ORACLE
ZFS STORAGE APPLIANCE

異機種環境における Oracle ZFS Storage Appliance のアクセス制御リスト機能の使用

Oracle テクニカル・ホワイト・ペーパー | 2016 年 8 月

ORACLE

目次

概要

Oracle ZFS Storage Appliance は、高度なアーキテクチャを持つハードウェアとソフトウェアを統合して、動作の軽快なマルチプロトコル・ストレージ・システムを実現しています。さまざまなアプリケーションやデータ・サービスの同時動作のような、要求の厳しいワークロードの処理が可能であり、業界標準の Storage Performance Council ベンチマークテスト（SPC-1、SPC-2、SPECsfs など）の結果により、最上級のパフォーマンス特性が実証されています。

Oracle ZFS Storage Appliance は、優れたネットワーク接続ストレージ（NAS）プラットフォームを提供する統合ストレージ・ソリューションです。NAS プラットフォームは、Active Directory 環境での Server Message Block（SMB）ファイル・ストレージとして Microsoft Windows クライアントから利用可能であると同時に、同じディレクトリやファイルに Oracle Solaris や Oracle Linux からアクセスすることもできます。

このホワイト・ペーパーでは、Network File System（NFS）プロトコルおよび SMB プロトコルでファイル・システムを共有する際の、Portable Operating System Interface（POSIX）ベースのオペレーティング・システム（Oracle Solaris など）と Windows 双方のアクセス制御リスト（ACL）間の連携について説明します。

はじめに

マルチユーザー・コンピュータ・システムが生まれた当初から、ファイルおよびディレクトリを複数のユーザーで共有するというニーズに加え、おそらくさらに重要なこととして、共有対象以外のファイルおよびディレクトリは他の誰とも共有しないというニーズがありました。初期の UNIX 環境では、ユーザーは UID（ユーザー識別子）という小さな整数および GID（グループ識別子）という大きなグループ・メンバーシップで内部的に識別されていました。ユーザーは、複数のグループのメンバーになることができます。たとえば、開発者は、取り組んでいるプロジェクトに該当する各グループのメンバーになることができます。

ファイルおよびディレクトリへのアクセスを許可することには、メリットとデメリットがあります。したがって、プライバシーを守るためにファイルおよびディレクトリへのアクセスを禁止するアクセス権の仕組みも必要だったのです。

初期の UNIX システムでは、この機能はファイルの所有者、グループ、その他すべてのユーザーのアクセス制御を行うための 3 つのビットから構成されるファイル・アクセス・マスクを使用して実現されていました。ユーザーは、複数のグループのメンバーになることはできますが、ファイルが所属できるグループは 1 つだけです。

アクセス権は、読取り、書込み、実行の各ビットの組み合わせで定義されていました。実行ビットの意味は、状況によって異なります。つまり、実行ビットがファイルに付加された場合は、そのファイルが実行可能という意味になります。ファイルがスクリプトやバイナリ実行可能ファイルの場合に該当します。実行ビットがディレクトリに付加された場合は、検索（再帰検索）可能という意味です。

次に、ディレクトリ一覧の表示例を示します。

```
$ ls -l
total 72
drwxr-xr-x  2 andrew  unixusers          68 10 Aug 10:49 directory1
-rw-r--r--  1 andrew  unixusers         270 10 Aug 10:49 jokes.txt
-rw-r-----  1 andrew  developergroup  20480 10 Aug  10:49 jokes-about-other-groups.txt
-rw-------  1 andrew  unixusers        9255 10 Aug 10:49 my-secret-santa-list.txt
drwxr-----  2 andrew  blackopsgroup      68 10 Aug 10:49 reallysecretdirectory
drwxrwx---  2 andrew  developergroup     68 10 Aug 10:49 secretdirectory
```

アクセス権セットが各行の最初のエントリに表示され、このセットの最初の文字がファイル・タイプを表します。「-」は普通のファイルを表し、「d」はディレクトリを表します。この文字列は、次の図に示すように分かれます。

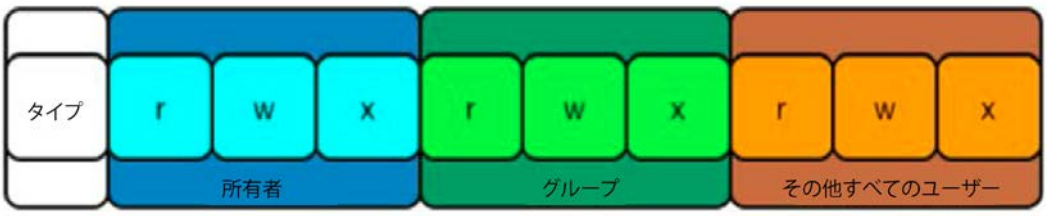

図1：初期のUNIXでのアクセス権

これらのアクセス権の意味を、次の表に示します。

表1：RWX方式のアクセス権ビット

アクセス権	表記	バイナリ表記	8進数表記
アクセス不可	`---`	000	0
読取り専用	`r--`	100	4
書込み専用	`-w-`	010	2
実行専用	`--x`	001	1
読取りと書込み	`rw-`	110	6
読取りと実行	`r-x`	101	5
書込みと実行	`-wx`	011	3
読取りと書込みと実行	`rwx`	111	7

上記のアクセス権の中には、実際の環境では無意味なものもありますが（「書込みと実行」など）、アクセス権モードとしては有効です。

ファイルのアクセス権セット（またはアクセス権モード）は、ユーザー、グループ、その他すべてのユーザーに対するアクセス権の組み合わせで、通常は 3 桁の 8 進数表記で指定します。

```
-rw-r--r-- 1 andrew unixusers 270 10 Aug 10:49 jokes.txt
```

