Sécurisation des utilisateurs et des processus dans Oracle® Solaris 11.2

Référence: E53952
Juillet 2014

ORACLE®

Table des matières

Liste des exemples

Utilisation de la présente documentation

- **Présentation** : décrit comment affecter des droits supplémentaires aux utilisateurs, comment créer et utiliser des rôles et comment affecter des droits pour les programmes et des ressources spécifiques sur les systèmes Oracle Solaris.
- **public visé :** les administrateurs de sécurité.
- **Connaissances requises** : exigences en matière de sécurité du site.

Bibliothèque de la documentation des produits

Les informations de dernière minute et les problèmes connus pour ce produit sont inclus dans la bibliothèque de documentation accessible à l'adresse : http://www.oracle.com/pls/topic/lookup?ctx=E56338.

Accès aux services de support Oracle

Les clients Oracle ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info ou le site http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs si vous êtes malentendant.

Commentaires en retour

Faites part de vos commentaires sur cette documentation à la page suivante : http://www.oracle.com/goto/docfeedback.

♦ ♦ ♦ CHAPITRE 1

A propos de l'utilisation de droits pour contrôle Users and Processes

Oracle Solaris fournit des droits qui peuvent être affectés aux utilisateurs, aux rôles, aux processus et aux ressources sélectionnées. Ces droits fournissent une méthode d'administration plus sécurisée au modèle de superutilisateur.

Ce chapitre fournit des informations sur les processus des éléments qui prennent en charge la gestion des droits utilisateur et traite et pour développer un utilisateur des droits manières, limitez à un utilisateur des droits, ajoutez des privilèges à des commandes, et limiter l'accès des applications aux seulement les privilèges dont ils ont besoin, procédez comme suit :

Nouveautés relatives aux droits dans Oracle Solaris 11.2

Cette section met en évidence importantes pour les clients existants sur les informations nouvelles fonctions de droits les droits de l'utilisateur, également appelé RBAC (contrôle d'accès basé sur les droits de processus), également appelée et privilèges.

- Un profil de droits qui l'administrateur attribue profil de droits en tant *qu'* elle oblige l'utilisateur authentifié à saisir un mot de passe avant d'exécuter une commande privilégiée. Si l'utilisateur ne fournit pas un mot de passe, la commande s'exécute sans privilège. Un mot de passe doit rester appliquée à la période de temps configurables. Reportez-vous à la section Exemple 3-11, "Le mot de passe, qui requiert un avant l'utilisateur à entrer Administering DHCP".

  Vous pouvez affecter un profil de droits chaque fois qu'une personne autorisée souhaite authentifiés se connecte au système en ajoutant la valeur du profil AUTH_PROFS_GRANTED en tant que mot-clé dans le fichier. `policy.conf`
- Vous pouvez limiter l'accès des utilisateurs et des groupes aux hôtes en fonction du temps et de access_times et fuseau horaire access_tz en affectant les droits. Pour consulter un un exemple, reportez-vous à la page de manuel `user_attr`(4).

- Oracle Solaris fournit les rôles d'autorisation gérés dans l'ensemble de rôles standard RBAC (ARMOR) compris dans le package `armor`. Pour plus d'informations, reportez-vous à “Fournir un élément de contrat et Processus de gestion des droits utilisateur la solution de substitution au modèle superutilisateur” à la page 14Exemple 3-1, “Les rôles à l'aide de ARMOR”
- L'interface graphique du gestionnaire d'utilisateurs permet de gérer les droits des utilisateurs et des rôles. Pour plus d'informations, reportez-vous au Chapitre 3, “ Gestion des comptes utilisateur dans l’interface graphique du Gestionnaire d’utilisateurs ” du manuel “ Gestion des comptes utilisateur et des environnements utilisateur dans Oracle Solaris 11.2 ”.

Utilisateur Rights Management

La gestion des droits des utilisateurs est une fonction de sécurité qui permet de contrôler l'accès des utilisateurs aux tâches qui incombent normalement au rôle `root`. En appliquant des attributs de sécurité, ou *droits* aux processus et aux utilisateurs, le site peut répartir les privilèges de superutilisateur entre plusieurs administrateurs. La gestion des droits des processus est mise en oeuvre par le biais des *privilèges*. La gestion des droits des utilisateurs est mise en oeuvre par le *biais* de droits, qui collecte les profils de droits qui sont ensuite affectés aux utilisateurs et aux rôles. Les droits de l'utilisateur peut également se limiter les kiosques ou ; par exemple, pour les utilisateurs invités.

- Pour en savoir plus sur ses droits sur des processus noyau, reportez-vous à la rubrique Processus Rights Management. “Processus Rights Management” à la page 24
- Pour plus d'informations sur les procédures, reportez-vous au Chapter pour gérer la section Affectation de droits Rights in Oracle Solaris 3,. Chapitre 3, Affectation de droits dans Oracle Solaris
- Pour obtenir des informations de référence, reportez-vous au Chapitre 8, Droits Oracle Solaris (référence).

Fournir un élément de contrat et Processus de gestion des droits utilisateur la solution de substitution au modèle superutilisateur

Dans les systèmes UNIX classiques, l'utilisateur `root`, également appelé superutilisateur, dispose de tous les pouvoirs. Les programmes exécutés comme root, `et` réciproquement pour la plupart des programmes, sont également tous les pouvoirs. `setuid` L'utilisateur `root` peut lire les fichiers et y écrire des données, exécuter tous les programmes et envoyer des signaux d'interruption aux processus. Concrètement, un superutilisateur peut changer le pare-feu d'un site, modifier la piste d'audit, consulter des informations confidentielles et arrêter l'ensemble du réseau. Un programme `setuid root` piraté peut avoir la mainmise sur le système.

L'affectation de droits aux utilisateurs, aux ressources et aux processus représente une alternative plus sûre au modèle tout ou rien des superutilisateurs. Les droits permettent d'appliquer des stratégies de sécurité à un niveau plus détaillé. Les droits respectent le principe de sécurité du *moindre privilège*. En d'autres termes, un utilisateur dispose ni plus ni moins des privilège exacts nécessaires à l'exécution d'un travail. Les utilisateurs standard ont suffisamment de privilèges pour utiliser leurs applications, vérifier l'état de leurs tâches, imprimer des fichiers, créer des fichiers, et ainsi de suite. Les droits qui dépassent ceux de l'utilisateur standard sont regroupées dans des profils de droits. Les utilisateurs qui doivent effectuer des tâches qui nécessitent une des droits de superutilisateur peuvent se voir attribuer un profil de droits.

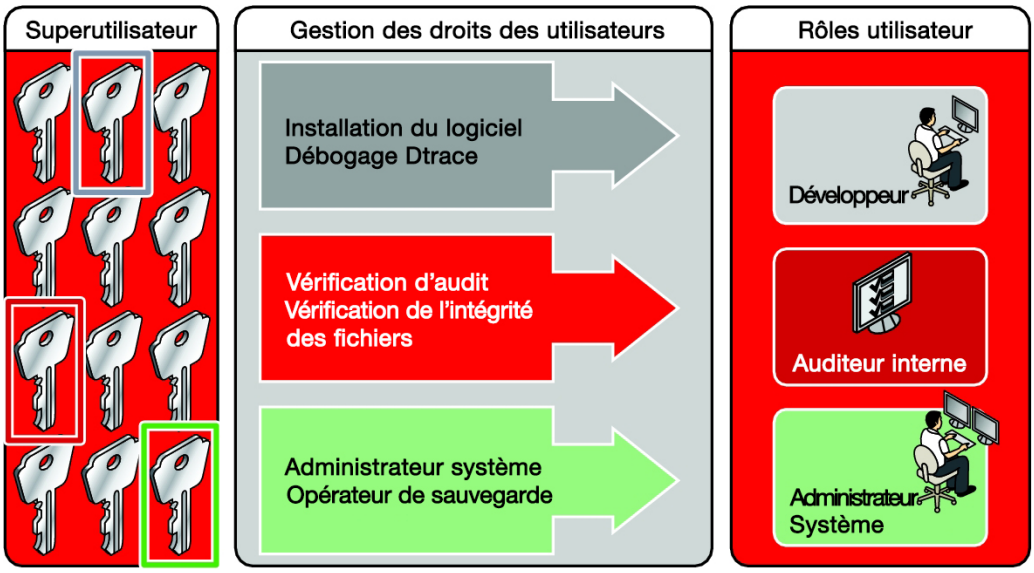
Droits qui sont regroupées dans un profil peut être affecté directement à des utilisateurs. Elles peuvent également être affectés indirectement en créant des comptes spéciaux, appelés rôles. Un utilisateur peut alors endosser un rôle pour effectuer une tâche qui requiert certains privilèges d'administration. Oracle Solaris fournit un grand nombre de profils de droits prédéfinis. Vous créez les rôles et affectez les profils.

Le package ARMOR contient un jeu de rôles standardisés. L'installation de ce package par l'interface et en affectant les rôles aux utilisateurs, vous ne pouvez créer un système qui fournit la séparation des tâches à l'initialisation. Pour plus d'informations, reportez-vous à la section Authorization Rules Managed On RBAC (ARMOR), “Modèle de gestion des droits de votre Sélectionné suivantes” à la page 45, et à l'Exemple 3-1, “Les rôles à l'aide de ARMOR”.

Les profils de droits peuvent fournir large les droits d'administration. Par exemple, le profil de droits System Administrator (administrateur système) permet à un compte d'effectuer des tâches qui ne sont pas liées à la sécurité, telles que la gestion des imprimantes et des tâches cron. Les profils de droits peuvent également être définis avec précision. Par exemple, le profil de droits Cron Management (gestion Cron) gère les tâches `at` et `cron`. Lorsque vous créez des rôles, les rôles peuvent être assignés droits des droits d'administration ou Etroit large.

La figure suivante illustre la manière dont Oracle Solaris peut répartir des droits aux Utilisateurs de confiance en créant des rôles. Des droits d'accès par le superutilisateur peut également imputer assigner des profils de droits directement à aux utilisateurs de confiance.

FIGURE 1-1 Répartition des droits



Dans le modèle de droits indiqués ci-après , le superutilisateur crée trois rôles. Les rôles sont basés sur des profils de droits. Le superutilisateur attribue ensuite les rôles aux utilisateurs autorisés à effectuer les tâches que ce rôle implique. Les utilisateurs se connectent avec leur nom d'utilisateur. Une fois connectés, les utilisateurs endossent les rôles qui autorisent l'utilisation des commandes d'administration et des outils d'interface graphique.

La flexibilité qui caractérise la configuration des rôles permet de définir une vaste gamme de stratégies de sécurité. Bien que Oracle Solaris soit livré avec peu de rôles, ces derniers peuvent être configurés facilement. Exemple 3-1, “Les rôles à l'aide de ARMOR”l' utilisation de rôles basés sur le ARMOR standard. En plus ou en remplacement des rôles ARMOR, vous pouvez créer vos propres rôles en fonction des profils de droits fournis par Oracle Solaris.

- **root** : rôle puissant, équivalent à l'utilisateur `root`. Cependant, à l'instar de tous les rôles, le rôle `root` ne peut pas se connecter. Un utilisateur standard doit se connecter, puis prendre le rôle `root` qui lui est affecté. Ce rôle est configuré et affecté à l'utilisateur initial par défaut.
- **System Administrator** : rôle moins puissant impliquant des tâches d'administration mais pas de sécurité. Ce rôle peut gérer des systèmes de fichiers, la messagerie électronique et l'installation de logiciels. Cependant, ce rôle ne permet pas de définir des mots de passe.
- **Operator** : rôle d'administrateur junior permettant d'effectuer des opérations telles que la gestion d'imprimantes et de sauvegardes.

Remarque - Le profil de droits Media Backup permet d'accéder à la totalité du système de fichiers root. Par conséquent, bien que les profils de droits Media Backup et Operator (opérateur) soient conçus pour les administrateurs débutants, les utilisateurs auxquels vous les affectez doivent être dignes de confiance.

Vous pouvez aussi être amené à configurer un ou plusieurs rôles de sécurité. Trois profils de droits et leurs profils supplémentaires prennent en charge la sécurité. Il s'agit des profils suivants : Information Security (sécurité des informations), User Security (sécurité des utilisateurs) et Zone Security (sécurité de la zone). Sécurité du réseau est un profil supplémentaire inclus dans le profil de droits Information Security (sécurité des informations).

Notez que les rôles n'est pas nécessaire de mettre en oeuvre. Les rôles sont fonction des besoins de sécurité d'une organisation. Une stratégie consiste à configurer les rôles pour les administrateurs ayant un objectif précis dans des domaines tels que la sécurité, la mise en réseau ou l'administration d'un pare-feu. Une autre stratégie consiste à créer un seul rôle d'administrateur puissant conjointement avec un rôle d'utilisateur avancé. Le rôle d'utilisateur avancé est affecté aux utilisateurs autorisés à réparer des parties de leur propre système. Vous pouvez également affecter des profils de droits directement de créer des rôles aux utilisateurs et pas du tout.

Le modèle superutilisateur et le modèle de droits peuvent coexister. Le tableau suivant résume les différents degrés (du superutilisateur à l'utilisateur standard limité) possibles dans le modèle de droits. Le tableau comprend les actions d'administration pouvant faire l'objet d'un suivi dans les deux modèles. Pour obtenir un récapitulatif de l'effet de, c'est - à - dire des droits des processus, reportez-vous au Table 1 -2 des privilèges. Tableau 1-2, “Différences visibles entre un système avec des privilèges et un système sans privilèges”

TABLEAU 1-1 Droits du modèle modèle de superutilisateur par rapport

Capacités d'un utilisateur sur un système	Modèle superutilisateur	Modèle de droits
Peut devenir superutilisateur avec tous les privilèges correspondants	Peut	Peut
Peut se connecter en tant qu'utilisateur disposant des droits utilisateur complets	Peut	Peut
Peut devenir superutilisateur avec des droits limités	Ne peut pas	Peut
Peut se connecter en tant qu'utilisateur et disposer des privilèges superutilisateur sporadiquement	Peut, avec les programmes `setuid root` uniquement	Peut, avec les programmes `setuid root` et avec des droits
Peut se connecter en tant qu'utilisateur disposant des droits d'administration, mais sans les privilèges superutilisateur complets	Ne peut pas	Pouvez, si vous disposez de profils de droits, des rôles ou avec les autorisations et privilèges affectés directement
Peut se connecter en tant qu'utilisateur disposant de moins de droits qu'un utilisateur standard	Ne peut pas	La suppression de droits peut, par

Capacités d'un utilisateur sur un système	Modèle superutilisateur	Modèle de droits
Peut suivre les actions superutilisateur	Peut, par l'audit de la commande `su`	Peut, par l'audit des appels à `pfexec()` En outre, le nom de l'utilisateur qui a pris le rôle `root` se trouve dans la piste d'audit.

Informations de base sur les droits des utilisateurs et les droits des processus

Droits lestermes sans privilège ne s'*appliquent pas ou sans ENT : OSOL entre accolades {}*. Chaque processus dans Oracle Solaris, y compris les processus utilisateurs normaux, possèdent certains privilèges ou des droits utilisateurs tels que des autorisations. Pour plus d'informations sur l'ensemble de privilèges de base qu'Oracle Solaris accorde à tous les processus UNIX, reportez-vous à la section “Processus Rights Management” à la page 24.

Les éléments suivants appliquent les droits d'utilisateur dans Oracle Solaris. Ces droits permissif peut être configuré pour appliquer les stratégies de sécurité ou des stratégies de sécurité restrictif.

- **Autorisation** : permission permettant à un utilisateur ou à un rôle de réaliser une classe d'actions nécessitant des droits supplémentaires. Par exemple, les utilisateurs de la console de la règle de sécurité par défaut donne l'autorisation solaris.device.cdrw. Cette autorisation offre aux utilisateurs les droits de lecture et d'écriture au niveau du périphérique de CD-ROM. Pour obtenir la liste des autorisations, utilisez la commande. `auths list` Les "feux verts" sont mises en oeuvre au niveau de l'application utilisateur, et non dans le noyau. Reportez - vous à la“En savoir plus sur les autorisations utilisateur” à la page 22
- **Privilège** : droit pouvant être accordé à une commande, un utilisateur, un rôle ou une ressource spécifique, telle qu'un port ou une méthode SMF. Les privilèges sont mis en oeuvre dans le noyau. Par exemple, le privilège `proc_exec` permet à un processus d'appeler `execve()`. Les utilisateurs standard disposent des privilèges de base. Pour connaître vos privilèges de base, exécutez la commande `ppriv -vl basic`. Pour plus d'informations, voir Processus Rights Management. “Processus Rights Management” à la page 24
- **Attributs de sécurité** : attributs autorisant un processus à exécuter une opération, ou implémentation d'un droit. Dans un environnement UNIX standard, un attribut de sécurité permet à un processus d'effectuer une opération qui serait autrement interdite aux utilisateurs standard. Par exemple, les programmes `setuid` et `setgid` ont des attributs de sécurité. Outre les programmes `setuid` et `setgid`, les autorisations et les privilèges sont également des attributs de sécurité dans le modèle de droits. Ces attributs, ou droits, peuvent être affectés à un utilisateur. Par exemple, un utilisateur avec l'autorisation `solaris.device.allocate` peut allouer un périphérique pour une utilisation exclusive. Les privilèges peuvent être placés sur un processus. Par exemple, un processus avec le privilège `file_flag_set` peut définir des attributs de fichier immutable, no-unlink ou append-only.

Les attributs de sécurité peuvent également limiter les droits. Par exemple, lesattributs de sécurité access_times et le week-end access_tz et heures de fin et le fuseau horaire quand les, si vous le souhaitez, une des opérations liées à la sécurité sont autorisés. Vous pouvez limiter le nombre d'utilisateurs en les affectant directement authentifié ou profil de droits contenant ces mots-clés . Pour plus d'informations, reportez-vous à la page de manuel `user_attr`(4).

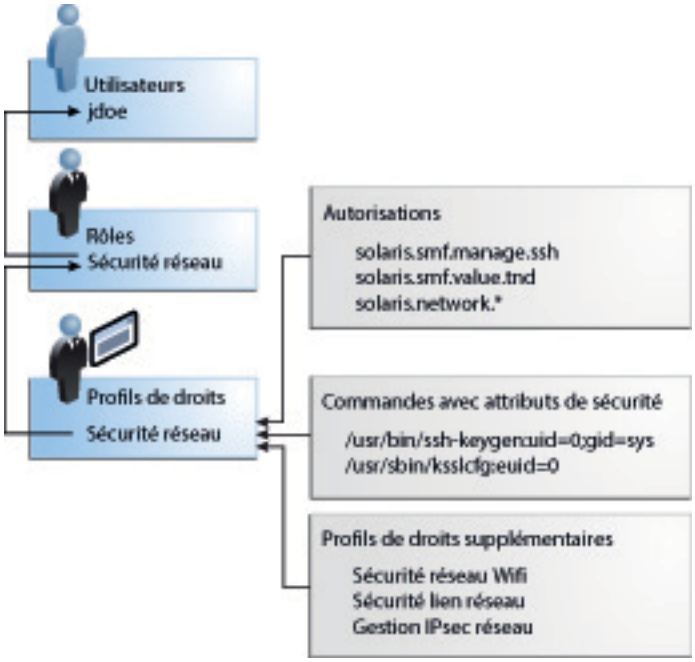
- **Application privilégiée** : application ou commande pouvant ignorer les contrôles système en vérifiant les droits. Pour plus d'informations, reportez-vous à la section “Applications vérifiant for Rights” à la page 38 et au manuel “ Developer’s Guide to Oracle Solaris 11 Security ”
- **Profil de droits** : ensemble de droits pouvant être affectés à un rôle ou à un utilisateur. . Un profil de droits peut inclure des autorisations, des privilèges affectés directement, des commandes avec des attributs de sécurité et d'autres profils de droits. Les profils au sein d'un autre profil sont appelés *profils de droits supplémentaires*. Les profils de droits offrent un moyen pratique de regrouper des droits. Ils peuvent être directement affectés aux utilisateurs et à des comptes spéciaux appelés *rôles*. Vous pouvez utiliser les commandes dans un profil de droits que si votre un traitement reconnaît les droits. En outre, vous pouvez vous être demandé d'indiquer un mot de passe. Vous pouvez également l'authentification par mot de passe peut être fourni par défaut. Reportez-vous à la section “En savoir plus sur les profils de droits” à la page 22.
- **Rôle** : identité spéciale permettant d'exécuter des *applications privilégiées*. L'identité spéciale peut être endossée par les utilisateurs assignés uniquement. Dans un système exécuté par les rôles, le superutilisateur peut être inutile après la phase de configuration initiale. Reportez - vous à la rubrique Autres A propos des rôles. “En savoir plus sur les rôles” à la page 23

La figure ci-dessous montre comment les droits des utilisateurs et des droits de processus fonctionnent ensemble.

FIGURE 1-2 Besoin en fonds de roulement Droits et de travail de processus de gestion des droits utilisateur de pair

La figure suivante utilise le rôle Network Security (Sécurité réseau) et le profil de droits Network Security pour illustrer le fonctionnement des droits attribués.

FIGURE 1-3 Et exemple de processus of a User Rights Assignment de droits



Le rôle Network Security (Sécurité du réseau) permet de gérer les liaisons réseau, IPsec et wifi. Ce rôle est assigné à l'utilisateur `jdoe`. Pour prendre le rôle, `jdoe` change de rôle, puis indique le mot de passe correspondant. L'administrateur peut activer le rôle à vous authentifier à l'aide de le mot de passe utilisateur plutôt qu'un mot de passe du rôle.

Dans cette illustration, le profil de droits Network Security (sécurité réseau) est affecté au rôle Sécurité réseau. Le profil de droits Network Security (sécurité réseau) contient des profils supplémentaires qui sont évalués dans l'ordre Network Wifi Security (sécurité wifi réseau), Network Link Security (sécurité liaison réseau) et Network IPsec Management (gestion IPsec réseau). Droits ces profils supplémentaires qui complètent la contenir tâches principales du rôle.

Le profil de droits Network Security (sécurité réseau) comporte trois autorisations attribuées directement, aucun privilège affecté directement et deux commandes avec des attributs de sécurité. Les profils de droits supplémentaires ont des autorisations attribuées directement et deux d'entre eux ont des commandes avec des attributs de sécurité.

Lorsque jdoe endosse lerôle, le shell Network Security (sécurité réseau) les modifications apportées à un shell de profil. Shell de profil Le shell de profil de processus est possible d'évaluer l'utilisation de droits. par conséquent, jdoe peut administrer la sécurité du réseau.

En savoir plus sur les droits utilisateur

Cette section fournit des informations supplémentaires sur l'implémentation et l'utilisation de droits au niveau de l'utilisateur.

En savoir plus sur les autorisations utilisateur

Une autorisationest un droite et qui peut être accordé à un rôle, une zone, un programme ou un utilisateur. Les autorisations permettent d'appliquer des stratégies au niveau de l'application utilisateur. Les affectations, telles que des privilèges d'autorisations permettent mistaken entraînent prévue les droits accordés à l'origine. Pour plus d'informations, reportez-vous à la section “Et droits utilisateurs privilège d'escalade” à la page 34.

La différence entre les autorisations et les privilèges réside dans le niveau auquel la stratégie de sécurité est appliquée. Sans le privilège adéquat, un processus peut se voir empêcher d'exécuter des opérations privilégiées par le noyau. Sans les autorisations adéquates, un utilisateur peut se voir empêcher d'utiliser une Application privilégiée ou d'exécuter des opérations liées à la sécurité au sein d'une application privilégiée. Pour en savoir plus sur les privilèges, reportez-vous à la rubrique Processus Rights Management. “Processus Rights Management” à la page 24

Les applications compatibles avec les droits peuvent vérifier les autorisations de l'utilisateur avant de lui accorder l'accès soit à l'application elle-même, soit à des opérations spécifiques au sein de l'application. Cette vérification remplace la vérification conventionnelle dans les applications UNIX pour `UID=0`.

Pour plus d'informations sur les autorisations, reportez-vous aux sections suivantes :

- “Les "feux verts" Reference (disponible en anglais uniquement)” à la page 119
- “Base de données `auth_attr`” à la page 123
- “Commandes sélectionnées nécessitant des autorisations” à la page 125

En savoir plus sur les profils de droits

Un *profil de droits* est un ensemble de droits pouvant être affecté à un rôle ou à un utilisateur pour effectuer des tâches requérant des droits d'administration. Un profil de droits peut se composer d'autorisations, de privilèges, de commandes auxquelles des attributs de sécurité ont été affectées et d'autres profils de droits. Les profils de droits peuvent également contenir des entrées afin de réduire ou d'étendre le premier ensemble de privilèges hérité et de réduire l'ensemble limite de privilèges.

un profil de droits est authentifié les profils de droits qui nécessite que l'utilisateur, ou à fournir un mot de passe de vous authentifier à nouveau. L'administrateur ne décide pas les profils qui

peuvent être utilisées sans la réauthentification. Vous trouverez un bon exemple d'un profil qui ne nécessite la réauthentification est le profil de droits de l'utilisateur Solaris de base. Selon les exigences de sécurité du site, des profils de droits pour les tâches liées à la sécurité peut avoir besoin de connaître la réauthentification.

Pour obtenir des informations de référence sur les profils de droits, reportez-vous aux sections suivantes :

- “Profils de droits Reference (disponible en anglais uniquement)” à la page 117
- “Base de données `prof_attr`” à la page 123
- “Base de données `exec_attr`” à la page 123

En savoir plus sur les rôles

Un *rôle* est un type spécial de compte utilisateur à partir duquel vous pouvez exécuter des applications privilégiées. Les rôles sont créés de la même manière que les comptes utilisateur. Les rôles ont un répertoire d'accueil, une affectation de groupe, un mot de passe, et ainsi de suite. Les profils de droits et les autorisations attribuent les capacités d'administration des rôles. Les rôles ne peuvent pas hériter de droits d'autres rôles ou de la part de l'utilisateur qui prend le rôle. Et distribuer les privilèges de superutilisateur, les rôles permettent ainsi des pratiques administratives plus sécurisées.

Un rôle peut être affecté à plusieurs utilisateurs. Tous les utilisateurs qui peuvent prendre le même rôle possèdent le même répertoire d'accueil de rôle, fonctionnent dans le même environnement et ont accès aux mêmes fichiers. Les utilisateurs peuvent endosser les rôles à partir de la ligne de commande, à l'aide de la commande `su` ainsi que du nom et du mot de passe des rôles. L'administrateur peut configurer le système pour permettre à un utilisateur de s'authentifier en fournissant le mot de passe de l' utilisateur . Reportez-vous à la section Exemple 3-16, “Autorisation d'un utilisateur le mot de passe pour l'autorisation d'utiliser son propre mot de passe du rôle”.

Un rôle ne peut pas se connecter directement. Un utilisateur se connecte, puis prend un rôle. Une fois que vous avez endossé un rôle, vous ne pouvez pas en prendre un autre sans avoir quitté votre rôle en cours.

De plus, alors qu'un profil de droits ajoute des droits à l'environnement de l'utilisateur, un rôle donne à ce dernier un environnement d'exécution propre qui est partagé avec les autres utilisateurs pouvant endosser ce rôle. Lorsqu'un utilisateur change d'un rôle de workflow se termine et aucune des profils de droits de l'utilisateur s'applique aux autorisations ou le rôle.

Les `passwd`, shadow et user_attr les informations sur les rôles. les bases de données stockent statique Vous pouvez, voire vous devez les actions des rôles d'audit.

Pour obtenir des informations détaillées sur la configuration des rôles, reportez-vous aux sections suivantes :

- Modèle de gestion des “Modèle de gestion des droits de votre Sélectionné suivantes” à la page 45
- “Attribution de droits aux utilisateurs” à la page 47

Le fait que `root` soit un rôle dans Oracle Solaris empêche toute connexion `root` anonyme. Si la commande shell de profil,, `pfexecest` en cours d'audit, la piste d'audit contient l'UID réel de l'utilisateur de connexion, quel que soit leur rôle que l'utilisateur a pris et les opérations privilégiées qui ont été effectuées. Pour auditer le système pour les opérations avec privilèges, reportez-vous aux sections Auditing Administrative Actions. “Actions administratives d'audit” à la page 86

Processus Rights Management

Le module de gestion des droits de processus Oracle Solaris est implémenté à l'aide de *privilèges*. Activez les processus d'être restreint les privilèges au niveau de commandes, des utilisateurs, des rôles et ceux propres ressource système. En termes de sécurité, les privilèges diminuent le risque qu'un utilisateur ou un processus puisse disposer des pouvoirs de superutilisateur complets sur un système. Les droits des processus et les droits des utilisateurs offrent une solution de substitution intéressante au modèle superutilisateur traditionnel.

Les privilèges sont utilisés pour aujourd'hui encore, les droits d'ajout. Toutefois, les privilèges peuvent également être utilisée pour limiter l'accès aux droits, par exemple, la modification d'une à un programme qui est programme prenant en charge les privilèges. `setuid root` En outre, avec une stratégie *de privilège étendue* spécifiée, les administrateurs peuvent autoriser uniquement les privilèges nécessaires pour être utilisé avec un ID objet fichier, utilisateur ou un port. Affecter des privilèges au groupe cette fin (FGA) à l'exception interdit à tous les privilèges de base d'autres privilèges à ces dernières.

- Pour plus d'informations sur les privilèges et privilèges restrictifs, reportez-vous à la section “L'utilisation d'Extended Privilege Policy to Utiliser Restreindre du privilège, procédez comme suit :” à la page 34.
- Pour plus d'informations sur les droits de l'utilisateur, reportez-vous à la section User Rights Management. “Utilisateur Rights Management” à la page 14
- Pour plus d'informations sur la gestion des privilèges, reportez-vous au Chapter 3,. la section Attribution de Rights in Oracle Solaris Chapitre 3, Affectation de droits dans Oracle Solaris
- Pour plus d'informations sur les privilèges, reportez-vous à la section “Privilèges (référence)” à la page 127.

Protection des processus noyau par les privilèges

Un privilège est une droite dont a besoin un processus pour réaliser une opération. Le droit est appliqué dans le noyau. Un programme qui s'exécute dans les limites du *jeu de base* de privilèges fonctionne dans les limites de la stratégie de sécurité système. Les programmes `setuid root` sont des exemples de programmes qui fonctionnent en dehors des limites de la stratégie de sécurité système. Les privilèges permettent aux programmes d'éliminer la nécessité d'appeler `setuid root`.

Les privilèges énumèrent les types d'opérations possibles sur un système. Les programmes peuvent être exécutés avec les privilèges exacts nécessaires à leur réussite. Par exemple, un programme qui manipule des fichiers peut nécessiter les privilèges `file_dac_write` et `file_flag_set`. Ces privilèges sur le processus éliminent la nécessité d'exécuter le programme en tant que `root`.

Historiquement, les systèmes n'ont pas suivi le modèleModèle de privilègeou le modèle de droits, tel que présenté dans la section Droits d'utilisateur et traitement notions de base relatives à Oracle. “Informations de base sur les droits des utilisateurs et les droits des processus” à la page 18 Ils ont plutôt utilisé le modèle de superutilisateur. Dans le modèle de superutilisateur, les processus s'exécutaient en tant que `root` ou en tant qu'utilisateur. L'action des processus utilisateur était limitée au niveau des répertoires et fichiers de l'utilisateur. Les processus `root` pouvaient créer des répertoires et fichiers en tout point du système. Un processus qui devait créer un répertoire en dehors du répertoire de l'utilisateur devait s'exécuter avec `UID=0`, c'est-à-dire, en tant que `root`. La stratégie de sécurité s'appuyait sur le contrôle d'accès discrétionnaire (DAC, Discretionary Access Control) pour protéger les fichiers système. Les noeuds de périphérique étaient protégés par DAC. Par exemple, les périphériques appartenant au groupe `sys` ne pouvaient être ouverts que par des membres de ce groupe.

Cependant, les programmes `setuid`, les autorisations de fichier et les comptes d'administration sont vulnérables à une utilisation abusive. Les actions qu'un processus `setuid` est autorisé à réaliser sont plus nombreuses qu'il n'est nécessaire au processus pour terminer son opération. Un programme `setuid root` peut être compromis par un intrus qui s'exécute ensuite en tant qu'utilisateur `root` disposant de tous les pouvoirs. De même, tout utilisateur ayant accès au mot de passe `root` peut compromettre l'ensemble du système.

En revanche, un système appliquant la stratégie avec des privilèges fournit les droits des utilisateurs et une racine graduation entre les droits. Un utilisateur peut se voir accorder des privilèges nécessaires à la réalisation d'activités dépassant les droits des utilisateurs standard et `root` peut voir le nombre de ses privilèges actuels réduit. Avec les droits, une commande qui s'exécute avec les privilèges peut être isolée dans un profil de droits et assignée à un utilisateur ou à un rôle. Tableau 1-1, “Droits du modèle modèle de superutilisateur par rapport” résume la gradation entre les droits utilisateur et les privilèges root que le modèle de droits fournit.

Le modèle de privilèges offre une plus grande sécurité que le modèle superutilisateur. Les privilèges supprimés d'un processus ne peuvent pas être exploités. Les privilèges de processus

peuvent fournir une protection supplémentaire pour les fichiers sensibles, où les protections DAC seules peuvent être exploitées pour obtenir l'accès.

Les privilèges peuvent donc restreindre les programmes et processus aux capacités qu'ils nécessitent Sur un système appliquant ce privilège, un intrus qui capture un processus ne peut accéder qu'aux privilèges dont dispose ce processus. Le reste du système ne peut pas être compromis.

Descriptions des privilèges

Les privilèges sont regroupés logiquement, sur la base du domaine de privilège.

- **Privilèges FILE (fichier)** : les privilèges dont le nom commence par la chaîne `file` agissent sur les objets du système de fichiers. Par exemple, le privilège `file_dac_write` remplace le contrôle d'accès discrétionnaire lors de l'écriture dans les fichiers.
- **Privilèges IPC** : les privilèges dont le nom débute par la chaîne `ipc` remplacent les contrôles d'accès aux objets IPC. Par exemple, le privilège `ipc_dac_read` permet à un processus de lire une mémoire partagée distante et protégée par le contrôle DAC.
- **Privilèges NET** : les privilèges dont le nom commence par la chaîne `net` offrent l'accès à des fonctionnalités réseau spécifiques. Par exemple, le privilège `net_rawaccess` permet à un périphérique de se connecter au réseau.
- **Privilèges PROC** : les privilèges dont le nom commence par la chaîne `proc` permettent aux processus de modifier les propriétés restreintes du processus lui-même. Les privilèges `PROC` comprennent des privilèges ayant un effet très limité. Par exemple, le privilège `proc_clock_highres` permet à un processus d'utiliser des horloges haute résolution.
- **Privilèges SYS** : les privilèges dont le nom commence par la chaîne `sys` offrent aux processus un accès illimité à diverses propriétés système. Par exemple, le privilège `sys_linkdir` permet à un processus de créer et de rompre des liens physiques vers des répertoires.

D'autres groupes logiques comprennent CONTRACT, CPC, DTRACE, GRAPHICS, VIRT et WIN.

