストのリストまたは複数の値を含む値です。各エントリは、Null 文字で区切られます。

BINARY

生のバイナリ データ。ハードウェア コンポーネントの情報の大部分は、バイナリ データとして格納され、16 進数形式または簡単に読み取れる形式で表示できます。

このプロパティを変更するには、newres コマンド、chres コマンド、または editres コマンドのパラメータとして上記のデータ型のいずれかを使用します。

VALUE

Windows レジストリ値が保持する値。

SEOS クラス

SEOS クラスは、ネイティブのローカル セキュリティ システムの動作を制御します。

クラスには、SEOS というレコードが 1 つだけ含まれます。このレコードは、一般的なネイティブ セキュリティ オプションを指定します。SEOS クラス プロパティのステータスを表示または変更するには、setoptions コマンドを使用します。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

AuditCategory

監査する認証されたイベントと不正なイベントの検出を指定します。

AccountLogon

このコンピュータがアカウントの検証に使用されているコンピュータに対するユーザのログオンまたはログオフの各インスタンスを監査するかどうかを指定します。

AccountManagement

コンピュータのアカウント管理の各イベントを監査するかどうかを指定します。アカウント管理イベントには以下のようなものがあります。

- ユーザ アカウントまたはグループが作成、変更、または削除された。
- ユーザ アカウントの名前が変更された、またはユーザ アカウントが無効または有効にされた。
- パスワードが設定または変更された。

DirectoryAccess

専用のシステム アクセス制御リスト（SACL）が定義されている Active Directory オブジェクトへのユーザによるアクセスのイベントを監査するかどうかを指定します。

Logon

ユーザのコンピュータに対するログオンまたはログオフの各インスタンスを監査するかどうかを指定します。

ObjectAccess

ユーザによるオブジェクトへのアクセスのイベントを監査するかどうかを指定します。オブジェクトの例としては、専用のシステム アクセス制御リスト(SACL)が定義されているファイル、フォルダ、レジストリ キー、プリンタなどが挙げられます。

PolicyChange

ユーザ権限の割り当てポリシー、監査ポリシー、または信頼ポリシーへの変更の各インシデントを監査するかどうかを指定します。

PrivilegeUse

ユーザによるユーザ権限の使用の各インスタンスを監査するかどうかを指定します。

DetailedTracking

プログラムのアクティブ化、プロセスの終了、ハンドルの複製、オブジェクトへの間接アクセスなどのイベントに関する詳細なトレース情報を監査するかどうかを指定します。

システム

ユーザがコンピュータを再起動またはシャットダウンしたとき、またはシステム セキュリティまたはセキュリティ ログに影響するイベントが発生したときに監査するかどうかを指定します。

History

ユーザ アカウントに一意の新しいパスワードを関連付ける数を指定します。この数に達すると、古いパスワードを再利用できるようになります。

制限: 1 ～ 24 までの整数。0 を指定すると、パスワードは保存されません。

Interval

ユーザがパスワードを使用できる有効期間(日単位)を指定します。この期間が過ぎると、システムがユーザに変更を要求します。

Min life

ユーザがパスワードを最低でも使用しなければならない期間(日単位)を指定します。この期間が過ぎると、ユーザはパスワードを変更できます。

Min length

ユーザ アカウントのパスワードに使用する最小文字数を定義します。

Password fails

ログオンの失敗数を定義します。この数になると、ユーザ アカウントがロックアウトされます。

Reset count after

ログオン失敗から失敗ログイン カウンタを 0 にリセットするまでの時間を分単位で指定します。

SERVICE クラス

SERVICE クラスの各レコードは、Windows の[コントロール パネル]-[サービス]で表示される Windows サービスを定義します。

SERVICE クラス レコードのキーは、制御されるサービスの名前です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

ACCOUNT

サービスのログイン アカウントを変更します。大部分のサービスは、システム アカウントでログインする必要がありますが、特別なユーザ アカウントでログインするように設定できるサービスもあります。詳細については、関連する Microsoft Windows のマニュアルを参照してください。デフォルト値は LocalSystem です。

このプロパティを変更するには、chres コマンドまたは editres コマンドで account パラメータを使用します。

BINARY_NAME

サービスの実行可能ファイルの場所を指す完全パスです。

IMAGEPATH

指定した実行可能モジュールの完全修飾パスです。

INTERACTIVE

サービスが開始されている状態のときに、ログインしたすべてのユーザが使用できるユーザ インターフェースをデスクトップに表示します。このインターフェースは、サービスが LocalSystem アカウントとして実行されている場合にのみ使用可能です。