上記の例では、ファイルのモードは 644 なので、所有者は読取りと書込みが可能で、グループとその他すべてのユーザーは、読取りのみが可能です。

次の例は、ユーザー`andrew` はこのファイルを読み書き可能で、`other` グループ・メンバーは読取りのみが可能であることを示しています。その他すべてのユーザーは、このファイルへのアクセス権がありません。

```
-rw-r----- 1 andrew unixusers 20480 10 Aug 10:49 jokes-about-other-groups.txt
```

setuid ビットのような付加ビットをアクセス権マスクに追加すると、実行可能ファイルを実行するユーザーが、実行中だけ一時的にそのファイルの所有者の UID を取得できます。setgid ビットも同様に実装されており、ユーザーは実行中だけ GID を取得できます。通常、setuid ビットや setgid ビットが必要になるのは、一時的に特別なアクセスを必要とするタスクや、特権昇格を必要とするタスクです。

繰り返しますが、アクセス権ビットの働きは適用する状況で決まります。setuid および setgid アクセス権はディレクトリに適用できます。setgid をディレクトリに使用すると、そのディレクトリで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そのディレクトリの GID を継承します。ディレクトリに setuid ビットを設定することも可能ですが、このビットは無視されます。ディレクトリに設定しても意味がないからです。

ニーズの高まりに応じてさらに新たなビットも追加されました。たとえば、スティッキー・ビットを設定すると、あるディレクトリ配下のファイルまたはサブディレクトリの所有者のみがファイルを削除できます。スティッキー・ビットをファイルに設定した場合、使用しているオペレーティング・システムのバージョンによってその意味は異なりますが、一般的に、普通のユーザー・ファイルには使用されませんでした。

この方式は、初期のマルチユーザー・アクセスには適していました。というのも、ホストごとのユーザー数が比較的少なく、分散ファイル・システムやネットワーク接続ストレージ・システムも登場して間がなかったからです。オペレーティング・システムやサーバーが進化、成長するにつれて、「ユーザー/グループ/その他すべてのユーザー」形式のアクセス権という制約の多い環境で複雑な協働プロジェクトを管理するのはますます困難になりました。

そこで、UNIX システムに ACL が実装され、複雑なプロジェクトをより実用的な方法で表せるようになりました。この頃、Windows システムのネットワーク対応も大幅に進化したため、Windows システムにおけるプロジェクトやユーザーの相互関連を表すアクセス権のシステムが必要になりました。

ACLの管理

ACL には、指定されたユーザーまたはグループが、ACL の適用されたオブジェクトに対してできる操作を示す、個別のアクセス権の定義が保持されています。

次の ACL の基本的な定義と原則は、ACL を理解するうえで重要です。

» アクセス制御リスト（ACL）は、0 個以上のアクセス制御エントリ（ACE）の集まりです。

» アクセス制御エントリ（ACE）には、アクセス権および ID（たとえば、ユーザーやグループ）、またはアクセス権を適用するトラスティ、さらにオプションで、ACE のタイプを指定するコンテキスト情報とその継承ステータスが設定されます。

» セキュリティ設定が可能なオブジェクトは、ACL が適用されるエンティティであり、通常はファイルまたはディレクトリです。

» ユーザーまたはグループの識別方法は、アクセス制御されている、セキュリティ設定が可能なオブジェクトを含むファイル・システムによって異なります。

オラクルの ZFS テクノロジーによって、アクセス権ビット（rwx 方式）の派生元である ACL がすべてのファイルをディレクトリに設定されます。これは、ファイルまたはディレクトリが ACL またはアクセス権ビットを設定する UNIX ファイル・システムに相当します。

ネットワーク・ファイル・システム・バージョン 4（NFSv4）プロトコルに採用された ACL モデルは、Microsoft Windows の ACL モデルと同等であり、両方とも豊富なアクセス権セットと継承管理をサポートしています。どちらのモデルも、ACE でアクセス権やアクセス・タイプ、継承フラグ、およびこれらの値を適用するエンティティを指定します。POSIX の世界で一般に理解されている名前のセットである、`owner@`、`group@`、および `everyone@`は、ファイルの所有者、グループの所有者、その他すべてのユーザーをそれぞれ表します。

`everyone@`エントリは、他の POSIX クラスと同じではありません。このエントリに含まれるのは、所有者を含むすべてのユーザーです。POSIX の `other` クラスのように、所有者を除くその他すべてのユーザーではありません。

次の表に、NFSv4 のアクセス権限の一覧を示します。

表2：NFSv4でのアクセス権

アクセス権	説明
read_data	ファイルのデータを読み取る権限。
list_data	ディレクトリの内容を一覧表示する権限。
write_data	ファイルのオフセット範囲ならどこでもファイルのデータを変更できる権限。これには、ファイルを拡張する権限、または任意のオフセットへの書込み権限が含まれます。
add_file	ディレクトリに新しいファイルを追加する権限。
append_data	データを変更する権限、ただしファイルの最後（EOF）への追加に限定。
add_subdirectory	ディレクトリにサブディレクトリを作成する権限。
read_xattr	ファイルの拡張属性を読み取る権限、または拡張属性ディレクトリの検索を実行する権限。
write_xattr	拡張属性を作成する権限、または拡張属性ディレクトリに書き込む権限。
Execute	ファイルを実行する権限。
delete_child	ディレクトリ内のファイルを削除する権限。
read_attributes	ファイルの基本属性（ACL 以外）を読み取る権限。
write_attributes	ファイルまたはディレクトリに関連付けられたタイムスタンプを任意の値に変更する権限。
Delete	ファイルを削除する権限。
read_acl	ACL を読み取る権限。
write_acl	ファイルの ACL に書き込む権限。
write_owner	所有者を変更する権限、すなわち chown (1)または chgrp (1)コマンドを実行する権限。
Synchronize	サーバーにあるファイルにローカルでアクセスして同期的に読み書きする権限。

次の表に示す継承フラグは、上記のアクセス権限の継承を制御します。

表3：NFSV4での継承フラグ

フラグ	説明
file_inherit	作成されたファイル（ディレクトリ以外）ごとに ACE が付与されます。
dir_inherit	新しく作成されたサブディレクトリごとに ACE が付与されます。
inherit_only	ディレクトリに設定すると、そのディレクトリ自体には適用されず、新しく作成されたファイルとサブディレクトリにのみ適用されます。このフラグを使用する場合は、何を継承するかを指定するために、file_inherit または dir_inherit を指定する必要があります。
no_propagate	ディレクトリにこのフラグが設定された場合、ACL エントリはそのディレクトリに含まれるフォルダおよびファイルに対してのみ継承され、その配下のサブフォルダや、サブフォルダに含まれるファイルには継承されないことを示します。