Certains privilèges ont un effet limité sur le système, d'autres un effet important. La définition du privilège `proc_taskid` indique son effet limité :

```
proc_taskid
        Allows a process to assign a new task ID to the calling process.
```

La définition du privilège `net_rawaccess` indique son large effet :

```
net_rawaccess
        Allows a process to have direct access to the network layer.
```

La page de manuel `privileges`(5) décrit chaque privilège. Reportez - vous également à la zone de la liste des privilèges. “Liste des privilèges” à la page 102

Différences administratives sur un système disposant de privilèges

Un système disposant de privilèges présente plusieurs différences visibles avec un système qui n'en possède pas. Le tableau suivant énumère certaines différences.

TABLEAU 1-2 Différences visibles entre un système avec des privilèges et un système sans privilèges

Caractéristiques	Aucun privilège	Droits
Démons	Les démons s'exécutent en tant que `root`.	Les démons s'exécutent au titre d'utilisateur `daemon`. Par exemple, les démons suivants ont reçu les privilèges appropriés et s'exécutent en tant que `daemon` : `lockd` et `rpcbind`.
Propriété du fichier journal	Les fichiers journaux appartiennent à `root`.	Les fichiers journaux sont désormais la propriété du `daemon`, qui a créé le fichier journal. L'utilisateur `root` ne détient pas la propriété du fichier.
Messages d'erreur	Les messages d'erreur se rapportent au superutilisateur. Par exemple, `chroot: not superuser`.	Les messages d'erreur reflètent l'utilisation des privilèges. Par exemple, le message d'erreur équivalent pour l'échec `chroot` est `chroot: exec failed`.
Programmes `setuid`	Les programmes utilisent `setuid` pour terminer les tâches que les utilisateurs standard ne sont pas autorisés à effectuer.	La plupart des programmes `setuid root` s'exécutent uniquement avec les privilèges dont ils ont besoin. Par exemple, les commandes suivantes utilisent des privilèges : `audit`, `ikeadm`, `ipadm`, `ipsecconf`, `ping`, `traceroute` et `newtask`.
Droits d'accès aux fichiers	Les autorisations d'accès aux périphériques sont contrôlées par DAC. Par exemple, les membres du groupe `sys` peuvent ouvrir `/dev/ip`.	Les autorisations de fichier (DAC) ne prédisent pas qui peut ouvrir un périphérique. Les périphériques sont protégés par DAC *et* par la stratégie de périphériques. Par exemple, le fichier `/dev/ip` a `666` autorisations, mais le périphérique ne peut être ouvert que par un processus disposant des privilèges appropriés.
Evénements d'audit	L'audit de l'utilisation de la commande `su` couvre de nombreuses fonctions d'administration.	L'audit de l'utilisation des privilèges couvre la plupart des fonctions d'administration. La classe d'audit `cusa` inclut des événements d'audit qui surveillent les fonctions d'administration.
Processus	Les processus sont protégés par les droits de l'responsable du processus.	Les processus sont protégés par des privilèges. Les privilèges de processus et les indicateurs de processus sont visibles sous la forme d'une nouvelle entrée dans le répertoire `/proc/<pid>/priv`.
Débogage	Aucune référence aux privilèges dans les dumps noyau.	La section de note ELF des dumps noyau comprend des informations sur les privilèges et indicateurs de processus dans les notes `NT_PRPRIV` et `NT_PRPRIVINFO`. La commande `ppriv` et d'autres commandes indiquent le nombre correct de jeux de taille adéquate. Les commandes font correspondre correctement les bits dans les jeux de bits avec les noms de privilège.

Plus la rubrique A propos des privilèges

Cette section couvre, l'utilisation, le privilège de mise en oeuvre et des détails de l'affectation.

Mise en oeuvre des privilèges

Chaque processus comporte quatre jeux de privilèges qui déterminent s'il peut utiliser un privilège particulier. Le noyau calcule automatiquement le *jeu effectif* de privilèges. Vous pouvez modifier le *jeu héritable* de privilèges. Un programme codé pour utiliser des privilèges peut réduire le *jeu autorisé* du programme. Vous pouvez réduire le *jeu limite* de privilèges.

- **Jeu de privilèges effectif ou E (Effective)** : jeu de privilèges actuellement en vigueur. Un processus peut ajouter des privilèges du jeu autorisé dans le jeu effectif. Un processus peut également supprimer des privilèges de E.
- **Jeu de privilèges autorisés ou P (Permitted)** : jeu de privilèges disponible pour l'utilisation. Un programme peut disposer de privilèges par héritage ou par affectation. Un profil d'exécution est un moyen d'affecter des privilèges à un programme. La commande `setuid` affecte tous les privilèges dont dispose `root` sur un programme. Des privilèges peuvent être supprimés du jeu autorisé, mais pas ajoutés. Les privilèges supprimés de P sont automatiquement supprimés de E.

  Un programme *conscient des privilèges* supprime de son jeu autorisé les privilèges qu'un programme n'utilise jamais. De cette manière, les privilèges superflus ne peuvent pas être exploités par le programme ou un processus malveillant. Pour plus d'informations sur les Conscient des privilèges, reportez-vous au Chapitre 2, “ Developing Privileged Applications ” du manuel “ Developer’s Guide to Oracle Solaris 11 Security ”.
- **Jeu de privilèges héritable ou I (Inheritable)** : jeu de privilèges qu'un processus peut hériter d'un appel à `exec`. Après l'appel à `exec`, les privilèges hérités sont placés dans le jeu autorisé et le jeu effectif, lesquels deviennent alors identiques, sauf dans le cas particulier d'un programme `setuid`.

  Pour un programme `setuid`, après l'appel à `exec`, le jeu héritable est d'abord restreint par le jeu limite. Ensuite, les privilèges hérités (I), moins les privilèges qui se trouvaient dans le jeu limite (L), sont affectés à P et E pour ce processus.
- **de limite, ou L (limit)** : Jeu qui définit la limite extérieure des privilèges qui sont disponibles pour un processus et ses enfants. Par défaut, le jeu limite contient tous les privilèges. Les processus peuvent réduire le jeu limite mais ne peuvent jamais l'étendre. L permet de limiter I. Par conséquent, L limite P et E au moment de `exec`.

  Si un utilisateur s'est vu affecter un profil qui inclut un programme qui a reçu des privilèges, l'utilisateur peut généralement exécuter ce programme. Sur un système non modifié, les privilèges affectés du programme se trouvent dans le jeu limite de l'utilisateur. Les privilèges affectés au programme deviennent partie intégrante du jeu autorisé de l'utilisateur. C'est à partir d'un Shell de profil que l'utilisateur doit exécuter le programme auquel des privilèges ont été affectés.

Le noyau reconnaît un jeu de privilèges de base. Sur un système non modifié, le jeu héritable initial de chaque utilisateur correspond au jeu de base défini au moment de la connexion. Si vous ne pouvez pas modifier le jeu de base, vous pouvez modifier les privilèges dont un utilisateur hérite du jeu de base.

Sur un système non modifié, les jeux de privilèges de l'utilisateur lors de la connexion ressemblent à ce qui suit :

```
E (Effective): basic
I (Inheritable): basic
P (Permitted): basic
L (Limit): all
```

Au moment de la connexion, tous les utilisateurs ont le jeu de base dans leur jeu hérité, leur jeu autorisé et leur jeu effectif. Le jeu limite d'un utilisateur est l'équivalent du jeu limite par défaut pour la zone globale ou non globale.

Vous pouvez affecter d'autres, Privilèges Directement à un utilisateur à un ou de l'utilisateur de la mémoire de manière plus précise, de manière indirecte à de nombreux processus de connexion des utilisateurs par le biais d'un profil de droits, et en affectant une commande privilégiée indirectement à un utilisateur. Vous pouvez également supprimer des privilèges du jeu de base d'un utilisateur. Pour plus d'informations sur les procédures et Affecter des exemples, reportez-vous au Chapter Rights in Oracle Solaris 3,. Chapitre 3, Affectation de droits dans Oracle Solaris

La façon dont les privilèges sont utilisés

{ENT privilèges :, sont regroupés en OSOL}. Cette section décrit la manière dont Oracle Solaris utilise les privilèges avec les périphériques, dans la gestion des ressources, ainsi qu'avec les applications héritées.

Comment les processus obtiennent des privilèges

Hérite des privilèges ou les processus peuvent se voir affecter des privilèges. Un processus hérite des privilèges de son processus parent. Au moment de la connexion, le jeu de privilèges héritable initial détermine les privilèges disponibles pour les processus de l'utilisateur. Tous les processus fils de la connexion initiale de l'utilisateur héritent de ce jeu.

Vous pouvez également affecter directement des privilèges à des programmes, des utilisateurs, des rôles et des ressources spécifiques. Lorsqu'un programme requiert des privilèges, vous affectez les privilèges à l'exécutable du programme dans un profil de droits. Les utilisateurs et les rôles autorisés à exécuter le programme se voient affecter le profil qui comprend le programme. Au moment de la connexion, ou lorsqu'un shell de profil est ouvert, le programme

s'exécute avec le privilège lorsque l'exécutable du programme est saisi dans le shell de profil. Par exemple, un rôle qui inclut le profil Access Management est en mesure de l'objet pour exécuter `chmod`la commande avec le privilège file_chown, et peut donc changer la propriété d'un fichier que le rôle n'est pas propriétaire.

Lorsqu'un rôle ou un utilisateur exécute un programme auquel un privilège supplémentaire a été affecté directement, celui-ci est ajouté au jeu héritable du rôle ou de l'utilisateur. Les processus fils du programme auquel des privilèges ont été affectés héritent des privilèges de leur parent. Si le processus fils nécessite plus de privilèges que le processus parent, ces privilèges doivent lui être affectés directement.

Les programmes qui sont codés pour faire usage de privilèges sont appelés Conscient des privilèges. Un programme conscient des privilèges active et désactive l'utilisation de privilèges lors de l'exécution du programme. Pour réussir dans un environnement de production, le programme doit se voir affecter les privilèges qu'il active et désactive. Avant d'apporter une programme disponible prenant en charge les privilèges que vous affectez à l'exécutable, seuls les privilèges dont le programme a besoin. Vous devez ensuite tester le programme pour vérifier qu'il accomplit ses tâches correctement. Vous devez également vérifier que le programme ne fait pas une utilisation abusive des privilèges.

Pour des exemples de code prenant en charge les privilèges, reportez-vous au Chapitre 2, “ Developing Privileged Applications ” du manuel “ Developer’s Guide to Oracle Solaris 11 Security ”. Pour affecter des privilèges à un programme qui en nécessite, voir l'Exemple 4-1, “Affecter des attributs de sécurité à une ancienne application” et l' Exemple 5-7, “Création d'un profil de droits qui inclut des commandes privilégiées”.

Privilèges et périphériques

Le modèle de droits utilise des privilèges pour protéger les interfaces système qui, dans le modèle de superutilisateur, sont uniquement protégées par des autorisations de fichier. Dans un système doté de privilèges, les autorisations de fichier sont trop faibles pour protéger les interfaces. Un privilège comme `proc_owner` peut remplacer les autorisations de fichier et donner ensuite l'accès complet au système.

Par conséquent, dans Oracle Solaris la propriété du répertoire de périphérique n'est pas suffisante pour ouvrir un périphérique. Par exemple, les membres du groupe `sys` ne sont plus automatiquement autorisés à accéder au périphérique `/dev/ip`. Les autorisations de fichier sur `/dev/ip` sont `0666`, mais le privilège `net_rawaccess` est requis pour ouvrir le périphérique.

Car la stratégie de périphériques est contrôlée par les privilèges, vous bénéficiez d'une plus grande souplesse dans l'octroi d'autorisations pour ouvrir des périphériques. Les exigences en matière de privilèges peuvent être modifiées pour la stratégie de périphérique et les propriétés du pilote. Vous pouvez configurer les exigences en matière de privilèges, ou lorsque la mise à jour de l'installation, de l'ajout d'un pilote de périphérique.

Pour plus d'informations, reportez-vous aux pages de manuel `add_drv`(1M), `devfsadm`(1M), `getdevpolicy`(1M) et `update_drv`(1M).

Les privilèges et de gestion des ressources

Dans Oracle Solaris, les contrôles de ressources `project.max-locked-memory` et `zone.max-locked-memory` peuvent être utilisés pour limiter la consommation mémoire des processus auxquels le privilège `PRIV_PROC_LOCK_MEMORY` est affecté. Ce privilège permet à un processus de verrouiller des pages dans la mémoire physique.

Si vous affectez le privilège `PRIV_PROC_LOCK_MEMORY` à un profil de droits, vous pouvez attribuer aux processus qui disposent de ce privilège la capacité de verrouiller la totalité de la mémoire. A titre de protection, définissez un contrôle de ressources pour empêcher l'utilisateur de ce privilège de verrouiller toute la mémoire. Pour les processus privilégiés qui s'exécutent dans une zone non globale, définissez le contrôle de ressources `zone.max-locked-memory`. Pour les processus privilégiés qui s'exécutent sur un système, créez un projet et définissez le contrôle de ressources `project.max-locked-memory` . Pour plus d'informations sur ces contrôles de ressources A propos de, reportez -Chapitre 6, “ A propos des contrôles de ressources ” du manuel “ Administration de la gestion des ressources dans Oracle Solaris 11.2 ”Chapitre 2, “ Présentation de la configuration des zones non globales ” du manuel “ Présentation d’Oracle Solaris Zones ”

L'utilisation d'anciennes applications et des privilèges

Pour s'adapter aux anciennes applications, l'implémentation de privilèges fonctionne à la fois avec le superutilisateur et les modèles de droits. Le noyau suit automatiquement l'indicateur `PRIV_AWARE`, qui indique qu'un programme a été conçu pour fonctionner avec des privilèges. Prenons un processus fils qui n'est pas conscient des privilèges. Les privilèges hérités du processus parent sont disponibles dans les jeux effectif et autorisé de l'enfant. Si le processus enfant définit un UID sur `0`, le processus enfant n'a peut-être pas toutes les capacités de superutilisateur. Les jeux effectif et autorisé du processus sont limités aux privilèges dans le jeu limite de l'enfant. Par conséquent, le jeu limite d'un processus conscient des privilèges restreint les privilèges root des processus fils qui ne sont pas conscients des privilèges.

Le débogage de l'utilisation des du privilège, procédez comme suit :

Oracle Solaris fournit des outils pour déboguer les défaillances des privilèges. Les commandes `ppriv` et `truss` fournissent le résultat du débogage. Pour consulter des exemples, reportez-vous à la page de manuel `ppriv`(1). Pour consulter des exemples, reportez-vous à la section

Troubleshooting Rights. “Dépannage de gestion des droits” à la page 107 Vous pouvez également utiliser la commande `dtrace`. Pour plus d'informations, reportez-vous à la page de manuel `dtrace`(1M) et au manuel “ Oracle Solaris 11.2 Dynamic Tracing Guide ”.

Affectation de privilèges

Traditionnellement, le terme "privilège" indique une augmentation de droits. Etant donné que chaque processus sur un système Oracle Solaris s'exécute avec certains droits, vous pouvez diminuer les droits sur un processus en supprimant des privilèges. Dans cette version, vous pouvez également utiliser une *stratégie de privilèges étendue* pour supprimer la plupart des privilèges, à l'exception de ceux attribués par défaut à certaines ressources.

Affectation de privilèges aux utilisateurs et processus

Administrateur de sécurité qu'en votre qualité, vous êtes responsable de l'affectation des privilèges. Déjà les profils de droits existants affectées à des commandes disposer de privilèges dans le profil. Vous affectez ensuite le profil de droits à un rôle ou un utilisateur.

Des privilèges peuvent également être directement affectés à un utilisateur, un rôle ou un profil de droits. Si vous estimez qu'un sous-ensemble d'utilisateurs est suffisamment responsable pour utiliser un privilège au cours de sessions, vous pouvez lui affecter directement ce privilège. Les bons candidats à l'affectation directe sont les privilèges ayant un effet limité, tels que `proc_clock_highres`. Les mauvais candidats à l'affectation directe sont les privilèges ayant des effets importants, tels que `file_dac_write`. Pour plus d'informations, reportez-vous à la section “Considérations relatives à la sécurité lors de la section Attribution de droits” à la page 40.

Les privilèges peuvent également être refusés à un utilisateur, d'un rôle ou d'un business process. Procédez avec prudence lorsque vous supprimez des privilèges du jeu héritable initial ou du jeu limite d'un utilisateur ou d'un système.

Extension des privilèges d'un utilisateur ou d'un rôle

Les utilisateurs et les rôles disposent d'un jeu héritable de privilèges. Le jeu limite peut uniquement être réduit, car il contient initialement tous les privilèges. Le jeu héritable initial peut être étendu pour les utilisateurs, les rôles et les processus en affectant un privilège qui ne figure pas dans le jeu héritable.

Vous pouvez étendre les privilèges qui sont disponibles de trois manières :

- Un privilège qui ne figure pas dans le jeu héritable initial est dans le jeu limite mais peut être affecté à des utilisateurs et rôles. Peut être indirecte, l'affectation par le biais d'une commande dans un profil de droits doté de privilèges, ou il peut s'agir d'accès direct.

- Un privilège qui ne figure pas dans le jeu héritable peut être affecté de façon explicite à un processus, p. ex. l'ajout de privilèges à un script ou de l'application.
- Un privilège qui ne figure pas dans le jeu héritable mais figure dans le jeu limite peut être affecté de façon explicite à un port réseau, un ID utilisateur ou un objet de fichier. Cette utilisation du privilège est connu sous le nom de la stratégie *de privilège étendue et est également un moyen de la restriction de privilèges disponibles.* Pour plus d'informations, reportez-vous “L'utilisation d'Extended Privilege Policy to Utiliser Restreindre du privilège, procédez comme suit :” à la page 34

Pour l'affectation d'une tâche au lieu de recevoir uniquement le privilège pour cela, vous devez disposer du privilège d'administration est le moyen le plus précis pour développer des privilèges d'un utilisateur ou d'un rôle. Vous créez un profil de droits incluant la commande ou le script avec ses privilèges requis. Puis vous affectez ce profil de droits à un utilisateur ou un rôle. Permet d'autoriser l'utilisateur ou telles d'affectation de rôle pour exécuter ce commande privilégiée. Le privilège est indisponible à l'utilisateur.

Le développement du jeu héritable initial de privilèges pour des utilisateurs, des rôles ou des systèmes est une manière plus risquée d'affecter des privilèges. Tous les privilèges dans le jeu héritable figurent dans le jeu autorisé et le jeu effectif. Toutes les commandes que l'utilisateur ou le rôle tape dans un shell peuvent utiliser les privilèges affectés directement. Pour plus d'informations, reportez-vous à la section “Considérations relatives à la sécurité lors de la section Attribution de droits” à la page 40.

Disponibilité afin de réduire inutiles, vous pouvez affecter le privilège de privilèges *étendus pour les ports réseau, les ID utilisateur et les objets de fichier.* Enlever les privilèges tels d'affectation qui ne figurent pas dans la, l'affectation de de privilège étendue le jeu effectif. Pour plus de détails, reportez-vous “L'utilisation d'Extended Privilege Policy to Utiliser Restreindre du privilège, procédez comme suit :” à la page 34

La limitation des privilèges pour un utilisateur ou d'un rôle

Les privilèges et les droits d'accès les profils peuvent aussi être appliquée aux pour limiter les droits d'accès des utilisateurs non autorisés. En supprimant des privilèges permet d'empêcher les utilisateurs et les rôles d'exécuter certaines tâches. Vous pouvez supprimer des privilèges du jeu héritable initial et du jeu limite. Vous devez soigneusement tester la suppression de privilèges avant de distribuer un jeu héritable initial ou un jeu limite, plus petit que le jeu par défaut. En supprimant des privilèges du jeu héritable initial, vous risquez d'empêcher les utilisateurs de se connecter. Lorsque des privilèges sont supprimés du jeu limite, un ancien programme `setuid root` peut échouer s'il nécessite un privilège qui a été supprimé. Pour obtenir des exemples de leur suppression, reportez-vous à l'Example 3 privilège et l'exemple 5 -6 -21. Exemple 3-21, “Suppression de privilèges du jeu limite d'un utilisateur”Exemple 5-6, “Création d'un profil de droits pour les utilisateurs Sun Ray”

Pour limiter les privilèges à la disposition d'un utilisateur, ID ou objet de fichier, le port, reportez-vous à Utilisation de Restreindre Privilège Extended Privilege Policy to Use.

“L'utilisation d'Extended Privilege Policy to Utiliser Restreindre du privilège, procédez comme suit :” à la page 34

Affectation de privilèges à un script

Les scripts sont exécutables, tout comme les commandes. Par conséquent, dans un profil de droits, vous pouvez ajouter des privilèges à un script de la même façon que vous pouvez ajouter des privilèges à une commande. Le script s'exécute avec les privilèges ajoutés lorsqu'un utilisateur ou un rôle à qui le profil de droits a été affecté exécute le script dans un shell de profil. Si le script contient des commandes qui nécessitent des privilèges, les commandes avec les privilèges ajoutés doivent également figurer dans un profil de droits affecté. Pour consulter des exemples, reportez-vous à“Affecter des scripts et des droits aux applications” à la page 67

L'utilisation d'Extended Privilege Policy to Utiliser Restreindre du privilège, procédez comme suit :

La stratégie de privilège étendue peut limiter l'accès à des ports, les objets fichiers d'ID d'utilisateur, à l'exception du ou des privilèges de base ainsi que les privilèges qui vous octroyez explicitement. , de sorte que certains privilèges avec la ressource de type "attaque de l'intercepteur ne peuvent pas être facilement être utilisé pour le système. En fait, les utilisateurs peuvent protéger les fichiers et les répertoires qui leur appartiennent les processus à partir de l'accès par potentiellement malveillants. Pour obtenir des exemples de la stratégie de privilège étendue Applications, les scripts,“Limitation des applications, scripts et ressources à des droits spécifiques” à la page 67

Et droits utilisateurs privilège d'escalade

Oracle Solaris offre aux administrateurs une grande flexibilité pour configurer la sécurité. Selon la configuration de l'installation, le logiciel empêche l'escalade de privilèges. L'*escalade des privilèges* se produit lorsqu'un utilisateur ou un processus obtient plus de droits d'administration qu'il n'était prévu de lui en accorder. Dans ce sens, "privilege", et non pas uniquement signifie que tous les privilèges de noyau de droits. Reportez “Escalade de privilèges et privilèges de noyau” à la page 35

Le logiciel Oracle Solaris comprend les droits qui sont affectés au rôle `root` uniquement. Si d'autres systèmes de sécurité sont en place, l'administrateur peut affecter les attributs conçus pour le rôle `root` à d'autres comptes, mais il doit procéder avec prudence.

Les profils de droits et ensembles d'autorisations suivants peuvent escalader les privilèges à un compte non root :

- **Profil de droits Media Restore (Restauration des médias)** : ce profil existe, mais il ne fait partie d'aucun autre profil de droits. Dans la mesure où Media Restore permet d'accéder à l'ensemble du système de fichiers root, son utilisation est une escalade possible de privilège. Des fichiers délibérément modifiés ou des médias de substitution peuvent être restaurés. Par défaut, le rôle `root` inclut ce profil de droits.
- `solaris.*.assign` **Autorisations** : ces autorisations ne sont affectées à aucun profil de droits. Un compte avec une autorisation `solaris.*.assign` peut affecter à d'autres des droits qui ne lui sont pas affectés à lui-même. Par exemple, un rôle avec l'autorisation `solaris.profile.assign` peut affecter des profils de droits à d'autres comptes auxquels le rôle lui-même n'est pas affecté. Par défaut, seul le rôle `root` dispose des autorisations `solaris.*.assign`.

  Affecter les autorisations`solaris.*.delegate`, au lieu d'autorisations `solaris.*.assign`. Une autorisation `solaris.*.delegate` permet au délégant de n'attribuer à d'autres comptes que les droits qu'il possède. Par exemple, un rôle avec l'autorisation `solaris.profile.delegate` peut affecter à d'autres utilisateurs et rôles des profils de droits qui lui sont affectés.

Pour éviter l'escalade des privilèges de noyau reportez-vous à “Escalade de privilèges et privilèges de noyau” à la page 35.

Escalade de privilèges et privilèges de noyau

Le noyau empêche Escalade des privilèges. Pour empêcher un processus d'acquérir plus de privilèges que ceux qui lui sont accordés normalement, le noyau vérifie que les modifications de système vulnérable disposent du jeu complet de privilèges. Par exemple, un fichier ou un processus détenu par `root` (`UID=0`) ne peut être modifié que par un processus ayant le jeu complet de privilèges. Le compte `root` n'a pas besoin de privilèges pour modifier un fichier appartenant à `root`. Toutefois, un utilisateur non root doit avoir tous les privilèges pour modifier un fichier appartenant à `root`.

De même, les opérations permettant d'accéder aux périphériques requièrent tous les privilèges du jeu effectif.

Spécifiquement, les privilèges `file_chown_self` et `proc_owner` sont soumis à l'escalade de privilèges.

- Le privilège `file_chown_self` permet à un processus d'abandonner ses fichiers. Le privilège `proc_owner` permet à un processus d'examiner des processus dont il n'est pas propriétaire.

  Le privilège `file_chown_self` est limité par la variable système `rstchown`. Lorsque la variable `rstchown` est définie sur `0`, le privilège `file_chown_self` est supprimé du jeu héritable initial de tous les utilisateurs de l'image du système. Pour plus d'informations sur la variable système `rstchown`, reportez-vous à la page de manuel `chown`(1).

Le privilège `file_chown_self` est attribué à une commande particulière, la commande placée dans un profil de droits, et l'option de profil affectée à un rôle ou un utilisateur de confiance.

- Le privilège `proc_owner` n'est pas suffisant pour définir un processus UID sur `0`. Basculer d'un processus de n'importe quel UID à `UID=0` exige tous les privilèges. Etant donné que le privilège `proc_owner` donne un accès illimité en lecture à tous les fichiers du système, le privilège est attribué, pour des raisons de sécurité, à une commande particulière qui est ensuite placée dans un profil affecté à un rôle.

Attention - Le compte d'un utilisateur peut être modifié afin d'inclure le privilège `file_chown_self` ou `proc_owner` dans le jeu héritable initial de l'utilisateur. Cependant, vous devez avoir des raisons de sécurité de poids pour placer ces privilèges puissants dans n'importe quel utilisateur ou le jeu héritable du rôle.

Pour plus d'informations sur la manière d'empêcher l'escalade de privilèges pour des périphériques, reportez-vous à la section “Privilèges et périphériques” à la page 30. Pour une description générale, reportez-vous à la page de manuel `privileges`(5).

Droits de vérification

Le shell qu'un processus s'exécute dans, la portée du service de noms, mais l'ordre des droits qui vous sont affectés peuvent avoir une incidence sur l'affichage de recherche sont évaluées. Les processus dont les droits ne peuvent pas être évalués échouent. Pour obtenir de l'aide des affectations dans la vérification des droits, reportez-vous à la section “Dépannage de gestion des droits” à la page 107.

Shells de profil et des droits de vérification

Les utilisateurs et les rôles peuvent exécuter des applications privilégiées à partir d'un shell de profil. Un *shell de profil* est un shell spécial qui reconnaît les droits. Les administrateurs peuvent affecter un shell de profil à des utilisateurs spécifiques en tant que shell de connexion, ou bien le shell de profil est démarré lorsqu'un utilisateur exécute la commande `pfexec` ou `su` pour endosser un rôle. Dans Oracle Solaris, chaque shell dispose d'un shell de profil homologue. Pour obtenir une liste des shells de profil, reportez-vous à la page de manuel `pfexec`(1).

Les utilisateurs auxquels un profil de droits est assigné directement et dont le shell de connexion n'est pas un shell de profil doivent ouvrir un shell de profil pour exécuter les commandes privilégiées qui leur sont affectées. Les utilisateurs et les rôles auxquels est affecté

un profil de droits sont authentifiés vous êtes invité à vous authentifier, c'est - à - dire à saisir un mot de passe avant l'exécution de la commande est autorisé à exécuter. Pour plus d'informations concernant l'utilisation et la sécurité, reportez-vous à la“Eléments à prendre en compte lors de la section Attribution de droits” à la page 40.

Champ d'application du service de noms et des droits de vérification

Le champ d'application du service de noms affecte la disponibilité des droits attribués. Le champ d'application d'un rôle peut être limité à un hôte unique. Il peut également inclure tous les hôtes pris en charge par un service de noms, tel que LDAP. Le champ d'application du service de noms pour un système est indiqué dans le service de commutation de nom, `svc:/system/name-service/switch`. La recherche s'arrête à la première correspondance. Si, par exemple, un profil de droits existe dans deux champs d'application du service de noms, seules les entrées dans le premier champ d'application du service de noms sont utilisées. Si `files` est la première correspondance, le champ d'application de ce rôle est limité à l'hôte local. Pour plus d'informations sur les services de noms, reportez-vous à la page de manuel`nsswitch.conf`(4), “ Utilisation des services de noms et d’annuaire Oracle Solaris 11.2 : DNS et NIS ”, et au manuel “ Utilisation des services de noms et d’annuaire Oracle Solaris 11.2 : LDAP ”.

Ordre de recherche pour Assigned Rights

Un utilisateur ou un rôle peut se voir affecter Attributs de sécurité directement ou par le biais d'un profil de droits. L'ordre de recherche a une incidence sur la valeur d'attribut de sécurité utilisée. La valeur de la première instance trouvée de l'attribut est utilisée.

Remarque - L'ordre des autorisations n'est pas important. Les autorisations sont cumulatives.

Lorsqu'un utilisateur se connecte, droits sont affectés dans l'ordre de recherche suivant :

- **Droits** affectés directement à l'utilisateur avec les commandes `useradd` et `usermod`. Pour obtenir la liste des affectations de droits possibles, reportez-vous à “Base de données `user_attr`” à la page 121.
- **Profils de droits** affectés à l'utilisateur avec les commandes `useradd` et `usermod`. Ces affectations sont recherchées dans l'ordre.
  - Les profils de droits authentifiés sont tout d'abord recherchés.

    L'ordre est le suivant : le premier profil de la liste des profils authentifiés suivi de ses profils supplémentaires, le deuxième profil de la liste des profils authentifiés suivi de ses profils supplémentaires, etc. La première instance d'une valeur est celle que le système

utilise, à l'exception des valeurs `auths`, qui sont cumulatives. Les attributs qui peuvent être affectés à des profils de droits incluent tous les droits qui peuvent être affectés à des utilisateurs, plus les profils supplémentaires. Pour obtenir la liste, reportez-vous à la section “Base de données `user_attr`” à la page 121.

  - Ensuite, les profils de droits d'accès qui la réauthentification font l'objet d'une recherche ne nécessitent pas de la même manière.
- **Profil de droits Utilisateur Console**, valeur. Pour obtenir une description, reportez-vous à la section “Profils de droits Reference (disponible en anglais uniquement)” à la page 117.
- Si le **profil de droits Stop (arrêt)** est affecté, l'évaluation des attributs de sécurité s'arrête. Aucun attribut n'est affecté après l'affectation du profil Stop (arrêt). Le profil Stop est évalué après le profil de droits Console User (utilisateur de la console) et avant les autres attributs de sécurité dans le fichier `policy.conf`, y compris `AUTHS_GRANTED`. Pour obtenir une description, reportez-vous à la section “Profils de droits Reference (disponible en anglais uniquement)” à la page 117.
- Valeur **Profil de droits Basic Solaris User (utilisateur Solaris de base)** dans le fichier `policy.conf`.
- Valeur `AUTHS_GRANTED` dans le fichier `policy.conf`.
- Valeur `AUTHS_GRANTED` dans le fichier `policy.conf`.
- Valeur `PROFS_GRANTED` dans le fichier `policy.conf`.
- Valeur `PRIV_DEFAULT` dans le fichier `policy.conf`.
- Valeur `PRIV_LIMIT` dans le fichier `policy.conf`.

Applications vérifiant for Rights

Les applications et les commandes pouvant ignorer les contrôles système sont considérées comme des applications privilégiées. Les attributs de sécurité tels que `UID=0`, les privilèges et les autorisations rendent une application privilégiée.

Applications vérifiant les UID et GID

Les applications privilégiées vérifiant l'ID utilisateur `root` (`UID=0`) ou autres UID (ID utilisateur) ou GID (ID de groupe) spéciaux existent depuis longtemps dans l'environnement UNIX. Le mécanisme de profil de droits vous permet d'isoler les commandes nécessitant un ID spécifique. Au lieu de modifier l'ID sur une commande accessible par tous, vous pouvez placer la commande avec un UID affecté dans un profil de droits. Un utilisateur ou un rôle avec ce profil de droits peut alors exécuter le programme au titre de cet UID sans avoir à devenir superutilisateur.

Les ID peuvent être spécifiés comme *réels* ou *effectifs*. On préférera une affectation d'ID effectifs à une affectation d'ID réels. Les ID effectifs correspondent à la fonction `setuid` dans

les bits d'autorisation de fichier. Les ID effectifs permettent également d'identifier l'UID pour l'audit. Cependant, étant donné que certains programmes et scripts shell nécessitent un UID réel de `root`, il est également possible de définir des ID utilisateur réels. Par exemple, la commande `reboot` requiert un ID utilisateur réel plutôt qu'un ID utilisateur effectif.

Astuce - Si un ID effectif n'est pas suffisant pour exécuter une commande, vous devez affecter l'ID réel à la commande.

Applications vérifiant les privilèges

Les applications privilégiées peuvent vérifier l'utilisation de privilèges. Le mécanisme de profil de droits permet de spécifier les privilèges pour des commandes spécifiques qui requièrent des attributs de sécurité. Ensuite, vous pouvez isoler la commande avec des attributs de sécurité affectés dans un profil de droits. Un utilisateur ou un rôle doté de ce profil de droits peut ensuite exécuter la commande avec les privilèges nécessaires à la commande.

Voici la liste des commandes vérifiant les privilèges :

- Commandes Kerberos, telles que `kadmin`, `kprop` et `kdb5_util`
- Commandes réseau, telles que `ipadm`, `routeadm` et `snoop`
- Commandes de fichier et de système de fichiers, telles que `chmod`, `chgrp` et `mount`
- Commandes qui contrôlent les processus, telles que `kill`, `pcred` et `rcapadm`

Pour ajouter des commandes avec des privilèges à un profil de droits, reportez-vous à la section “Création d'un profil de droits” à la page 87 et à la page de manuel `profiles`(1). Pour déterminer quelles commandes vérifient les privilèges dans un profil particulier, reportez-vous au Chapter 6, Rights in Oracle Solaris sous forme de liste. Chapitre 6, Liste des droits dans Oracle Solaris

Applications vérifiant les autorisations

Voici la liste des commandes de Oracle Solaris qui vérifient les autorisations :

- Commandes d'administration d'audit, telles que `auditconfig` et `auditreduce`
- Commande d'administration d'imprimantes, telles que `cupsenable` et `lpadmin`
- Commandes de traitement batch, telles que `at`, `atq`, `batch` et `crontab`
- Commandes orientées périphérique, telles que `allocate`, `deallocate`, `list_devices` et `cdrw`.