このプロパティを変更するには、chres コマンドまたは editres コマンドで interactive パラメータを使用します。

PROFILE

ユーザのプロファイルへのパスを指定する文字列です。この文字列には、ローカルの絶対パスまたは UNC パスを含めることができます。

このプロパティを変更するには、chusr コマンド、editusr コマンド、または newusr コマンドで profile パラメータを使用します。

REG_KEY

Windows レジストリのサービス定義の場所を指します。

STARTUPTYPE

サービスを開始する方法（また、いつ開始するか）を定義します。以下のオプションがあります。

- **automatic** - システムの起動中にデバイスを自動的に起動する。
- **disabled** - ユーザまたは依存関係にあるサービスによってサービスを開始できないようにする。
- **manual** - ユーザまたは依存関係にあるサービスによるサービスの開始を許可する。
- このプロパティを変更するには、chres コマンドまたは editres コマンドで startuptype パラメータを使用します。

STATUS

現在のサービスの状態を変更します。オプションには、started、stopped、および paused があります。

このプロパティを変更するには、chres コマンドまたは editres コマンドで status パラメータを使用します。

例: サービスを手動で開始する設定

SeOSAgent サービスを手動で開始するように変更するには、以下の selang コマンドを入力します。

```
chres SERVICE "SeosAgent" starttype(manual)
```

例: ディレクトリ ログイン アカウントの変更

Directory Replicator のログイン アカウントを ReplAdmin に変更し、パスワードを abcde とするには、以下の selang コマンドを入力します。

```
chres SERVICE directory replicator account(repladmin) domainpwd(abcde)
```

SESSION クラス

SESSION クラスの各レコードは、ローカル ホスト上のユーザ セッションを定義します。このレコードには、ユーザ名、コンピュータ名、接続経過時間、および使用中のリソースが含まれます。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

CNAME

セッションが確立されたホスト名です。

GUEST

セッションが Guest アカウントで作成されたかどうかを示します。

IDLE

サーバとワークステーションとの間のネットワーク セッションを終了します。

このプロパティを変更するには、chres コマンドまたは editres コマンドで disconnect パラメータを使用します。

OPENS

開かれているファイル セッションの数を示します。

RESOURCES

サーバ上の共有ファイルに関する情報を提供するプロパティです。この情報には、開かれている共有リソースのパスや、リソースを開いたユーザまたはコンピュータが含まれます。

TIME

セッションが確立されてから経過した時間です。

USER

ユーザの相対 ID（RID）を含む値です。RID は、ユーザの作成時にセキュリティ アカウント マネージャ（SAM）によって決定されます。RID によって、ユーザ アカウントがドメイン内の SAM に対して一意に定義されます。

例: ローカル セッションからのユーザの切断

ローカル ホストのセッションからユーザ ZORRO を切断するには、以下の selang コマンドを入力します。

```
chres SESSION zorro disconnect
```

注: ユーザの接続を切断すると、データが失われる可能性があります。接続を切断する前に、ユーザに警告することをお勧めします。

SHARE クラス

SHARE クラスの各レコードは、1 つ以上のデバイスまたはプログラムで使用するデバイス、データ、またはプログラムに指定できる共有リソースを定義します。Windows の場合、共有リソースとは、ディレクトリ、ファイル、プリンタ、および名前付きパイプなど、ネットワーク ユーザが使用可能な任意のリソースを指します。また、共有はネットワーク ユーザが使用可能なサーバ上のリソースも指します。

SHARE クラス レコードのキーは、リソースの共有名です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

CURR_USERS

（情報のみ）。リソースへの現在の接続数です。

DACL

標準アクセス制御リストを定義します。ここには、リソースへのアクセスを許可されたユーザとグループの名前、およびユーザまたはグループごとに与えられたアクセス権のレベルが登録されています。

このプロパティを変更するユーザは、リソースの所有者であるか、またはリソースへの特別なアクセス権（ACL を変更する権限）を持っている必要があります。

アクセス制御リストの各要素には、以下の情報が含まれます。

アクセス タイプ

リソースに以下のアクセス権を指定します。

- **Allowed** - リソースへの特別なアクセスを許可する。
- **Denied** - リソースへの特別なアクセスを拒否する。

アクセサ

アクセス権の許可または拒否の対象になるユーザまたはグループです。

アクセス

アクセサに与えられる、リソースに対するアクセス権限です。

注: ACL が空の場合、アクセス権が明示的に与えられません。したがって、アクセスは暗黙的に拒否されます。ACL を持たないリソースの場合、保護がオブジェクトに割り当てられません。したがって、すべてのアクセス要求が許可されます。