Windowsオペレーティング・システムでのACLの使用

Windows 環境では、次の表に示すとおり、主に 3 つの ACE タイプが採用されています。

表4：WINDOWS環境でのアクセス制御エントリのタイプ

タイプ	使用例	説明
随意 ACE（DACE）	Access-Denied	セキュリティ設定が可能なオブジェクトへのアクセスを拒否
	Access-Allowed	セキュリティ設定が可能なオブジェクトへのアクセスを許可
システム ACE（SACE）	System Audit	トラスティがアクセス権の実行を試みた場合に、システム監査を生成します。

注：これらの主要なタイプに加えて、オブジェクト固有の ACE もあります。ただし、サポート対象はディレクトリ・サービス・オブジェクトのみであるため、このドキュメントでは取り上げません。

ACL は、ACE のリストで構成され、セキュリティ設定が可能な特定のオブジェクトへのアクセスを要求するプロセスに対し、そのアクセスを許可するかどうかを決定する重要な命令が含まれています。セキュリティ設定が可能なオブジェクトへのアクセスをプロセスが要求すると、アクセスを許可または拒否する ACE が現れるまで、ACL のエントリが順番に調べられます。

随意 ACE（別名 DACE）では、セキュリティ設定が可能なオブジェクトに対して単なるこうした許可または拒否のユーザー権限しか設定されません。Windows Server 2003 の登場とともに、ACL の自動継承機能が導入され、この仕組みは中断されました。

評価の順番は次のとおりです。

1. 明示的な随意 ACE が、グループの継承 ACE の前に配置されます。
2. 継承 ACE は継承順に配置されます。親から継承された ACE は、親の親から継承された ACE の前に配置されます。以降も同様です。
3. 各グループの中で、アクセス拒否 ACE は、アクセス許可 ACE の前に配置されます。
4. すべての ACE を検索しても合致するエントリがなかった場合は、アクセスが拒否されます。

次の図は、上から下へと順に示したものです。プロセス、またはプロセスが属するグループの ID に合致するエントリが見つかると、検索が停止して、その随意 ACE がアクセス権として返されます。

図2：ACLにおけるDACEの順番

オブジェクトへのすべてのアクセスが許可される、NULL 随意 ACL（DACL）を作成することが可能です。これは、ACE を 1 つも含まない ACL である空の DACL とは対照的です。空の DACL は、前述のルール 4 に記載のとおり、すべてのアクセスを拒否します。

ACLの設定

Microsoft は、生成される ACL が意味的にもコンテキスト的にも正しくなるように、標準ツールを使って Windows 環境で ACL を管理することを強く推奨しています。可能な場合、低い権限レベルでも ACL を操作できるツールが数多く提供されていますが、注意して使わないと、セキュリティ・モデルに損害をもたらすおそれがあります。継承 ACL を使用する場合はなおさらです。Windows エクスプローラは、Windows で ACL の表示や変更を行うのにもっとも手軽なツールです。

前述のディレクトリー覧表示例と同様に、ファイルおよびディレクトリへのアクセスを許可したり拒否したりするために、次のアクセス権限が設定されています。

```
drwxr-xr-x  2 andrew  unixusers          68 10 Aug 10:49 directory1
-rw-r--r--  1 andrew  unixusers         270 10 Aug 10:49 jokes.txt
-rw-r-----  1 andrew  developergroup  20480 10 Aug 10:49 jokes-about-other-groups.txt
-rw-------  1 andrew  unixusers        9255 10 Aug 10:49 my-secret-santa-list.txt
drwxr-----  2 andrew  blackopsgroup      68 10 Aug 10:49 reallysecretdirectory
drwxrwx---  2 andrew  developergroup     68 10 Aug 10:49 secretdirectory
```

次の表は、前述のコマンドラインで設定された許可アクセスを示しています。

表5：ユーザーまたはグループに対してファイルまたはフォルダごとに許可されたアクセス

ファイル/フォルダ	ユーザー/グループ	アクセス
directory1	andrew	フル
	他のすべて	読取り/検索
jokes.txt	andrew	フル
	他のすべて	読取り/検索
jokes-about-other-groups.txt	andrew	フル
	developergroup	読取り専用
my-secret-santa-list.txt	andrew	フル
	他のすべて	アクセス拒否
reallysecretdirectory	andrew	フル
	blackopsgroup	読取り専用
secretdirectory	andrew	フル
	developergroup	フル
	auditgroup	読取り専用

これらのアクセス権は比較的単純なものですが、もっと複雑なアクセス権セットの設定に必要な手順もまったく同じです。

次の例（`reallysecretdirectory` と `my-secret-santa-list.txt`）で、これらのアクセス権の設定方法を説明します。

最初に、個別の所有権およびグループ・アクセス権の設定を必要とする項目の継承オプションをオフにします。もっとも制限の厳しいディレクトリの例として、`reallysecretdirectory` を次の図に示します。