Pour obtenir des instructions sur la vérification d'un script ou de programme dans le cadre des autorisations, reportez-vous à l'Example 4 -3. Exemple 4-3, “Recherche d'autorisations dans un script ou un programme” Pour écrire un programme nécessitant des autorisations, reportez-

vous à la rubrique “ About Authorizations ” du manuel “ Developer’s Guide to Oracle Solaris 11 Security ”.

Eléments à prendre en compte lors de la section Attribution de droits

Les problèmes de sécurité peuvent influer sur la manière dont l'utilisabilité et de les administrateurs des droits d'accès.

Considérations relatives à la sécurité lors de la section Attribution de droits

Obtenir en règle générale, les utilisateurs des droits d'administration ou des rôles l'intermédiaire d'un profil de droits, mais l'affectation directe de droits est également possible.

- Les privilèges peuvent être attribués directement aux utilisateurs et aux rôles.

  Toutefois, l'affectation directe de privilèges ne constitue pas une pratique sécurisée. Les utilisateurs et les rôles avec un privilège affecté peuvent remplacer la stratégie de sécurité partout où ce privilège est requis par le noyau. En outre, les processus malveillants qui compromettent le processus de rôle ou d'utilisateur peuvent utiliser ce privilège partout où il est requis par le noyau.

  Une pratique plus sécurisée consiste à attribuer le privilège en tant qu'attribut de sécurité d'une commande dans un profil de droits. Ce privilège n'est alors disponible que pour cette commande par un utilisateur doté de ce profil de droits.
- Les privilèges et les autorisations peuvent être attribués directement aux utilisateurs et aux rôles.

  Etant donné que les autorisations sont évaluées au niveau de l'utilisateur, l'affectation directe d'autorisations constitue un moindre risque que l'affectation directe de privilèges. Cependant, les autorisations peuvent permettent à un utilisateur d'effectuer des tâches hautement sécurisées, comme l'affectation d'indicateurs d'audit par exemple. Pour plus de sécurité, affectez des autorisations dans un profil de droits authentifié à l'endroit où l'utilisateur doit fournir un mot de passe avant l'exécution de la commande.

Considérations relatives à l'utilisation lors de la section Attribution de droits

L'affectation directe de droits peuvent avoir une incidence sur l'utilisation.

- Les autorisations affectées directement et les commandes et autorisations dans le profil de droits d'un utilisateur doivent être interprétés par un shell de profil pour être effectifs. Par défaut, un shell de profil n'est pas affecté aux utilisateurs. L'utilisateur ne doit donc pas oublier d'ouvrir un shell de profil, puis d'exécuter les commandes dans ce shell.
- L'affectation individuelle d'autorisations n'est pas évolutive. Les autorisations affectées directement peuvent ne pas suffire pour réaliser une tâche. Une tâche peut requérir des commandes privilégiées.

  Les profils de droits sont conçus pour lier des autorisations et des commandes privilégiées. De plus, ces derniers sont correctement redimensionnée à des groupes d'utilisateurs.

♦ ♦ ♦ CHAPITRE 2

Planification de la configuration de vos droits d'administration

Ce chapitre fournit des informations pour vous aider à décider si vous voulez utiliser un modèle de droits traditionnel ou si vous voulez profiter pleinement du modèle de droits d'Oracle Solaris lors de l'administration du système. Ce chapitre comprend les sections suivantes :

- “Modèle par rapport à décider Utiliser pour quel (s) de gestion des droits d'administration” à la page 43
- Modèle de gestion des “Modèle de gestion des droits de votre Sélectionné suivantes” à la page 45

Pour obtenir un aperçu des droits, reportez-vous à la section “Utilisateur Rights Management” à la page 14. Pour obtenir des informations de référence, reportez-vous au Chapitre 8, Droits Oracle Solaris (référence).

Modèle par rapport à décider Utiliser pour quel (s) de gestion des droits d'administration

Les droits dans Oracle Solaris incluent les profils de droits, les autorisations et les privilèges. Oracle Solaris offre différentes manières de configurer les droits d'administration sur un système.

La liste suivante est ordonnée du niveau le plus sécurisé au niveau le moins sécurisé du modèle de superutilisateur traditionnel.

1. Diviser les tâches d'administration entre plusieurs Utilisateurs de confiance ayant chacun des droits limités. Cette approche correspond au modèle de droits d'Oracle Solaris.

   Pour plus d'informations sur la façon de suivre cette approche reportez-vous à la section“Modèle de gestion des droits de votre Sélectionné suivantes” à la page 45.

   Pour en savoir plus sur les avantages de cette approche, reportez-vous au Chapitre 1, A propos de l'utilisation de droits pour contrôle Users and Processes.
2. Utilisez la configuration de droits par défaut. Le modèle de droits cette approche utilise mais n'effectue pas les personnaliser de façon à ce qu'il votre site.

Par défaut, l'utilisateur initial dispose de certains droits d'administration et peut prendre le rôle `root`. Le cas échéant, le rôle `root` peut attribuer le rôle `root` à un autre utilisateur de confiance. Pour plus de sécurité, le rôle `root` doit permettre l'audit des commandes d'administration.

Les tâches qui sont utiles pour les administrateurs qui utilisent ce modèle sont les suivants :

- “A l'aide de vos droits administratifs attribués” à la page 82
- “Attribution de droits aux utilisateurs” à la page 47
- “Actions administratives d'audit” à la page 86
- “Modification d'un mot de passe de rôle” à la page 55
- Chapitre 6, Liste des droits dans Oracle Solaris

3. Utilisation de la commande `sudo`.

   Les administrateurs qui maîtrisent la commande `sudo` peuvent configurer `sudo` et l'utiliser. Le cas échéant, ils peuvent configurer le fichier`/etc/sudoers` pour permettre aux utilisateurs de `sudo` d'exécuter des commandes d'administration sans réauthentification pendant une durée définie.

   Les tâches qui peuvent s'avérer utiles pour les utilisateurs de `sudo` sont les suivantes :

   - “A l'aide de vos droits administratifs attribués” à la page 82
   - “Actions administratives d'audit” à la page 86
   - Mise en cache de l'authentification : Exemple 5-2, “Mise en cache de l'authentification pour faciliter l'utilisation des rôles”

   La commande `sudo`n'est pas aussi implantée dans le noyau que les profils de droits. La commande s'exécute en tant qu'utilisateur `root` avec tous les privilèges, si bien qu'elle permet d'accorder à l'utilisateur en cours les droits qui sont spécifiés pour chaque programme dans le fichier `/etc/sudoers`. Bien que `sudo` ne permette pas d'indiquer les attributs des processus enfants suivants du programme, il peut en bloquer exécution. La version Oracle Solaris de `sudo` supprime le privilège `PRIV_PROC_EXEC` du processus. Pour plus d'informations, reportez-vous à la version pour Oracle Solaris de la page de manuel `sudo`(1M).

4. Utilisez le modèle de superutilisateur en transformant le rôle `root` en utilisateur.

   Les administrateurs qui utilisent le modèle traditionnel UNIX doivent suivre la procédure décrite dans la section “Modification du rôle `root` en utilisateur” à la page 93. Le cas échéant, l'utilisateur `root` peut configurer les audits.

Modèle de gestion des droits de votre Sélectionné suivantes

Et la gestion des droits de processus utilisateur peut faire partie intégrante de la gestion de vos systèmes de déploiement. La planification requiert une connaissance approfondie des exigences en matière de sécurité de votre organisation, ainsi qu'une bonne compréhension des droits dans Oracle Solaris. Cette section décrit un processus général permettant de planifier l'utilisation de droits du site.

1. Il est important que vous connaissiez les concepts de base sur les droits.

   Consultez le chapitre Chapitre 1, A propos de l'utilisation de droits pour contrôle Users and Processes. L'utilisation des droits pour administrer un système est très différente de l'utilisation des pratiques administratives UNIX conventionnelles.

2. Examinez votre stratégie de sécurité.

   La stratégie de sécurité de votre entreprise détaille les menaces potentielles pour votre système, mesure les risques de chaque menace et fournit des stratégies pour les contrer. L'isolation des tâches liées à la sécurité par le biais de droits peut faire partie de la stratégie.

   Par exemple, un site de sécurité peuvent exiger que vous distincte d'administration de l'administration à partir d'autres que la sécurité. Afin de mettre en oeuvre une séparation des tâches, reportez-vous à l'Exemple 3-3, “Création de rôles pour la séparation des tâches”.

   Si votre stratégie de sécurité repose sur des rôles ARMOR (Authorization Rules Managed On RBAC), vous devez installer et utiliser le package ARMOR. Pour plus d'informations sur son utilisation dans Oracle Solaris, voir l'Exemple 3-1, “Les rôles à l'aide de ARMOR”.

3. Passez en revue les profils de droits par défaut.

   Collecter la valeur par défaut les droits dont les profils de droits sont requises pour réaliser une tâche. Pour consulter les profils de droits disponibles, reportez-vous à la section “Liste des profils de droits” à la page 99

4. Décidez si vous avez prévu d'utiliser des rôles ou affecter des profils de droits à directement des utilisateurs.

   Les rôles peuvent simplifier l'administration de droits. Le nom du rôle rôle identifie les tâches qui isole le rôle est autorisé à effectuer et des droits d'utilisateur des droits à partir du Si vous avez prévu d'utiliser des rôles, vous avez le choix entre trois options :

   - Vous pouvez installer le package, qui installe le ARMOR sept géré rôles que l'd'autorisation ARMOR Rôles (RBAC) standard sur définit. Reportez-vous à la section Exemple 3-1, “Les rôles à l'aide de ARMOR”.
   - Vous pouvez définir vos propres rôles et d'employer ARMOR des rôles. Reportez-vous à la section “Création d'un rôle” à la page 49 and Exemple 3-1, “Les rôles à l'aide de ARMOR”.
   - Vous pouvez définir vos propres rôles et ne pas utiliser ARMOR des rôles. Reportez - vous à la section “Création d'un rôle” à la page 49.

Lorsque l'existence des rôles ne sont pas requis sur votre site, vous pouvez directement affecter des profils de droits aux utilisateurs. Pour exiger un mot de passe lorsque l'utilisateur effectue une tâche d'administration à partir de leurs authentifié les profils de droits, utilisez des profils de droits. Reportez-vous à la section Exemple 3-11, “Le mot de passe, qui requiert un avant l'utilisateur à entrer Administering DHCP”.

5. Décidez si vous devez créer d'autres profils de droits.

   Recherchez d'autres applications ou familles d'applications sur votre site susceptibles de bénéficier d'un accès limité. Les applications affectant la sécurité, pouvant entraîner des problèmes de déni de service ou nécessitant une formation d'administrateur système particulière constituent de bons candidats pour l'utilisation de droits. Par exemple, des utilisateurs de systèmes Sun Ray ne nécessite pas que tous les privilèges de base. Pour obtenir un exemple d'un profil de droits qui limite les utilisateurs, reportez-vous à l'Exemple 3-22, “Suppression d'un privilège de base d'un profil de droits”.

   a. Déterminez les droits nécessaires pour la nouvelle tâche.
   b. Le cas échéant, choisissez un profil de droits existant approprié pour cette tâche.
   c. Classez le nouveau profil de droits de sorte que les commandes s'exécutent avec les privilèges requis.

      Pour plus d'informations sur le classement, reportez-vous à la section “Ordre de recherche pour Assigned Rights” à la page 37.

6. Déterminez à quels utilisateurs doivent être affectés certains droits.

   Selon le principe du principe du moindre privilège, vous devez affecter des utilisateurs à des rôles adaptés à leur niveau de confiance. Lorsque vous empêchez les utilisateurs d'effectuer des tâches qu'ils n'ont pas besoin d'effectuer, vous réduisez les problèmes potentiels.

Remarque - Les droits qui s'appliquent à tous les utilisateurs d'une image système sont spécifiés dans le fichier `/etc/security/policy.conf` file.

Une fois que vous disposez d'un plan, créez des connexions pour des Utilisateurs de confiance auxquels peuvent être affectés des profils de droits ou des rôles. Pour plus d'informations sur la création d'utilisateurs, consultez la section “ Configuration et gestion des comptes utilisateur avec l’interface de ligne de commande ” du manuel “ Gestion des comptes utilisateur et des environnements utilisateur dans Oracle Solaris 11.2 ”.

Pour affecter des droits, commencez par les procédures décrites dans la section “Attribution de droits aux utilisateurs” à la page 47. Les sections qui suivent fournissent des exemples de droits, à l'aide de ces caractères, de droits Affectation de droits, définir des critères et la résolution de problèmes de droits pour les affectations de ressources.

♦ ♦ ♦ CHAPITRE 3

Affectation de droits dans Oracle Solaris

Ce chapitre décrit les tâches de Affectation de droits aux utilisateurs et aux rôles. Ce chapitre comprend les sections suivantes :

- “Attribution de droits aux utilisateurs” à la page 47
- Droits à l'“Droits à l'aide de ces caractères, des utilisateurs "” à la page 56
- Droits“Droits Restriction des utilisateurs "” à la page 61

Pour obtenir un aperçu des droits, reportez-vous à la section “Utilisateur Rights Management” à la page 14. Pour obtenir des informations de référence, reportez-vous au Chapitre 8, Droits Oracle Solaris (référence).

Attribution de droits aux utilisateurs

OSOL dans {ENT : droits existent sur chacun des processus.} Vous pouvez ajouter des droits aux utilisateurs et rôles, et la suppression des droits. Les droits comprennent des privilèges sur des privilèges au processus de l'utilisateur, les SIT, commande que celle invoquée par l'ID utilisateur sur un s'exécute, et les autorisations permettant d'effectuer une action donnée. Pour faciliter l'administration de l'affectation des droits, Oracle Solaris rassemble les droits relatifs aux services et aux actions d'administration dans des *profils de droits*. Droits au lieu d'affecter à des utilisateurs et à des rôles individuels, vous pouvez affecter un profil de droits incluant toutes les autorisations et privilèges de la tâche d'administration exige que.

Les rôles d'administration donner un nom au qu'un utilisateur est autorisé à effectuer de tâches, telles que auditadm sont. Pour effectuer une action d'administration, l'utilisateur endosse un rôle qui leur a été assigné pour exécuter l'action. Les rôles peuvent être requis par une stratégie de sécurité et vous conviennent le mieux. il leur suffit de être Vous pouvez créer des rôles ou installer le package `armor` qui crée sept rôles et leurs répertoires personnels locaux. Pour plus d'informations sur les rôles, reportez-vous à la“Fournir un élément de contrat et Processus de gestion des droits utilisateur la solution de substitution au modèle superutilisateur” à la page 14

Droits qui peut d'affectation, procédez comme suit :

Au départ, vous devez être dans le rôle root pour créer des utilisateurs disposant de droits ajouté.

Si le rôle roota procédé à une distribution des tâches d'administration pour ou sous forme d'un utilisateur de confiance vous par l'affectation d'un rôle à des affectations, les profils de droits d'accès suivants vous permettent de créer des utilisateurs et des rôles ou des droits d'accès à ces derniers :

- Pour créer un utilisateur ou un rôle, vous devez vous connecter en tant qu'administrateur disposant du profil de droits User Management (gestion des utilisateurs).
- Pour affecter la plupart des droits à un utilisateur ou à un rôle, vous devez vous connecter en tant qu'administrateur disposant du profil de droits User Security (sécurité des utilisateurs).

  Vous ne pouvez pas attribuer des indicateurs d'audit. Seul le `rôle` root peut attribuer des indicateurs d'audit à un utilisateur ou un rôle.

  Vous ne pouvez pas modifier le mot de passe d'un rôle. Seul le rôle root peut `modifier` le mot de passe d'un rôle.

Si vous disposez des droits d'administration, passez en revue par le“A l'aide de vos droits administratifs attribués” à la page 82

Affecter des droits pour des utilisateurs et des rôles

Cette section décrit les commandes permettant de créer et modifier les rôles et les utilisateurs. Pour créer ou modifier des profils de droits, reportez-vous“Création d'un profil de droits” à la page 87“Clonage et modification d'un profil de droits système” à la page 89

Pour plus d'informations sur les rôles, reportez-vous à Droits d'utilisateur et traitement notions de base relatives à Oracle. “Informations de base sur les droits des utilisateurs et les droits des processus” à la page 18

Les actions principales lors de la création de ainsi que la modification des rôles et les utilisateurs sont les suivantes :

- Création d'un rôle
- Création d'un utilisateur de confiance
- Modification des droits d'un rôle
- Modification des droits d'un utilisateur
- Afin de permettre aux utilisateurs d'utiliser leur propre mot de passe pour endosser un rôle
- Modification d'un mot de passe de rôle
- Suppression d'un rôle

Création d'un rôle

Si vous avez l'intention d'utiliser les rôles, vous disposez de plusieurs options. Vous pouvez installer le ARMOR les rôles prédéfinis à partir de et de les utiliser en mode exclusif. En outre, vous pouvez créer des rôles et leur attribuer les mots de passe. ARMOR les rôles avec vous pouvez utiliser les rôles que vous créez.

Pour utiliser, reportez-vous à l'Example 3 ARMOR -1 des rôles. Exemple 3-1, “Les rôles à l'aide de ARMOR”

Pour créer vos propres rôles, vous utilisez la commande. `roleadd` de cette commande, reportez-vous à la page de manuel `roleadd`(1M).

Par exemple, les commandes suivantes créent un rôle d'administrateur des utilisateurs local avec un répertoire d'accueil et un shell de connexion `pfbash`, puis créent un mot de passe pour le rôle :

```
# roleadd -c "User Administrator role, local" \
-m -K profiles="User Security,User Management"  useradm
80 blocks
# ls /export/home/useradm
local.bash_profile     local.login     local.profile
# passwd useradm
Password: xxxxxxxx
Confirm Password: xxxxxxxx
```

avec

`-c` , *commentaire*	Décrit le rôle.
`-m`	Crée un répertoire personnel.
`-K profiles=`	Affecte une ou plusieurs profils de droits au rôle. Pour la liste des profils de droits, reportez-vous à la section Listing Rights Profiles. “Liste des profils de droits” à la page 99
rolename	Nom du rôle. Pour connaître les restrictions sur les chaînes acceptables, reportez-vous à la page de manuel `roleadd`(1M).

Remarque - Un compte de rôle peut être affecté à plusieurs utilisateurs. Par conséquent, un administrateur crée généralement un mot de passe du rôle et fournit aux utilisateurs le mot de passe du rôle hors bande. Le rôle d'un autre mot de passe, reportez-vous à Enabling Users “Le mot de passe habilitation des utilisateurs pour l'autorisation d'utiliser son propre mot de passe du rôle” à la page 54Exemple 3-16, “Autorisation d'un utilisateur le mot de passe pour l'autorisation d'utiliser son propre mot de passe du rôle”Exemple 3-17, “Modification d'un profil de droits to Enable a User to Use Own Password pour mot de passe du rôle”

EXEMPLE 3-1 Les rôles à l'aide de ARMOR

Dans cet exemple, les rôles l'administrateur de sécurité installe ARMOR standard sont définis par le. L'administrateur vérifie premièrement que des noms de rôle ne sont pas en conflit avec les comptes existants, puis procède à l'installation du module, affiche les définitions de rôle, et affectez-les à aux utilisateurs de confiance.

Tout d'abord, l'administrateur s'assure que les ID utilisateur et les noms suivants n'existent pas dans le service de noms, procédez comme suit :

- `57 auditadm`
- `55 fsadm`
- `58 pkgadm`
- `53 secadm`
- `56 svcadm`
- `59 sysop`
- `54 useradm`

Après vérification que les UID et noms ne sont pas utilisés, l'administrateur installe le package.

```
# pkg install system/security/armor
```

Les rôles et le package sept locale (Home) crée les répertoires du répertoire. `/export/home`

Pour en visualiser les droits de chaque rôle, l'administrateur permet d'établir la liste des profils qui sont affectés à chaque rôle.

```
# profiles auditadm
# profiles fsadm
# profiles pkgadm
# profiles secadm
# profiles svcadm
# profiles sysop
# profiles useradm
```

Ces droits les affectations ne peuvent pas être modifié. Pour créer une configuration différente de droits, vous devez créer de nouveaux rôles, puis nouveaux profils de droits en suivant la procédure expliquée dans Comment clonage et modification d'un profil de droits système.
“Clonage et modification d'un profil de droits système” à la page 89

Enfin, l'administrateur affectez-les à aux utilisateurs de confiance. Mot de passe de l'utilisateur sont utilisées pour l'authentification par rapport au rôle. Certains utilisateurs sont d'affectation de plusieurs rôles. Les rôles dont les tâches sont liés à des échéances sont affectés à plus d'un utilisateur de confiance.

```
# usermod -R=auditadm adal
# usermod -R=fsadm,pkgadm bdewey
```

```
# usermod -R=secadm,useradm cfoure
# usermod -R=svcadm ghamada
# usermod -R=svcadm yjones
# usermod -R=sysop hmurtha
# usermod -R=sysop twong
```

EXEMPLE 3-2 Création d'un rôle d'administrateur des utilisateurs dans le référentiel LDAP

L'administrateur crée un rôle d'administrateur des utilisateurs dans LDAP. L'utilisateur fournit un mot de passe lorsqu'ils prennent le rôle, puis n'a pas besoin de fournir un mot de passe pour différentes commandes.

```
# roleadd -c "User Administrator role, LDAP" -m -S ldap \
-K profiles="User Security,User Management"  useradm
```

EXEMPLE 3-3 Création de rôles pour la séparation des tâches

L'administrateur crée deux rôles. Le rôle `usermgt` peut créer des utilisateurs, leur attribuer des répertoires d'accueil et effectuer d'autres tâches non liées à la sécurité. Le rôle usersec ne peut pas créer d'utilisateurs, mais peut affecter les affectations des mots de passe et modifier d'autres droits. Ni le rôle peut définir des indicateurs d'audit pour des utilisateurs ou des rôles, ou modifier un mot de passe du rôle. Le rôleroot doivent effectuer ces actions.

```
# roleadd -c "User Management role, LDAP" -s /usr/bin/pfksh \
-m -S ldap -K profiles="User Management"  usermgt
# roleadd -c "User Security role, LDAP" -s /usr/bin/pfksh \
-m -S ldap -K profiles="User Security"  usersec
```

L'administrateur s'assure que deux personnes sont nécessaires à la création dans l'exemple 3 chaque -5 utilisateur standard. Exemple 3-5, “L'ajout d'un rôle à un utilisateur”

EXEMPLE 3-4 Création et attribution d'un rôle pour administrer Cryptographic Services

Dans cet exemple, l'administrateur sur un réseau LDAP crée un rôle pour administrer la structure cryptographique et affecte le rôle à l'UID `1111`.

```
# roleadd -c "Cryptographic Services manager" \
-g 14 -m -u 104 -S ldap -K profiles="Crypto Management" cryptmgt
# passwd cryptmgt
New Password: xxxxxxxx
Confirm password: xxxxxxxx
# usermod -u 1111 -R +cryptmgt
```

L'utilisateur avec l'UID `1111` se connecte, puis endosse le rôle et affiche les attributs de sécurité affectés.

```
% su - cryptmgt
Password: xxxxxxxx
# profiles -l
      Crypto Management
          /usr/bin/kmfcfg            euid=0
          /usr/sbin/cryptoadm        euid=0
          /usr/sfw/bin/CA.pl         euid=0
          /usr/sfw/bin/openssl       euid=0
#
```

Pour plus d'informations sur la structure cryptographique Structure cryptographique d ',Chapitre 1, “ Structure cryptographique ” du manuel “ Gestion du chiffrement et des certificats dans Oracle Solaris 11.2 ” Pour l'administration de la structure, reportez-vous “ Administration de la structure cryptographique ” du manuel “ Gestion du chiffrement et des certificats dans Oracle Solaris 11.2 ”

Pour une création d'un utilisateur sécurisé de connexion

Vous utilisez la commande permettant de créer des informations de connexion. `useradd` Pour consulter la liste complète des arguments de la commande `useradd`, reportez-vous à la page de manuel `useradd`(1M). Les arguments liés aux droits de la commande sont similaires à la commande `roleadd`, avec l'ajout de l'option `-R` *rolename*.

Si vous affectez un rôle à un utilisateur, il pourra l'utiliser après avoir assumé le rôle de droits d'accès du rôle. Par exemple, la commande suivante crée un utilisateur de confiance useradm qui peut prendre le rôle que vous avez créé pour un dans la section Création d'un utilisateur sécurisé de connexion. “Pour une création d'un utilisateur sécurisé de connexion” à la page 52

```
# useradd -c "Trusted Assistant User Manager user" -m -R useradm jdoe
80 blocks
# ls /export/home/jdoe
local.bash_profile     local.login     local.profile
```

où

`-s` *shell* — Détermine le shell de connexion pour le nom d'utilisateur. Ce shell peut être un shell de profil, tel que `pfbash`. Pour savoir dans quels affecter un shell de profil à un utilisateur de confiance, reportez-vous à Considérations lors de la section Attribution de droits, qui ont été entièrement. “Considérations relatives à l'utilisation lors de la section Attribution de droits” à la page 40 Pour obtenir une liste des shells de profil, reportez-vous à la page de manuel `pfexec`(1).

`-R` *rolename* — Affecte le nom d'un rôle existant.

Pour voir d'autres exemples, consultez la section “ Configuration et gestion des comptes utilisateur avec l’interface de ligne de commande ” du manuel “ Gestion des comptes utilisateur et des environnements utilisateur dans Oracle Solaris 11.2 ”.

Modification des droits d'un utilisateur

Vous utilisez la commande`usermod` pour modifier un compte utilisateur. Pour consulter la liste complète des arguments de la commande `usermod`, reportez-vous à la page de manuel `usermod`(1M). Arguments la liés aux droits de la commande sont similaires à la commande. `useradd`

Si vous affectez un profil de droits à un utilisateur, l'utilisateur peut utiliser les droits d'accès nécessaires une fois que l'utilisateur ouvre un shell de profil. Par exemple, pour affecter un profil de droits à un utilisateur :

```
# usermod -K profiles="User Management" kdoe
```

Les modifications sont prises en compte à la prochaine connexion de l'utilisateur. Pour les utilisateurs pour plus d'informations sur l'utilisation, reportez-vous les droits leur étant affectés à la section Using Your Assigned Administrative Rights. “A l'aide de vos droits administratifs attribués” à la page 82

EXEMPLE 3-5 L'ajout d'un rôle à un utilisateur

Dans cet exemple, l'administrateur s'assure que deux requises pour créer des utilisateurs de confiance de l'autorisation des utilisateurs standard. La création des rôles dans l'exemple 3 -3. Exemple 3-3, “Création de rôles pour la séparation des tâches”

```
# usermod -R +useradm jdoe
# usermod -R +usersec mdoe
```

Modification des droits d'un rôle

Vous utilisez la commande permettant`rolemod` de modifier un compte de rôle. Pour consulter la liste complète des arguments de la commande `rolemod`, reportez-vous à la page de manuel `rolemod`(1M). Arguments la liés aux droits de la commande sont similaires à la commande. `roleadd`

Les valeurs de paireskey = `value`,ainsi que les options A, P et R peuvent être modifiées par le signe moins () ou plus (). ----+ Le signe - indique que la valeur doit être soustraite des

valeurs actuellement affectées. Le signe + indique que la valeur doit être ajoutée aux valeurs actuellement affectées. Pour les profils de droits, la valeur est ajouté au début de la liste actuelle des profils. Pour connaître les effets de profil de droits en cours, reportez-vous à une version antérieure à la section Order of Search for Assigned Rights. “Ordre de recherche pour Assigned Rights” à la page 37

EXEMPLE 3-6 Comme l'ajout d'un profil de droits de premier profil de droits de rôle

Par exemple, ajoutez le préfixe useradm un profil de droits à ce rôle, procédez comme suit :

```
# rolemod -K profiles+="Device Management" useradm
# profiles useradm
useradm:
Device Management
User Management
User Security
```

EXEMPLE 3-7 Affecté (e) en remplaçant les profils d'un rôle local

Dans cet exemple, l'administrateur de sécurité modifie le rôle `prtmgt` afin d'inclure le profil de droits VSCAN Management (gestion VSCAN) après le profil de droits Printer Management (Gestion d'imprimantes).

```
# rolemod -c "Handles printers and virus scanning" \
-P "Printer Management,VSCAN Management,All" prtmgt
```

EXEMPLE 3-8 Affectation de privilèges directement à un rôle

Dans cet exemple, l'administrateur de sécurité confie au rôle `realtime` un privilège très spécifique qui affecte le temps système. Pour affecter le privilège à un utilisateur, reportez-vous à l'Exemple 3-14, “Affectation de privilèges directement à un utilisateur”

```
# rolemod -K defaultpriv+='proc_clock_highres' realtime
```

Les valeurs pour le mot-clé `defaultpriv` se trouvent dans la liste des privilèges dans les processus du rôle à tout moment.

Le mot de passe habilitation des utilisateurs pour l'autorisation d'utiliser son propre mot de passe du rôle

Pour permettre aux utilisateurs d'utiliser leur propre mot de passe au lieu d'un mot de passe du rôle lorsqu'un utilisateur endosse un rôle, modifier le rôle.

La commande suivante vous permet d'activer tous les utilisateurs auxquels le rôle àutiliser leur propre useradm propre mot de passe lorsqu'ils endossent affectés à un rôle, y compris les useradm rôle.

```
# rolemod -K roleauth=user useradm
```

Modification d'un mot de passe de rôle

Dans la mesure où un rôle peut être assigné à de nombreux utilisateurs, les utilisateurs concernés ne peuvent pas modifier le mot de passe du rôle. Vous devez être dans le rôle root pour modifier un mot de passe du rôle.

```
# passwd useradm
Enter useradm's password: xxxxxxxx
New: xxxxxxxx
Confirm: xxxxxxxx
```

Si vous ne spécifiez pas de référentiel, le mot de passe est modifié dans tous les référentiels.

Pour obtenir d'autres options de commande, reportez-vous à la page de manuel `passwd`(1).

EXEMPLE 3-9 Modification du mot de passe d'un rôle dans un référentiel propres à

Dans l'exemple suivant, le rôle`root` modifie le mot de passe du rôle `devadmin` local.

```
# passwd -r files  devadmin
New password: xxxxxxxx
Confirm password: xxxxxxxx
```

Dans l'exemple suivant, le rôle`root` modifie le mot de passe du rôle `devadmin` dans le service de noms LDAP.

```
# passwd -r ldap devadmin
New password: xxxxxxxx
Confirm password: xxxxxxxx
```

Suppression d'un rôle

Lorsque vous supprimez un rôle, le rôle immédiatement devient inutilisable.

```
# roledel useradm
```

Les utilisateurs qui sont actuellement la réalisation de tâches d'administration dans le rôle n'ont pas la possibilité de se poursuivre. La commande `profiles`présente la sortie suivante :

```
useradm # profiles
Unable to get user name
```

Droits à l'aide de ces caractères, des utilisateurs "

Les tâches et les exemples de cette section les droits d'ajout aux droits que les utilisateurs reçoivent par défaut. Pour plus d'informations à propos des droit, reportez-vous au Chapitre 1, A propos de l'utilisation de droits pour contrôle Users and Processes.

- Affecter un rôle à un utilisateur de confiance Exemple-Exemple 3 -4 -1, Exemple, -5 3 3 Exemple 3-1, “Les rôles à l'aide de ARMOR”Exemple 3-4, “Création et attribution d'un rôle pour administrer Cryptographic Services”Exemple 3-5, “L'ajout d'un rôle à un utilisateur”
- Affecter un profil de droits à un utilisateur de confiance, Exemple-Exemple 3 -19 -10 4 3 -1, Exemple Exemple 3-10, “Création d'un utilisateur pouvant Administrer DHCP”Exemple 3-19, “L'activation d'un utilisateur sécurisé des fichiers à lire la comptabilisation étendue”Exemple 4-1, “Affecter des attributs de sécurité à une ancienne application”
- Affecter un profil de droits authentifiés à un utilisateur de confiance, - Exemple 3 -2 -11 l'Example 4 Exemple 3-11, “Le mot de passe, qui requiert un avant l'utilisateur à entrer Administering DHCP”Exemple 4-2, “Exécuter une application avec des droits attribués”
- Affecter une ou de rôle à un utilisateur de confiance, Exemple-Exemple 3 -13 -12 3 Exemple 3-12, “Affectation d'autorisations directement à un utilisateur”Exemple 3-13, “Affectation d'autorisations à un rôle”
- Affecter des privilèges directement à un utilisateur ou à un rôle, Exemple-Exemple 3 -14 -8 3 3 -15, Exemple Exemple 3-8, “Affectation de privilèges directement à un rôle”Exemple 3-14, “Affectation de privilèges directement à un utilisateur”Exemple 3-15, “De l'ajout de privilèges de base à un rôle”

Attention - L'utilisation inappropriée de privilèges et autorisations attribués directement peut entraîner des violations involontaires de sécurité. Pour plus de détails, reportez-vous à la section Security Considerations When la section Attribution de droits. “Considérations relatives à la sécurité lors de la section Attribution de droits” à la page 40

- Autoriser un utilisateur à utiliser propre mot de passe lorsqu'ils endossent un rôle, Exemple-Exemple 3 -17 -16 3 Exemple 3-16, “Autorisation d'un utilisateur le mot de passe pour l'autorisation d'utiliser son propre mot de passe du rôle”Exemple 3-17, “Modification d'un profil de droits to Enable a User to Use Own Password pour mot de passe du rôle”
- Modification d'un profil de droits -22 - Exemple 3 Exemple 3-22, “Suppression d'un privilège de base d'un profil de droits”
- Ajouter l'attribut de sécurité à une commande dans un profil de droits-Exemple 3 -27 -26, Exemple, Exemple -7 3 5 Exemple 3-26, “Lancement dynamique empêchant sélectionnés les nouveaux processus des applications à partir de”Exemple 3-27, “Les invités à partir

de lancement dynamique de l'éditeur, évitant ainsi que le sous-processus”Exemple 5-7, “Création d'un profil de droits qui inclut des commandes privilégiées”

- Activer l'utilisateur doit lire unepossédé par un utilisateur root, fichier Exemple-Exemple 3 -20 -19 3 Exemple 3-19, “L'activation d'un utilisateur sécurisé des fichiers à lire la comptabilisation étendue”Exemple 3-20, “Compte, de façon àdisposer d'un non-racine la route pour lire un fichier racine ”
- Permettre à des utilisateurs ou rôles possédé par un utilisateur root pour modifier un fichier -9 : Example 5 Exemple 5-9, “Clonage et suppression de droits sélectionnés d'un profil de droits”
- Attribution d'un profil de droits est dotée d'une nouvelle -11 Authozation (Autorisation) : l'Example 5 Exemple 5-11, “Ajout d'autorisations à un profil de droits”

EXEMPLE 3-10 Création d'un utilisateur pouvant Administrer DHCP

L'administrateur de sécurité crée un utilisateur qui peut administrer DHCP.

```
# useradd -P "DHCP Management" -s /usr/bin/pfbash -S ldap  jdoe
```

La mesure où pfbash est affecté, car l'utilisateur comme shell de connexion, les droits d'accès dans la profil de droits de gestion de DHCP sont toujours évaluées comme devant DHCP d'administration, de sorte que les commandes aboutissent.