このプロパティを変更するには、auth コマンドまたは auth- コマンドを使用します。

MAX_USERS

共有リソースに対して可能な最大同時接続数です。

注: このプロパティの値としてゼロ（0）を指定することはできません。ゼロを指定すると、Windows によって無視されます。

このプロパティを変更するには、newres コマンド、chres コマンド、または editres コマンドで max_users パラメータを使用します。

NAME

共有の名前を定義します。

PATH

共有リソースのローカル パスを指定する文字列です。ディスクの場合、これは共有になっているパスです。印刷キューの場合、これは共有になっている印刷キューの名前です。

このプロパティを変更するには、newres コマンド、chres コマンド、または editres コマンドで path パラメータを使用します。

PERMISSION

(情報のみ)。共有レベルのセキュリティで実行しているサーバに対する共有リソースのアクセス許可を示す値です。このプロパティは、以下の表に示す値のいずれかです。

ACCESS_READ

リソースのデータを読み取り、デフォルトで実行できます。

ACCESS_WRITE

リソースへのデータの書き込みができます。

ACCESS_CREATE

リソース(ファイルなど)のインスタンスを作成できる。つまり、リソースを作成したら、そのリソースにデータを書き込むことができる。

ACCESS_EXEC

リソースを実行できます。

ACCESS_DELETE

リソースを削除できます。

ACCESS_ATRIB

リソースの属性(ファイルを最後に変更した日時など)を変更できます。

ACCESS_PERM

ユーザまたはアプリケーションのリソースに割り当てられたアクセス許可（読み取り、書き込み、作成、実行、および削除）を変更できます。

ACCESS_ALL

リソースの読み取り、書き込み、作成、実行、および削除ができ、リソースの属性およびアクセス許可を変更できます。

ACCESS_NONE

アクセス許可を与えません。

REMARK

レコードに含める追加情報です。この文字列には、255 文字までの英数字を指定できます。この情報が権限付与に使用されることはありません。

このプロパティを変更するには、newres コマンド、chres コマンド、および editres コマンドで comment パラメータまたは comment- パラメータを使用します。

RESOURCES

（情報のみ）。サーバ上の共有ファイルに関する情報を提供するプロパティです。この情報には、開かれている共有リソースのパスや、リソースを開いたユーザまたはコンピュータが含まれます。

TYPE

（情報のみ）。共有のタイプです。共有リソースには、以下のタイプのいずれかを使用します。

ファイル フォルダ

ディスクドライブ。サーバのリモート管理（ADMIN$）や、C$、D$ などの管理共有も該当します。

印刷キュー

印刷キュー

通信デバイス

通信デバイス

プロセス間通信（IPC）

プロセス間通信用に予約された特別な共有（IPC$）

USERS

共有リソースに現在アクセス中のユーザに関する情報です。 この情報には、接続を確立したユーザの名前(USER)、サーバの共有リソースの共有名、またはクライアントのコンピュータ名(MACHINE)が含まれます。また、接続が確立されている秒数(TIME)、および接続の結果として現在開かれているファイル数(INUSE)も含まれます。

USER クラス

USER クラスには、Windows オペレーティング システムに定義されているすべてのユーザ レコードが含まれます。USER クラスのレコードのキーは、ユーザがシステムへのログイン時に入力したユーザ名です。

以下の定義では、このクラス レコードに含まれるプロパティについて説明します。ほとんどのプロパティは変更可能で、selang インターフェースまたは管理インターフェースを使用して操作することができます。 変更できないプロパティには、「*情報のみ*」と記載されます。

BAD_PW_COUNT

(情報のみ)。ユーザが間違ったパスワードを使用してアカウントにログインしようとした回数です。値 -1 は、その値が不明であることを示します。

COMMENT

レコードに含める追加情報です。この情報が権限付与に使用されることはありません。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドの comment[-] パラメータを使用します。

制限: 255 文字。

COUNTRY

ユーザの国記述子を指定する文字列です。この文字列は、X.500 ネーミング スキーマの一部です。この情報が権限付与に使用されることはありません。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドの country パラメータを使用します。

DAYTIME

ユーザがリソースにアクセスできる日時を管理する、曜日と時間帯の制限です。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドの restrictions パラメータを使用します。