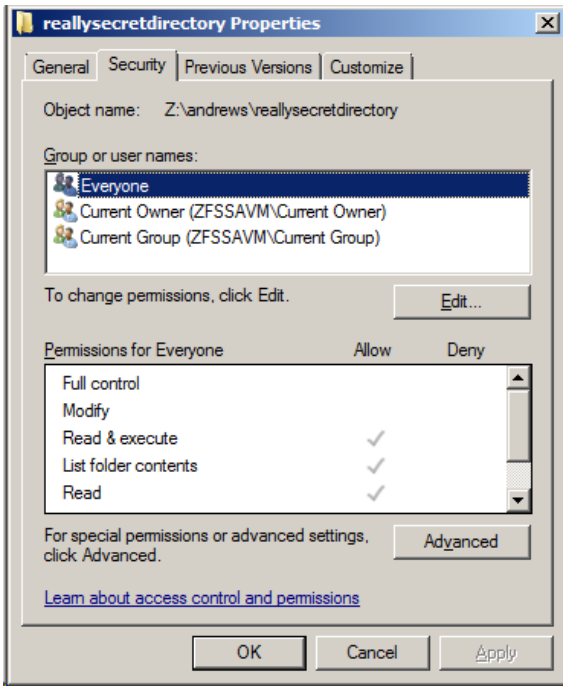

図3：reallysecretdirectoryのアクセス権の初期設定

「**詳細設定**」をクリックして、既存の継承可能なアクセス権（次の図）を表示して、アクセス権を変更します。

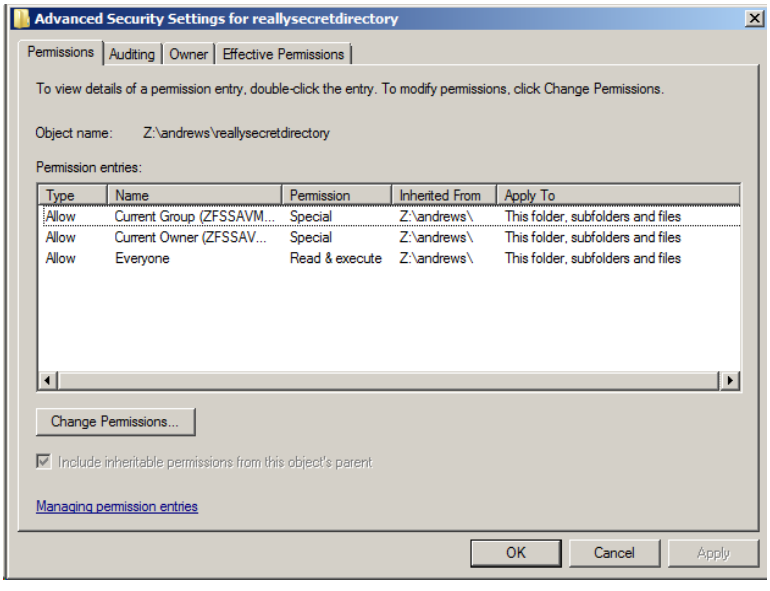

図4：継承の詳細表示画面

「**アクセス許可の変更**」をクリックすると、アクセス権の追加と削除のオプションを含む、以下のような「セキュリティの詳細設定」画面が表示されます。

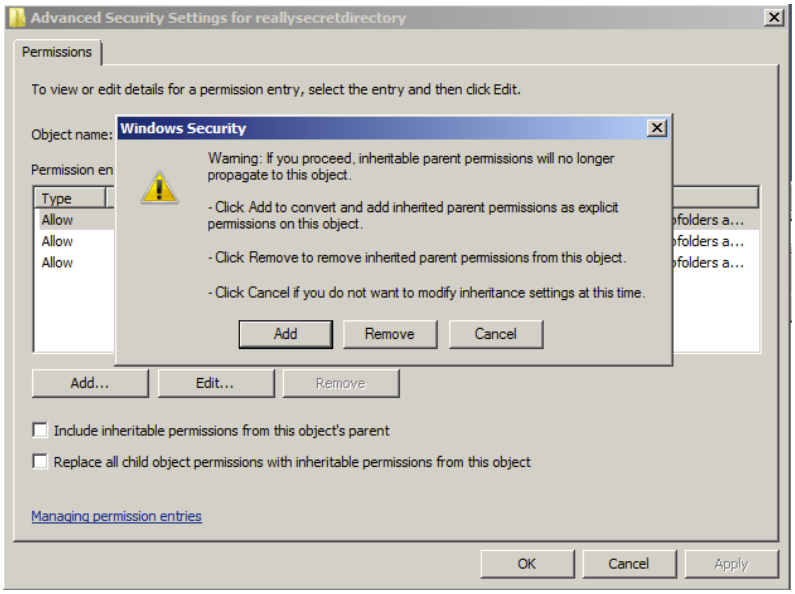

図5：Windowsエクスプローラのアクセス権変更画面

アクセス権を削除するには、「**削除**」をクリックします。後で、継承されたアクセス権を権限リストに追加して編集できるように、「**このオブジェクトの親からの継承可能なアクセス許可を含める**」チェックボックスをオフにします。これにより、継承されたアクセス権が効果的に明示されます。

親から継承したアクセス権を削除することで、セキュリティ設定が可能なオブジェクトにアクセス権を一から追加できます。この例では、アクセス権の制限が非常に厳しいため、「このオブジェクトの親からの継承可能なアクセス許可を含める」チェックボックスをオフにした後、次のようなプロパティ・パネルが表示されます。

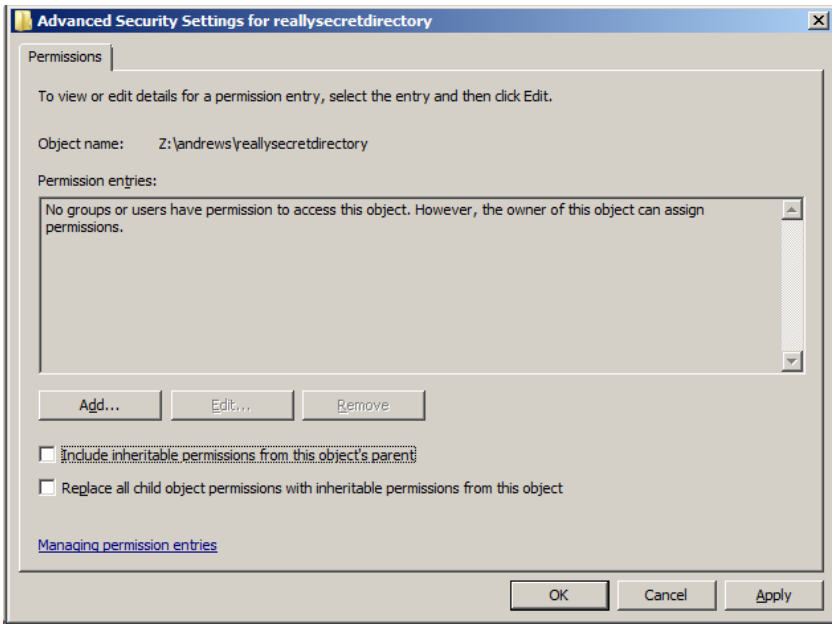

図6：セキュリティの詳細設定画面（権限が設定されていない状態）

アクセス権を追加するには、「セキュリティの詳細設定」画面で「**追加**」をクリックします。Windows エクスプローラの標準的なダイアログ・ボックスが以下のように表示され、これを使用してユーザーやグループを権限リストに追加できます。