EXEMPLE 3-11 Le mot de passe, qui requiert un avant l'utilisateur à entrer Administering DHCP

Dans cet exemple jdoe, l'administrateur de sécurité requiert avant de gérer des utilisateurs à saisir un mot de passe DHCP.

```
# usermod -K auth_profiles="DHCP Management" profiles="Edit Administrative Files" jdoe
```

Lorsque l'utilisateur `jdoe` saisit une commande DHCP, une invite de mot de passe s'affiche. Le DHCP `après` l'authentification d'exécution de la commande de jdoe,. En ordre de recherche, les profils de droits sont traitées avant authentifié profils standard.

```
jdoe% dhcpconfig -R 120.30.33.7,120.30.42.132
Password: xxxxxxxx
     /** Command completes **/
```

EXEMPLE 3-12 Affectation d'autorisations directement à un utilisateur

Dans cet exemple, l'administrateur de sécurité créer un utilisateur local qui peut contrôler la luminosité de l'écran.

```
# useradd -c "Screened KDoe, local" -s /usr/bin/pfbash \
-A solaris.system.power.brightness  kdoe
```

Cette autorisation est ajoutée aux affectations d'autorisation existantes de l'utilisateur.

EXEMPLE 3-13 Affectation d'autorisations à un rôle

Dans cet exemple, l'administrateur de sécurité crée un rôle pouvant modifiez les données de configuration du service du serveur DNS.

```
# roleadd -c "DNS administrator role" -m -A solaris.smf.manage.bind" dnsadmin
```

EXEMPLE 3-14 Affectation de privilèges directement à un utilisateur

Dans cet exemple, l'administrateur de sécurité confie à l'utilisateur `jdoe` un privilège très spécifique qui affecte le temps système. Pour savoir comment attribuer le privilège à un rôle, consultez l'Exemple 3-8, “Affectation de privilèges directement à un rôle”.

```
# usermod -K defaultpriv='basic,proc_clock_highres' kdoe
```

Les valeurs pour le mot-clé `defaultpriv` remplacent les valeurs existantes. Par conséquent, pour que l'utilisateur conserve les privilèges `basic`, la valeur `basic` est spécifiée. Dans la configuration par défaut, tous les utilisateurs disposent de privilèges de base. Pour obtenir la liste des privilèges, reportez-vous à la section “Liste des privilèges” à la page 102.

L'utilisateur peut visualiser que les nouvelles privilège et sa définition.

```
kdoe% ppriv -v $$
1800:   pfksh
flags = <none>
        E: file_link_any,…,proc_clock_highres,sys_ib_info
        I: file_link_any,…,proc_clock_highres,sys_ib_info
        P: file_link_any,…,proc_clock_highres,sys_ib_info
        L: cpc_cpu,dtrace_kernel,dtrace_proc,dtrace_user,...,win_upgrade_sl
% ppriv -vl proc_clock_highres
        Allows a process to use high resolution timers.
```

EXEMPLE 3-15 De l'ajout de privilèges de base à un rôle

Dans l'exemple suivant, des privilèges ont été directement attribués au rôle `realtime` pour gérer les programmes de date et d'heure. Vous avez affecté `proc_clock_highres` à `realtime` dans l'Exemple 3-8, “Affectation de privilèges directement à un rôle”.

```
# rolemod -K defaultpriv='basic,sys_time' realtime
```

```
% su - realtime
Password: xxxxxxxx
# ppriv -v $$
1600:   pfksh
flags = <none>
        E: file_link_any,...,proc_clock_highres,sys_ib_info,sys_time
        I: file_link_any,...,proc_clock_highres,sys_ib_info,sys_time
        P: file_link_any,...,proc_clock_highres,sys_ib_info,sys_time
```

```
L: cpc_cpu,dtrace_kernel,dtrace_proc,dtrace_user,...,sys_time
```

EXEMPLE 3-16 Autorisation d'un utilisateur le mot de passe pour l'autorisation d'utiliser son propre mot de passe du rôle

Par défaut, les utilisateurs doivent entrer le mot de passe du rôle pour prendre un rôle. En exigeant un mot de passe de l'utilisateur, l'administrateur rend la prise d'un rôle dans Oracle Solaris similaire à la prise d'un rôle dans un environnement Linux.

```
# rolemod -K roleauth=user auditrev
```

Pour endosser ce rôle, les utilisateurs affectés peuvent désormais utiliser leur propre mot de passe, et non pas le mot de passe qui a été spécifiquement créé pour le rôle.

Si d'autres rôles ont été affectés à l'utilisateur, celui-ci utilise son propre mot de passe pour ces rôles également.

EXEMPLE 3-17 Modification d'un profil de droits to Enable a User to Use Own Password pour mot de passe du rôle

```
# profiles -p "Local System Administrator"
profiles:Local System Administrator> set roleauth="user"
profiles:Local System Administrator> end
profiles:Local System Administrator> exit
```

Lorsqu'un utilisateur bénéficiant du profil de droits Local System Administrator (administrateur système local) veut endosser le rôle, cet utilisateur est invité à saisir un mot de passe. Dans la séquence suivante, le nom du rôle est `admin` :

```
% su - admin
Password: xxxxxxxx
#       /** You are now in a profile shell with administrative rights**/
```

EXEMPLE 3-18 Modification de la valeur `roleauth` pour un rôle dans le référentiel LDAP

Dans cet exemple, le rôle `root` permet à tous les utilisateurs pouvant prendre le rôle `secadmin` d'utiliser leur propre mot de passe lorsqu'ils prennent un rôle. Cette capacité est accordée à ces utilisateurs pour tous les systèmes qui sont gérés par le serveur LDAP.

```
# rolemod -S ldap -K roleauth=user secadmin
```

EXEMPLE 3-19 L'activation d'un utilisateur sécurisé des fichiers à lire la comptabilisation étendue

Vous pouvez activer un utilisateur ou un groupe d'utilisateurs de confiance de lire un fichier dont le propriétaire est le compte root. Ce droit peut être utile pour les utilisateurs qui peuvent

exécuter une application administrative possédé par un utilisateur root qui fait l'objet d'un fichier. Cet exemple ajoute un ou plusieurs scripts Perl net au profil de droits de gestion de la comptabilisation étendue.

Après avoir assumé le `rôle` root, l'administrateur crée un profil de droits qui ajoute la prise en charge la possibilité de lire le fichiers dont le nom commence par comptable. `network`

Le profil suivant utilise la stratégie de privilège étendue pour accorder le privilège file_dac_read à un script qui pourra ensuite accéder à des fichiers uniquement. `/var/adm/exacct/network*` Ce profil ajoute le profil de droits existant Extended Accounting Net Management (gestion de réseau de comptabilité étendu) en tant que profil de droits supplémentaire.

```
# profiles -p "Extended Accounting Perl Scripts"
profiles:Extended Accounting Perl Scripts >
set desc="Perl Scripts for Extended Accounting"
... Scripts> add profiles="Extended Accounting Net Management"
... Scripts> add cmd=/usr/local/bin/exacctdisp.pl
... Scripts:exacctdisp.pl> set privs={file_dac_read}:/var/adm/exacct/network*
... Scripts:exacctdisp.pl> end
... Scripts> commit
... Scripts> exit
```

Pour consulter des exemples de scripts, reportez - " Utilisation de l'interface Perl pour accéder à libexacct " du manuel " Administration de la gestion des ressources dans Oracle Solaris 11.2 "

Après avoir vérifié s'il existe des erreurs dans le profil de droits, tels que les entrées des erreurs typographiques, les omissions ou les répétitions, l'administrateur affecte un profil de droits la comptabilisation étendue scripts Perl à un rôle ou à un utilisateur.

```
# profiles -p "Extended Accounting Perl Scripts" info
Found profile in files repository.
name=Extended Accounting Perl Scripts
desc=Perl Scripts for Extended Accounting
profiles=Extended Accounting Net Management
cmd=/usr/local/bin/exacctdisp.pl
privs={file_dac_read}:/var/adm/exacct/network*
```

```
# rolemod -K profiles+="Extended Accounting Perl Scripts" rolename
# usermod -K profiles+="Extended Accounting Perl Scripts" username
```

EXEMPLE 3-20 Compte, de façon àdisposer d'un non-racine la route pour lire un fichier racine

Dans cet exemple, l'administrateur crée un profil de droits qui utilise la stratégie de privilège étendue des rôles pour permettre aux utilisateurs autorisés à lire le fichier et appartenant à root. `/var/adm/sulog` L'administrateur ajoute les commandes que l'utilisateur peut se servir pour lire le fichier. Commandes ne peuvent pas être utilisées non répertorié, telles que la commande. `head`

```
# profiles -p "Read sulog File"
profiles:Read sulog File
set desc="Read sulog File"
... File> add profiles="Read Log Files"
... File> add cmd=/usr/bin/cat
... File:cat> set privs={file_dac_read}:/var/adm/sulog
... File:cat> end
... File> add cmd=/usr/bin/less
... File:less> set privs={file_dac_read}:/var/adm/sulog
... File:less> end
... File> add cmd=/usr/bin/more
... File:more> set privs={file_dac_read}:/var/adm/sulog
... File:more> end
... File> add cmd=/usr/bin/page
... File:page> set privs={file_dac_read}:/var/adm/sulog
... File:page> end
... File> add cmd=/usr/bin/tail
... File:tail> set privs={file_dac_read}:/var/adm/sulog
... File:tail> end
... File> add cmd=/usr/bin/view
... File:head> set privs={file_dac_read}:/var/adm/sulog
... File:head> end
... File> commit
... File> exit
```

La commande `view`permet à l'utilisateur de lire un fichier mais pas pour la modifier.

Droits Restriction des utilisateurs "

Les exemples de cette limite de la section, les droits ou enlever des utilisateurs standard de certains droits d'administration de la part d'un administrateur. Elles indiquent comment modifier des utilisateurs, rôles et les profils de droits. Pour plus d'informations à propos des droit, reportez-vous au Chapitre 1, A propos de l'utilisation de droits pour contrôle Users and Processes.

- A partir d'un enlever les privilèges limites -21 utilisateur-Exemple 3 Exemple 3-21, “Suppression de privilèges du jeu limite d'un utilisateur”
- Enlever les privilèges d'un utilisateur de base-Exemple 3 -22 Exemple 3-22, “Suppression d'un privilège de base d'un profil de droits”
- Enlever les privilèges de base à partir de votre propre -23 du processus de shell-Exemple 3 Exemple 3-23, “Suppression d'un privilège de base d'vous-même”
- Pour créer un système -24 à usage limité - Exemple 3 Exemple 3-24, “Modification d'un système pour limiter les droits disponibles pour ses utilisateurs”
- Limiter un administrateur aux droits affectés de manière explicite -25 droits-Exemple 3 Exemple 3-25, “Limitation d'un administrateur aux droits affectés de manière explicite”

- Suppression des droits à partir de tous les utilisateurs d'un système, Exemple-Exemple 3 -28 -24 3 Exemple 3-24, “Modification d'un système pour limiter les droits disponibles pour ses utilisateurs”Exemple 3-28, “Profil de droits de l'affectation de l'Editeur de restrictions à tous les utilisateurs”
- Empêcher les applications d'-26 la création de sous-processus-Exemple 3 Exemple 3-26, “Lancement dynamique empêchant sélectionnés les nouveaux processus des applications à partir de”
- La génération dynamique d'empêcher les processus utilisateur des -27 les sous-processus-Exemple 3 Exemple 3-27, “Les invités à partir de lancement dynamique de l'éditeur, évitant ainsi que le sous-processus”
- Créer une liste restreinte-Exemple 3 éditeur -27 pour les invités Exemple 3-27, “Les invités à partir de lancement dynamique de l'éditeur, évitant ainsi que le sous-processus”
- Editeur affecter limitant l'accès à un -28 système public-Exemple 3 Exemple 3-28, “Profil de droits de l'affectation de l'Editeur de restrictions à tous les utilisateurs”
- Supprimer des privilèges du jeu limite d'un profil de droits -6 : Example 5 Exemple 5-6, “Création d'un profil de droits pour les utilisateurs Sun Ray”
- Création d'un profil de droits pour l'exemple 5 Sun Ray -6 Users (Utilisateurs) : Exemple 5-6, “Création d'un profil de droits pour les utilisateurs Sun Ray”
- Suppression des droits d'un profil de droits, Exemple : Example -9 -6 5 5 Exemple 5-6, “Création d'un profil de droits pour les utilisateurs Sun Ray”Exemple 5-9, “Clonage et suppression de droits sélectionnés d'un profil de droits”
- Supprimer une autorisation de l'Example 5 -10 par l'utilisateur Exemple 5-10, “Le test d'un nouveau d'autorisation”
- Supprimer un rôle d'affectation -13 : Example 5 Exemple 5-13, “Interdiction de l'utilisation du rôle `root` pour gérer un système”

EXEMPLE 3-21 Suppression de privilèges du jeu limite d'un utilisateur

Dans l'exemple suivant, toutes les sessions qui proviennent de la connexion initiale de `jdoe` sont empêchées d'utiliser le privilège `sys_linkdir`. L'utilisateur ne peut pas créer de liens physiques vers les répertoires ou rompre un lien vers des répertoires, même une fois que l'exécution de la commande. `su`

```
# usermod -K 'limitpriv=all,!sys_linkdir' jdoe
# userattr limitpriv jdoe
all,!sys_linkdir
```

EXEMPLE 3-22 Suppression d'un privilège de base d'un profil de droits

Dans l'exemple suivant, après des tests approfondis, l'administrateur de la sécurité supprime un autre privilège de base du profil de droits Sun Ray Users (utilisateurs Sun Ray). Lorsque l'administrateur a créé le profil de l'Exemple 5-6, “Création d'un profil de droits pour les utilisateurs Sun Ray”-6, l'administrateur a supprimé un privilège du jeu limite. Cette fois-ci ,

l'administrateur supprime deux privilèges de base. Les utilisateurs auxquels ce profil est affecté ne peuvent examiner aucun processus hors de leur session en cours, ni ajouter une autre session.

```
# profiles -p "Sun Ray Users"
profiles:Sun Ray Users> set defaultpriv="basic,!proc_session,!proc_info"
profiles:Sun Ray Users> end
profiles:Sun Ray Users> exit
```

EXEMPLE 3-23 Suppression d'un privilège de base d'vous-même

Dans l'exemple suivant, un utilisateur standard modifie proc_info de privilèges de base pour supprimer le. `.bash_profile` Et la sortie de contenir`ps` que des programmes comme propres`prstat` processus, de l'utilisateur qui peut mettent en évidence des informations utiles.

```
##  .bash_profile
## Remove proc_info privilege from my shell
##
ppriv -s EI-proc_info $$
```

`ppriv`supprime le privilège proc_info la ligne à partir `d'` effectif de l'utilisateur (EE et des jeux de privilèges héritable -) dans le processus de shell ($$) en cours.

Dans la sortie suivante à partir de, les `prstat`totaux pour l'opération de réduction des trois processus suivants : 74

```
## With all basic privileges
Total: 74 processes, 527 lwps, load averages: 0.01, 0.00, 0.00

## With proc_info removed from the effective and inheritable set
Total: 3 processes, 3 lwps, load averages: 0.00, 0.00, 0.00
```

EXEMPLE 3-24 Modification d'un système pour limiter les droits disponibles pour ses utilisateurs

Dans cet exemple, l'administrateur crée un système qui n'est utile que pour administrer le réseau. L'administrateur supprime le profil de droits Basic Solaris User (utilisateur de base Solaris) et toutes les autorisations du fichier `policy.conf`. Le profil de droits Console User (utilisateur de la console) n'est pas supprimé. Les lignes concernées dans le fichier `policy.conf` résultant sont les suivantes :

```
...
##AUTHS_GRANTED=
##AUTH_PROFS_GRANTED=
##PROFS_GRANTED=Basic Solaris User
CONSOLE_USER=Console User
...
```

Seul un utilisateur auquel des autorisations, commandes ou profils de droits sont explicitement affectés est capable d'utiliser ce système. Après la connexion, l'utilisateur autorisé peut effectuer

des tâches d'administration. Si l'utilisateur autorisé se trouve devant la console système, il dispose des droits de l'utilisateur de la console.

EXEMPLE 3-25 Limitation d'un administrateur aux droits affectés de manière explicite

Vous pouvez limiter un rôle ou un utilisateur à un nombre restreint d'actions d'administration de deux manières.

- Affectez le profil de droits Stop (arrêt) comme dernier profil dans la liste des profils de l'utilisateur.

  Ce profil constitue le moyen le plus simple de créer un shell limité. Les autorisations et les profils de droits dans le fichier ne sont pas affectés à l'utilisateur ou au rôle. `policy.conf`
- Vous pouvez modifier le fichier `policy.conf` sur un système et exiger que le rôle ou l'utilisateur utilisent ce système pour les tâches d'administration. Reportez-vous à la section Exemple 3-24, “Modification d'un système pour limiter les droits disponibles pour ses utilisateurs”.

La commande suivante permet de limiter le rôle `auditrev` aux révisions d'audit uniquement.

```
# rolemod -P "Audit Review,Stop" auditrev
```

Etant donné que le rôle `auditrev` ne dispose pas du profil de droits Console User (utilisateur de la console), l'auditeur ne peut pas arrêter le système. Etant donné que ce rôle ne dispose pas de l'autorisation `solaris.device.cdrw`, l'auditeur ne peut pas lire ou écrire sur l'unité de CD-ROM. Etant donné que ce rôle ne dispose pas du profil de droits de l'utilisateur Solaris de base des commandes à partir de ce profil, il n'y a pas de peut être exécutée dans ce rôle. Dans la mesure où le profil de droits Tous n'est pas affecté, la commande `ls` ne sera pas exécuté. Le rôle utilise le navigateur de fichiers pour sélectionner les fichiers d'audit à des fins de consultation.

Pour plus d'informations, reportez-vous à la“Ordre de recherche pour Assigned Rights” à la page 37“Profils de droits Reference (disponible en anglais uniquement)” à la page 117

EXEMPLE 3-26 Lancement dynamique empêchant sélectionnés les nouveaux processus des applications à partir de

Dans cet exemple, l'administrateur crée un profil de droits pour les sous-processus des applications qui n'ont pas besoin pour fonctionner correctement. Pour plus de commodité, l'administrateur crée un répertoire dans lequel stocker ces fichiers exécutables. Lorsque de nouvelles applications sont ajoutés, qui n'ont pas besoin de la version actuelle, les sous-processus peuvent être ajoutés à ce répertoire. Ou, si l'exécutable doit se trouver dans un répertoire spécifique, l'administrateur peut la lier à partir de app-exécutable . `/opt/local/noex/`

```
# profiles -p "Prevent App Subprocess"
```

```
profiles:Prevent App Subprocess> set desc="Keep apps from execing processes"
profiles:Prevent App Subprocess> add cmd=/opt/local/noex/mkmod
... Subprocess:mkmod> set limitprivs=all,!proc_exec
... Subprocess:mkmod> end
... Subprocess> add cmd=/opt/local/noex/gomap
... Subprocess:gomap> set limitprivs=all,!proc_exec
... Subprocess:gomap> end
... Subprocess> commit
... Subprocess> exit
```

EXEMPLE 3-27 Les invités à partir de lancement dynamique de l'éditeur, évitant ainsi que le sous-processus

Dans cet exemple, l'administrateur empêche les utilisateurs de créer à partir d'une ou plusieurs éditeurs subshells en supprimant le privilège de base proc_exec. a partir de l'éditeur de commande

1. L'administrateur crée un profil de droits qui supprime `proc_exec` du jeu limite de privilèges de l'éditeur `vim`.

   ```
   # profiles -p -S ldap "Editor Restrictions"
   profiles:Editor Restrictions> set desc="Site Editor Restrictions"
   ... Restrictions> add cmd=/usr/bin/vim
   ... Restrictions:vim> set limitprivs=all,!proc_exec
   ... Restrictions:vim> end
   ... Restrictions> commit
   ... Restrictions> exit
   ```

2. D'autres éditeurs de l'administrateur ajoute au profil de droits populaires.

   ```
   # profiles -p "Editor Restrictions"
   profiles:Editor Restrictions> add cmd=/usr/bin/gedit
   ... Restrictions:gedit> set limitprivs=all,!proc_exec
   ... Restrictions:gedit> end
   ... Restrictions> add cmd=/usr/bin/gconf-editor
   ... Restrictions:gconf-editor> set limitprivs=all,!proc_exec
   ... Restrictions:gconf-editor> end
   ... Restrictions> add cmd=/usr/bin/ed
   ... Restrictions:ed> set limitprivs=all,!proc_exec
   ... Restrictions:ed> end
   ... Restrictions> add cmd=/usr/bin/ex
   ... Restrictions:ex> set limitprivs=all,!proc_exec
   ... Restrictions:ex> end
   ... Restrictions> add cmd=/usr/bin/edit
   ... Restrictions:edit> set limitprivs=all,!proc_exec
   ... Restrictions:edit> end
   ... Restrictions> commit
   ```

```
... Restrictions> exit
```

3. Il recherche n'y a pas d'erreur dans le profil de droits, tels que les entrées des erreurs typographiques, les omissions ou les répétitions.

```
# profiles -p "Editor Restrictions" info
Found profile in files repository.
name=Editor Restrictions
desc=Site Editor Restrictions
cmd=/usr/bin/vim
limitprivs=all,!proc_exec
...
```

4. Restrictions l'administrateur attribue le profil de droits de l'éditeur pour l'utilisateur guest.

```
# usermod -K profiles+="Editor Restrictions" guest
```

   A l'aide de `profils` +, l'administrateur ajoute ce profil de droits d'accès au compte en cours de la base de données des profils de droits.

5. Des privilèges pour vérifier que l'éditeur sont limitées, l'administrateur ouvre l'éditeur et dans une fenêtre distincte, éditeur examine les privilèges associés au processus.

```
# ppriv -S $(pgrep vi)
2805:   vi .bash_profile
flags = PRIV_PFEXEC       User is running a profile shell
        E: basic,!proc_info       proc_info is removed from basic set
        I: basic,!proc_info
        P: basic,!proc_info
        L: all,!proc_exec       proc_exec is removed from limit set
```

EXEMPLE 3-28 Profil de droits de l'affectation de l'Editeur de restrictions à tous les utilisateurs

Dans cet exemple, l'administrateur ajoute le profil de droits à l'Editeur de fichier restrictions. `policy.conf` L'administrateur s'assure que ce fichier est distribué à tous les systèmes sur lesquels les invités public peuvent se connecter.

```
# cd /etc/security; cp policy.conf policy.conf.orig
# pfedit /etc/security/policy.conf
...
AUTHS_GRANTED=
AUTH_PROFS_GRANTED=
#PROFS_GRANTED=Basic Solaris User
PROFS_GRANTED=Editor Restrictions,Basic Solaris User
```

L'administrateur a affecté de la sécurité utilisateur d'un shell de profil pour tous les utilisateurs. La procédure sur la base des motifs et des droits aux utilisateurs de, reportez-vous aux rubriques Affectation. “Attribution de droits aux utilisateurs” à la page 47

♦ ♦ ♦ CHAPITRE 4

Affectation des droits aux applications, aux scripts et aux ressources

Ce chapitre traite des tâches qui s'appliquent les privilèges, la stratégie de privilège étendue, ainsi que d'autres droits aux utilisateurs, ainsi que les ports et les applications, procédez comme suit :

- “Affecter des scripts et des droits aux applications” à la page 67
- “Verrouiller les ressources à l'aide de privilèges étendus” à la page 70
- “Les utilisateurs verrouillent les applications qu'ils exécutent” à la page 77

Pour obtenir un aperçu des droits, reportez-vous à la section “Utilisateur Rights Management” à la page 14.

Limitation des applications, scripts et ressources à des droits spécifiques

Les tâches et les exemples de cette section affectent des privilèges aux exécutables et aux ressources système. En règle générale, vous affectez un privilège à un exécutable pour activer un utilisateur de confiance afin d'exécuter cet exécutable. Dans “Affecter des scripts et des droits aux applications” à la page 67, l'affectation d'un privilège permet à l'application ou au script d'être exécuté par un utilisateur de confiance dans un shell de profil. Dans “Verrouiller les ressources à l'aide de privilèges étendus” à la page 70, la stratégie de privilèges étendus limite un ID d'utilisateur, un port ou un objet de fichier à un jeu de privilèges plus réduit que le jeu par défaut. Les privilèges qui n'ont pas été spécifiés sont refusés aux, le port de processus de l'utilisateur ou l'objet. Une telle affectation correspond le moins à une stratégie de privilèges.

Affecter des scripts et des droits aux applications

Les scripts et les applications s'exécutent une commande ou une série de commandes. Pour affecter des droits, vous définissez les attributs de sécurité, tels que ID de jeu ou les

privilèges, pour chaque commande dans un profil de droits. Les applications peuvent vérifier les autorisations, le cas échéant.

Remarque - Si une commande d'un script doit avoir le bit `setuid` ou `setgid` défini pour s'exécuter correctement, les attributs de sécurité du fichier exécutable du script *et* de la commande doivent être ajoutés dans un profil de droits. Si le script est exécuté dans un shell de profil, la commande s'exécute avec les attributs de sécurité.

- Exécuter un script nécessitant des droits : “Exécution d'un script shell avec des commandes privilégiées” à la page 68
- Autoriser des applications prenant en charge les privilèges à être exécutées par des utilisateurs non `root` : Exemple 4-1, “Affecter des attributs de sécurité à une ancienne application”
- Autoriser des applications `root`à être exécutées par des utilisateurs non `root` : Exemple 4-2, “Exécuter une application avec des droits attribués”
- Vérifier les autorisations à l'aide d'un script : Exemple 4-3, “Recherche d'autorisations dans un script ou un programme”

▼ Exécution d'un script shell avec des commandes privilégiées

Pour exécuter un script shell privilégié, vous ajoutez des privilèges au script et aux commandes du script. Ensuite, le profil de droits approprié doit contenir les commandes avec les privilèges qui leur sont attribués.

Avant de commencer Vous devez prendre le rôle `root`. Pour plus d'informations, voir “A l'aide de vos droits administratifs attribués” à la page 82.

1. **Commencez le script avec `/bin/pfsh`, ou tout autre shell de profil, sur la première ligne.**

```
#!/bin/pfsh
# Copyright (c) 2013 by Oracle
```

2. **En tant qu'utilisateur standard, déterminez les privilèges requis pas les commandes du script.**

   Si le script est exécuté sans privilèges, l'option `debug` de la commande `ppriv` permet d'afficher les privilèges manquants.

```
% ppriv -eD script-full-path
```

   Pour plus d'informations, voir “Détermination des privilèges requis par un programme” à la page 113.

3. **Créez ou modifiez un profil de droits pour le script.**

Ajoutez le script de shell et les commandes dans le script de shell, avec les attributs de sécurité requis au profil de droits. Voir “Création d'un profil de droits” à la page 87.

4. **Attribuez le profil de droits à un utilisateur ou à un rôle de confiance.**

   Pour consulter des exemples, reportez-vous à la section “Attribution de droits aux utilisateurs” à la page 47.

5. **Pour exécuter le script, effectuez l'une des opérations suivantes :**

   - **Si l'on vous a affecté le script en tant qu'utilisateur, ouvrez un shell de profil, puis exécutez le script.**

     ```
     % pfexec script-full-path
     ```

   - **Si vous disposez du script en tant que rôle, prenez le rôle correspondant et exécutez le script.**

     ```
      % su - rolename
     Password: xxxxxxxx
     # script-full-path
     ```

Exemple 4-1 Affecter des attributs de sécurité à une ancienne application

Dans la mesure où une application ancienne ne prend pas en charge les privilèges, l'administrateur affecte l'attribut de sécurité `euid=0` à l'exécutable de l'application dans un profil de droits. L'administrateur attribue ensuite le profil à un utilisateur de confiance.

```
# profiles -p LegacyApp
profiles:LegacyApp> set desc="Legacy application"
profiles:LegacyApp> add cmd=/opt/legacy-app/bin/legacy-cmd
profiles:LegacyApp:legacy-cmd> set euid=0
profiles:LegacyApp:legacy-cmd> end
profiles:LegacyApp> exit
# profiles -p LegacyApp 'select cmd=/opt/legacy-app/bin/legacy-cmd;info;end'
 id=/opt/legacy-app/bin/legacy-cmd
 euid=0

# usermod -K profiles+="Legacy application" jdoe
```

Exemple 4-2 Exécuter une application avec des droits attribués

Dans cet exemple, l'administrateur affecte le profil de droits de l'Exemple 5-7, “Création d'un profil de droits qui inclut des commandes privilégiées” à un utilisateur de confiance. Celui-ci doit fournir un mot de passe pour exécuter le script.

```
# usermod -K auth_profiles+="Site application" jdoe
```

Exemple 4-3 Recherche d'autorisations dans un script ou un programme

Pour vérifier des autorisations, écrivez un test basé sur la commande `auths`. Pour plus d'informations sur cette commande, reportez-vous à la page de manuel `auths`(1).

Par exemple, la ligne suivante vérifie si l'utilisateur dispose de l'autorisation fournie comme argument `$1` :

```
if [ `/usr/bin/auths|/usr/xpg4/bin/grep $1` ]; then
        echo Auth granted
else
        echo Auth denied
fi
```

Un test plus complet doit comporter une routine recherchant des caractères génériques. Par exemple, pour tester si l'utilisateur dispose de l'autorisation `solaris.system.date`, vous devez rechercher les chaînes suivantes :

- `solaris.system.date`
- `solaris.system.*`
- `solaris.*`

Si vous écrivez un programme, utilisez la fonction `getauthattr()` pour effectuer un test d'autorisation.

Verrouiller les ressources à l'aide de privilèges étendus

Une stratégie de sécurité restrictive peut limiter l'accès des pirates à un ordinateur en cas d'attaque réussie sur une application. Une stratégie étendue règle limite l'étendue de l'effet de l'affectation de privilèges seulement les étapes intra-opérations une ressource dans la règle. Stratégie étendue les règles sont exprimées en plaçant les privilèges entre accolades, suivi de deux points et la ressource associée. Pour plus d'informations, reportez-vous à la section “Extension des privilèges d'un utilisateur ou d'un rôle” à la page 32. Pour des exemples de syntaxe, reportez-vous aux pages de manuel `ppriv`(1) et `privileges`(5).

Les utilisateurs standard peuvent aussi bien les administrateurs vers le bas et des ressources à l'aide de verrouillage des privilèges étendus. Les administrateurs peuvent créer de règles pour les utilisateurs de privilège étendue, les ports et d'applications. Les utilisateurs standard peuvent se servir de la ligne de commande ou de scripts utilisant la commande `ppriv -r` pour empêcher les applications d'écrire dans des fichiers situés hors des répertoires spécifiés par l'utilisateur.

- Limiter tout accès d'un utilisateur malveillant passant par un port : “Application d'une stratégie de privilège étendue à un port” à la page 71

- Exécuter une base de données en tant que démon non `root` : “Procédure de verrouillage du service MySQL” à la page 72
- Exécuter le serveur Web Apache en tant que démon non `root` : “Procédure d'assignation de privilèges propres au shell Apache du serveur Web” à la page 74
- Vérifier que le serveur Web Apache est en cours d'exécution avec des privilèges : “Procédure de détermination des privilèges que le serveur Web Apache utilise” à la page 76
- Empêchez Firefox d'écrire dans des répertoires de votre système : Exemple 4-4, “Exécution d'un navigateur dans un environnement protégé”
- Limiter les applications à des sous-répertoires spécifiques de votre système : Exemple 4-5, “Protection du système à partir de répertoires sur les processus d'application”

▼ Application d'une stratégie de privilège étendue à un port

Le service du protocole NTP (Network Time Protocol) utilise le port privilégié `123` pour le trafic `udp`. Il faut disposer des privilèges pour ce service à exécuter. Cet exemple de procédure modifie le manifeste du service afin d'empêcher l'accès à d'autres ports par un utilisateur malveillant susceptible d'obtenir les privilèges affectés aux ports en question.

Avant de commencer Vous devez prendre le rôle `root`. Pour plus d'informations, voir “A l'aide de vos droits administratifs attribués” à la page 82.

1. **Reportez-vous à l'entrée de manifeste de service par défaut pour le port.**

   A partir de l'entrée de méthode `/lib/svc/manifest/network/ntp.xml start`, les privilèges `net_privaddr`, `proc_lock_memory` et `sys_time` pourraient être utilisés sur d'autres processus :

   ```
   privileges='basic,!file_link_any,!proc_info,!proc_session,net_privaddr,
   proc_lock_memory,sys_time'
   ```

   Les privilèges supprimés spécifiés par `!file_link_any,!proc_info,!proc_session` empêchent le service de signaler ou d'observer tout autre processus, ou de créer des liens physiques afin de renommer des fichiers. Par conséquent, le processus démarré par le service ne peut établir de liaison qu'avec le port NTP `123`, c'est-à-dire avec aucun autre port privilégié. Si un pirate peut exploiter le service afin de démarrer un autre processus, celui-ci être la même façon limitée.

2. **Modifiez les méthodes `start` et `restart` afin de limiter le privilège `net_privaddr` à ce port uniquement.**

   ```
   # svccfg -s ntp editprop
   ```

   a. **Recherchez la chaîne `net_privaddr`.**

   b. **Supprimer la mise en commentaire des entrées qui contiennent `net_privaddr`.**

c. **Dans les deux entrées, remplacez `net_privaddr` par `{net_privaddr}:123/udp`.**
La stratégie de privilège étendue service supprime tous les privilèges octroyés sauf les privilèges spécifiés et les privilièges de base qui ne sont pas spécifiés. Par conséquent, l'ensemble des quatre-vingt privilèges exploitables est réduit à moins de huit.

3. **Redémarrez le service pour utiliser la stratégie de privilège étendu.**

```
# svcadm restart ntp
```

4. **Assurez-vous que le service utilise le privilège étendu.**

```
# svccfg -s ntp listprop | grep privileges
start/privileges    astring  basic,!file_link_any,!proc_info,!proc_session,
                    {net_privaddr}:123/udp,proc_lock_memory,sys_time
restart/privileges  astring  basic,!file_link_any,!proc_info,!proc_session,
                    {net_privaddr}:123/udp,proc_lock_memory,sys_time
```

▼ Procédure de verrouillage du service MySQL

Lors de l'installation, la base de données MySQL est configurée pour être exécutée avec les privilèges d'accès complet `root` sur un port non protégé. Dans cette tâche, vous affectez la stratégie de privilège étendu au service MySQL dans un profil de droits. Lorsque le profil de droits devient la méthode `exec` du service, MySQL s'exécute sur un port protégé en tant que l'utilisateur `mysql`, avec un accès limité à la base de données par les processus non MySQL.

Avant de commencer L'utilisateur initial peut installer le package. Les étapes restantes doivent être effectuées par le rôle `root`. Pour plus d'informations, voir “A l'aide de vos droits administratifs attribués” à la page 82.