注: このプロパティの情報は、入力された分単位の値が切り捨てられること以外は、AC 環境の DAYTIME プロパティの情報と同一です。

DIAL_CALLBACK

ユーザに提供するコール バック権限の種類。以下のオプションが定義されています。

NoCallBack

ユーザにはコール バック権限がありません。

SetByCaller

リモート ユーザは、ダイヤル イン時にコール バック用の電話番号を指定できます。

Call-back Phone Number

管理者はコール バック用の番号を設定します。

このプロパティを変更するには、chusr コマンドまたは editusr コマンドで gen_prop パラメータまたは gen_val パラメータを使用します。

DIAL_PERMISSION

RAS サーバにダイヤル インするためのアクセス許可です。値に 0 を指定すると、ユーザは RAS サーバにダイヤル インできません。

このプロパティを変更するには、chusr コマンドまたは editusr コマンドで gen_prop パラメータまたは gen_val パラメータを使用します。

EXPIRE_DATE

USER クラスのレコードが有効期限切れで無効になる日付です。USER クラスのレコードの EXPIRE_DATE プロパティの値は、GROUP クラスのレコードの値より優先されます。有効期限切れのレコードを再び有効にするには、chusr コマンドの expire- パラメータを使用します。有効期限切れのユーザを再開することはできません。一時停止したユーザは、再開日を指定することで再開できます。

このプロパティを変更するには、chusr コマンド、editusr コマンド、または newusr コマンドの expire パラメータあるいは expire- パラメータを使用します。

FLAGS

特定の属性を指定するためにユーザのアカウントに割り当てることができるフラグです。各アカウントに複数のフラグを適用できます。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドで flags パラメータを使用します。

FULL_NAME

ユーザに関連付けられたフル ネームです。フル ネームは、CA Access Control の監査ログ メッセージでユーザを識別するために使用されますが、権限付与に使用されることはありません。

このプロパティを変更するには、chusr コマンド、editusr コマンド、または newusr コマンドの name パラメータを使用します。

GID

グループの相対識別子を含む値。相対識別子は、グループの作成時にアカウント データベースによって決定されます。相対識別子によって、ドメイン内のアカウント マネージャに対してグループを一意に識別できます。

GROUPS

ユーザが所属するグループのリストです。このプロパティで設定するグループ リストは、AC 環境の GROUPS プロパティで設定するユーザ リストとは異なる場合があります。

このプロパティを変更するには、join[-] コマンドの group パラメータを使用します。

HOME

ホーム ディレクトリは、該当ユーザがアクセスでき、該当ユーザのファイルやプログラムが保存されるフォルダです。ホーム ディレクトリはユーザごとに割り当てることができ、複数のユーザで共有することもできます。

HOMEDIR

ユーザのホーム ディレクトリを指定する文字列です。ユーザは、自分のホーム ディレクトリに自動的にログインできます。

このプロパティを変更するには、chusr コマンド、editusr コマンド、または newusr コマンドの homedir パラメータを使用します。

HOME_DRIVE

ユーザのホーム ディレクトリのドライブを指定する文字列です。ユーザは、自分のホームドライブおよびホーム ディレクトリに自動的にログインできます。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドで homedrive パラメータを使用します。

ID

ユーザの相対 ID（RID）を含む値です。RID は、ユーザの作成時にセキュリティ アカウント マネージャ（SAM）によって決定されます。RID によって、ユーザ アカウントがドメイン内の SAM に対して一意に定義されます。

LAST_ACC_TIME

（情報のみ）。最後にログインが実行された日時です。

LAST_LOGOFF

（情報のみ）。最後にログオフが実行された日時です。

LOCATION

ユーザの所在地を格納するために使用する文字列です。この情報が権限付与に使用されることはありません。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドの location パラメータを使用します。

LOGON_SERVER

ユーザのログイン情報を確認するサーバを指定する文字列です。ユーザがドメイン ワークステーションにログインすると、ログイン情報がサーバに送信され、サーバによってユーザがワークステーションを使用することが許可されます。

MAX_LOGINS

（情報のみ）。 ユーザがこのアカウントに正常にログインした回数。 値 -1 は、その値が不明であることを示します。

NAME

ユーザの名前です。

ORGANIZATION

ユーザが所属する組織に関する情報を格納する文字列です。この文字列は、X.500 ネーミング スキーマの一部です。この情報が権限付与に使用されることはありません。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドの organization パラメータを使用します。

ORG_UNIT

ユーザが所属する組織単位に関する情報を格納する文字列です。この文字列は、X.500 ネーミング スキーマの一部です。この情報が権限付与に使用されることはありません。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドで org_unit パラメータを使用します。