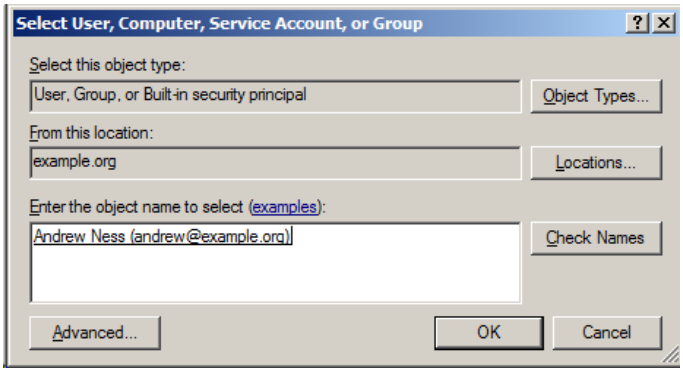

図7：権限リストにユーザーを追加

次の図に、選択したオブジェクト `andrew@example.org` の権限オプションを示します。

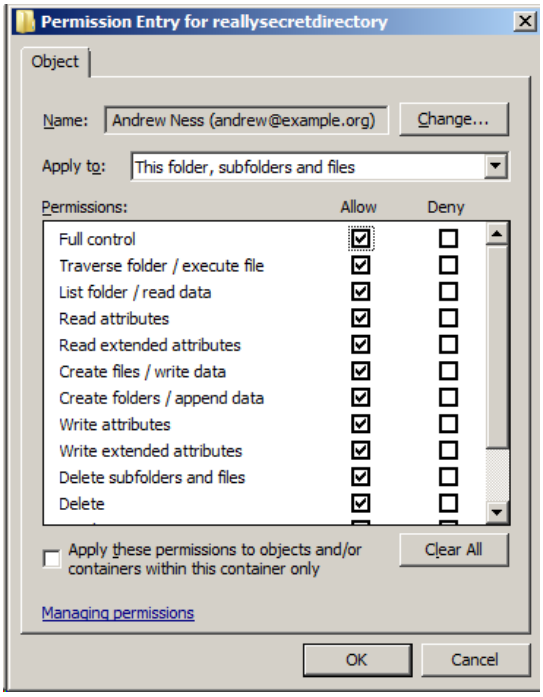

図8：ユーザー`andrew`にフルコントロールを許可

同様に、グループ `blackopsgroup` に読取り専用権限を設定すると、許可されたアクセス権は次のように表示されます。

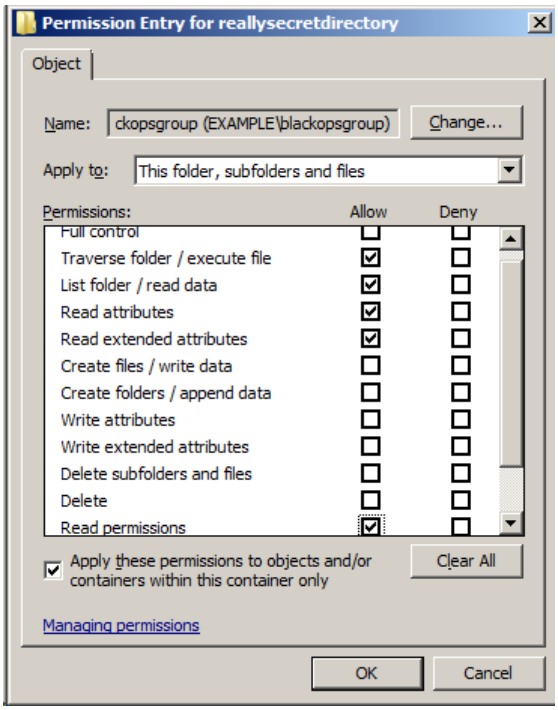

図9：グループ**blackopsgroup**に読取り専用権限を設定

「**適用**」と「**OK**」をクリックすると、最初のアクセス権プロパティ・ダイアログに、次に示すような新しいアクセス権の設定内容が表示されます。

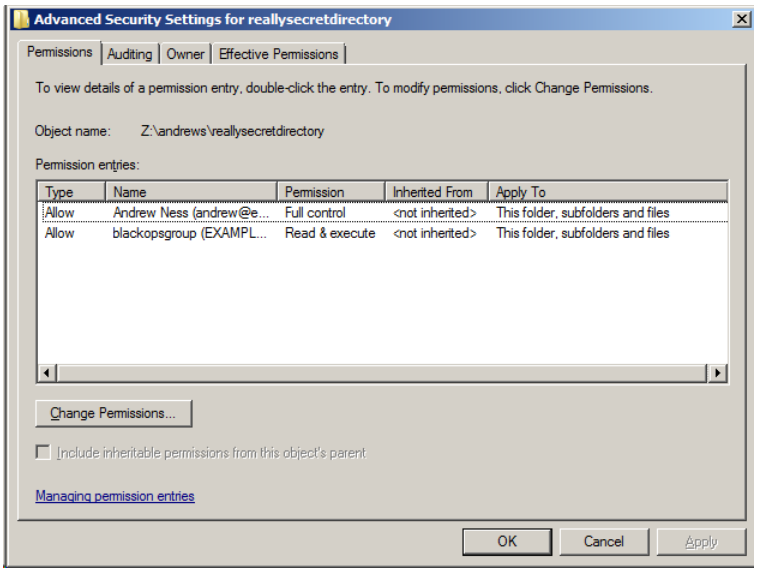

図10：指定どおりに設定されたアクセス権

Oracle ZFS Storage Appliance では、セキュリティと権限に関する統合モデルが採用されています。個々のオペレーティング環境（OE）すべてで、複数の OE 間での一貫した認証を統合モデルで実現するための必要条件として、Lightweight Directory Access Protocol（LDAP）、Active Directory、ネットワーク情報サービス（NIS）といったその環境にふさわしい共通ディレクトリ・サービスを共有する必要があります。