1. **Installation du package MySQL.**

```
# pkg search basename:mysql
...
basename ... pkg:/database/mysql-51@version
# pfexec pkg install mysql-51
```

Remarque - Si vous mettez la base de données à niveau vers la version 5.5 de MySQL, modifiez toutes les étapes afin d'utiliser `5.5` et `55` au lieu de `5.1` et `51`.

2. **Affichez FMRI et l'état du service MySQL.**

```
# svcs mysql
STATE          STIME    FMRI
disabled       May_15   svc:/application/database/mysql:version_51
```

3. **Créez un profil de droits qui modifie la méthode d'exécution du service.**

Le manifeste de ce service spécifie que la méthode d'execution est un wrapper de script shell, `/lib/svc/method/mysql_51`.

```
# svccfg -s mysql listprop | grep manifest
... astring     /lib/svc/manifest/application/database/mysql_51.xml
# grep exec= /lib/svc/manifest/application/database/mysql_51.xml
                exec='/lib/svc/method/mysql_51 start'
                exec='/lib/svc/method/mysql_51 stop'
```

Utilisez le wrapper `/lib/svc/method/mysql_51` pour la commande dans le profil.

```
% su -
Password: xxxxxxxx
# profiles -p "MySQL Service"
MySQL Service> set desc="Locking down the MySQL Service"
MySQL Service> add cmd=/lib/svc/method/mysql_51
MySQL Service:mysql_51> set privs=basic
MySQL Service:mysql_51> add privs={net_privaddr}:3306/tcp
MySQL Service:mysql_51> add privs={file_write}:/var/mysql/5.1/data/*
MySQL Service:mysql_51> add privs={file_write}:/tmp/mysql.sock
MySQL Service:mysql_51> add privs={file_write}:/var/tmp/ib*
MySQL Service:mysql_51> end
MySQL Service> set uid=mysql
MySQL Service> set gid=mysql
MySQL Service> exit
```

Le privilège `file_write` est un privilège de base accordé par défaut à tous les processus. Lors de l'énumération explicite des chemins, l'accès en mode écriture est limité à ces chemins d'accès uniquement. Cette contrainte s'applique à l'exécutable et ses processus enfants.

4. Définissez le port par défaut pour MySQL comme port privilégié.

```
# ipadm set-prop -p extra_priv_ports+=3306 tcp
# ipadm show-prop -p extra_priv_ports tcp
PROTO PROPERTY            PERM CURRENT     PERSISTENT   DEFAULT    POSSIBLE
tcp   extra_priv_ports    rw   2049,4045,  3306         2049,4045  1-65535
                                 3306
```

Le privilège`net_privaddr` est obligatoire pour établir la liaison avec un port privilégié. Pour MySQL, se connecter au numéro de port par défaut, 3306, ne nécessite normalement pas ce privilège.

5. Affectez le profil de droits au service MySQL et dites au service de l'utiliser.

```
# svccfg -s mysql:version_51
...version_51> setprop method_context/profile="MySQLService"
...version_51> setprop method_context/use_profile=true
...version_51> refresh
...version_51> exit
```

6. Activez le service.

Le dernier composant du FMRI, `mysql:version_51`, est suffisant pour spécifier le service de manière unique.

```
# svcadm enable mysql:version_5
```

7. **(Facultatif) Vérifiez que le service est en cours d'exécution avec les droits spécifiés dans le profil de droits MySQL Service.**

```
# ppriv $(pgrep mysql)
103697:   /usr/mysql/5.1/bin/mysqld --basedir=/usr/mysql/5.1
                                    --datadir=/var/mysql/5.1/data
flags =   PRIV_XPOLICY
    Extended policies:
              {net_privaddr}:3306/tcp
              {file_write}:/var/mysql/5.1/data/*
              {file_write}:/tmp/mysql.sock
              {file_write}:/var/tmp/ib*
         E: basic,!file_write
         I: basic,!file_write
         P: basic,!file_write
         L: all
103609:  /bin/sh /usr/mysql/5.1/bin/mysqld_safe --user=mysql
                               --datadir=/var/mysql/5.1/data
flags =  PRIV_XPOLICY
    Extended policies:
              {net_privaddr}:3306/tcp
              {file_write}:/var/mysql/5.1/data/*
              {file_write}:/tmp/mysql.sock
              {file_write}:/var/tmp/ib*
         E: basic,!file_write
         I: basic,!file_write
         P: basic,!file_write
         L: all
```

▼ Procédure d'assignation de privilèges propres au shell Apache du serveur Web

Cette procédure verrouille le démon de serveur Web par le biais d'une affectation à celle-ci est uniquement les privilèges dont il a besoin. Le serveur Web ne peut être lié qu'au port 80 et peut uniquement effectuer des opérations d'écriture dans des fichiers dont le démon `webservd` est propriétaire. Aucun processus du service `apache22` ne s'exécute en tant que `root`.

Avant de commencer Vous devez prendre le rôle `root`. Pour plus d'informations, voir “A l'aide de vos droits administratifs attribués” à la page 82.

1. **Créez le profil de droits du serveur Web.**

```
# profiles -p "Apache2"
```

```
profiles:Apache2> set desc="Apache Web Server Extended Privilege"
profiles:Apache2> add cmd=/lib/svc/method/http-apache22
profiles:Apache2:http-apache22> add privs={net_privaddr}:80/tcp
...http-apache22> add privs={zone}:/system/volatile/apache2
...http-apache22> add privs={zone}:/var/apache2/2.2/logs/*
...http-apache22> add privs={zone}:/var/user
...http-apache22> add privs={file_write}:/var/user/webserv*
...http-apache22> add privs={file_write}:/tmp/*
...http-apache22> add privs={file_write}:/system/volatile/apache*
...http-apache22> add privs={file_write}:/proc/*
...http-apache22> add privs=basic,proc_priocntl
...http-apache22> set uid=webservd
...http-apache22> set gid=webservd
...http-apache22> end
---Apache2> exit
```

2. **(Facultatif) Si vous utilisez le service proxy SSL au niveau du noyau avec Apache 2, vous devez ajouter les ports SSL à votre stratégie étendue `webservd`.**

```
# profiles -p "Apache2"
profiles:Apache2> add privs={net_privaddr}:443/tcp
profiles:Apache2> add privs={net_privaddr}:8443/tcp
profiles:Apache2:http-apache22> end
```

La procédure du proxy SSL au niveau du noyau est décrite dans la section " Configuration d'un serveur Web Apache 2.2 afin qu'il utilise le proxy SSL au niveau du noyau " du manuel " Sécurisation du réseau dans Oracle Solaris 11.2 ".

3. **Ajoutez le profil de droits à la méthode de démarrage du service apache22.**

```
# svccfg -s apache22
svc:/network/http:Apache2> listprop start/exec
start/exec  astring "/lib/svc/method/http-apache22 start"
...
svc:/network/http:Apache2> setprop start/profile="Apache2"
svc:/network/http:Apache2> setprop start/use_profile=true
svc:/network/http:Apache2> refresh
svc:/network/http:Apache2> exit
```

Lorsque le service `apache22` est activé, le profil Apache 2 est utilisé.

4. **Activez le service `apache22`.**

```
# svcadm enable apache22
```

5. **Vérifiez que le serveur Web fonctionne.**

Ouvrez Firefox et tapez `localhost` dans le champ d'adresse.

Étapes suivantes Pour vérifier que ces privilèges s'appliquent correctement, reportez-vous à la section "Procédure de détermination des privilèges que le serveur Web Apache utilise" à la page 76.

▼ Procédure de détermination des privilèges que le serveur Web Apache utilise

Cette opération consiste à déterminer les privilèges par le serveur Web est de débogage à l'aide de la création d'un profil de droits version Apache 2.

Avant de commencer Vous avez terminé la section “Procédure d'assignation de privilèges propres au shell Apache du serveur Web” à la page 74. Le service `apache22` est désactivé. Vous êtes dans le rôle `root`.

1. **Clonez le profil Apache 2 pour appeler une autre commande.**

   Il est plus simple de déboguer une commande qu'un service SMF. La commande `apachectl`démarre le service Apache de façon interactive.

   ```
   # profiles -p "Apache2"
   profiles:Apache2> set name="Apache-debug"
   profiles:Apache-debug> sel <Tab><Tab>
   profiles:Apache-debug:http-apache22> set id=/usr/apache2/2.2/bin/apachectl
   profiles:Apache-debug:apachectl> end
   profiles:Apache-debug> exit
   ```

   Pour plus d'informations, reportez-vous à la page de manuel `apachectl`(8).

2. **Affectez le profil cloné au compte `webservd`.**

   ```
   # usermod -K profiles+=Apache-debug webservd
   ```

3. **Prenez l'identité`webservd`.**

   ```
   # su - webservd
   ```

4. **(Facultatif) Vérifiez l'identité.**

   ```
   # id
   uid=80(webservd) gid=80(webservd)
   ```

5. **Démarrez le service Web en mode de débogage dans un shell de profil.**

   N'utilisez pas SMF directement. Utilisez la commande du profil de droits`Apache-debug`.

   ```
   % pfbash
   # ppriv -De /usr/apache2/2.2/bin/apachectl start
   ```

6. **Dans le rôle `root`, examinez les privilèges du premier démon `http`.**

   ```
   # ppriv $(pgrep httpd|head -1)
   2999:   httpd
   flags = PRIV_DEBUG|PRIV_XPOLICY|PRIV_EXEC
     5         Extended policies:
     6                 {net_privaddr}:80/tcp
   ```

```
 7                {zone}:/system/volatile/apache2
 8                {zone}:/var/apache2/2.2/logs/*
 9                {zone}:/var/user
10                {file_write}:/var/user/webserv*
11                {file_write}:/tmp/*
12                {file_write}:/system/volatile/apache*
13                {file_write}:/proc/*
14        E: basic,!file_write,!proc_info,proc_priocntl
15        I: basic,!file_write,!proc_info,proc_priocntl
16        P: basic,!file_write,!proc_info,proc_priocntl
17        L: all
```

Les utilisateurs verrouillent les applications qu'ils exécutent

Les utilisateurs peuvent supprimer des privilèges de base à partir de demandes à l'aide de la stratégie de privilège étendue. La stratégie empêche l'accès aux répertoires auxquels les applications ne devraient pas accéder.

Remarque - L'ordre est important. Pour les répertoires tels que des `privilèges plus étendus $HOME / Téléchargerplus` restrictif * doit être de privilèges pour la plupart de transport affectés à $HOME /. les répertoires *.

EXEMPLE 4-4 Exécution d'un navigateur dans un environnement protégé

Cet exemple illustre la façon dont les utilisateurs peuvent exécuter le navigateur Firefox dans un environnement protégé. Dans cette configuration, le répertoire `Documents` de l'utilisateur est masqué pour Firefox.

A l'aide de la commande suivante, l'utilisateur supprime des privilèges de base de la commande. `/usr/bin/firefox` Les arguments de privilèges étendus de la commande `ppriv -r` limitent le navigateur Web à la lecture et l'écriture uniquement dans les répertoires spécifiés par l'utilisateur. L'option -e et ses arguments ouvrent le navigateur avec la stratégie de privilèges étendus.

```
% ppriv -r "\
{file_read}:/dev/*,\
{file_read}:/etc/*,\
{file_read}:/lib/*,\
{file_read}:/usr/*,\
{file_read}:/var/*,\
{file_read}:/proc,\
{file_read}:/proc/*,\
{file_read}:/system/volatile/*,\
```

```
{file_write}:$HOME,\
{file_read}:$HOME/.*,\
{file_read,file_write}:$HOME/.mozill*,\
{file_read,file_write}:$HOME/.gnome*,\
{file_read,file_write}:$HOME/Downloa*,\
{file_read,file_write}:/tmp,\
{file_read,file_write}:/tmp/*,\
{file_read,file_write}:/var/tmp,\
{file_read,file_write}:/var/tmp/*,\
{proc_exec}:/usr/*\
" -e /usr/bin/firefox file:///$HOME/Desktop
```

Lorsque les privilèges `file_read` et `file_write` sont utilisés dans une stratégie étendue, il est nécessaire d'accorder un accès explicite à chaque fichier dans lequel il doit être possible de lire ou d'écrire. L'utilisation du caractère générique `*` est essentielle dans des stratégies de ce type.

Pour gérer des répertoires d'accueil montés automatiquement, l'utilisateur ajoute une entrée explicite au chemin de montage automatique, par exemple :

```
{file_read,file_write}:/export/home/$USER
```

Si le site n'utilise pas `automount`, la liste initiale des répertoires protégés est suffisante.

Les utilisateurs peuvent automatiser ce navigateur Web par ligne de commande en créant un interpréteur de commandes protégé par un script. Pour lancer un navigateur, l'utilisateur appelle alors le script et non pas la commande. `/usr/bin/firefox`.

EXEMPLE 4-5 Protection du système à partir de répertoires sur les processus d'application

Dans cet exemple, un utilisateur standard crée un modèle d'environnement restreint pour les applications wrapper à l'aide d'un script shell. La première partie du script quelques répertoires les limites des infos de paramétrage d'applications. Les exceptions, tels que Firefox, sont traitées plus loin dans le script. Script des commentaires sur suivez les parties du script.

```
1 #!/bin/bash
2
3 # Using bash because ksh misinterprets extended policy syntax
4
5 PATH=/usr/bin:/usr/sbin:/usr/gnu/bin
6
7 DENY=file_read,file_write,proc_exec,proc_info
8
9 SANDBOX="\
10 {file_read}:/dev/*,\
11 {file_read}:/etc/*,\
12 {file_read}:/lib/*,\
13 {file_read,file_write}:/usr/*,\
14 {file_read}:/proc,\
15 {file_read,file_write}:/proc/*,\
```

```
16 {file_read}:/system/volatile/*,\
17 {file_read,file_write}:/tmp,\
18 {file_read,file_write}:/tmp/*,\
19 {file_read,file_write}:/var/*,\
20 {file_write}:$HOME,\
21 {file_read}:$HOME/.*,\
22 {file_read,file_write}:$PWD,\
23 {file_read,file_write}:$PWD/*,\
24 {proc_exec}:/usr/*\
25 "
26
27 # Default program is restricted bash shell
28
29 if [[ ! -n $1 ]]; then
30     program="/usr/bin/bash --login --noprofile
       --restricted"
31 else
32     program="$@"
33 fi
34
35
36 # Firefox needs more file and network access
37 if [[ "$program" =~ firefox ]]; then
38     SANDBOX+=",\
39 {file_read,file_write}:$HOME/.gnome*,\
40 {file_read,file_write}:$HOME/.mozill*,\
41 {file_read,file_write}:$HOME/.dbu*,\
42 {file_read,file_write}:$HOME/.puls*\
43 "
44
45 else
46     DENY+=",net_access"
47 fi
48
49 echo Starting $program in sandbox
50 ppriv -s I-$DENY -r $SANDBOX -De $program
```

La stratégie peuvent être modifiés de façon à autoriser l'accès des applications spécifiques plus ou moins. Un ajustement aux lignes 38 à 42, où Firefox a des droits d'accès en écriture à plusieurs fichiers points qui conservent les informations de session dans le répertoire de l'utilisateur. En outre, Firefox n'est pas soumis à la ligne 46, ce qui supprimer l'accès au réseau. Toutefois, Firefox est toujours limitée de lire des fichiers arbitraires sur le répertoire personnel de l'utilisateur et peut enregistrer des fichiers uniquement dans son répertoire en cours.

En tant que niveau supplémentaire de protection, le programme par défaut, à la ligne 30, est un shell Bash limité. Un shell restreint n'est plus possible de changer son répertoire en cours ou cliquez sur le point de l'utilisateur des fichiers. Par conséquent, toutes les commandes qui s'y trouvent sont aussi démarrées à partir du verrouillé dans le shell le modèle d'environnement restreint.

Dans la dernière ligne du script, la commande `ppriv` reçoit en argument deux jeux de privilèges sous forme de variables de shell, `$DENY` et `$SANDBOX`.

Le premier ensemble, `$DENY`, empêche le processus de lire ou d'écrire dans un fichier, d'exécuter un sous-processus, d'observer les processus d'autres utilisateurs et, de manière conditionnelle, d'accéder au réseau. Ces restrictions sont trop importantes, c'est pourquoi dans le second jeu, `$SANDBOX`, la stratégie est affinée en énumérant des répertoires qui sont disponibles pour la lecture, l'écriture et l'exécution.

En outre, l'option de débogage `-D` est spécifiée à la ligne 50. Les défaillances d'accès s'affichent dans la fenêtre de terminal en temps réel et inclure le privilège objet nommé et correspondant, ce dernier étant requis pour indiquer la réussite. Cette des informations de débogage pour aider l'utilisateur à personnaliser la stratégie destinée à d'autres applications.

♦ ♦ ♦ CHAPITRE 5

Gestion de l'utilisation des droits

Ce chapitre traite des tâches permettant de tenir à jour des systèmes qui utilisent le modèle de droits pour l'administration. Plusieurs tâches {ENT étendre : les droits qui en créant de nouveaux OSOL} fournit les profils de droits et les autorisations.

Ce chapitre comprend les sections suivantes :

Pour plus d'informations à propos des droit, reportez-vous au Chapitre 1, A propos de l'utilisation de droits pour contrôle Users and Processes. Pour plus d'informations sur la mise à jour des droits affectés à des utilisateurs et des rôles, reportez-vous au Chapitre 3, Affectation de droits dans Oracle Solaris.

Gestion de l'utilisation des droits

Les tâches et les exemples de cette section indiquent comment utiliser les droits qui vous ont été affectés, et comment modifier la configuration des droits fournie par défaut.

Remarque - Pour consulter des informations sur l'assistance de dépannage, reportez-vous à la section "Dépannage de gestion des droits" à la page 107.

- Utilisez les droits qui vous sont affectés "A l'aide de vos droits administratifs attribués" à la page 82
- Actions administratives d'audit – Exemple 5-5, "Utiliser deux rôles pour la configuration d'un audit"
- Ajout de profils de droits et d'autorisations :"La création des profils de droits et autorisations" à la page 87
- Configurer`root` pour être utilisateur "Modification du rôle `root` en utilisateur" à la page 93

- Faites passer `root` à un rôle – Exemple 5-12, “Transformation de l'utilisateur `root` en rôle `root`”
- Empêchez `root` d'administrer un système – Exemple 5-13, “Interdiction de l'utilisation du rôle `root` pour gérer un système”

A l'aide de vos droits administratifs attribués

Dans le rôle `root`, l'utilisateur initial dispose de tous les droits d'administration. En tant qu'`utilisateur` root, cet utilisateur peut affecter les droits d'administration, tels que les rôles, un profil de droits, des privilèges et des autorisations à ou spécifique aux utilisateurs de confiance. Cette section décrit la façon dont ces utilisateurs peuvent se servir de leurs droits affectés.

Remarque - Oracle Solaris fournit un éditeur spécial pour les fichiers d'administration. Lorsque vous modifiez les fichiers d'administration, utilisez la commande. `pfedit` Exemple 5-1, “La modification d'un système de fichiers” montre comment activer les utilisateurs non `root` pour modifier les fichiers système spécifiés.

Pour effectuer vos tâches administratives, ouvrez une fenêtre de terminal et choisissez l'une des options suivantes :

- Si vous utilisez `sudo`, saisissez la commande `sudo`.

  Pour les administrateurs qui maîtrisent la commande,`sudo` exécutez la commande avec le nom d'une commande d'administration, qui vous ont été affectés dans le fichier. `sudoers`
  Pour plus d'informations, consultez les pages de manuel `sudo`(1M) et `sudoers`(4).
- Si votre tâche exige des privilèges de superutilisateur, devenez l'utilisateur `root`.

```
% su -
Password: xxxxxxxx
#
```

Remarque - Cette méthode fonctionne aussi bien lorsque `root` est un utilisateur que lorsqu'il est un rôle. L'invite avec le signe dièse (#) indique que vous êtes maintenant l'utilisateur `root`.

- Si votre tâche est affectée à un rôle, prenez le rôle qui ne peuvent pas effectuer cette tâche.

  Dans l'exemple suivant, vous prenez un rôle de configuration d'audit. Ce rôle inclut le profil de droits Audit Configuration (configuration d'audit). Vous avez reçu le mot de passe du rôle à votre administrateur.

```
% su - audadmin
```

```
Password: xxxxxxxx
#
```

Astuce - Si vous n'avez pas reçu un mot de passe du rôle, cela signifie que l'administrateur a configuré le rôle de façon à exiger votre mot de passe utilisateur. Saisissez votre mot de passe de l'utilisateur à assumer un rôle. Pour plus d'informations sur cette option, reportez-vous à l'Exemple 3-16, “Autorisation d'un utilisateur le mot de passe pour l'autorisation d'utiliser son propre mot de passe du rôle”.

Le shell dans lequel vous avez tapé cette commande est désormais un shell de profil. Dans ce shell, vous pouvez exécuter la commande `auditconfig`. Pour en savoir plus sur les shells de profil, reportez-vous à la section “Shells de profil et des droits de vérification” à la page 36.

Astuce - Pour en visualiser les droits de votre rôle, reportez-vous à la section “Liste des profils de droits” à la page 99

- Si votre tâche vous est directement attribuée, créez un shell de profil de l'une des manières suivantes :
  - Utilisez la commande `pfbash` pour créer un shell qui évalue les droits d'administration.

    Dans l'exemple suivant, le profil de droits Audit Configuration (configuration d'audit) vous a été directement assigné. Par exemple, le jeu de commandes ci-après vous permet de visualiser les valeurs de présélection d'audit et la stratégie d'audit dans le shell de profil `pfbash` :

    ```
    % pfbash
    # auditconfig -getflags
    active user default audit flags = ua,ap,lo(0x45000,0x45000)
    configured user default audit flags = ua,ap,lo(0x45000,0x45000)
    # auditconfig -getpolicy
    configured audit policies = cnt
    active audit policies = cnt
    ```

  - Utilisez la commande `pfexec` pour exécuter une commande d'administration.

    Dans l'exemple suivant, le profil de droits Audit Configuration (configuration d'audit) vous a directement été assigné en tant que profil de droits authentifié. Vous pouvez exécuter une commande privilégiée de ce profil à l'aide de la commande avec le nom de cette commande. `pfexec` Par exemple, vous pouvez visualiser les indicateurs d'audit présélectionnés de l'utilisateur, procédez comme suit :

    ```
    % pfexec auditconfig -getflags
    Enter password:       Type your user password
    active user default audit flags = ua,ap,lo(0x45000,0x45000)
    ```

```
configured user default audit flags = ua,ap,lo(0x45000,0x45000)
```

En règle générale, pour exécuter une autre commande privilégiée concernée par les droits, vous devez entrer une nouvelle fois avant de taper la commande privilégiée.

`pfexec` Pour plus d'informations, reportez-vous à la page de manuel `pfexec`(1). Si vous êtes configurés avec une mise ne cache de mot de passe, vous pouvez exécuter les commandes suivante dans un intervalle configurable sans fournir de mot de passe, comme l'indique l'Exemple 5-2, “Mise en cache de l'authentification pour faciliter l'utilisation des rôles”.

EXEMPLE 5-1 La modification d'un système de fichiers

Si vous n'êtes pas connecté en tant qu'utilisateur `root` avec l'ID utilisateur `0`, par défaut, vous ne pouvez pas modifier les fichiers système. Toutefois, si l'autorisation `solaris.admin.edit/`*`path-to-system-file`* vous est affectée, vous pouvez modifier *system-file* . Par exemple, si l'autorisation `solaris.admin.edit/etc/security/audit_warn` vous a été attribuée, vous pouvez modifier le fichier `audit_warn` à l'aide de la commande `pfedit`.

```
# pfedit /etc/security/audit_warn
```

Pour plus d'informations, reportez-vous à la page de manuel `pfedit`(4). Cette commande est destiné à être utilisé par tous les administrateurs.

EXEMPLE 5-2 Mise en cache de l'authentification pour faciliter l'utilisation des rôles

Dans cet exemple, l'administrateur configure un rôle pour gérer la configuration d'audit et facilite son utilisation mettant en cache l'authentification de l'utilisateur. Tout d'abord, l'administrateur crée et affecte le rôle.

```
# roleadd -K roleauth=user -P "Audit Configuration" audadmin
# usermod -R +audadmin jdoe
```

Quand `jdoe` utilise l'option `-c` lors du changement de rôle, un mot de passe est requis avant que la sortie `auditconfig` ne s'affiche :

```
% su - audadmin -c auditconfig option
Password: xxxxxxxx
     auditconfig output
```

Si l'authentification n'est pas mise en cache, lorsque `jdoe` réexécute la commande, une invite de mot de passe s'affiche.

L'administrateur crée un fichier `dans` le répertoire pam.d pour contenir une de pile d'authentification qui permet la mise en cache. `su` Lorsque celle-ci est mis en mémoire cache, le mot de passe est requis initialement mais pas par la suite jusqu'à ce qu'une certaine quantité de temps s'est écoulé.

```
# pfedit /etc/pam.d/su
## Cache authentication for switched user
#
auth required           pam_unix_cred.so.1
auth sufficient         pam_tty_tickets.so.1
auth requisite          pam_authtok_get.so.1
auth required           pam_dhkeys.so.1
auth required           pam_unix_auth.so.1
```

Après avoir créé le fichier, l'administrateur vérifie les fautes de frappe, omissions ou répétitions dans les entrées.

L'administrateur doit fournir la pile `su` précédente entière. La module `pam_tty_tickets.so.1` met en oeuvre le cache. Pour plus d'informations sur PAM, consultez les pages de manuel `pam_tty_tickets`(5) et `pam.conf`(4), ainsi que le Chapitre 1, “ Utilisation de modules d’authentification enfichables ” du manuel “ Gestion de Kerberos et d’autres services d’authentification dans Oracle Solaris 11.2 ”.

Une fois que l'administrateur a ajouté le fichier PAM `su` et réinitialisé le système, tous les rôles, y compris le rôle `audadmin`, sont invités une seule fois à saisir un mot de passe lors de l'exécution d'une série de commandes.

```
% su - audadmin -c auditconfig option
Password: xxxxxxxx
     auditconfig output
% su - audadmin -c auditconfig option
     auditconfig output
...
```

EXEMPLE 5-3 Prise du rôle `root`

Dans l'exemple suivant, l'utilisateur initial prend le rôle `root` et répertorie les privilèges dans le shell du rôle.

```
% roles
root
% su - root
Password: xxxxxxxx
#       Prompt changes to root prompt
# ppriv $$
1200:   pfksh
flags = <none>
        E: all
        I: basic
        P: all
        L: all
```

Pour plus d'informations sur les privilèges, reportez-vous à la page de manuel “Processus Rights Management” à la page 24 and the `ppriv`(1).

EXEMPLE 5-4 Prise du rôle ARMOR

Dans cet exemple, l'utilisateur endosse un rôle que l'administrateur a attribué ARMOR.

Dans une fenêtre de terminal, l'utilisateur détermine les rôles attribués.

```
% roles
fsadm
sysop
```

L'utilisateur endosse le rôle `fsadm` et fournit le mot de passe de l'utilisateur.

```
% su - fsadm
Password: xxxxxxxx
#
```

La commande `su - `*rolename* remplace le shell du terminal par un shell de profil. L'utilisateur est désormais le rôle `fsadm` dans cette fenêtre de terminal.

Pour déterminer quelles commandes ne peut être exécutée dans ce rôle, l'utilisateur suit les instructions à la section Listing Rights Profiles. “Liste des profils de droits” à la page 99

Actions administratives d'audit

Exige que la stratégie de sécurité du site pour effectuer l'audit souvent des actions d'administration. L'événement d'audit `116:AUE_PFEXEC:execve(2) with pfexec enabled:ps,ex,ua,as` capture ces actions. La métaclasse `cusa`, qui fournit un groupe d'événements qui ne sont adaptés pour être utilisés avec les rôles, est une autre solution pour gérer l'audit des actions d'administration. Pour plus d'informations, reportez-vous aux commentaires dans le fichier `/etc/security/audit_class`

EXEMPLE 5-5 Utiliser deux rôles pour la configuration d'un audit

Dans cet exemple, deux administrateurs implémentent le plan de configuration d'audit du responsable local de la sécurité du site. Le plan consiste à utiliser la classe `pf` pour tous les utilisateurs et indiquer la métaclasse `cusa` pour les rôles individuels. Le `rôle` root affecte les indicateurs d'audit à des rôles. Le premier administrateur configure l'audit et la seconde permet la nouvelle configuration.

Le premier administrateur se voit attribuer le profil de droits Audit Configuration (configuration d'audit). Cet administrateur affiche la configuration d'audit en cours, procédez comme suit :

```
# auditconfig -getflags
active user default audit flags = lo(0x1000,0x1000)
configured user default audit flags = lo(0x1000,0x1000)
```

La classe`pf` n'inclut pas la classe `lo`, l'administrateur ajoute la classe à la configuration système.

```
# auditconfig -setflags lo,pf
```

Pour lire la nouvelle configuration d'audit dans le noyau, l'administrateur disposant du profil de droits Audit Control (contrôle d'audit) actualise le service d'audit.

```
# audit -s
```

La création des profils de droits et autorisations

Vous pouvez créer ou modifier un profil de droits lorsque les profils de droits fournis ne contiennent pas l'ensemble des droits dont vous avez besoin. Vous pouvez créer un profil de droits pour les utilisateurs ayant des droits limités dans le cadre d'une nouvelle application, ou différentes à d'autres motifs.

Les profils de droits fournis par Oracle Solaris sont en lecture seule. Vous pouvez cloner un profil de droits fourni à des fins de modification si son ensemble de droits est insuffisant. Par exemple, vous pouvez ajouter l'autorisation `solaris.admin.edit/` *path-to-system-file* à un profil de droits fourni. Pour plus d'informations, reportez-vous à la section “En savoir plus sur les profils de droits” à la page 22.

Vous pouvez créer une autorisation lorsque les autorisations fournies ne sont pas pris en compte dans plus d'informations sur les autorisations sont codées sur votre des applications privilégiées. Vous ne pouvez pas modifier une autorisation. Pour plus d'informations, reportez-vous à la section “En savoir plus sur les autorisations utilisateur” à la page 22.

▼ Création d'un profil de droits

Avant de commencer Pour créer un profil de droits, vous devez vous connecter en tant qu'administrateur disposant du profil de droits File Security (sécurité des fichiers). Pour plus d'informations, voir “A l'aide de vos droits administratifs attribués” à la page 82.

1. **Création d'un profil de droits.**

   ```
   # profiles -p [-S repository] profile-name
   ```

   Vous êtes invité à saisir une description.

2. **Ajoutez du contenu au profil de droits.**

   Utilisez la sous-commande `set` pour les propriétés de profil qui possèdent une valeur unique, comme `set desc`. Utilisez la sous-commande `add` pour les propriétés qui peuvent posséder plusieurs valeurs, comme `add cmd`.

La commande suivante crée le profil de droits PAM personnalisé dans la section " Assignation d'une stratégie PAM modifiée " du manuel " Gestion de Kerberos et d'autres services d'authentification dans Oracle Solaris 11.2 ". Le nom est raccourci à des fins d'affichage.

```
# profiles -p -S LDAP "Site PAM LDAP"
profiles:Site PAM LDAP> set desc="Profile which sets pam_policy=ldap"
...LDAP> set pam_policy=ldap
...LDAP> commit
...LDAP> end
...LDAP> exit
```

Exemple 5-6 Création d'un profil de droits pour les utilisateurs Sun Ray

Dans cet exemple, l'administrateur crée un profil de droits pour les utilisateurs Sun Ray dans le référentiel LDAP. L'administrateur a déjà créé une version Sun Ray du profil de droits Basic Solaris User (utilisateur Solaris de base), et a supprimé tous les profils de droits du fichier `policy.conf` sur le serveur Sun Ray.

```
# profiles -p -S LDAP "Sun Ray Users"
profiles:Sun Ray Users> set desc="For all users of Sun Rays"
... Ray Users> add profiles="Sun Ray Basic User"
... Ray Users> set defaultpriv="basic,!proc_info"
... Ray Users> set limitpriv="basic,!proc_info"
... Ray Users> end
... Ray Users> exit
```

L'administrateur vérifie le contenu.

```
# profiles -p "Sun Ray Users" info
Found profile in LDAP repository.
        name=Sun Ray Users
        desc=For all users of Sun Rays
        defaultpriv=basic,!proc_info,
        limitpriv=basic,!proc_info,
        profiles=Sun Ray Basic User
```

Exemple 5-7 Création d'un profil de droits qui inclut des commandes privilégiées

Dans cet exemple, l'administrateur de la sécurité ajoute des privilèges à une application dans un profil de droits créé par l'administrateur. L'application est consciente des privilèges.

```
# profiles -p SiteApp
profiles:SiteApp> set desc="Site application"
profiles:SiteApp> add cmd="/opt/site-app/bin/site-cmd"
profiles:SiteApp:site-cmd> add privs="proc_fork,proc_taskid"
profiles:SiteApp:site-cmd> end
profiles:SiteApp> exit
```

Pour vérifier, l'administrateur sélectionne la commande `site-cmd`.

```
# profiles -p SiteApp "select cmd=/opt/site-app/bin/site-cmd; info;end"
Found profile in files repository.
  id=/opt/site-app/bin/site-cmd
  privs=proc_fork,proc_taskid
```

Étapes suivantes Attribuez le profil de droits à un utilisateur ou à un rôle de confiance. Pour consulter des exemples, reportez-vous à l'Exemple 3-10, “Création d'un utilisateur pouvant Administrer DHCP” et à l'Exemple 3-19, “L'activation d'un utilisateur sécurisé des fichiers à lire la comptabilisation étendue”.

Voir aussi Pour dépanner les attributions de droits, reportez-vous à la section, “Dépannage d'attributions de droits” à la page 107. Pour plus d'informations, reportez-vous à la section “Ordre de recherche pour Assigned Rights” à la page 37.

▼ Clonage et modification d'un profil de droits système

Avant de commencer Pour créer ou modifier un profil de droits, vous devez vous connecter en tant qu'administrateur disposant du profil de droits File Security (sécurité des fichiers). Pour plus d'informations, voir “A l'aide de vos droits administratifs attribués” à la page 82.

1. **Créez un nouveau profil de droits à partir d'un profil existant.**

   `# profiles -p [-S` *repository*`]` *existing-profile-name*

   - **Pour ajouter du contenu à un profil de droits existant, créez un profil.**
     Ajoutez dans celui-ci le profil de droits existant en tant que profil de droits supplémentaire, puis ajoutez les améliorations. Reportez-vous à la section Exemple 5-8, “Clonage et optimisation du profil de droits Network IPsec Management (gestion IPsec du réseau)”.

   - **Pour supprimer du contenu d'un profil de droits existant, clonez ce dernier et renommez-le et apportez les modifications.**
     Reportez-vous à la section Exemple 5-9, “Clonage et suppression de droits sélectionnés d'un profil de droits”.

2. **Modifier le nouveau profil de droits en ajoutant ou en supprimant des profils de droits supplémentaires, les autorisations et d'autres droits.**

Exemple 5-8 Clonage et optimisation du profil de droits Network IPsec Management (gestion IPsec du réseau)

Dans cet exemple, l'administrateur ajoute une autorisation `solaris.admin.edit` à un profil de droits IPsec Management de site, si bien que le rôle `root` n'est pas requis. Ce profil de droits sera affecté uniquement aux utilisateurs autorisés à modifier le fichier `/etc/hosts`.