PASSWD_EXPIRED

ユーザ アカウントが失効する日付です。

PGROUP

ユーザのプライマリ グループ ID です。プライマリ グループは、ユーザが定義されているグループの 1 つです。プライマリ グループはグローバル グループである必要があります。この文字列には、スペースまたはカンマを指定できません。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドで pgroup パラメータを使用します。

PHONE

ユーザの電話番号を格納するために使用できる文字列です。この情報が権限付与に使用されることはありません。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドの phone パラメータを使用します。

PRIVILEGES

ユーザに割り当てられた Windows 権限です。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドで privileges パラメータを使用します。

PROFILE

ユーザのプロファイルへのパスを指定する文字列です。この文字列には、ローカルの絶対パスまたは UNC パスを含めることができます。

このプロパティを変更するには、chusr コマンド、editusr コマンド、または newusr コマンドで profile パラメータを使用します。

PW_LAST_CHANGE

（情報のみ）。パスワードが更新された日時です。

RESUME_DATE

一時停止された USER アカウントが有効になる日付です。

RESUME_DATE および SUSPEND_DATE を組み合わせて指定する方法については、SUSPEND_DATE の説明を参照してください。

SCRIPT

ユーザのログオン スクリプト ファイルのパスを指定する文字列です。スクリプト ファイルには、.CMD ファイル、.EXE ファイル、.BAT ファイルを指定できます。

TERMINALS

ユーザがログインできる端末のリストを指定する文字列です。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドで terminals パラメータを使用します。

TS_CONFIG_PGM

クライアントが初期プログラムを指定できるかどうかを示す値です。

TS_INITIAL_PGM ユーザ プロパティは、初期プログラムを示します。ユーザの初期プログラムを指定すると、ユーザが実行することができるプログラムがそれだけになります。そのプログラムを終了したユーザは、ターミナル サーバによってログオフされます。

この値を 1 に設定すると、クライアントが初期プログラムを指定することができます。この値を 0 に設定すると、クライアントは初期プログラムを指定することができません。

このプロパティを変更するには、chusr コマンドと editusr コマンドで gen_prop パラメータおよび gen_val パラメータを使用します。

TS_HOME_DIR

ターミナル サーバにログオンするためのユーザのホーム ディレクトリのパスです。この文字列には、ローカルのパスまたは UNC パス（¥¥machine¥share¥path）を指定できます。

このプロパティを変更するには、chusr コマンドと editusr コマンドで gen_prop パラメータおよび gen_val パラメータを使用します。

TS_HOME_DRIVE

UNC パスが TS_HOME_DIR プロパティで指定されるドライブ（コロンの後にドライブ文字を指定）です。

このプロパティを変更するには、chusr コマンドと editusr コマンドで gen_prop パラメータおよび gen_val パラメータを使用します。

TS_INITIAL_PGM

ターミナル サービスがユーザのログオン時に実行する初期プログラムのパス。

ユーザの初期プログラムを指定すると、ユーザが実行することができるプログラムはそれだけになります。そのプログラムを終了したユーザは、ターミナル サーバによってログオフされます。

TS_CONFIG_PGM プロパティを 1 に設定すると、クライアントが初期プログラムを指定することができます。

このプロパティを変更するには、chusr コマンドと editusr コマンドで gen_prop パラメータおよび gen_val パラメータを使用します。

TS_PROFILE_PATH

ターミナル サーバにログオンするためのユーザのプロファイルのパスです。パスで識別されるディレクトリは、ログオン前に手動で作成する必要があります。

このプロパティを変更するには、chusr コマンドと editusr コマンドで gen_prop パラメータおよび gen_val パラメータを使用します。

TS_WORKING_DIR

ターミナル サービスがユーザのログオン時に実行する初期プログラムの作業ディレクトリのパス。

このプロパティを変更するには、chusr コマンドと editusr コマンドで gen_prop パラメータおよび gen_val パラメータを使用します。

WORKSTATIONS

ユーザがログインできるワークステーションのリスト。

このプロパティを変更するには、chusr コマンド、editusr コマンド、および newusr コマンドで workstations パラメータを使用します。

詳細情報:

chusr コマンド - Windows ユーザの変更 (P. 219)
Windows のアカウント フラグフラグ (P. 566)
Windows の権限 (P. 569)

UNIX 環境のクラス

このセクションでは、UNIX システム ファイルに存在するすべての UNIX クラス (unix 環境のクラス) をアルファベット順に説明します。これらネイティブ クラスのプロパティはオペレーティング システムが管理しており、システムによって異なります。