Windows アクティブ ドメイン・コントローラは、複数の POSIX OE に LDAP サービスを提供し、それにより信頼できる一元的な ID ソースを提供するよう構成できます。Microsoft Server 2008 R2 以前のバージョンで、Active Directory を構成する方法については、次のホワイト・ペーパーを参照してください。

http://www.oracle.com/technetwork/jp/server-storage/sun-unified-storage/documentation/activedir-ldap-source-1124-2372474-ja.pdf

Microsoft Server 2012 以降には、POSIX システムに RFC2307 の LDAP 機能を提供するために必要な UNIX 用 ID 管理（IDMU）ソフトウェアが搭載されていませんが、IDMU に代わる無料のソフトウェアや市販品の選択肢もいくつかあります。

Oracle SolarisのACLとWindows Active Directoryの統合

Oracle Solaris は、標準搭載の LDAP クライアント・インタフェースを使用して、Windows Active Directory LDAP サーバーのクライアントになるように構成できます。

このドキュメントに記載した例では、ドメインは `example.org` です。Oracle Solaris には、`ldapclient(1M)`コマンドが用意されており、必要な構成ファイルをすべて設定できます。

Windows Active Directory を Oracle Solaris と統合するために必要なコマンドの詳細は、多くのホワイト・ペーパーや My Oracle Support Note に記載されているため、操作の詳細はここでは説明しません。

ユーザー名を `ldap-proxy` にして非特権 LDAP プロキシ・ユーザーを構成する場合、前述の例について LDAP アクセスを有効化するコマンドは次のようになります。

```
ldapclient -a defaultSearchBase=dc=example,dc=org \
     -a defaultServerList=<IP-ADDRESS>:389 \
     -a proxyDN=cn=ldap-proxy,cn=users,dc=example,dc=org \
     -a proxyPassword=<ldap-proxy Password> \
     -a domainName=example \
     -a defaultSearchScope=sub \
     -a objectclassMap=passwd:posixAccount=user \
     -a objectclassMap=shadow:posixAccount=user \
     -a objectclassMap=group:posixGroup=group \
     -a enableShadowUpdate=true \
     -a adminDN=cn=ldap-admin,cn=users,dc=example,dc=org \
     -a adminPassword=<ldap-admin Password> \
     -a followReferrals=true \
     -a serviceSearchDescriptor=passwd:cn=users,dc=example,dc=org \
     -a serviceSearchDescriptor=shadow:cn=users,dc=example,dc=org \
     -a serviceSearchDescriptor=group:cn=users,dc=example,dc=org \
```

```
    -a attributeMap=passwd:homeDirectory=unixHomeDirectory \
    -a attributeMap=shadow:homeDirectory=unixHomeDirectory \
    -a attributeMap=passwd:gecos=cn \
    -a attributeMap=shadow:gecos=cn \
    -a attributeMap=passwd:userPassword=unixUserPassword \
    -a attributeMap=shadow:userPassword=unixUserPassword \
  manual
```

想定される動作を検証するには、`ldaplist(1M)`コマンドだけを実行してください。

```
$ ldaplist
dn: CN=Administrator,CN=Users,DC=example,DC=org
dn: CN=Alaina Ashman,CN=Users,DC=example,DC=org
dn: CN=Alex Castillo,CN=Users,DC=example,DC=org
dn: CN=Alexander Mann,CN=Users,DC=example,DC=org
dn: CN=Allowed RODC Password Replication Group,CN=Users,DC=example,DC=org
dn: CN=Amber Brock,CN=Users,DC=example,DC=org
dn: CN=Andrew Ness,CN=Users,DC=example,DC=org
dn: CN=Angela Rogers,CN=Users,DC=example,DC=org
dn: CN=Anna Hunt,CN=Users,DC=example,DC=org
dn: CN=Anne Mathis,CN=Users,DC=example,DC=org
dn: CN=Annie Cooper,CN=Users,DC=example,DC=org
[…]
```

Oracle SolarisでのACLの確認

次の出力結果は、`ls(1)`コマンドに標準フラグを付けて実行し、Windows 環境で作成したデモ用ディレクトリを表示したものです。

```
  $ ls -al
  total 85
  drwxrwxr-x+ 5 andrew  unixusers           8  Dec  3 13:02 .
  drwxrwxr-x  4 andrew  developergroup          4 Nov 9 19:22 ..
  drwxrwxr-x+ 2 andrew  unixusers           2  Oct 29 13:56 directory1
  -rwx------+ 1 andrew  unixusers       20480  Aug 10 10:49 jokes-about-other-
                                               groups.txt
  -rwxrwxr-x+ 1 andrew  unixusers         270  Aug 10 10:49 jokes.txt
  -rwx------+ 1 andrew  unixusers        9255  Aug 10 10:49 my-secret-santa-
                                               list.txt
  drwx------+ 2 andrew  unixusers           2  Dec  3 13:02 reallysecretdirectory
  drwx------+ 2 andrew  unixusers           2  Oct 29 13:57 secretdirectory
```

所有者は想定どおり、ユーザー`andrew` です。ただし、グループは必ずしも想定どおりではありません。特定のグループ・アクセス権が付与されているからです。上記の例で重要な点は、従来型の `rwx` アクセス権の後に+記号が付いていることです。この記号は、ACL が設定されていることを示しています。