1. L'administrateur vérifie que le profil de droits Network IPsec Management ne peut pas être modifié.

```
# profiles -p "Network IPsec Management"
profiles:Network IPsec Management> add auths="solaris.admin.edit/etc/hosts"
Cannot add. Profile cannot be modified
```

2. Il crée ensuite un profil de droits incluant le profil Network IPsec Management.

```
# profiles -p "Total IPsec Mgt"
... IPsec Mgt> set desc="Network IPsec Mgt plus /etc/hosts"
... IPsec Mgt> add profiles="Network IPsec Management"
... IPsec Mgt> add auths="solaris.admin.edit/etc/hosts"
... IPsec Mgt> end
... IPsec Mgt> exit
```

3. L'administrateur vérifie le contenu.

```
# profiles -p "Total IPsec Mgt" info
        name=Total IPsec Mgt
        desc=Network IPsec Mgt plus /etc/hosts
        auths=solaris.admin.edit/etc/hosts
        profiles=Network IPsec Management
```

Exemple 5-9 Clonage et suppression de droits sélectionnés d'un profil de droits

Dans cet exemple, l'administrateur sépare la gestion des propriétés du service VSCAN de la capacité à activer et désactiver le service.

Tout d'abord, l'administrateur répertorie le contenu du profil de droits fourni par Oracle Solaris.

```
# profiles -p "VSCAN Management" info
        name=VSCAN Management
        desc=Manage the VSCAN service
        auths=solaris.smf.manage.vscan,solaris.smf.value.vscan,
              solaris.smf.modify.application
        help=RtVscanMngmnt.html
```

L'administrateur crée ensuite un profil de droits pour activer et désactiver le service.

```
# profiles -p "VSCAN Management"
profiles:VSCAN Management> set name="VSCAN Control"
profiles:VSCAN Control> set desc="Start and stop the VSCAN service"
... VSCAN Control> remove auths="solaris.smf.value.vscan"
... VSCAN Control> remove auths="solaris.smf.modify.application"
... VSCAN Control> end
... VSCAN Control> exit
```

Ensuite, l'administrateur crée un profil de droits pouvant modifier les propriétés du service.

```
# profiles -p "VSCAN Management"
profiles:VSCAN Management> set name="VSCAN Properties"
profiles:VSCAN Properties> set desc="Modify VSCAN service properties"
... VSCAN Properties> remove auths="solaris.smf.manage.vscan"
```

```
... VSCAN Properties> end
... VSCAN Properties> exit
```

L'administrateur vérifie le contenu des nouveaux profils de droits.

```
# profiles -p "VSCAN Control" info
        name=VSCAN Control
        desc=Start and stop the VSCAN service
        auths=solaris.smf.manage.vscan
# profiles -p "VSCAN Properties" info
        name=VSCAN Properties
        desc=Modify VSCAN service properties
        auths=solaris.smf.value.vscan,solaris.smf.modify.application
```

Étapes suivantes Attribuez le profil de droits à un utilisateur ou à un rôle de confiance. Pour consulter des exemples, reportez-vous à l'Exemple 3-10, “Création d'un utilisateur pouvant Administrer DHCP” et à l'Exemple 3-19, “L'activation d'un utilisateur sécurisé des fichiers à lire la comptabilisation étendue”.

Voir aussi Pour dépanner les attributions de droits, reportez-vous à la section, “Dépannage d'attributions de droits” à la page 107. Pour plus d'informations, reportez-vous à la section “Ordre de recherche pour Assigned Rights” à la page 37.

▼ Création d'une autorisation

Avant de commencer Les développeurs avez défini et utilisé l'autorisation dans les applications que vous êtes en train d'installer. Pour obtenir des instructions, reportez-vous au manuel “ Developer’s Guide to Oracle Solaris 11 Security ” et à la section “ About Authorizations ” du manuel “ Developer’s Guide to Oracle Solaris 11 Security ”.

1. (Facultatif) Créez le fichier d'aide de la nouvelle autorisation.

Par exemple, créez le fichier d'aide d'une autorisation pour permettre à l'utilisateur de modifier les données dans une application.

```
# pfedit /docs/helps/NewcoSiteAppModData.html
<HTML>
-- Copyright 2013 Newco.  All rights reserved.
-- NewcoSiteAppModData.html
-->
<HEAD>
     <TITLE>NewCo Modify SiteApp Data Authorization</TITLE>
</HEAD>
<BODY>
The com.newco.siteapp.data.modify authorization authorizes you
to modify existing data in the application.
<p>
Only authorized accounts are permitted to modify data.
Use this authorization with care.
<p>
```

```
</BODY>
</HTML>
```

2. **Créer l'autorisation à l'aide de la commande. `auths add`**

   Par exemple, la commande suivante permet de créer l'autorisation `com.newco.siteapp.data.modify` sur le système local.

   ```
   # auths add -t "SiteApp Data Modify Authorized" \
   -h /docs/helps/NewcoSiteAppModData.html com.newco.siteapp.data.modify
   ```

   Vous pouvez ensuite tester l'autorisation, puis l'ajouter à un profil de droits et attribuer ce dernier à un rôle ou un utilisateur.

Exemple 5-10 Le test d'un nouveau d'autorisation

Dans cet exemple, l'administrateur teste l'autorisation `com.newco.siteapp.data.modify` avec le profil de droits SiteApp de l'Exemple 5-7, “Création d'un profil de droits qui inclut des commandes privilégiées”.

```
# usermod -A com.newco.siteapp.data.modify -P SiteApp tester1
```

Lorsque le test réussit, l'administrateur supprime l'autorisation.

```
# rolemod -A-=com.newco.siteapp.data.modify siteapptester
```

Pour une gestion plus facile, l'administrateur ajoute l'autorisation à l'dans l'exemple 5 SiteApp -11 profil de droits. Exemple 5-11, “Ajout d'autorisations à un profil de droits”

Exemple 5-11 Ajout d'autorisations à un profil de droits

Après avoir testé le bon fonctionnement de l'autorisation, l'administrateur de sécurité ajoute l'autorisation `com.newco.siteapp.data.modify` à un profil de droits existant. L'Exemple 5-7, “Création d'un profil de droits qui inclut des commandes privilégiées” illustre la manière dont l'administrateur a créé le profil.

```
# profiles -p "SiteApp"
profiles:SiteApp> add auths="com.newco.siteapp.data.modify"
profiles:SiteApp> end
profiles:SiteApp> exit
```

Pour le vérifier, l'administrateur répertorie le contenu du profil.

```
# profiles -p SiteApp
Found profile in files repository.
  id=/opt/site-app/bin/site-cmd
  auths=com.newco.siteapp.data.modify
```

Étapes suivantes Attribuez le profil de droits à un utilisateur ou à un rôle de confiance. Pour consulter des exemples, reportez-vous à l'Exemple 3-10, “Création d'un utilisateur pouvant Administrer

DHCP” et à l'Exemple 3-19, “L'activation d'un utilisateur sécurisé des fichiers à lire la comptabilisation étendue”.

Voir aussi Pour dépanner les attributions de droits, reportez-vous à la section, “Dépannage d'attributions de droits” à la page 107. Pour plus d'informations, reportez-vous à la section “Ordre de recherche pour Assigned Rights” à la page 37.

Modification selon si root est un utilisateur ou un rôle

Par défaut, root correspond un rôle dans Oracle Solaris. Vous avez la possibilité pour la remplacer par une, modifiez-le (check in) l'utilisateur à un rôle, ou annuler son utilisation.

Vous devez changer root en utilisateur si vous utilisez Oracle Enterprise Managerou si vous suivez le modèle d'administration du superutilisateur traditionnel plutôt que le modèle de droits. Pour plus d'informations, reportez-vous à la section“Modèle par rapport à décider Utiliser pour quel (s) de gestion des droits d'administration” à la page 43.

Si vous suivez une, il peut être utile de modifier le modèle de droits root en utilisateur lors de la mise hors service d'un système a été supprimé du réseau. Dans ce cas, la connexion au système en tant que root simplifie le nettoyage.

Remarque - Si vous procédez à des tâches d'administration à distance avec le rôle root, consultez la section “ Administration à distance de ZFS avec le shell sécurisé ” du manuel “ Gestion de l’accès au shell sécurisé dans Oracle Solaris 11.2 ” pour des instructions sur la connexion à distance sécurisée.

De certains sites légitime , compte root n'est pas sur les systèmes de production. Pour enlever l'utilisateur root, reportez-vous à l'Exemple 5-13, “Interdiction de l'utilisation du rôle root pour gérer un système”

▼ Modification du rôle root en utilisateur

Vous devez exécuter cette procédure sur les systèmes sur lesquels root doit être en mesure de se connecter directement au système.

Avant de commencer Vous devez prendre le rôle root.

1. **Supprimez l'affectation du rôle root des utilisateurs locaux.**

Par exemple, supprimez l'affectation du rôle de deux utilisateurs.

```
% su -
Password: xxxxxxxx
# roles jdoe
root
# roles kdoe
root
# roles ldoe
secadmin
# usermod -R "" jdoe
# usermod -R "" kdoe
#
```

2. **Modification du rôle `root` en utilisateur.**

```
# rolemod -K type=normal root
```

Les utilisateurs qui sont actuellement dans le rôle `root` restent dans le rôle. Les autres utilisateurs disposant d'un accès root peuvent utiliser la commande `su` pour accéder à `root` ou se connecter au système en tant qu'utilisateur `root`.

3. **Vérifiez l'application de la modification.**

Vous pouvez utiliser l'une des commandes suivantes.

- **Reportez-vous à l'entrée pour root. `user_attr`**

```
# getent user_attr root
root::::auths=solaris.*;profiles=All;audit_flags=lo\:no;lock_after_retries=no;
min_label=admin_low;clearance=admin_high
```

Si le mot-clé `type` n'est pas présent dans la sortie ou s'il est égal à `normal`, le compte n'est pas un rôle.

- **Visualiser les résultats de la commande. `userattr`**

```
# userattr type root
```

Si la sortie est vide ou si elle répertorie `normal`, le compte n'est pas un rôle.

Exemple 5-12 Transformation de l'utilisateur `root` en rôle `root`

Dans cet exemple, l'utilisateurroot transforme à nouveau l'utilisateur root en un rôle.

Tout d'abord, l'utilisateur `root` transforme le compte `root` en un rôle et vérifie la modification.

```
# usermod -K type=role root
# getent user_attr root
root::::type=role...
```

Ensuite, root affecte le rôle root à un utilisateur local.

```
# usermod -R root jdoe
```

Exemple 5-13 Interdiction de l'utilisation du rôle root pour gérer un système

Dans cet exemple, la stratégie de sécurité du site requiert que le compte root ne puisse pas effectuer la maintenance du système. L'administrateur a créé et testé les rôles qui mettent à jour le système. Ces rôles incluent tous les profils de sécurité et le profil de droits System Administrator (administrateur système). Un rôle pouvant restaurer une sauvegarde a été affecté à un utilisateur de confiance. Ni aucun autre rôle ne peut modifier les indicateurs d'audit pour un utilisateur, rôle ou un profil de droits ou de modifier le mot de passe d'un rôle.

Pour empêcher que le compte root soit utilisé pour effectuer la maintenance du système, l'administrateur de sécurité supprime l'affectation du rôle root. Dans la mesure où le compte root doit être en mesure de se connecter au système en mode monoutilisateur, le compte conserve un mot de passe.

```
# usermod -K roles= jdoe
# userattr roles jdoe
```

Erreurs fréquentes

Dans un environnement bureautique, vous ne pouvez pas vous connecter directement en tant que root si root est un rôle. Un message de diagnostic indique que root est un rôle sur votre système.

Si vous ne disposez pas d'un compte local pouvant endosser le rôle root en procédant comme suit :

- En tant qu'utilisateurroot, connectez-vous au système en mode monoutilisateur, créez un compte utilisateur local et un mot de passe.
- Affectez le rôle root au nouveau compte.
- Connectez-vous en tant que ce nouvel utilisateur et endossez le rôle root.

♦ ♦ ♦ CHAPITRE 6

Liste des droits dans Oracle Solaris

Ce chapitre décrit la manière de répertorier tous les droits apparaissant dans le système, droits affectés à des utilisateurs donnés et vos propres droits, procédez comme suit :

Pour obtenir un aperçu des droits, reportez-vous à la section “Utilisateur Rights Management” à la page 14. Pour obtenir des informations de référence, reportez-vous au Chapitre 8, Droits Oracle Solaris (référence).

Liste des droits et leurs définitions

Les commandes de cette section vous permettront de choisir droits défini sur le système et répertorier les droits qui sont en vigueur sur un processus de l'utilisateur.

Pour obtenir une description des commandes de cette section, voir les pages de manuel suivantes :

- `auths`(1)
- `getent`(1M)
- `ppriv`(1)
- `profiles`(1)
- `privileges`(5)
- `roles`(1)

Liste les "feux verts"

- `auths` : répertorie les autorisations de l'utilisateur en cours
- `auths list` : répertorie les autorisations de l'utilisateur en cours
- `auths list -u` *username* : répertorie les autorisations accordes à *username*
- `auths list -x`: répertorie les autorisations de l'utilisateur en cours et devant être authentifiés
- `auths list -xu` *username* : répertorie les autorisations de *username* devant être authentifiées
- `auths info` : répertorie tous les noms d'autorisations dans le service de noms
- `getent auth_attr` : répertorie les définitions complètes de toutes les autorisations du service de noms

EXEMPLE 6-1 Affichage de toutes les autorisations

```
$ auths info
    solaris.account.activate
    solaris.account.setpolicy
    solaris.admin.edit
...
    solaris.zone.login
    solaris.zone.manage
```

EXEMPLE 6-2 Répertorier le contenu de la base de données des autorisations

```
$ getent auth_attr | more
solaris.:::All Solaris Authorizations::help=AllSolAuthsHeader.html
solaris.account.:::Account Management::help=AccountHeader.html
...
solaris.zone.login:::Zone Login::help=ZoneLogin.html
solaris.zone.manage:::Zone Deployment::help=ZoneManage.html
```

EXEMPLE 6-3 Répertorier les autorisations par défaut des utilisateurs

Les autorisations sont incluses dans les profils de droits qui sont assignés à tous les utilisateurs par défaut.

```
$ auths
solaris.device.cdrw,solaris.device.mount.removable,solaris.mail.mailq
solaris.network.autoconf.read,solaris.admin.wusb.read
solaris.smf.manage.vbiosd,solaris.smf.value.vbiosd
```

Liste des profils de droits

- `profiles` : répertorie les profils de droits de l'utilisateur en cours
- `profiles -a` répertorie tous les noms des profils de droits
- `profiles -l` : répertorie les définitions complètes des profils de droits de l'utilisateur en cours
- `profiles` *username* : répertorie les profils de droits de *username*
- `profiles -x` : répertorie les profils de droits de l'utilisateur actuel devant être authentifiés
- `profiles -x` *username* : répertorie les profils de droits de *username* devant être authentifiés
- `profiles -p` *profile-name* `info` : imprime le contenu des profils de droits spécifiés
- `getent prof_attr` : répertorie les définitions complètes de tous les profils de droits figurant dans le service de noms

EXEMPLE 6-4 Répertorier les noms de tous les profils de droits

```
$ profiles -a
        Console User
        CUPS Administration
        Desktop Removable Media User
...
        VSCAN Management
        WUSB Management
```

EXEMPLE 6-5 Affichage du contenu de la base de données de profils de droits

```
$ getent prof_attr | more
All:::Execute any command as the user or role:help=RtAll.html
Audit Configuration:::Configure Solaris Audit:auths=solaris.smf.value.audit;
help=RtAuditCfg.html
...
Zone Management:::Zones Virtual Application Environment Administration:
help=RtZoneMngmnt.html
Zone Security:::Zones Virtual Application Environment Security:auths=solaris.zone.*,
solaris.auth.delegate;help=RtZoneSecurity.html ...
```

EXEMPLE 6-6 Répertorier les profils de droits par défaut des utilisateurs

Répertoriez vos profils de droits. Les profils de droits suivants sont affectés à tous les utilisateurs par défaut.

```
$ profiles
Basic Solaris User
All
```

EXEMPLE 6-7 Répertorier les profils de droits de l'utilisateur initial

Plusieurs profils de droits sont affectés à l'utilisateur initial.

```
$ profiles Initial user
System Administrator
Audit Review
...
CPU Power Management
Basic Solaris User
All
```

Pour afficher tous les attributs de sécurité affectés aux profils de l'utilisateur initial, utilisez l'option `-l`.

```
$ profiles -l Initial user | more
Initial user:
System Administrator
 profiles=Install Service Management,Audit Review,Extended Accounting
Flow Management,Extended Accounting Net Management,Extended Accounting Process
Management,Extended Accounting Task Management,Printer Management,Cron Managem
ent,Device Management,File System Management,Log Management,Mail Management,
Maintenance and Repair,Media Catalog,Name Service Management,Network Management,
Project Management,RAD Management,Service Operator,Shadow Migration Monitor,So
Software Installation,System Configuration,User Management,ZFS Storage Management
          /usr/sbin/gparted          uid=0
Install Service Management
 auths=solaris.autoinstall.service
 profiles=Install Manifest Management,Install Profile Management,
Install Client Management
...
```

EXEMPLE 6-8 Affichage du contenu de an Assigned Rights Profile

L'utilisateur initial énumère les droits accordés par le profil Audit Review (vérification d'audit).

```
$ profiles -l
Audit Review
  solaris.audit.read

  /usr/sbin/auditreduce  euid=0
  /usr/sbin/auditstat    privs=proc_audit
  /usr/sbin/praudit      privs=file_dac_read
```

EXEMPLE 6-9 Liste the Security Attributes of a Command in a Rights Profile

Cette variante de la commande `profiles` est utile pour voir les attributs de sécurité d'une commande dans un profil de droits qui ne vous est pas affecté.

Dans un premier temps, répertoriez les commandes dans le profil.

```
% profiles -p "Audit Review" info
```

```
 name=Audit Review
 desc=Review Solaris Auditing logs
 help=RtAuditReview.html
 cmd=/usr/sbin/auditreduce
 cmd=/usr/sbin/auditstat
 cmd=/usr/sbin/praudit
```

Ensuite, répertoriez les attributs de sécurité de l'une des commandes répertoriées dans le profil.

```
% profiles -p "Audit Review" "select cmd=/usr/sbin/praudit ; info; end;"
select: command is read-only
 id=/usr/sbin/praudit
 privs=file_dac_read
end: command is read-only
```

EXEMPLE 6-10 Affichage du contenu de Rights Profiles That Are récemment créé (e) s

L'option `less` affiche d'abord les profils de droits ajoutés le plus récemment. Cette variante de la commande`profiles` est utile lorsque vous créez ou modifiez les profils de droits sur votre site. La sortie suivante indique le contenu du profil qui a été ajoutée dans l'Exemple 4-1, “Affecter des attributs de sécurité à une ancienne application”. Un utilisateur standard peut exécuter cette commande.

```
$ profiles -la | less
LegacyApp
         /opt/legacy-app/bin/legacy-cmd
                                euid=0
OpenLDAP...
```

Liste des rôles

- `roles` : répertorie les rôles de l'utilisateur en cours
- `roles` *username* : répertorie les rôles de *username*
- `logins -r`: affiche la liste de tous les rôles disponibles

EXEMPLE 6-11 Liste des rôles qui vous sont affectés

Le rôle `root` est affecté à l'utilisateur initial par défaut. `Aucun rôle` indique qu'aucun rôle ne vous est affecté.

```
$ roles
root
```

Liste des privilèges

- `man privileges` : répertorie les définitions de privilège et leurs noms tels qu'ils sont utilisés par les développeurs
- `ppriv -vl` : répertorie les définitions de privilège et leurs noms tels qu'ils sont utilisés par les administrateurs
- `ppriv -vl basic` : répertorie les noms et les définitions des privilèges figurant dans le jeu de privilèges de base
- `ppriv $$`: répertorie les privilèges dans le shell actuel (`$$`)
- `getent exec_attr` : répertorie par noms de profils de droits toutes les commandes ayant des attributs de sécurité (`setuid` ou privilèges)

```
$ getent exec_attr | more
All:solaris:cmd:::*:
Audit Configuration:solaris:cmd:::/usr/sbin/auditconfig:privs=sys_audit
...
Zone Security:solaris:cmd:::/usr/sbin/txzonemgr:uid=0
Zone Security:solaris:cmd:::/usr/sbin/zonecfg:uid=0 ...
```

EXEMPLE 6-12 Etablissement de la liste de tous les privilèges et leurs définitions

Le format de privilège décrit dans la page de manuel `privileges`(5) est utilisé par les développeurs.

```
$ man privileges
Standards, Environments, and Macros                 privileges(5)

NAME
     privileges - process privilege model
...
     The defined privileges are:

     PRIV_CONTRACT_EVENT

         Allow a process to request reliable delivery  of  events
         to an event endpoint.

         Allow a process to include events in the critical  event
         set  term  of  a  template  which  could be generated in
         volume by the user.
...
```

EXEMPLE 6-13 Dans l'établissement de la liste Privilèges de l'affectation de privilèges That Are Used

La commande `ppriv`répertorie tous les privilèges par leurs noms. Pour obtenir une définition, utilisez l'option `-v`.

Ce format de privilège est utilisé pour affecter des privilèges à des utilisateurs et des rôles à l'aide des commandes `useradd`, `roleadd`, `usermod` et `rolemod`, et à des profils de droits à l'aide de la commande `profiles`.

```
$ ppriv -lv | more
contract_event
 Allows a process to request critical events without limitation.
 Allows a process to request reliable delivery of all events on
 any event queue.
...
win_upgrade_sl
        Allows a process to set the sensitivity label of a window
        resource to a sensitivity label that dominates the existing
        sensitivity label.
        This privilege is interpreted only if the system is configured
        with Trusted Extensions.
```

EXEMPLE 6-14 Liste des privilèges dans votre shell actuel

Le jeu de privilèges de base est affecté par défaut à tous les utilisateurs. Le jeu limite par défaut est constitué de tous les privilèges.

Les lettres dans la sortie font référence aux jeux de privilèges suivants :

`E` Jeu de privilèges effectif

`I` Jeu de privilèges héritable

`P` Privilèges autorisés jeu

`L` Jeu de privilèges de limite

```
$ ppriv $$
1200:   -bash
flags = <none>
        E: basic
        I: basic
        P: basic
        L: all
$ ppriv -v $$
1200:   -bash
flags = <none>
 E: file_link_any,file_read,file_write,net_access,proc_exec,proc_fork,
          proc_info,proc_session,sys_ib_info
 I: file_link_any,file_read,...,sys_ib_info
 P: file_link_any,file_read,...,sys_ib_info
 L: contract_event,contract_identity,...,sys_time
```

Utilisez le symbole double dollar (`$$`) pour transmettre le numéro de processus du shell parent à la commande. Cette liste n'inclut pas les privilèges sont restreintes aux commandes dans un profil de droits affecté.

EXEMPLE 6-15 Répertoriant les privilèges de base et leurs définitions

```
$ ppriv -vl basic
file_link_any
 Allows a process to create hardlinks to files owned by a uid
 different from the process' effective uid.
file_read
 Allows a process to read objects in the filesystem.
file_write
 Allows a process to modify objects in the filesystem.
net_access
 Allows a process to open a TCP, UDP, SDP or SCTP network endpoint.
proc_exec
 Allows a process to call execve().
proc_fork
 Allows a process to call fork1()/forkall()/vfork()
proc_info
 Allows a process to examine the status of processes other
 than those it can send signals to.  Processes which cannot
 be examined cannot be seen in /proc and appear not to exist.
proc_session
 Allows a process to send signals or trace processes outside its
 session.
sys_ib_info
 Allows a process to perform read InfiniBand MAD (Management Datagram)
 operations.
```

EXEMPLE 6-16 A l'aide d'attributs de sécurité répertoriant les dans votre commandes des profils de droits

Le profil d'utilisateur de base Solaris comprend des commandes permettant de lire et d'écrire sur des CD-ROM.

```
$ profiles -l
  Basic Solaris User
...
   /usr/bin/cdrecord.bin   privs=file_dac_read,sys_devices,
     proc_lock_memory,proc_priocntl,net_privaddr
   /usr/bin/readcd.bin     privs=file_dac_read,sys_devices,net_privaddr
   /usr/bin/cdda2wav.bin   privs=file_dac_read,sys_devices,
     proc_priocntl,net_privaddr
  All
   *
```

Attributs liste qualifié

- `man user_attr` : définit les qualificatifs des attributs de sécurité

- `getent` : répertorie les attributs de sécurité qualifiés d'un utilisateur ou d'un rôle sur le système sur lequel la commande est exécutée
- `ldapaddent` : répertorie tous les attributs de sécurité qualifiés d'un utilisateur ou d'un rôle

EXEMPLE 6-17 Le fait d'ajouter un de l'utilisateur d'attributs sur ce système qualifié

```
machine1$ getent user_attr | jdoe:
jdoe:machine1:::profiles=System Administrator
```

EXEMPLE 6-18 Etablissement de la liste de tous les dans des attributs d'un utilisateur qualifié LDAP

```
machine1$ ldapaddent -d user_attr | grep ^jdoe:
jdoe:machine1:::profiles=System Administrator
jdoe:sysopgroup:::profiles=System Operator
```

♦ ♦ ♦ CHAPITRE 7

Résolution des problèmes liés aux droits dans Oracle Solaris

Ce chapitre fournit des suggestions pour la résolution de problèmes lors de la gestion et de l'utilisation des droits administratifs dans Oracle Solaris :

- “Dépannage d'attributions de droits” à la page 107
- “Réorganisation Assigned Rights” à la page 112
- “Détermination des privilèges requis par un programme” à la page 113

Pour plus d'informations sur l'utilisation de droits, passez en revue les informations suivantes :

- Chapitre 3, Affectation de droits dans Oracle Solaris
- “Droits qui peut d'affectation, procédez comme suit :” à la page 48
- “Utilisateur Rights Management” à la page 14
- “Processus Rights Management” à la page 24

Dépannage de gestion des droits

Les tâches et les exemples de cette section suggèrent des méthodes afin de résoudre les problèmes, avec des droits d'accès leur affectations. Pour plus d'informations d'ordre général, reportez-vous à la section “Droits de vérification” à la page 36.

▼ Dépannage d'attributions de droits

Les processus d'un utilisateur ou d'un rôle peuvent ne pas s'exécuter avec les attributs de sécurité qui leurs sont affectés pour différentes raisons. Cette procédure aide à résoudre les problèmes lorsque des utilisateurs, des rôles ou des processus ne disposent pas de droits qui leur ont été affectés. Plusieurs étapes sont basées sur la section “Ordre de recherche pour Assigned Rights” à la page 37.

Avant de commencer

Vous devez prendre le rôle `root`. Pour plus d'informations, voir “A l'aide de vos droits administratifs attribués” à la page 82.

1. **Vérification et redémarrage du service de noms.**

   a. **Vérifiez que les affectations de sécurité pour l'utilisateur ou le rôle sont dans le service de noms activé sur le système.**

   ```
   # svccfg -s name-service/switch

   svc:/system/name-service/switch>
   listprop config

   config                      application
   config/value_authorization  astring  solaris.smf.value.name-service.switch
   config/default              astring  files ldap
   config/host                 astring  "files dns mdns ldap"
   config/netgroup             astring  ldap
   config/printer              astring  "user files"
   ```

   Dans cette sortie, tous les services qui ne sont pas explicitement mentionnés héritent de la valeur par défaut, `files ldap`. Par conséquent, la recherche porte tout d'abord sur `passwd` et ses bases de données d'attributs apparentées, `user_attr`, `auth_attr` et `prof_attr` dans les fichiers, puis dans LDAP.

   b. **Redémarrez le cache du service de noms, `svc:/system/name-service/cache`.**

   Le démon `nscd` peut présenter un long intervalle de durée de vie. En redémarrant le démon, vous mettez à jour le service de noms avec les données en cours.

   ```
   # svcadm restart name-service/cache
   ```

2. **Pour déterminer où un droit est attribué à l'utilisateur, exécuter la commande `userattr -v`.**

   Par exemple, les commandes suivantes indiquent les attributs de sécurité attribués et le lieu de cette attribution pour l'utilisateur `jdoe` : Indique que jdoe se voit qu'il n'y ait aucune sortie à l'aide des valeurs par défaut.

   ```
   % userattr -v access_times jdoe
   % userattr -v access_tz jdoe
   % userattr -v auth_profiles jdoe
   % userattr -v defaultpriv jdoe
   % userattr -v limitpriv jdoe
   % userattr -v idlecmd jdoe
   % userattr -v idletime jdoe
   % userattr -v lock_after_retries jdoe
   % userattr -v pam_policy jdoe

   % userattr -v auths jdoe      Output indicates authorizations from rights profiles
   Basic Solaris User :solaris.mail.mailq,solaris.network.autoconf.read,
   solaris.admin.wusb.read
   Console User :solaris.system.shutdown,solaris.device.cdrw,
   solaris.device.mount.removable,solaris.smf.manage.vbiosd,solaris.smf.value.vbiosd
   ```

```
% userattr -v audit_flags jdoe
user_attr: fw:no       Output indicates jdoe is individually assigned audit flags
# userattr -v profiles jdoe
user_attr: Audit Review,Stop       Output indicates two assigned rights profiles
# userattr roles jdoe
user_attr : cryptomgt,infosec       Output indicates two assigned roles
```

La sortie indique que des indicateurs d'audit, deux profils de droits et deux rôles sont directement affectés à `jdoe`. Les autorisations affectées sont à partir de profils de droits par défaut dans le fichier. `policy.conf`

- Etant donnéque les indicateurs d'audit sont directement affectés à jdoe ne pas lancer d'audit les valeurs d'indicateur, les profils de droits d'accès sera utilisée.
- Les profils de droits d'accès sont évalués dans l'ordre, tout d'abord le profil Audit Review (vérification d'audit), puis le profil Stop (arrêt).
- Pour tous les autres droits sont affectés et jdoe infosec parmi les rôles cryptomgt. Pour visualiser, jdoe devez vous connecter entant qu'ces droits leur étant associés, puis répertoriez les droits.

Si le droit n'est directement affectés à l'utilisateur aux vérifications suivantes, poursuivez en vous reportant à la section.

3. **Vérifiez que les autorisations affectées sont orthographiées correctement.**

   La source d'une attribution d'autorisation n'est pas importante, car les autorisations s'accumulent pour les utilisateurs. Une autorisation mal orthographiée échoue toutefois discrètement.

4. **Pour les profils de droits que vous avez créés, vérifiez que vous avez affecté les attributs de sécurité appropriés aux commandes dans ce profil.**

   Par exemple, certaines commandes requièrent `uid=0` plutôt que `euid=0` pour s'exécuter correctement. Passez en revue la page de manuel de la commande pour savoir si la commande, ou n'importe lequel de ses options requérir des autorisations.

5. **Recherchez les droits figurant dans les profils de droits de l'utilisateur.**

   a. **Dans l'ordre, recherchez l'authentifié droits dans la liste des profils de droits.**

      La valeur de l'attribut dans les premiers profils de droits de la liste est la valeur dans le noyau. Si cette valeur est incorrecte, modifiez-la dans ce profil de droits ou réaffectez les profils dans l'ordre correct. Reportez - vous à la “Réorganisation Assigned Rights” à la page 112

      Pour les commandes dotées de privilèges, vérifiez que les privilèges ne sont pas supprimés du mot-clé `defaultpriv` ou `limitpriv`.

   b. **Dans l'ordre, recherchez les droits d'accès dans la liste de titres normaux des profils de droits.**

Au fur et à mesure que vous sont identiques à celles de la configuration effectuée pour authentifié vérifie les profils de droits.

c. Si les droits que vous recherchez n'est répertorié, vérifiez les rôles affectés à l'utilisateur.

Si le droit est affecté à un rôle, l'utilisateur doit endosser le rôle pour obtenir ce droit.

6. Vérifiez si l'échec d'une commande requiert des autorisations pour s'exécuter correctement.

a. Vérifiez si un profil de droits existant inclut l'autorisation obligatoire.

Si le profil, il est préférable d'utiliser celle-ci . L'affecter à l'utilisateur sous forme d'un profil de droits ou régulier authentifié profil de droits. Trier les profil avant les autres profils de droits incluant la commande nécessitant cette autorisation s'exécute correctement.

b. Vérifiez si une option de la commande nécessite une autorisation.

Affectez le privilège à la commande qui le nécessite, ajoutez les autorisations nécessaires, placez la commande et les autorisations dans un profil de droits, et affectez ce profil à l'utilisateur.

7. Si une commande échoue continuellement pour un utilisateur, vérifiez que cet utilisateur exécute la commande dans un shell de profil.

Les commandes d'administration doivent être exécutées dans un shell de profil. Exemple 7-1, “Déterminer si vous utilisez un shell de profil”comment tester -1 pour un shell de profil.

De façon à minimiser la probabilité de les erreurs de l'utilisateur, vous pouvez essayer l'opération suivante :

- Affecter un shell de profil comme shell de connexion d'utilisateur.
- Demandez à l'utilisateur de faire précéder tous les commandes privilégiées à l'aide de la commande. `pfexec`
- Rappelez à l'utilisateur d'exécuter les commandes d'administration dans un shell de profil.
- Si votre site, rappeler à l'utilisateur à l'aide des rôles pour prendre ce rôle avant d'exécuter des commandes d'administration. Commande pour obtenir un exemple de participants dont la présence a été plutôt que l'exécution en tant que rôle défini en tant qu'utilisateur, reportez-vous à l'Example 7 -3. Exemple 7-3, “Exécution de commandes privilégiées dans votre rôle”

8. En cas d'échec d'une commande pour un rôle, prenez le rôle et réalisez les étapes effectuées lors de la vérification des droits dont dispose l'utilisateur.