注: 用語「*unix 環境*」は、selang の env unix コマンドでアクセスされるシステム ファイルのことです。これは、UNIX オペレーティング システムがユーザやグループに対して維持しているシステム ファイルと同じであり、システム上のファイルです。

FILE クラス

FILE クラスの各レコードは、ファイル システム上のコンピュータの物理ドライブまたは論理ドライブを定義します。

注: CA Access Control を使用してディスク上に物理的にファイルを作成することはできません。

FILE クラス レコードのキーは、レコードが保護するファイルまたはディレクトリの名前です。完全パスを指定する必要があります。

このネイティブ クラスのプロパティはオペレーティング システムが管理しており、システムによって異なります。chfile コマンドを実行すると、selang を使用して変更できるネイティブ プロパティが一覧表示されます。

GROUP クラス

GROUP クラスには、UNIX オペレーティング システムに定義されているすべてのグループ レコードが含まれます。GROUP クラスのレコードは、ユーザのすべてのグループを表します。

このネイティブ クラスのプロパティはオペレーティング システムが管理しており、システムによって異なります。chgrp コマンドを実行すると、selang を使用して変更できるネイティブ プロパティが一覧表示されます。

USER クラス

USER クラスには、UNIX オペレーティング システムに定義されているすべてのユーザ レコードが含まれます。USER クラスのレコードのキーは、ユーザがシステムへのログイン時に入力したユーザ名です。

このネイティブ クラスのプロパティはオペレーティング システムが管理しており、システムによって異なります。chusr コマンドを実行すると、selang を使用して変更できるネイティブ プロパティが一覧表示されます。

カスタム クラス

このセクションでは、ユーザ定義のクラスとプロパティについて説明します。

ユーザ定義クラス

ユーザ定義クラスの各レコードは、必要に応じて独自に作成したクラスへのアクセスを定義します。ユーザ定義のクラス名に関する唯一の制限は、すべて大文字の名前を指定できないことです。

ユーザ定義クラスのレコードのキーは、レコードの名前です。

Unicenter TNG ユーザ定義クラス

CA Access Control では、Unicenter TNG アセット クラスをリソースとして定義できます。Unicenter TNG のユーザ定義クラスは、作成、削除、アクティブ化、無効化が可能です。

Unicenter TNG のユーザ定義クラスは UACC クラスにあります。

注: 標準の CA Access Control クラスに定義される任意のプロパティをユーザ定義クラスに使用できます。

付録 A: Windows の値

このセクションには、以下のトピックが含まれています。

Windows のファイル属性

chfile コマンド、editfile コマンド、または newfile コマンドを使用して、ファイルに属性を割り当てることができます。属性によってファイルの特性が決まります。

注: これらのファイル属性のフル ネームは FILE_ATTRIBUTE_*name* ですが、CA Access Control で入力する必要があるのは *name* の部分（ARCHIVE や COMPRESSED など）のみです。

Windows で変更可能なファイル属性の一覧とその説明を以下に示します。

FILE_ATTRIBUTE_ARCHIVE

バックアップ対象または削除対象としてマークされたアーカイブ ファイル。

FILE_ATTRIBUTE_HIDDEN

隠しファイル。通常、隠しファイルは標準ディレクトリの内容一覧に含まれません。

FILE_ATTRIBUTE_NORMAL

他の属性がないファイル。この値は、単独で使用した場合にのみ有効です。

FILE_ATTRIBUTE_READONLY

読み取り専用ファイル。読み取り専用ファイルは、アプリケーションで読み取りはできますが、書き込みまたは削除はできません。

FILE_ATTRIBUTE_SYSTEM

オペレーティング システム ファイルまたはオペレーティング システムのみが使用するファイル。

FILE_ATTRIBUTE_TEMPORARY

一時的な保存に使用されているファイル。

Windows で変更できないファイル属性の一覧とその説明を以下に示します。

FILE_ATTRIBUTE_COMPRESSED

圧縮ファイルまたは圧縮ディレクトリ。ファイルの場合は、ファイル内のすべてのデータが圧縮されていることを示します。ディレクトリの場合は、新規に作成されたすべてのファイルおよびサブディレクトリがデフォルトで圧縮されていることを示します。

FILE_ATTRIBUTE_DIRECTORY

ディレクトリ。

詳細情報:

chfile コマンド - Windows ファイル設定の変更 (P. 212)

Windows のアカウント フラグフラグ

chusr コマンド、editusr コマンド、および newusr コマンドを使用すると、ユーザのアカウントにフラグを割り当てることによって、アカウントの特定の属性を指定できます。各アカウントに複数のフラグを適用できます。