`v` または `V` フラグを `ls(1)`コマンドに付けると ACL が表示されます。`V` フラグでは、確認対象のオブジェクトに関連付けられている ACL 情報のみが表示されます。`-v` フラグでは、ACL 情報に加えて `rwx` 権限の解釈も表示されます。ACL を表示するために指定する文字の意味について詳しくは、

ls(1)コマンドのマニュアル・ページの記載を参照してください。

```
$ ls -dV secretdirectory
 drwx------+ 2 andrew   unixusers      2 Oct 29 13:57 secretdirectory/
           user:andrew:rwxpdDaARWcCos:fd-----:allow
   group:developergroup:rwxpdDaARWcCos:fd-----:allow
       group:auditgroup:r-x---a-R-c--s:fd-----:allow
```

上記の画面出力を Windows 環境と比較すると、情報は一致しているように見えます。次の図は、フル・アクセス権を持つ所有者 Andrew Ness (andrew@example.org)の権限を示しています。情報は、両方の環境で一致しています。

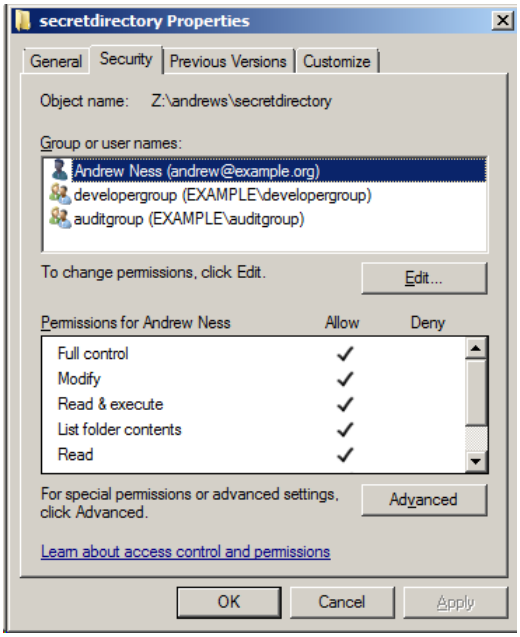

図11：フル・アクセス権を有する所有者のWindowsでの表示

次の図は、グループ`developergroup (EXAMPLE\developergroup)`のフル・アクセス権を示しています。

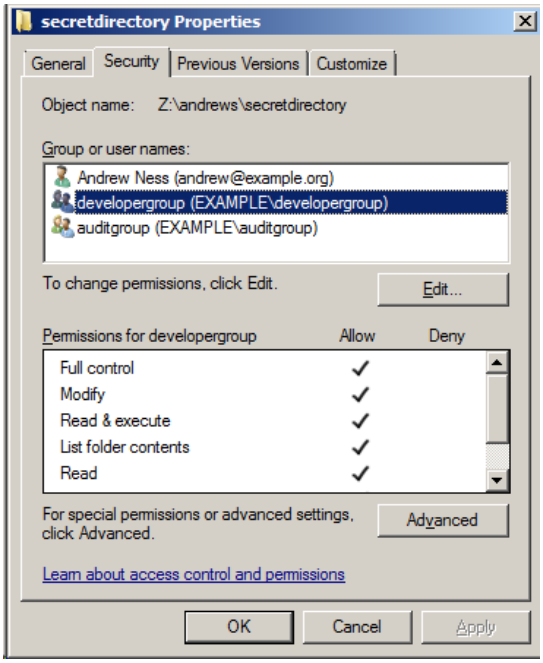

図12：フル・アクセス権を有するグループのWindowsでの表示

次の図は、`auditgroup (EXAMPLE\auditgroup)`の制限付きアクセス権を示しています。

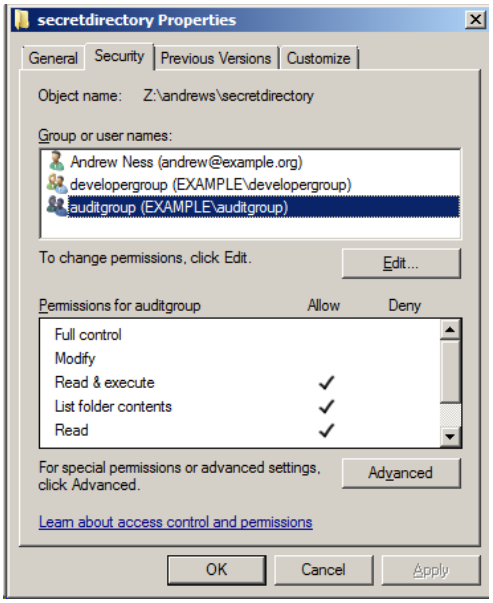

図13：読取り専用アクセス権を有するグループのWindowsでの表示

Oracle SolarisでのACLの変更

Oracle Solaris 環境で、chmod(1)コマンドを使って ACL を変更します。このコマンドは、ACL の操作に対応するように拡張されています。詳しくは、chmod(1)コマンドのマニュアル・ページを参照してください。

次のコード例に示すように、chmod(1)コマンドで A オプションを使って ACL の操作を指定します。

```
chmod [options] A[index]- file ...
chmod [options] A-acl_specification file ...
chmod [options] A[index]{+|=}acl_specification file ...
```

acl_specification の部分は、カンマ区切りのリストです。

インデックスが指定されていない場合は、A[index]+を使って ACE をファイルおよびディレクトリの ACL の前に付加することができます。インデックスが指定されている場合は、ACE index の前に配置されます。

注：ACL 内の ACE の順序が重要であることに留意してください。

A[index]-を使う場合、index を付ければ指定された ACE だけが削除され、付けなければ ACL 全体が削除されます。

A-acl_specification を使うと、acl_specification で指定された ACE が、現在のファイルの ACL 内に存在する場合は削除されます。