Exemple 7-1 Déterminer si vous utilisez un shell de profil

Lorsqu'une commande ne fonctionne pas privilégiées et informations d'identification et de l'utilisateur, tests du exécute ensuite ces PRIV_PFEXEC commande indicateur. Indiquer le message d'erreur indiquant que le problème risque de ne pas s'agit d'un privilège problème.

```
% praudit 20120814200247.20120912213421.example-system
praudit: Cannot associate stdin with 20120814200247.20120912213421.example-system:
Permission denied

% ppriv $$
107219: bash
flags = <none>
...

% pfbash
# ppriv $$
1072232: bash
flags = PRIV_PFEXEC
...

# praudit 20120814200247.20120912213421.example-system
     /** Command succeeds **/
```

Exemple 7-2 Détermination des commandes privilégiées d'un rôle

Dans cet exemple, un utilisateur endosse un rôle affecté et répertorie les droits compris dans l'un des profils de droits. Mettre en valeur les droits sont tronqués de façon à respecter les commandes.

```
% roles
devadmin

% su - devadmin
Password: xxxxxxxx

# profiles -l
Device Security
        ...
 profiles=Service Configuration
          /usr/sbin/add_drv          uid=0
          /usr/sbin/devfsadm         uid=0
                                     privs=sys_devices,sys_config,
                                           sys_resource,file_owner,
                                           file_chown,file_chown_self,
                                           file_dac_read
          /usr/sbin/eeprom           uid=0
          /usr/bin/kbd
          /usr/sbin/list_devices     euid=0
          /usr/sbin/rem_drv          uid=0
          /usr/sbin/strace           euid=0
          /usr/sbin/update_drv       uid=0
```

```
          /usr/sbin/add_allocatable  euid=0
          /usr/sbin/remove_allocatable   euid=0
Service Configuration
          /usr/sbin/svcadm
          /usr/sbin/svccfg
```

Exemple 7-3 Exécution de commandes privilégiées dans votre rôle

Dans l'exemple suivant, le rôle `admin` peut modifier les autorisations pour le fichier `useful.script`.

```
% whoami
jdoe
% ls -l useful.script
-rwxr-xr-- 1 elsee eng 262 Apr 2 10:52 useful.script

% chgrp admin useful.script
chgrp: useful.script: Not owner

% su - admin
Password: xxxxxxxx

# chgrp admin useful.script
# chown admin useful.script
# ls -l useful.script
-rwxr-xr-- 1 admin admin 262 Apr 2 10:53 useful.script
```

▼ Réorganisation Assigned Rights

Vous devez réorganiser les affectations d'un utilisateur sans privilège les profils de droits lorsqu'une commande en vigueur pour la version plutôt que son doté de privilèges utilisateur. Pour plus d'informations, reportez-vous à la section “Ordre de recherche pour Assigned Rights” à la page 37.

Avant de commencer Vous devez vous connecter en tant qu'administrateur disposant du profil de droits User Security (sécurité des utilisateurs). Pour plus d'informations, voir “A l'aide de vos droits administratifs attribués” à la page 82.

1. **Affichez la liste des profils de droits qui sont actuellement affectés à l'utilisateur ou au rôle.**

   La liste s'affiche dans l'ordre.

   ```
   % profiles username | rolename
   ```

2. **Affectez les profils de droits dans l'ordre approprié.**

   ```
   # usermod | rolemod -P "list-of-profiles"
   ```

Exemple 7-4 Affectation de profils de droits dans un ordre spécifique

Dans cet exemple, l'administrateur détermine si un profil de droits avec des commandes privilégiées est répertorié après le profil de droits All (tous) pour le rôle `devadmin`.

```
# profiles devadmin

    Basic Solaris User
    All
    Device Management
```

Par conséquent, le rôle `devadmin` ne peut pas exécuter les commandes de gestion des périphériques avec les privilèges affectés au rôle.

L'administrateur réaffecte les profils de droits à `devadmin`. Dans le nouvel ordre d'affectation, les commandes de gestion des périphériques sont exécutées avec les privilèges qui leur sont affectés.

```
# rolemod -P "Device Management,Basic Solaris User,All"

# profiles devadmin

    Device Management
    Basic Solaris User
    All
```

▼ Détermination des privilèges requis par un programme

Utilisez cette procédure de débogage lorsqu'une commande ou un processus est défectueux. Après avoir identifié le premier privilège appliquée, puis le réparer, il peut être nécessaire d'exécuter la commande à nouveau pour trouver d'autres des exigences en matière de privilèges.
`ppriv -eD` *command*

1. **Saisissez la commande ayant échoué en tant qu'argument de la commande de débogage `ppriv`.**

```
% ppriv -eD touch /etc/acct/yearly

touch[5245]: missing privilege "file_dac_write"
     (euid = 130, syscall = 224) needed at zfs_zaccess+0x258
touch: cannot create /etc/acct/yearly: Permission denied
```

2. **Utilisez le numéro `syscall` à partir de la sortie de débogage pour déterminer l'appel système défaillant.**

   Pour rechercher le nom d'le numéro syscall dans le fichier. `/etc/name_to_sysnum`

```
% grep 224 /etc/name_to_sysnum

creat64                224
```

Creat64 dans cet exemple, l'appel est défectueux. () Pour s'exécuter correctement, le droit de créer un fichier dans le répertoire /etc/acct/yearly doit être affecté au processus.

Exemple 7-5 Utilisation de la commande truss pour examiner l'utilisation des privilèges

La commande truss peut déboguer l'utilisation des privilèges dans un shell standard. Par exemple, la commande suivante débogue le processus touch défaillant :

```
% truss -t creat touch /etc/acct/yearly

creat64("/etc/acct/yearly", 0666)
                      Err#13 EACCES [file_dac_write
]
touch: /etc/acct/yearly cannot create
```

Les interfaces /proc étendues signalent les privilèges file_dac_write manquants après le code d'erreur dans la sortie truss.

Exemple 7-6 Utilisation de la commande ppriv pour l'examen de l'utilisation des privilèges dans un shell de profil

Dans cet exemple, l'utilisateur jdoe peut endosser le rôle objadmin. Le rôle objadmin comprend le profil de droits Object Access Management (gestion de l'accès aux objets). Ce profil de droits permet au rôle objadmin de modifier les autorisations pour les fichiers dont objadmin n'est pas propriétaire.

Dans l'exemple ci-dessous, jdoe ne parvient pas à changer les autorisations pour le fichier useful.script :

```
jdoe% ls -l useful.script

-rw-r--r--  1 aloe  staff  2303 Apr 10 10:10 useful.script
jdoe%
chown objadmin useful.script

chown: useful.script: Not owner
jdoe%
ppriv -eD chown objadmin useful.script

chown[11444]: missing privilege "file_chown"
            (euid = 130, syscall = 16) needed at zfs_zaccess+0x258
chown: useful.script: Not owner
```

Lorsque jdoe endosse le rôle objadmin, les autorisations pour le fichier sont modifiées :

```
jdoe% su - objadmin
Password: xxxxxxxx

# ls -l useful.script
-rw-r--r--  1 aloe  staff  2303 Apr 10 10:10 useful.script

# chown objadmin useful.script
# ls -l useful.script
-rw-r--r--  1 objadmin  staff  2303 Apr 10 10:10 useful.script
# chgrp admin useful.script

# ls -l objadmin.script
-rw-r--r--  1 objadmin  admin  2303 Apr 10 10:11 useful.script
```

Exemple 7-7 Modification d'un fichier appartenant à l'utilisateur `root`

Cet exemple illustre les protections contre l'escalade des privilèges. Pour plus de détails, reportez-vous à Privilèges Privilege Escalation et noyau. “Escalade de privilèges et privilèges de noyau” à la page 35 Le fichier appartient à l'utilisateur `root`. Le rôle le moins puissant, `objadmin`, a besoin de tous les privilèges pour modifier la propriété du fichier, de sorte que l'opération échoue.

```
jdoe% su - objadmin
Password: xxxxxxxx

# cd /etc; ls -l system
-rw-r--r--  1 root  sys   1883 Oct 10 10:20 system

# chown objadmin system
chown: system: Not owner
# ppriv -eD chown objadmin system
chown[11481]: missing privilege "ALL"
     (euid = 101, syscall = 16) needed at zfs_zaccess+0x258
chown: system: Not owner
```

♦ ♦ ♦ CHAPITRE 8

Droits Oracle Solaris (référence)

Ce chapitre fournit des informations de référence sur l'utilisation des droits d'administration dans Oracle Solaris :

Pour plus d'informations sur l'utilisation des droits et des privilèges, reportez-vous auChapitre 3, Affectation de droits dans Oracle Solaris. Pour des informations générales, reportez-vous aux sections“Utilisateur Rights Management” à la page 14 et “Processus Rights Management” à la page 24.

Profils de droits Reference (disponible en anglais uniquement)

Cette section décrit les principaux profils de droits. Les profils de droits sont des ensembles pratiques comprenant des autorisations et d'autres attributs de sécurité, des commandes avec des attributs de sécurité, et des profils droits supplémentaires. Oracle Solaris fournit de nombreux profils de droits. S'ils ne répondent pas à vos besoins, vous pouvez modifier des profils existants et en créer de nouveaux.

Les profils de droits doivent être affectés dans l'ordre du plus puissant au moins puissant. Pour plus d'informations, reportez-vous à la section “Ordre de recherche pour Assigned Rights” à la page 37.

Pour visualiser le contenu des profils de droits suivants, reportez-vous à la section “Affichage du contenu des profils de droits” à la page 118.

- **Profil de droits Administrateur système** : il permet d'accéder à la plupart des tâches qui ne concernent pas la sécurité. Ce profil inclut plusieurs autres profils permettant de créer un rôle puissant. Notez que le profil de droits All (tous) est attribué à la fin de la liste des profils de droits supplémentaires.

- **Profil de droits Opérateur** : ce profil aux capacités limitées permet de gérer les fichiers et les médias hors ligne. Ce profil inclut des profils de droits supplémentaires pour créer un rôle simple.
- **Profil de droits Gestion d'imprimantes** : ce profil offre un nombre limité de commandes et d'autorisations pour la gestion de l'impression. Ce profil est l'un des nombreux profils couvrant une seule partie de l'administration.
- **Profil de droits Utilisateur Solaris de base** : ce profil permet aux utilisateurs d'utiliser l'ordinateur dans les limites de la stratégie de sécurité. Ce profil est répertorié par défaut dans le fichier `policy.conf`. Notez que les avantages proposés par le profil de droits de l'utilisateur Solaris de base doivent être contrebalancés avec les exigences en matière de sécurité du site. Les sites nécessitant une sécurité plus stricte préféreront peut-être supprimer ce profil du fichier `policy.conf` ou affecter le profil de droits Stop (arrêt). Pour l'implémentation du profil de droits de l'utilisateur Solaris de base, reportez-vous à l'Exemple 6-16, “A l'aide d'attributs de sécurité répertoriant les dans votre commandes des profils de droits”.
- **Profil de droits Utilisateur de la console** : pour les propriétaires de station de travail, ce profil fournit à la personne placée devant l'ordinateur l'accès aux autorisations, commandes et actions.
- **Tous les profils de droits** : permet aux rôles d'accéder aux commandes n'ayant pas d'attribut de sécurité. Ce profil peut être approprié pour les utilisateurs ayant des droits limités.
- **Profil de droits Arrêt** : ce profil de droits spécial arrête l'évaluation d'autres profils. Ce profil empêche l'évaluation des variables `AUTHS_GRANTED`, `PROFS_GRANTED` et `CONSOLE_USER` dans le fichier `policy.conf`. Avec ce profil, vous pouvez fournir aux rôles et aux utilisateurs un shell de profil limité.

Remarque - Le profil de droits Stop affecte de manière indirecte l'affectation de privilèges. Les profils de droits indiqués après le profil de droits Stop (arrêt) ne sont pas évalués. Par conséquent, les commandes avec des privilèges dans ces profils ne sont pas prises en compte. Voir Exemple 3-25, “Limitation d'un administrateur aux droits affectés de manière explicite”.

Chaque profil de droits est associé à un fichier d'aide. Les fichiers d'aide sont au format HTML et sont personnalisables. Les fichiers sont stockés dans le répertoire `/usr/lib/help/profiles/locale/C`.

Affichage du contenu des profils de droits

Vous disposez de trois vues sur le contenu des profils de droits :

- La commande `getent` vous permet de consulter le contenu de tous les profils de droits sur le système. Vous trouverez un exemple de sortie dans le Chapitre 6, Liste des droits dans Oracle Solaris.

- La commande `profiles -p "`*Profile Name*`" info` vous permet d'afficher le contenu d'un profil de droits spécifique.
- La commande `profiles -l` *account-name* vous permet de visualiser le contenu des profils de droits qui sont affectés à un utilisateur ou rôle spécifique.

Pour plus d'informations, reportez-vous au Chapitre 6, Liste des droits dans Oracle Solaris and the `getent`(1M) et à la page de manuel `profiles`(1).

Les "feux verts" Reference (disponible en anglais uniquement)

Une *autorisation* est un droit discret qui peut être accordé à un rôle ou à un utilisateur. Les autorisations sont vérifiées par les applications conformes aux normes RBAC avant de donner à l'utilisateur un accès à l'application ou à des opérations spécifiques au sein de l'application.

Les autorisations sont de niveau utilisateur et sont par conséquent extensibles. Vous pouvez écrire un programme qui requiert une autorisation, ajouter les autorisations à votre système, créer un profil de droits pour ces autorisations et affecter le profil de droits aux rôles ou utilisateurs autorisés à utiliser le programme.

Conventions de nommage des autorisations

Une autorisation a un nom qui est utilisé en interne. Par exemple, `solaris.admin.usermgr.pswd` est le nom d'une autorisation. Une autorisation est accompagnée d'une description courte qui s'affiche dans les interfaces graphiques. Par exemple, `Set Date & Time` est la description de l'autorisation `solaris.system.date`.

Par convention, les noms des autorisation sont composés en inversant le nom Internet du fournisseur, le domaine, l'éventuel sous-domaine et la fonction. Les parties du nom d'autorisation sont séparées par des points. Un exemple serait `com.xyzcorp.device.access`. Les exceptions à cette convention sont les autorisations d'Oracle, lesquelles utilisent le préfixe `solaris` au lieu d'un nom Internet. La convention de nommage permet aux administrateurs d'appliquer des autorisations de manière hiérarchique. Un caractère générique (*) peut représenter des chaînes à droite d'un point.

Exemple d'utilisation des autorisations : le profil de droits Sécurité liaison réseau est limité à l'autorisation `solaris.network.link.security`, alors que le profil de droits Sécurité réseau dispose du profil de droits Sécurité liaison réseau comme profil supplémentaire et des autorisations `solaris.network.*` et `solaris.smf.manage.ssh`.

Pouvoir de délégation dans les autorisations

Une autorisation se terminant par le suffixe `delegate` permet à un utilisateur ou à un rôle de déléguer à d'autres utilisateurs des autorisations attribuées commençant par le même préfixe.

L'autorisation `solaris auth.delegate` permet à un utilisateur ou un rôle de déléguer à d'autres utilisateurs toute autorisation affectée à ce même utilisateur ou rôle. Par exemple, un rôle avec les autorisations `solaris auth.delegate` et `solaris.network.wifi.wep` peut déléguer l'autorisation `solaris.network.wifi.wep` à un autre utilisateur ou rôle.

Droits Bases de données

Les bases de données suivantes stockent les données relatives aux droits dans Oracle Solaris :

- **Base de données des attributs utilisateur étendus** `(user_attr)` **:** associe des utilisateurs et des rôles à des autorisations, des privilèges et des profils de droits, entre autres mots-clés.
- **Base de données d'attributs de profils de droits** `(prof_attr)` **:** définit les profils de droits, répertorie les autorisations, privilèges et mots-clés affectés aux profils et identifie le fichier d'aide associé.
- **Base de données d'attributs d'autorisations** `(auth_attr)` **:** définit les autorisations et leurs attributs, et identifie le fichier d'aide associé.
- **Base de données d'attributs d'exécution** `(exec_attr)` **:** identifie les commandes portant des attributs de sécurité attribués à des profils de droits spécifiques.

La base de données `policy.conf` contient les autorisations, les privilèges et les profils de droits appliqués à tous les utilisateurs. Pour plus d'informations, reportez-vous à la section “Fichier `policy.conf`” à la page 124.

Bases de données de droits et services de noms

La portée du service de noms des bases de données des droits est définie dans le service SMF pour le commutateur du service de noms, `svc:/system/name-service/switch`. Les propriétés de ce service pour les bases de données de droits sont `auth_attr`, `password` et `prof_attr`. La propriété `password` définit la priorité d'un service de noms pour les bases de données `passwd` et `user_attr`. La propriété `prof_attr` définit la priorité d'un service de noms pour les bases de données `prof_attr` et `exec_attr`.

Dans la sortie suivante, les entrées `auth_attr`, `password` et `prof_attr` ne sont pas répertoriées. Par conséquent, les bases de données de droits utilisent le service de noms `files`.

```
# svccfg -s name-service/switch listprop config
config                      application
config/value_authorization  astring       solaris.smf.value.name-service.switch
config/default              astring       files
config/host                 astring       "files ldap dns"
config/printer              astring       "user files ldap"
```

Base de données `user_attr`

La base de données `user_attr` contient des informations sur l'utilisateur et le rôle qui complètent les bases de données `passwd` et `shadow`. Le champ *attr* contient les attributs de sécurité, et le champ *qualifier* contient les attributs qui qualifient ou limitent l'effet des attributs de sécurité sur un système ou un groupe de systèmes.

Les attributs de sécurité figurant dans le champ `attr` peuvent être définis à l'aide des commandes `roleadd`, `rolemod`, `useradd`, `usermod` et `profiles`. Ils peuvent être définis localement et dans la portée de dénomination LDAP.

- Dans le cas d'un utilisateur, le mot-clé `roles` affecte un ou plusieurs rôles définis.
- Pour un rôle, la valeur `user` du mot-clé `roleauth` permet au rôle de s'authentifier avec le mot de passe utilisateur plutôt qu'avec le mot de passe du rôle. Par défaut, la valeur est `role`.
- Pour un utilisateur ou un rôle, les attributs suivants peuvent être définis :
  - Mot-clé `access_times` : indique les jours et heures auxquels les applications et services sont accessibles. Pour plus d'informations, reportez-vous à la page de manuel `getaccess_times`(3C).
  - `access_tz`, mot-clé : spécifie le fuseau horaire à utiliser pour l'interprétation de l'heure dans les entrées de `access_times`. Pour plus d'informations, reportez-vous à la page de manuel `pam_unix_account`(5).
  - `audit_flags`, mot-clé : modifiele masque d'audit. Pour plus d'informations, reportez-vous à la page de manuel `audit_flags`(5).
  - `auths`, mot-clé : attribue les autorisations. Pour plus d'informations, reportez-vous à la page de manuel `auths`(1).
  - `auth_profiles`, mot-clé : affecte des profils de droits authentifiés. Pour consulter des références, reportez-vous à la page de manuel `profiles`(1).
  - `defaultpriv`, mot-clé : ajoute des privilèges ou en supprime du jeu de privilèges de base par défaut.
  - `limitpriv`, mot-clé : ajoute des privilèges ou en supprime du jeu limite de privilèges par défaut.

    Les privilèges `defaultpriv` et `limitpriv` sont toujours en vigueur parce qu'ils sont affectés au processus initial de l'utilisateur. Pour plus d'informations, reportez-

vous à la page de manuel privileges(5) et à la section “Mise en oeuvre des privilèges” à la page 28.

- idlecmd, mot-clé : déconnecte l'utilisateur ou verrouille l'écran lorsque la valeur de idletime est atteinte.
- idletime, mot-clé : définit le délai pendant lequel le système reste disponible sans activité clavier. La valeur de idletime doit être définie si une valeur est définie pour idlecmd.
- lock_after_retries, mot-clé : si la valeur est yes, le système est verrouillé une fois que le nombre de tentatives a atteint le nombre autorisé dans le fichier /etc/default/login. Pour plus d'informations, reportez-vous à la page de manuel login(1).
- profiles, mot-clé : affecte des profils de droits. Pour plus d'informations, reportez-vous à la page de manuel profiles(1).
- project, mot-clé : ajoute un projet par défaut. Pour plus d'informations, reportez-vous à la page de manuel project(4).

Remarque - Dans la mesure où les attributs access_times et access_tz sont des attributs PAM, ils sont vérifiés lors de l'authentification. En conséquence, il doit être affectés directement à un utilisateur ou un rôle authentifié, ou dans un profil de droits. Standard ne seront pas pris en compte dans un profil de droits.

---

il est possible de définir des attributs qualifiés pour les utilisateurs et les rôles uniquement dans la portée de dénomination LDAP. Ces qualificatifs d'un utilisateur ou d'un rôle de limite d'affectation de l'attribut, tel qu'un profil de droits, afin d'un ou plusieurs systèmes. Pour consulter des exemples, reportez-vous aux pages de manuel useradd(1M) et user_attr(4).

Les qualificatifs sont host et netgroup:

- host, qualificatif : identifie le système sur lequel l'utilisateur ou le rôle peut accomplir les actions spécifiées.
- netgroup, qualificatif : dresse la liste des systèmes sur lesquels l'utilisateur ou le rôle peut exécuter les actions spécifiées. Les attributions de host ont la priorité sur les attributions de netgroup.

Pour plus d'informations, reportez-vous à la page de manuel user_attr(4). Pour afficher le contenu de cette base de données, utilisez la commande getent user_attr. Pour plus d'informations, reportez-vous à getent(1M) getent 1M et au Chapitre 6, Liste des droits dans Oracle Solaris.

Base de données auth_attr

Les définitions de la auth_attrbase de données stocke d'autorisation. Les autorisations peuvent être affectées à des utilisateurs, des rôles ou aux profils de droits. La meilleure méthode consiste à placer les autorisations dans un profil de droits, puis pour affecter le profil de droits à un rôle ou un utilisateur.

Pour afficher le contenu de cette base de données, utilisez la commande getent auth_attr. Pour plus d'informations, reportez-vous à getent(1M) getent 1M et Chapitre 6, Liste des droits dans Oracle Solaris.

Base de données prof_attr

La base de données prof_attr contient le nom, la description, l'emplacement du fichier d'aide, les privilèges et les autorisations qui sont affectés à des profils de droits. Les commandes et les attributs de sécurité qui sont affectés à des profils de droits sont stockés dans la base de données exec_attr. Pour plus d'informations, reportez-vous à la section “Base de données exec_attr” à la page 123.

Pour plus d'informations, reportez-vous à la page de manuel prof_attr(4). Pour visualiser le contenu de cette base de données, utilisez la commande getent exec_attr. Pour plus d'informations, reportez-vous à getent(1M) et au Chapitre 6, Liste des droits dans Oracle Solaris.

Base de données exec_attr

La base de données exec_attr définit les commandes nécessitant des attributs de sécurité pour la réussite de l'opération. Les commandes font partie d'un profil de droits. Une commande avec ses attributs de sécurité peut être exécutée par les rôles ou utilisateurs auxquels le profil est attribué.

Pour plus d'informations, reportez-vous à la page de manuel exec_attr(4). Pour visualiser le contenu de cette base de données, utilisez la commande getent. Pour plus d'informations, reportez-vous à getent(1M) getent 1M et au Chapitre 6, Liste des droits dans Oracle Solaris.

Fichier `policy.conf`

Le fichier `/etc/security/policy.conf` permet d'accorder des profils de droits, des autorisations et des privilèges spécifiques à tous les utilisateurs d'un système. Les entrées correspondantes dans le fichier sont constitués de paires *key=value* :

- `AUTHS_GRANTED=`*authorizations* : fait référence à une ou plusieurs autorisations.
- `AUTH_PROFS_GRANTED=`, *profils de droits* : fait référence à un ou plusieurs profils de droits.
- `PROFS_GRANTED=`*Profils de droits* : fait référence à un ou plusieurs profils de droits non authentifiés.
- `CONSOLE_USER=Console User` : fait référence au profil de droits Utilisateur de la console. Ce profil est fourni avec un ensemble pratique d'autorisations pour l'utilisateur de la console. Vous pouvez personnaliser ce profil.
- `PRIV_DEFAULT=`*privileges* : fait référence à un ou plusieurs privilèges.
- `PRIV_LIMIT=`*privileges* : fait référence à tous les privilèges.

L'exemple suivant illustre certaines valeurs de droits issues d'une base de données `policy.conf` :

```
##
AUTHS_GRANTED=
AUTH_PROFS_GRANTED=
CONSOLE_USER=Console User
PROFS_GRANTED=Basic Solaris User
#PRIV_DEFAULT=basic
#PRIV_LIMIT=all
```

Commandes pour l'administration de gestion des droits

Cette section répertorie les commandes utilisées pour administrer les droits. Elle comprend également un tableau des commandes dont l'accès peut être contrôlé via des autorisations.

Commandes qui génèrent une, Gérer les autorisations et des rôles, Rights Profiles

Les commandes répertoriées dans le tableau suivant récupèrent et définissent des droits sur tous les processus utilisateur.

TABLEAU 8-1 Droits Administration Commands

Commande	Description
auths(1)	Affiche les autorisations d'un utilisateur. Crée des autorisations.
getent(1M)	Répertorie le contenu des bases de données de droits.
nscd(1M)	Name service cache daemon (nscd), permet de mettre en cache les bases de données de droits. Utilisez la commande `svcadm` pour redémarrer le démon.
pam_roles(5)	Module de gestion des comptes de rôles pour PAM. Vérifie la présence de l'autorisation d'endosser un rôle.
pam_unix_account(5)	Module de gestion des comptes UNIX pour PAM. Compte des restrictions détermine s'il y a plus d'informations sur les resttrictions et, lorsque le délai d'inactivité a expiré, comme un intervalle de temps.
pfbash(1)	Permet de créer un processus shell de profil pouvant évaluer les droits.
pfedit(1M)	Permet de modifier des fichiers d'administration.
pfexec(1)	Permet d'exécuter une commande avec des attributs de sécurité.
policy.conf(4)	Fichier de configuration de la stratégie de sécurité du système. Répertorie les autorisations accordées, les privilèges accordés et d'autres informations de sécurité.
profiles(1)	Affiche les profils de droits de l'utilisateur spécifié. Crée ou modifie un profil de droits.
roles(1)	Affiche les rôles pouvant être endossés par l'utilisateur spécifié.
roleadd(1M)	Ajoute un rôle à un système local ou à un réseau LDAP.
roleadd(1M)	Ajoute un rôle à un système local ou à un réseau LDAP.
rolemod(1M)	Modifie les propriétés d'un rôle sur un système local ou sur un réseau LDAP.
userattr(1)	Affiche la valeur d'un droit spécifique qui est affecté à un utilisateur ou à un compte de rôle.
useradd(1M)	Ajoute un compte utilisateur au système ou à un réseau LDAP. L'option `-R` attribue un rôle au compte d'un utilisateur.
userdel(1M)	Supprime une connexion d'utilisateur du système ou d'un réseau LDAP.
usermod(1M)	Modifie les propriétés du compte d'un utilisateur sur le système.

Commandes sélectionnées nécessitant des autorisations

Le tableau suivant fournit des exemples de la façon dont les autorisations sont utilisées pour limiter les options de commande sur un système Oracle Solaris. Pour plus d'informations sur les autorisations, reportez-vous à la section “Les "feux verts" Reference (disponible en anglais uniquement)” à la page 119.

TABLEAU 8-2 Commandes et autorisations associées

Commande	Autorisations requises
at(1)	`solaris jobs.user` requise pour toutes les options (lorsque ni le fichier `at.allow` ni le fichier `at.deny` n'existent)
atq(1)	`solaris.jobs.admin` requise pour toutes les options
cdrw(1)	`solaris.device.cdrw` requise pour toutes les options qui sont accordées par défaut dans le fichier `policy.conf`
crontab(1)	`solaris jobs.user` requise pour l'option permettant de soumettre une tâche (lorsque ni le fichier `crontab.allow` ni le fichier `crontab.deny` n'existent) `solaris jobs.admin` requise pour les options permettant de répertorier ou de modifier les fichiers `crontab` d'autres utilisateurs
allocate(1)	`solaris.device.allocate` (ou toute autre autorisation spécifiée dans le fichier `device_allocate`) requise pour attribuer un périphérique `solaris.device.revoke` (ou toute autre autorisation spécifiée dans le fichier `device_allocate`) requise pour allouer un périphérique à un autre utilisateur (option `-F`)
deallocate(1)	`solaris.device.allocate` (ou toute autre autorisation spécifiée dans le fichier `device_allocate`) requise pour libérer un périphérique d'un autre utilisateur `solaris.device.revoke` (ou toute autre autorisation spécifiée dans `device_allocate`) requise pour forcer la libération du périphérique spécifié (option `-F`) ou de tous les périphériques (option `-I`)
list_devices(1)	`solaris.device.revoke` requise pour répertorier les périphériques d'un autre utilisateur (option `-U`)
roleadd(1M)	`solaris.user.manage` requise pour créer un rôle. `solaris.account.activate` requise pour définir le mot de passe initial. `solaris.account.setpolicy` requise pour définir une stratégie de mot de passe, comme le verrouillage de compte et le vieillissement du mot de passe.
roledel(1M)	Autorisation `solaris.passwd.assign` nécessaire pour supprimer le mot de passe.
rolemod(1M)	Autorisation `solaris.passwd.assign` nécessaire pour modifier le mot de passe. `solaris.account.setpolicy` requis pour modifier une stratégie de mot de passe, comme le verrouillage de compte et le vieillissement du mot de passe.
sendmail(1M)	`solaris.mail` requise pour accéder aux fonctions du sous-système de messagerie ; `solaris.mail.mailq` requise pour afficher la file d'attente de messagerie
useradd(1M)	`solaris.user.manage` requise pour créer un rôle. `solaris.account.activate` requise pour définir le mot de passe initial. `solaris.account.setpolicy` requise pour définir une stratégie de mot de passe, comme le verrouillage de compte et le vieillissement du mot de passe.
userdel(1M)	Autorisation `solaris.passwd.assign` nécessaire pour supprimer le mot de passe.
usermod(1M)	Autorisation `solaris.passwd.assign` nécessaire pour modifier le mot de passe. `solaris.account.setpolicy` requis pour modifier une stratégie de mot de passe, comme le verrouillage de compte et le vieillissement du mot de passe.

Privilèges (référence)

Des privilèges limitant les processus sont mis en oeuvre dans le noyau et peuvent limiter les processus au niveau des commandes, des utilisateurs, des rôles ou du système.

Commandes de gestion des privilèges

Le tableau suivant répertorie les commandes disponibles pour gérer les privilèges.

TABLEAU 8-3 Commandes de gestion des privilèges

Description	Commande	Page de manuel
Déboguage des défaillances des privilèges	`ppriv -eD` *failed-operation*	`ppriv`(1)
Dresser la liste des privilèges du système	`ppriv -l`	`ppriv`(1)
Répertorier un privilège et sa description	`ppriv -lv` *priv*	`ppriv`(1)
Répertorier la stratégie de privilège étendue sur un ID utilisateur, un processus ou un port	`ppriv -lv` *extended-policy*	`ppriv`(1)
Examiner les privilèges de processus	`ppriv -v` *pid*	`ppriv`(1)
Ajouter une stratégie de privilège étendu à un ID utilisateur, un processus ou un port	`ppriv -r` *rule*	`privileges`(5)
Définir les privilèges des processus	`ppriv -s` *spec*	`ppriv`(1)
Supprimer une règle d'une stratégie de privilège étendu	`ppriv -X` *rule*	`privileges`(5)
Affecter des privilèges à un profil de droits	`profiles -p` *profile-name*	`profiles`(1)
Affecter des privilèges à un nouveau rôle	`roleadd -K defaultpriv=`	`roleadd`(1M)
Ajouter des privilèges à un rôle existant	`rolemod -K defaultpriv+=`	`rolemod`(1M)
Attribuer des privilèges à un nouvel utilisateur	`useradd -K defaultpriv=`	`useradd`(1M)
Ajouter des privilèges à un utilisateur existant	`usermod -K defaultpriv+=`	`usermod`(1M)
Ajouter une stratégie de périphérique à un périphérique	`add_drv -p` *policy driver*	`add_drv`(1M)
Définir une stratégie de périphérique	`devfsadm`	`devfsadm`(1M)
Afficher une stratégie de périphérique	`getdevpolicy`	`getdevpolicy`(1M)
Mettre à jour la stratégie de périphérique sur des périphériques ouverts	`update_drv -p` *policy driver*	`update_drv`(1M)

Fichiers contenant des informations sur les privilèges

Les fichiers `policy.conf`et contiennent des informations sur les privilèges. `syslog.conf`

- :`/etc/security/policy.conf` contient les éléments suivants des informations sur les privilèges
  - `PRIV_DEFAULT` : jeu de privilèges héritable pour le système
  - `PRIV_LIMIT` : jeu limite de privilèges pour le système

  Pour plus d'informations, reportez-vous à la page de manuel `policy.conf`(4).
- `/etc/syslog.conf`est le fichier journal du système pour les messages de débogage qui sont liées à Débogage de privilège. Le chemin d'accès pour les messages de débogage priv.debug, entrée est défini dans le.

  Pour plus d'informations, reportez-vous à la page de manuel `syslog.conf`(4).

La région Actions de la doté de privilèges d'enregistrement d'audit

L'utilisation des privilèges peut être auditée. A chaque fois qu'un processus utilise un privilège, l'utilisation du privilège est enregistrée dans la piste d'audit du jeton d'audit `upriv`. Lorsque les noms de privilèges font partie de l'enregistrement, leur représentation textuelle est utilisée. Les événements d'audit suivants enregistrent l'utilisation de privilèges :

- `AUE_SETPPRIV` **(événement d'audit)** : génère un enregistrement d'audit lorsqu'un jeu de privilèges est modifié. L'événement d'audit `AUE_SETPPRIV` se trouve dans la classe `pm`.
- `AUE_MODALLOCPRIV`**(événement d'audit) :** génère un enregistrement d'audit lorsqu'un privilège est ajouté depuis l'extérieur du noyau. L'événement d'audit `AUE_MODALLOCPRIV` se trouve dans la classe `ad`.
- `AUE_MODDEVPLCY` **(événement d'audit) :** génère un enregistrement d'audit lorsque la stratégie liée au périphérique est modifiée. L'événement d'audit `AUE_MODDEVPLCY` se trouve dans la classe `ad`.
- `AUE_PFEXEC` **(événement d'audit)** : génère un enregistrement d'audit lorsqu'un appel est effectué à `execve()` avec `pfexec()` activé. L'événement d'audit `AUE_PFEXEC` se trouve dans les classes d'audit `as`, `ex`, `ps` et `ua`. Les noms des privilèges sont inclus dans l'enregistrement d'audit.

L'utilisation réussie de privilèges inclus dans le jeu de base n'est pas auditée. La tentative d'utilisation d'un privilège de base qui a été supprimé du jeu de base d'un utilisateur fait l'objet d'un audit.

Glossaire de la sécurité

AES Standard de chiffrement avancé (Advanced Encryption Standard). Technique de chiffrement de données symétrique par blocs de 128 bits. Le gouvernement des Etats-Unis a adopté la variante Rijndael de l'algorithme comme norme de chiffrement en octobre 2000. AES remplace le chiffrement Principal d'utilisateur comme norme administrative.

algorithme Algorithme cryptographique. Il s'agit d'une procédure de calcul récursive établie qui chiffre ou hache une entrée.

algorithme cryptographique Voir algorithme.

allocation de périphériques Protection des périphériques au niveau de l'utilisateur. L'allocation de périphériques met en oeuvre l'utilisation exclusive d'un périphérique par un utilisateur à la fois. Les données des périphériques sont purgées avant toute réutilisation d'un périphérique. Des autorisations peuvent être utilisées pour limiter les utilisateurs autorisés à allouer un périphérique.

Amorce Valeur numérique de départ pour la génération de nombres aléatoires. Lorsque cette valeur provient d'une source aléatoire, l'amorce est appelée *amorce aléatoire*.

Application privilégiée Application pouvant remplacer les contrôles système. L'application vérifie les attributs de sécurité, tels que des UID spécifiques, des ID de groupe, des autorisations ou des privilèges.