注: CA Access Control では、フラグの完全名を入力する必要はありません。次の表に示すショートカットを使用できます。

Windows で使用できるアカウント フラグは以下のとおりです。

ショートカット	フラグ	説明
blank	UF_PASSWRD_NOTREQD	ユーザのアカウントにパスワードが不要であることを示します。
cant_change	UF_PASSWORD_CANT_CHANGE	アカウントのパスワードをユーザが変更できないことを示します。
disable	UF_ACCOUNTDISABLE	ユーザのアカウントが無効であることを示します。

ショートカット	フラグ	説明
dont_expire	UF_DONT_EXPIRE_PASSWORD	このアカウントのパスワードが有効期限切れにならないことを示します。
homedir	UF_HOMEDIR_REQUIRED	ホーム ディレクトリが必要なことを示します。Windows ではこの値は無視されます。
interdomain	UF_INTERDOMAIN_TRUST_ACCOUNT	アカウントを信頼するための許可を示します。
lockout	UF_LOCKOUT	ユーザのアカウントが現在ロックアウトされていることを示します。ロックアウトを解除するには、このフラグを削除します。
normal	UF_NORMAL_ACCOUNT	通常のユーザを表すデフォルトのアカウント タイプを示します。
notreq	UF_PASSWRD_NOTREQD	ユーザのアカウントにパスワードが不要であることを示します。
protect	UF_PASSWORD_CANT_CHANGE	アカウントのパスワードをユーザが変更できないことを示します。
script	UF_SCRIPT	ユーザがアプリケーションを起動したときに、ディスク マッピングを実行するログイン スクリプトがアクティブになることを示します。LAN Manager 2.0 または Windows では、このフラグを設定する必要があります。
server	UF_SERVER_TRUST_ACCOUNT	このドメイン内の Windows NT バックアップ ドメイン コントローラのアカウントを示します。
temp	UF_TEMP_DUPLICATE_ACCOUNT	他のドメインにアカウントを持つユーザを示します。このアカウントに対して、そのドメインへのアクセス権を与えますが、このアカウントは信頼できるアカウントではありません。
trust	UF_INTERDOMAIN_TRUST_ACCOUNT	アカウントを信頼するための許可を示します。

ショートカット	フラグ	説明
workstation	UF_WORKSTATION_TRUST_ACCOUNT	このドメインのメンバであるワークステーションまたはサーバのアカウントを示します。

詳細情報:

chusr コマンド - Windows ユーザの変更 (P. 219)

Windows のアクセス許可

SHARE リソース タイプでは、アクセサに対してアクセス許可を与えることができます。

Windows で使用できるアクセス許可は以下のとおりです。

ACCESS_ALL

リソースの読み取り、書き込み、作成、実行、および削除ができ、リソースの属性およびアクセス許可を変更できます。

ACCESS_ATTRIB

リソースの属性を変更できます。

ACCESS_CREATE

リソースを作成できます(作成時にデータを書き込む許可を含む)。

ACCESS_DELETE

リソースを削除できます。

ACCESS_EXEC

リソースを実行できます。

ACCESS_NONE

アクセス許可がありません。

ACCESS_PERM

ユーザまたはアプリケーションに割り当てられたリソースに対するアクセス許可を変更できます。

ACCESS_READ

リソースのデータを読み取り、デフォルトで実行できます。

ACCESS_WRITE

リソースへのデータの書き込みができます。

詳細情報:

SHARE クラス (P. 551)

Windows の権限

Windows の権限は、個々のユーザ アカウントおよびグループに割り当てることができます。管理者は、chusr コマンドまたは editusr コマンドを使用してユーザに、chgrp コマンドまたは editgrp コマンドを使用してグループに、それぞれ権限を割り当てることができます。グループに追加されたユーザには、そのグループに割り当てられたすべての権限が自動的に与えられます。

一覧に示されているとおりの権限名 (ユーザ権限名) を使用できます。または名前の先頭に Se を、最後に Privilege を追加することもできます (BatchLogon、InteractiveLogon、NetworkLogon、および ServiceLogon は例外で、Privilege の代わりに Right を追加します)。

Windows で使用できる権限は以下のとおりです。

権限	デフォルトの割り当て	説明
AssignPrimaryToken	なし	プロセスのセキュリティ アクセストークンの変更をユーザに許可します。
監査	なし	セキュリティ監査を生成します。
Backup	Administrators Backup Operators	ファイルおよびディレクトリのバックアップをユーザに許可します。この権限はすべてのファイル許可およびディレクトリ許可を置き換えます。
BatchLogon	なし	バッチ ジョブとしてのログオンをユーザに許可します。