次の例では、新しいファイル todo-list を作成しています。このファイルにアクセスできるのは、所有者の andrew のみで、ファイルに書込みが可能です。グループ developergroup とグループ auditgroup は、ファイルの読取りのみが可能です。

```
$ touch todo-list
$ chmod A+user:andrew:rwx:allow,group:developergroup:r:allow,group:auditgroup:r:allow todo-list
$ ls -V todo-list
-rwxr--r--+ 1 andrew   unixusers      0 Jan 8 13:26 todo-list
          user:andrew:rwx-----------:-------:allow
   group:developergroup:r-------------:-------:allow
      group:auditgroup:r-------------:-------:allow
                owner@:rw-p--aARWcCos:-------:allow
                group@:r-----a-R-c--s:-------:allow
             everyone@:r-----a-R-c--s:-------:allow
```

グループ unixusers やその他すべてのユーザーは、標準的な rwx のセマンティックでは、ファイルの読取り権限を持つことに変わりはないことにご注意ください。

従来型の chmod g-r,a-r や chmod 600 を使うと、ACL は削除されますが、この場合望ましくありません。

こうした不要なアクセス権を削除するには、ls(1)コマンドに-v スイッチを使用して、削除する ACE のインデックスを特定します。

```
$ ls -v todo-list
```

```
-rwxr--r--+ 1 andrew   unixusers      0 Jan 8 13:43 todo-list
     0:user:andrew:read_data/write_data/execute:allow
     1:group:developergroup:read_data:allow
     2:group:auditgroup:read_data:allow
     3:owner@:read_data/write_data/append_data/read_xattr/write_xattr
          /read_attributes/write_attributes/read_acl/write_acl/write_owner
          /synchronize:allow
      4:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
      5:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
        :allow
$ chmod A5- todo-list
$ chmod A4- todo-list
$ ls –v todo-list
-rwx------+ 1 andrew   unixusers      0 Jan 8 13:43 todo-list
     0:user:andrew:read_data/write_data/execute:allow
     1:group:developergroup:read_data:allow
     2:group:auditgroup:read_data:allow
     3:owner@:read_data/write_data/append_data/read_xattr/write_xattr
          /read_attributes/write_attributes/read_acl/write_acl/write_owner
          /synchronize:allow
```

注：ACE のインデックスが更新されるのは、ACE を ACL に追加したか、ACL から削除したときです。そのため、逆索引順で ACE を削除するのがもっとも適切な方法です。前の ACE の削除によって、後の操作で削除される ACE の番号付けに影響しないようにするためです。

共有レベルACLの使用

共有レベル ACL は、Oracle ZFS Storage Appliance の共有に適用できます。共有レベル ACL として提供される ACL は、共有内のどのファイルおよびディレクトリの ACL とも組み合わせることができます。共有レベル ACL は、共有のマスターACL と考えられ、すべてのファイルまたはディレクトリの ACL を制限強化方向または許可拡大方向に拡張するための標準フレームワークを提供します。

デフォルトでは、共有レベル ACL は、すべてのユーザーにフルコントロール権限を付与します。そのため、アクセス権によって操作を許可したり拒否したりできるようにすることは、すべてファイルおよびディレクトリの ACL の役目です。共有レベル ACL は、SMB プロトコルを使って共有をエクスポートしない限り有効ではありません。

共有レベル ACL の使用例には、デフォルトでは、監査グループやセキュリティグループなどがあります。

結論

適切に共有ネーミング・サービスを構成すれば、Oracle ZFS Storage Appliance によって POSIX 環境と Windows 環境の間で、ファイルやディレクトリのセキュリティを確実に維持できます。つまり、ファイルやディレクトリのセキュリティが維持されるのは、NFS または SMB で共有する場合だということです。一方の環境に加えられた変更が、直ちに他方の環境に反映されるので、ユーザーと管理者はセキュリティの整合性を維持できます。

この統合型のセキュリティ手法のメリットは、一方の環境ではデータが保護されないままであることに気付かないのに、他方の環境ではきちんと保護されているということはあり得ないということです。この状況は、多角的な保護メカニズムが実装されている場合に発生する可能性があります。

参考資料

このドキュメントで取り上げた製品に関して詳しくは、次の資料を参照してください。

» Oracle ZFS Storage Appliance ホワイト・ペーパーとテーマ別の資料
http://www.oracle.com/technetwork/jp/server-storage/sun-unified-storage/documentation/index.html

技術的なホワイト・ペーパーは、次のとおりです。

» " Oracle ZFS Storage Appliance で LDAP ソースとして Microsoft Active Directory を使用する方法"
http://www.oracle.com/technetwork/jp/server-storage/sun-unified-storage/documentation/activedir-ldap-source-1124-2372474-ja.pdf

» "How to Configure IDMU on the Oracle ZFS Storage Appliance"
http://www.oracle.com/technetwork/articles/servers-storage-admin/o11-051-zffsa-idmu-mapping-405716.pdf

» "ストレージのパフォーマンスを最大化するパーティション・アライメント"
http://www.oracle.com/technetwork/jp/server-storage/sun-unified-storage/documentation/partitionalign-111512-1875560-ja.pdf

» Oracle ZFS Storage Appliance 製品情報
http://www.oracle.com/jp/storage/nas/overview/index.html

» Oracle ZFS Storage Appliance ドキュメント・ライブラリ（インストール、分析、カスタマ・サービス、管理ガイドなど）
http://docs.oracle.com/en/storage/

» Oracle ZFS Storage Appliance の管理ガイドは、Oracle ZFS Storage Appliance のコンテキスト・ヘルプでも提供されています。
Oracle ZFS Storage Appliance のヘルプ機能は、ブラウザのユーザー・インタフェースから利用できます。

» Oracle Solaris ダウンロード
http://www.oracle.com/technetwork/jp/server-storage/solaris11/downloads/index.html

» NFSv4 プロトコル仕様書
http://www.ietf.org/rfc/rfc3530.txt

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

海外からのお問い合わせ窓口
電話：+1.650.506.7000
ファクシミリ：+1.650.506.7200

CONNECT WITH US

blogs.oracle.com/oracle
facebook.com/oracle
twitter.com/oracle
oracle.com

Integrated Cloud Applications & Platform Services

異機種環境における Oracle ZFS Storage Appliance のアクセス制御リスト機能の使用（2016 年 8 月バージョン 1.0）
著者：Andrew Ness
Oracle Application Integration Engineering