権限	デフォルトの割り当て	説明
ChangeNotify	Everyone	通常、ファイルおよびサブディレクトリへのアクセス権は、上位から下位に向かって設定されます。つまり、ある特定のディレクトリへのアクセス権がないユーザは、そのディレクトリの下にあるサブディレクトリへのアクセス権も持ちません。しかし、この権限を使用すると、ユーザは親ディレクトリへのアクセス権がない場合でも、サブディレクトリにアクセスできます。
CreatePagefile	なし	ページ ファイルの作成をユーザに許可します。セキュリティは、次のキーに対するユーザのアクセス権によって決定されます。 ¥CurrentControlSet¥Control¥SessionManagement
CreatePermanent	なし	¥¥Device などの特別で永続的なオブジェクトの作成をユーザに許可します。
CreateToken	なし	トークン オブジェクトを作成します。これを実行できるのは Local Security Authority のみです。Local Security Authority は、ユーザがシステムへのアクセスを許可されていることを確認します。この権限の使用を監査することはできません。C2 レベルの認証については、この権限をどのユーザにも割り当てないことをお勧めします。
Debug	管理者	スレッドなどのプログラムまたはオブジェクトをデバッグします。この権限を監査することはできません。C2 レベルの認証については、システム管理者を含めてどのユーザにもこの権限を割り当てないことをお勧めします。
IncreaseBasePriority	Administrators Power Users	プロセスの実行優先順位を上げることをユーザに許可します。
IncreaseQuota	なし	オブジェクトのクォータを増やすことをユーザに許可します。
InteractiveLogon	Most groups	対話形式のログインをユーザに許可します。
LoadDriver	管理者	デバイスドライバのインストールおよび削除をユーザに許可します。

権限	デフォルトの割り当て	説明
LockMemory	なし	コンピュータのメモリにページをロックし、PAGEFILE.SYS などのバッキング ストア ファイルにページが自動的にバックアップされないようにすることをユーザに許可します。
MachineAccount	なし	ドメインに新しいマシンを追加することをユーザに許可します。
NetworkLogon	Everyone	ユーザがネットワークのどこからでもコンピュータに接続することを許可します。したがって、ユーザは、コンピュータにログオンするために特定の場所または特定の端末を使用する必要がありません。
ProfileSingleProcess	Administrators Power Users	ある 1 つのプロセスのパフォーマンスを監視するためにパフォーマンス監視ツールを使用することをユーザに許可します。
RemoteShutdownPrivilege	Administrators Power Users	Windows システムのリモートでの停止をユーザに許可します。
Restore	Administrators Backup Operators	バックアップされたファイルおよびディレクトリのリストアをユーザに許可します。この権限はすべてのファイルおよびディレクトリのアクセス権を置き換えます。
Security	管理者	監査の対象とするリソース アクセス権の種類(ファイル アクセス権など)を指定すること、またセキュリティ ログを表示および消去することをユーザに許可します。 **注**: この権限は、Windows のユーザー マネージャで[原則]メニューの[監査]コマンドを使用してシステム監査ポリシーを設定することをユーザに許可するものではありません。管理者にはセキュリティ ログを表示および消去する権限が常に与えられます。
ServiceLogon	なし	プロセスをサービスとしてシステムに登録できるようにします。
Shutdown	Administrators Backup Operators Everyone Power Users ユーザ	システム コンソールからのシステム停止をユーザに許可します。

権限	デフォルトの割り当て	説明
SystemEnvironment	管理者	システム環境変数の変更をユーザに許可します。ユーザは各自のワークステーションでシステム環境を設定できます。また、同じワークステーションで作業する他のすべてのユーザが確実に同じ設定を使用できます。
SystemProfile	管理者	システムに対するプロファイリング（パフォーマンスのサンプリング）の実行をユーザに許可します。
SystemTime	Administrators Power Users	コンピュータの内部時計の時間設定をユーザに許可します。
TakeOwnership	管理者	ファイル、ディレクトリ、プリンタ、およびコンピュータ上のその他のオブジェクトの所有者になることをユーザに許可します。この権限は、オブジェクトを保護するすべての許可を置き換えます。
Tcb	なし	オペレーティング システムで、安全で信頼できる部分としてプロセスを実行できるようにします。いくつかのサブシステムにこの権限が与えられます。

詳細情報:

chusr コマンド - Windows ユーザの変更 (P. 219)
chgrp コマンド - Windows グループの変更 (P. 213)