Attributs de sécurité Remplacements de la stratégie de sécurité qui permettent à une commande d'administration de s'exécuter correctement lorsqu'elle est exécutée par un utilisateur autre que le superutilisateur. Dans le modèle de superutilisateur, les programmes `setuid root` et `setgid` sont des attributs de sécurité. Lorsque ces attributs sont appliqués à une commande, la commande s'exécute correctement, quel que soit l'utilisateur qui l'exécute. Dans le Modèle de privilège, les privilèges du noyau et les autres droits remplacent les programmes `setuid root` comme attributs de sécurité. Le modèle de privilège est compatible avec le modèle de superutilisateur dans la mesure où le modèle de privilège reconnaît également les programmes `setuid` et `setgid` comme des attributs de sécurité.

Authentificateur Des authentificateurs sont transmis par des clients lors de la demande de tickets (à un KDC) et de services (à un serveur). Ils contiennent des informations générées par l'utilisation d'une clé de session connue uniquement du client et du serveur, dont l'origine récente peut être prouvée, indiquant ainsi que la transaction est sécurisée. Lorsqu'un authentificateur est utilisé avec un

ticket, il peut permettre d'authentifier un principal d'utilisateur. Un authentificateur inclut le nom du principal de l'utilisateur, l'adresse IP de l'hôte de l'utilisateur, ainsi qu'un horodatage. A la différence d'un ticket, un authentificateur ne peut servir qu'une seule fois, généralement lorsque l'accès à un service est demandé. Un authentificateur est chiffré à l'aide de la clé de session pour ce client et ce serveur.

Authentification Processus de vérification de l'identité déclarée d'un principal.

Autorisation 1. Dans Kerberos, processus consistant à déterminer si un principal peut utiliser un service et à définir les objets auxquels il peut accéder, ainsi que le type d'accès autorisé pour chaque objet.

2. Dans la gestion des droits des utilisateurs, autorisation pouvant être attribuée à un rôle ou un utilisateur (ou intégrée dans un profil de droits) en vue de l'exécution d'une classe d'opérations autrement interdites par la stratégie de sécurité. Les "feux verts" sont mises en oeuvre au niveau de l'application utilisateur, et non dans le noyau.

Blowfish Algorithme de chiffrement par bloc symétrique de longueur de clé variable (entre 32 et 448 bits). Son créateur, Bruce Schneier, affirme que Blowfish est optimisé pour les applications pour lesquelles la clé n'a pas besoin d'être régulièrement modifiée.

Cache d'informations d'identification Espace de stockage (généralement un fichier) contenant des informations d'identification reçues de KDC.

champ d'application du service de noms Champ d'application dans lequel un rôle est autorisé à fonctionner, c'est-à-dire un hôte particulier ou tous les hôtes desservis par un service de noms spécifié, tel que NIS LDAP.

Chiffrement par clé privée Dans le cas du chiffrement par clé privée, l'expéditeur et le destinataire utilisent la même clé de chiffrement. Voir également chiffrement par clé publique.

chiffrement par clé publique Modèle de chiffrement où chaque utilisateur dispose de deux clés, l'une publique et l'autre privée. Dans le cas du chiffrement par clé publique, l'expéditeur chiffre le message à l'aide de la clé publique du destinataire, et ce dernier se sert d'une clé privée pour le déchiffrer. Le service Kerberos est un système à clé privée. Voir également Chiffrement par clé privée.

clé de service Clé de chiffrement partagée par un principal de service et le KDC, et distribuée en dehors des limites du système. Voir également Touche.

Clé de session Clé générée par le service d'authentification ou le service d'octroi de ticket. Une clé de session est générée dans le but de sécuriser les transactions entre un client et un service. La durée de vie d'une clé de session est limitée à une seule session de connexion. Voir également Touche.

Clé privée Chaque principal utilisateur reçoit une clé qui n'est connue que du KDC et de l'utilisateur du principal. Pour les principaux d'utilisateur, la clé est basée sur le mot de passe de l'utilisateur. Voir également Touche.

Clé secrète Voir Clé privée.

client Au sens strict, il s'agit d'un processus utilisant un service réseau pour le compte d'un utilisateur, par exemple, une application utilisant `rlogin`. Dans certains cas, un serveur peut être lui-même le client d'un autre serveur ou service.

Au sens large, il s'agit d'un hôte qui a) reçoit des informations d'identification Kerberos et b) utilise un service fourni par un serveur.

Dans la pratique, ce terme désigne un principal utilisant un service.

Confidentialité Voir confidentialité.

confidentialité Un service de sécurité, dans lequel les données transmises sont chiffrées avant leur envoi. La confidentialité inclut également l'intégrité des données et l'authentification de l'utilisateur. Voir également Authentification, intégrité, service.

Conscient des privilèges Programmes, scripts et commandes qui activent et désactivent l'utilisation des privilèges dans leur code. Dans un environnement de production, les privilèges qui sont activés doivent être fournis au processus, par exemple, en demandant aux utilisateurs du programme d'utiliser un profil de droits qui ajoute les privilèges au programme. Pour une description complète des privilèges, reportez-vous à la page de manuel `privileges`(5).

credential Package d'informations comprenant un ticket et une clé de session correspondante. Informations utilisées pour authentifier l'identité d'un principal. Voir également ticket, Clé de session.

DES Standard de chiffrement de données (Data Encryption Standard). Méthode de chiffrement à clé symétrique développée en 1975 et standardisée par l'ANSI en 1981 comme ANSI X.3.92. Le DES utilise une clé de 56 bits.

Destinataire Dans la fonctionnalité Structure cryptographique d'Oracle Solaris, un consommateur est un utilisateur des services cryptographiques provenant de fournisseurs. Les consommateurs peuvent être des applications, des utilisateurs finaux ou des opérations de noyau. Kerberos, IKE et IPsec sont des exemples de consommateurs. Pour consulter des exemples de fournisseurs, reportez-vous à la section fournisseur.

domaine (realm) 1. Réseau logique desservi par une seule base de données Kerberos et un ensemble de centre de distribution des clés (KDC).

2. Troisième partie d'un nom de principal. Pour le nom de principal `jdoe/admin@CORP.EXAMPLE.COM`, le domaine est `CORP.EXAMPLE.COM`. Voir également Nom de principal.

droits Alternative au modèle tout ou rien des superutilisateurs. La gestion des droits des utilisateurs et la gestion des droits de processus permettent à une organisation de répartir les privilèges du superutilisateur et que vous les affectez aux utilisateurs ou rôles. Les droits dans Oracle Solaris sont implémentés en tant que privilèges de noyau, d'autorisation et de capacité à exécuter un

processus sous la forme d'un UID ou GID spécifique. Les droits peuvent être regroupés dans un Profil de droits et un rôle.

DSA Algorithme de signature numérique (Digital Signature Algorithm). Algorithme de clé publique dont la longueur de clé varie de 512 à 4 096 bits. La norme du gouvernement américain, DSS, atteint 1 024 bits. L'algorithme DSA repose sur l'algorithme SHA1 en entrée.

écart d'horloge Ecart maximal toléré entre les horloges système internes de tous les hôtes participant au système d'authentification Kerberos. Si l'écart d'horloge est dépassé entre des hôtes participants, les demandes sont rejetées. L'écart d'horloge est spécifié dans le fichier `krb5.conf`.

ECDSA Algorithme de signature numérique de courbe elliptique. Algorithme de clé publique basé sur une courbe elliptique mathématique. Une clé ECDSA est de beaucoup plus petite taille qu'une clé publique DSA nécessaire pour générer une signature de la même longueur.

Escalade des privilèges Accès aux ressources situées en dehors de la plage, les ressources qui y compris les droits qui vous sont affectés qui remplacent les valeurs par défaut, droits y autorisent. Il en résulte qu'un processus peut effectuer des actions non autorisées.

Evénement d'audit asynchrone Les événements asynchrones constituent la minorité des événements système. Ces événements ne sont associés à aucun processus, de sorte qu'aucun processus ne peut être bloqué puis réactivé ultérieurement. L'initialisation système initiale et les événements d'entrée et de sortie PROM sont des exemples d'événements asynchrones.

événement d'audit non allouable Evénement d'audit dont l'initiateur ne peut pas être déterminé, tel que l'événement AUE_BOOT.

événement d'audit synchrone Majorité des événements d'audit. Ces événements sont associés à un processus dans le système. Un événement non allouable associé à un processus est un événement synchrone, tel que l'échec d'une connexion.

fichier de ticket Voir Cache d'informations d'identification.

fichier keytab Fichier de table de clés contenant une ou plusieurs clés (principaux). Un hôte ou un service utilise un fichier keytab à peu près de la même façon qu'un utilisateur se sert d'un mot de passe.

fichier stash Un fichier stash contient une copie chiffrée de la clé principale pour le KDC. Cette clé principale est utilisée lorsqu'un serveur est réinitialisé pour authentifier automatiquement le KDC avant qu'il ne démarre les processus `kadmind` et `krb5kdc`. Etant donné que le fichier stash inclut la clé principale, ce fichier et toutes ses sauvegardes doivent être sécurisés. Si le chiffrement est compromis, la clé peut être utilisée pour accéder à la base de données KDC ou la modifier.

fichiers d'audit Journaux d'audit binaires. Les fichiers d'audit sont stockés séparément dans un système de fichiers d'audit.

fournisseur Dans la fonctionnalité Structure cryptographique d'Oracle Solaris, service de cryptographie fourni aux consommateurs. Les bibliothèques PKCS #11, les modules cryptographiques du noyau et les accélérateurs de matériel sont des exemples de fournisseurs. Les fournisseurs se connectent à la structure cryptographique et sont donc également appelés *plug-ins*. Pour consulter des exemples de consommateurs, voir Destinataire.

Fournisseur de logiciels Dans la fonctionnalité Structure cryptographique d'Oracle Solaris, module logiciel de noyau ou bibliothèque PKCS#11 fournissant des services cryptographiques. Voir également fournisseur.

Fournisseur de matériel Dans la fonctionnalité Structure cryptographique d'Oracle Solaris, un pilote de périphérique et son accélérateur matériel. Les fournisseurs de matériel déchargent le système informatique d'opérations cryptographiques coûteuses, libérant ainsi les ressources de l'unité centrale pour d'autres utilisations. Voir également fournisseur.

FQDN Nom de domaine complet. Par exemple, `central.example.com` (et pas simplement `denver`).

GSS-API Interface de programmation d'application générique de service de sécurité. Couche réseau assurant la prise en charge de différents services de sécurité modulaires, y compris le service Kerberos. GSS-API fournit des services d'authentification, d'intégrité et de confidentialité. Voir également Authentification, intégrité, confidentialité.

hôte Système accessible par l'intermédiaire d'un réseau.

Hôte principal Instance spécifique d'un principal de service dans lequel le principal (indiqué par le nom primaire `host`) est configuré de manière à fournir une gamme de services réseau, comme `ftp`, `rcp` ou `rlogin`. `host/central.example.com@EXAMPLE.COM` est un exemple d'hôte principal. Voir également principal de serveur.

Image système unique Une image système unique est utilisée dans l'audit d'Oracle Solaris afin de décrire un groupe de systèmes audités utilisant le même service de noms. Ces systèmes envoient leurs enregistrements d'audit à un serveur d'audit central où les enregistrements peuvent être comparés comme s'ils provenaient d'un même système.

instance Deuxième partie d'un nom de principal, une instance qualifie le primaire du principal. Dans le cas d'un principal de service, l'instance est requise. L'instance est le nom de domaine complet de l'hôte, comme dans `host/central.example.com`. Pour les principaux d'utilisateur, une instance est facultative. Notez, cependant, que `jdoe` et `jdoe/admin` sont des principaux uniques. Voir également primaire, Nom de principal, principal de service, Principal d'utilisateur.

intégrité Service de sécurité qui assure, outre l'authentification de l'utilisateur, la validité des données transmises par le biais de sommes de contrôle cryptographiques. Voir également Authentification, confidentialité.

jeu autorisé Jeu des privilèges disponibles pour l'utilisation par un processus.

jeu de base Jeu de privilèges affecté à un processus d'utilisateur lors de la connexion. Sur un système non modifié, le jeu héritable initial de chaque utilisateur correspond au jeu de base défini au moment de la connexion.

Jeu de privilèges Ensemble de privilèges. Chaque processus comporte quatre jeux de privilèges qui déterminent s'il peut utiliser un privilège particulier. Voir jeu limite, jeu effectif, jeu autorisé et jeu hérité.

De même, le jeu de base de privilèges est l'ensemble des privilèges affectés aux processus d'un utilisateur au moment de la connexion.

jeu effectif Jeu de privilèges actuellement en vigueur sur un processus.

jeu hérité Jeu de privilèges dont un processus peut hériter en appelant `exec`.

jeu limite Limite extérieure des privilèges disponibles pour un processus et ses enfants.

KDC Centre de distribution de clés (Key Distribution Center). Machine disposant de trois composants Kerberos V5.

- Principal et base de données de clés
- Service d'authentification
- Service d'octroi de tickets

Chaque domaine dispose d'un KDC maître et doit avoir un ou plusieurs KDC esclaves.

KDC esclave Copie d'un KDC maître capable de réaliser la plupart des fonctions du maître. Chaque domaine dispose généralement de plusieurs KDC esclaves et d'un seul KDC maître. Voir également KDC, KDC maître.

KDC maître KDC maître dans chaque domaine, comprenant un serveur d'administration Kerberos, `kadmind` et un démon d'authentification et d'octroi de tickets, `krb5kdc`. Chaque domaine doit disposer d'au moins un KDC maître, et peut avoir plusieurs KDC de duplication ou esclaves fournissant des services d'authentification aux clients.

Kerberos Service d'authentification, protocole utilisé par ce service ou code servant à mettre en oeuvre ce service.

Mise en oeuvre Kerberos d'Oracle Solaris étroitement basée sur la mise en oeuvre Kerberos V5.

Bien que techniquement différents, "Kerberos" et "Kerberos V5" sont souvent utilisés de façon interchangeable dans la documentation Kerberos.

Dans la mythologie grecque, Kerberos (en français Cerbère) était un chien féroce tricéphale qui gardait la porte des Enfers.

keystore Un keystore contient des mots de passe, des phrases de passe, des certificats et d'autres objets d'authentification pour l'extraction par des applications. Il peut être spécifique à une technologie, ou à un emplacement utilisé par plusieurs applications.

kvno Numéro de version de la clé. Numéro de série faisant le suivi d'une clé spécifique selon l'ordre dans lequel elle a été générée. Plus le numéro kvno est élevé, plus la clé est récente.

liste de contrôle d'accès (ACL) Une liste de contrôle d'accès (ACL) offre une sécurité des fichiers plus précise que la protection de fichier UNIX conventionnelle. Par exemple, une ACL vous permet d'autoriser l'accès en lecture de groupe à un fichier, tout en autorisant un seul membre de ce groupe à écrire dans le fichier.

MAC 1. Voir MAC.

2. Egalement appelé étiquetage. Dans la terminologie de sécurité gouvernementale, MAC signifie Mandatory Access Control (contrôle d'accès obligatoire). Les étiquettes telles que Top Secret et Confidential sont des exemples de MAC. MAC diffère de DAC (Discretionary Access Control, contrôle d'accès discrétionnaire). Les autorisations UNIX constituent un exemple de DAC.

3. Dans le matériel, il s'agit de l'adresse système unique sur un réseau local (LAN). Si le système est sur un réseau Ethernet, le MAC est l'adresse Ethernet.

MAC MAC garantit l'intégrité des données et authentifie leur origine. MAC ne protège aucunement contre l'écoute frauduleuse des informations échangées.

MD5 Fonction de hachage cryptographique répétitive utilisée pour authentifier les messages, y compris les signatures numériques. Elle a été développée en 1991 par Rivest. Son utilisation a été désapprouvée.

mécanisme 1. Package logiciel spécifiant des techniques cryptographiques pour assurer l'authentification ou la confidentialité des données. Exemples : Kerberos V5, clé publique Diffie-Hellman

2. Dans la fonctionnalité Structure cryptographique d'Oracle Solaris, mise en oeuvre d'un algorithme destiné à un usage particulier. Par exemple, un mécanisme DES appliqué à l'authentification, tel que CKM_DES_MAC, est un mécanisme distinct d'un mécanisme DES appliqué au chiffrement, CKM_DES_CBC_PAD.

Mécanisme de sécurité Voir mécanisme.

minimisation Installation du système d'exploitation minimal nécessaire pour l'exécution du serveur. Tout logiciel n'étant pas directement lié au fonctionnement du serveur n'est pas installé, ou est supprimé après l'installation.

Modèle de privilège Modèle de sécurité plus strict que le modèle superutilisateur sur un système informatique. Dans le modèle de privilège, les processus nécessitent des privilèges pour s'exécuter. L'administration du système peut être divisée en quatre parties discrètes basées sur les privilèges dont les administrateurs disposent dans leurs processus. Des privilèges peuvent être affectés au processus de connexion d'un administrateur, ou des privilèges peuvent être affectés de manière à ne s'appliquer qu'à certaines commandes.

modèle de superutilisateur Modèle de sécurité UNIX standard sur un système informatique. Dans le modèle de superutilisateur, un administrateur dispose d'un contrôle de type tout ou rien sur le système.

En règle générale, pour l'administration de la machine, un utilisateur se connecte en tant que superutilisateur (`root`) et peut effectuer toutes les activités d'administration.

moindre privilège Modèle de sécurité qui octroie à un processus spécifié uniquement un sous-ensemble de pouvoirs de superutilisateur. Le modèle de moindre privilège octroie aux utilisateurs standard les privilèges suffisants pour qu'ils puissent effectuer des tâches d'administration personnelles, telles que le montage de systèmes de fichiers et la modification de l'appartenance des fichiers. Par ailleurs, les processus s'exécutent uniquement avec les privilèges dont ils ont besoin pour terminer la tâche, et non avec tous les pouvoirs du superutilisateur, c'est-à-dire, tous les privilèges. Les dommages causés par les erreurs de programmation, comme les débordements de la mémoire tampon, peuvent être limités à un utilisateur non root, qui n'a pas accès à des fonctions essentielles comme la lecture ou l'écriture de fichiers système protégés ou l'arrêt de la machine.

Moteur d'analyse Application tierce, résidant sur un hôte externe, qui examine un fichier à la recherche de virus connus.

Nom de principal 1. Le nom d'un principal, au format *primary/instance@REALM*. Voir également instance, primaire, domaine (realm).

2. (RPCSEC_GSS API) Voir principal de client, principal de serveur.

NTP Network Time Protocol. Logiciel de l'Université de l'Etat du Delaware qui vous permet de gérer avec précision la synchronisation de l'heure ou de l'horloge réseau, ou les deux, dans un environnement réseau. Vous pouvez utiliser le protocole NTP pour préserver l'écart d'horloge dans un environnement Kerberos. Voir également écart d'horloge.

Objet public Fichier appartenant à l'utilisateur `root` et lisible par tout le monde, tel que n'importe quel fichier du répertoire `/etc`.

PAM Module d'authentification enfichable (Pluggable Authentication Module). Structure permettant d'utiliser plusieurs mécanismes d'authentification sans recompilation des services recourant à ces mécanismes. PAM permet l'initialisation de la session SEAM au moment de son ouverture

phrase de passe Phrase utilisée pour vérifier qu'une clé privée a été créée par l'utilisateur de la phrase de passe. Une bonne phrase de passe contient 10 à 30 caractères alphanumériques et évite les noms et proses simples. Vous êtes invité à saisir la phrase de passe pour authentifier l'utilisation de la clé privée pour chiffrer et déchiffrer les communications.

piste de vérification Collection de tous les fichiers d'audit provenant de tous les hôtes.

Politique d'audit Paramètres globaux et par utilisateur qui déterminent les événements d'audit enregistrés. Les paramètres globaux s'appliquant au service d'audit déterminent généralement les informations facultatives à inclure dans la piste d'audit. Deux paramètres, `cnt` et `ahlt`, affectent le fonctionnement du système lorsque la file d'attente de l'audit est pleine. Par exemple, la stratégie d'audit peut exiger qu'un numéro de séquence fasse partie de chaque enregistrement d'audit.

primaire Première partie du nom d'un nom de principal. Voir également instance, Nom de principal, domaine (realm).

principal 1. Client/utilisateur dont le nom est unique ou instance de serveur/service participant à une communication en réseau. Les transactions Kerberos impliquent des interactions entre des principaux (principaux de service et d'utilisateur) ou entre des principaux et des KDC. En d'autres termes, un principal est une entité unique à laquelle Kerberos peut attribuer des tickets. Voir également Nom de principal, principal de service, Principal d'utilisateur.

2. (RPCSEC_GSS API) Voir principal de client, principal de serveur.

Principal admin Principal d'utilisateur dont le nom est au format *nom_utilisateur*`/admin` (comme dans `jdoe/admin`). Un principal admin peut disposer de davantage de privilèges (de modification de stratégies par exemple) qu'un principal d'utilisateur standard. Voir également Nom de principal et Principal d'utilisateur.

Principal d'utilisateur Principal attribué à un utilisateur particulier. Le nom primaire d'un principal d'utilisateur est un nom d'utilisateur et son instance facultative est un nom utilisé pour décrire l'utilisation prévue des informations d'identification correspondantes (`jdoe` ou `jdoe/admin` par exemple). Egalement appelé instance d'utilisateur. Voir également principal de service.

principal de client (RPCSEC_GSS API) Client (utilisateur ou application) utilisant des services réseau sécurisés RPCSEC_GSS. Les noms de principaux client sont stockés sous forme de structures `rpc_gss_principal_t`.

principal de serveur (RPCSEC_GSS API) Principal fournissant un service. Le principal de serveur est stocké sous forme d'une chaîne de caractères ASCII dont le format est *service*`@`*hôte*. Voir également principal de client.

principal de service Principal assurant l'authentification Kerberos pour un ou plusieurs services. Pour les principaux de service, le nom primaire est un nom de service, tel que `ftp`, et son instance est le nom d'hôte complet du système fournissant le service. Voir également Hôte principal, Principal d'utilisateur.

principe du moindre privilège Voir moindre privilège.

privilège 1. En règle générale, la puissance ou la capacité d'exécuter une opération sur un système d'ordinateur qui est au-delà des pouvoirs d'un utilisateur régulier Les privilèges de superutilisateur sont tous les droits octroyés au superutilisateur. Application privilégiée par un utilisateur doté de privilèges ou est un utilisateur ou application qui a reçu des droits supplémentaires.

2. Droit discret dans le cadre d'un processus dans un système Oracle Solaris. Les privilèges offrent un contrôle des processus plus détaillé que `root`. Les privilèges sont définis et appliqués dans le noyau. Les privilèges sont également appelés *privilèges de processus* ou *privilèges*

de noyau. Pour une description complète des privilèges, reportez-vous à la page de manuel `privileges`(5).

Profil de droits On parle aussi de profil. Ensemble de remplacements de sécurité pouvant être affecté à un rôle ou à un utilisateur. Un profil de droits peut se composer d'autorisations, privilèges, de commandes avec des attributs de sécurité et d'autres profils de droits sont appelés les profils supplémentaires.

Profil de droits authentifiés Profil de droits qui nécessite que l'utilisateur ou le rôle qui lui est affecté saisisse un mot de passe avant d'exécuter une opération à partir du profil. Ce comportement est similaire au comportement de `sudo`. La durée de temps pendant laquelle le mot de passe est valide est configurable.

protocole Diffie-Hellman Egalement appelé cryptographie par clé publique. Protocole d'accord de clés cryptographiques asymétriques mis au point par Diffie et Hellman en 1976. Ce protocole permet à deux utilisateurs d'échanger une clé secrète via un moyen non sécurisé sans secrets préalables. Diffie-Hellman est utilisé par Kerberos.

QOP Qualité de la protection. Paramètre servant à sélectionner les algorithmes cryptographiques utilisés en association avec le service d'intégrité ou de confidentialité.

RBAC Contrôle d'accès basé sur les rôles, fonction de gestion des droits des utilisateurs d'Oracle Solaris. Reportez-vous à la section droits.

Réauthentification Mot de passe qu'il soit nécessaire de fournir une opération. pour effectuer un ordinateur En règle générale, les opérations `sudo` nécessitent une réauthentification. Les profils de droits peuvent contenir authentifiés des commandes qui nécessitent la réauthentification. Voir Profil de droits authentifiés .

relation Variable ou relation de configuration définie dans les fichiers `kdc.conf` ou `krb5.conf`.

renforcement Modification de la configuration par défaut du système d'exploitation pour supprimer les failles de sécurité inhérentes à l'hôte.

rôle Identité spéciale destinée à l'exécution d'applications privilégiées et ne pouvant être prise que par des utilisateurs assignés.

RSA Méthode permettant d'obtenir des signatures numériques et des systèmes de cryptographie par clé publique. Cette méthode datant de 1978 a été décrite par trois développeurs (Rivest, Shamir et Adleman).

SEAM Nom de produit de la version initiale de Kerberos sur les systèmes Solaris. Ce produit est à partir de la technologie Kerberos V5 développé au Massachusetts Institute of Technology. SEAM est maintenant appelé le service Kerberos. Il continue d'être légèrement différent de l'version MIT.

Secure Shell, shell sécurisé Un protocole particulier pour une connexion à distance sécurisée et d'autres services de réseau sécurisé via un réseau non sécurisé.

Séparation des tâches Composante de la notion de moindre privilège. La séparation des tâches permet d'empêcher un utilisateur d'exécuter ou d'approuver les opérations terminant une transaction. Par exemple, dans RBAC, vous pouvez séparer la création d'un utilisateur de connexion de l'affectation des remplacements de sécurité. Un rôle crée l'utilisateur. Un autre rôle peut affecter les attributs de sécurité, tels que des profils de droits, des rôles et des privilèges aux utilisateurs existants.

Serveur Serveur principal fournissant une ressource aux clients réseau. Si vous vous connectez par `ssh` au système `central.example.com` par exemple, ce système est le serveur fournissant le service `ssh`. Voir également principal de service.

serveur d'application réseau Serveur fournissant une application réseau, telle que `ftp`. Un domaine peut contenir plusieurs serveurs d'application réseau.

Serveur d'applications Voir serveur d'application réseau.

service 1. Ressource fournie aux clients du réseau, souvent par plusieurs serveurs. Si vous vous connectez par `rlogin` à la machine `central.example.com` par exemple, cette machine est le serveur fournissant le service `rlogin`.

2. Service de sécurité (d'intégrité ou de confidentialité) fournissant un niveau de protection supérieur à l'authentification. Voir également intégrité et confidentialité.

service de sécurité Voir service.

SHA1 Secure Hashing Algorithm, algorithme de hachage sécurisé. L'algorithme s'applique à toute longueur d'entrée inférieure à 2^{64} afin d'obtenir une synthèse des messages. L'algorithme SHA1 sert d'entrée à DSA.

Shell de profil Dans la gestion des droits, shell permettant à un rôle (ou un utilisateur) d'exécuter, à partir de la ligne de commande, toutes les applications privilégiées affectées au profil de droits du rôle. Les versions du shell de profil correspondent aux shells disponibles sur le système, comme la version `pfbash` de `bash`

stratégie En règle générale, plan ou ensemble d'actions qui influence ou détermine les décisions et actions. Pour les systèmes informatiques, la stratégie fait généralement référence à la stratégie de sécurité. La stratégie de sécurité de votre site constitue un ensemble de règles qui définissent la sensibilité des informations traitées et les mesures prises pour protéger les informations contre tout accès non autorisé. Par exemple, la stratégie de sécurité peut exiger que les systèmes soient audités, que les périphériques soient protégés par des privilèges et que les mots de passe soient modifiés toutes les six semaines.

Pour l'implémentation des stratégies dans des zones spécifiques du SE Oracle Solaris, voir Politique d'audit, stratégie dans la structure cryptographique, stratégie de périphériques, Stratégie Kerberos, stratégie de mot de passe et Stratégie de droits.

stratégie dans la structure cryptographique Dans la fonctionnalité Structure cryptographique d'Oracle Solaris, la stratégie correspond à la désactivation de mécanismes de chiffrement existants. Les mécanismes ne peuvent ensuite plus être utilisés. La stratégie de la structure cryptographique peut empêcher l'utilisation d'un mécanisme particulier tel que `CKM_DES_CBC` provenant d'un fournisseur tel que DES.

Stratégie de droits Stratégie de sécurité associée à une commande. Actuellement, `solaris` est la stratégie valide pour Oracle Solaris. La stratégie `solaris` reconnaît les privilèges et la stratégie de privilège étendue, les autorisations et les attributs de sécurité de `setuid`.

stratégie de mot de passe Algorithmes de chiffrement pouvant être utilisés pour générer des mots de passe. Peut également faire référence à des questions plus générales concernant les mots de passe, telles que la fréquence à laquelle le mot de passe doit être modifié, le nombre de tentatives autorisées de saisie d'un mot de passe et d'autres considérations relatives à la sécurité. La stratégie de sécurité requiert des mots de passe. La stratégie de mot de passe peut requérir des mots de passe chiffrés avec l'algorithme AES, et imposer d'autres exigences relatives à la force des mots de passe.

stratégie de périphériques Protection d'un périphérique au niveau du noyau. La stratégie de périphériques repose sur deux jeux de privilèges pour un périphérique. Un jeu de privilèges contrôle l'accès en lecture au périphérique. Le deuxième jeu de privilèges contrôle l'accès en écriture sur le périphérique. Voir également stratégie.

stratégie de sécurité Voir stratégie.

Stratégie Kerberos Ensemble de règles régissant l'utilisation des mots de passe dans le service Kerberos. Les stratégies peuvent réguler les accès des principaux ou des paramètres de ticket, tels que la durée de vie.

stratégie pour technologies à clé publique Dans la structure de gestion des clés (KMF), la stratégie correspond à la gestion de l'utilisation des certificats. La base de données de stratégies KMF peut définir des contraintes s'appliquant à l'utilisation des clés et des certificats gérés par la bibliothèque KMF.

Stratégie RBAC Reportez-vous à la section Stratégie de droits.

stratégies de réseau Paramètres configurés par les utilitaires réseau pour protéger le trafic réseau. Pour plus d'informations sur la sécurité réseau, reportez-vous à la section “ Sécurisation du réseau dans Oracle Solaris 11.2 ”.

synthèse Voir Synthèse de message.

Synthèse de message Valeur de hachage calculée à partir d'un message. La valeur de hachage identifie le message de manière presque unique. Une synthèse permet de vérifier l'intégrité d'un fichier.

TGS Service d'octroi de tickets (Ticket-Granting Service) Partie du KDC responsable de l'émission des tickets.

TGT Ticket d'octroi de tickets (Ticket-Granting Ticket) Ticket émis par le KDC, permettant au client de demander des tickets pour d'autres services.

ticket Paquet d'informations servant à transmettre en toute sécurité l'identité d'un utilisateur à un serveur ou un service. Un ticket n'est valable que pour un client et un service particulier sur un serveur spécifique. Il contient le nom de principal du service, le nom de principal de l'utilisateur, l'adresse IP de l'hôte de l'utilisateur, un horodatage et une valeur définissant la durée de vie du ticket. La création d'un ticket s'effectue à l'aide d'une clé de session aléatoire utilisée par le client et le service. Une fois le ticket créé, il peut être réutilisé jusqu'à son expiration. Un ticket sert uniquement à authentifier un client lorsqu'il est présenté avec un nouvel authentificateur. Voir également Authentificateur, credential, service, Clé de session.

ticket initial Ticket émis de manière directe, et pas à partir d'un ticket d'octroi de tickets. Certains services, tels que les applications modifiant les mots de passe, peuvent nécessiter des tickets marqués comme étant `initiaux` afin d'assurer que le client peut démontrer qu'il connait sa clé secrète. Cette garantie est importante car un ticket initial indique que le client s'est récemment authentifié et ne dépend pas d'un ticket d'octroi de tickets qui peut exister depuis un certain temps.

ticket non valide Ticket postdaté n'étant pas encore utilisable. Un ticket non valide est rejeté par un serveur d'application jusqu'à ce qu'il soit validé. Pour être validé, un ticket non valide doit être présenté au KDC par le client dans une demande de TGS, avec l'indicateur `VALIDATE`, après l'heure de début. Voir également Ticket postdaté.

Ticket postdaté Un ticket postdaté ne devient valide qu'un certain temps après sa création. Par exemple, un tel ticket peut être utile avec les tâches exécutées par lots la nuit car le ticket, s'il est volé, ne peut pas être utilisé tant que l'exécution de ces tâches n'a pas eu lieu. Lorsqu'un ticket postdaté est émis, il est émis en tant que `non valide` et le reste jusqu'à ce que a) son heure de début soit dépassée et b) le client demande la validation par le KDC. Un ticket postdaté est normalement valide jusqu'à l'heure d'expiration du ticket d'octroi de tickets. Toutefois, si le ticket postdaté est marqué comme `renouvelable`, sa durée de vie est normalement égale à la durée de vie entière du ticket d'octroi de tickets. Voir également ticket non valide, Ticket renouvelable.

Ticket renouvelable Etant donné que tout ticket dont la durée de vie est très longue peut présenter un risque pour la sécurité, les tickets peuvent être conçus pour être `renouvelables`. Un ticket renouvelable possède deux moments d'expiration : a) l'heure à laquelle l'instance courante du ticket expire et b) la durée de vie maximale de tout ticket. Si un client souhaite continuer à utiliser un ticket, il peut le renouveler avant sa première expiration. Par exemple, un ticket peut être valide pendant une heure, et tous les tickets ont une durée de vie maximale de dix heures. Si le client détenant le ticket souhaite le conserver plus d'une heure, il doit le renouveler. Lorsqu'un ticket atteint sa durée de vie maximale, celui-ci expire automatiquement et ne peut pas être renouvelé.

Ticket transmissible Ticket pouvant être utilisé par un client pour demander un ticket sur un hôte distant sans devoir se soumettre au processus complet d'authentification sur l'hôte concerné. Par exemple, si l'utilisateur `david` obtient un ticket transmissible sur la machine de l'utilisatrice `jennifer`, il peut se connecter à sa propre machine sans devoir demander un nouveau ticket (et donc s'authentifier à nouveau). Voir également Ticket utilisable avec proxy.

Ticket utilisable avec proxy Ticket pouvant être utilisé par un service pour le compte d'un client afin d'effectuer une opération pour ce dernier. On dit alors que le service agit en tant que proxy du client. Grâce à ce ticket, le service peut prendre l'identité du client. Le service peut utiliser un tel ticket afin d'obtenir un ticket de service d'un autre service, mais non un ticket d'octroi de tickets. La différence entre un ticket utilisable avec proxy et un ticket transmissible réside dans le fait que le premier n'est valide que pour une seule opération. Voir également Ticket transmissible.

Touche

1. En règle générale, l'un des deux principaux types de clés :

   - *clé symétrique* : clé de chiffrement identique à la clé de déchiffrement. Les clés symétriques sont utilisées pour chiffrer des fichiers.
   - *clé asymétrique* ou *clé publique* : clé utilisée dans les algorithmes à clé publique, tels que Diffie-Hellman ou RSA. Les clés publiques contiennent une clé privée, connue uniquement d'un utilisateur, une clé publique utilisée par le serveur ou des ressources générales, et une paire de clés publique-privée combinant les deux. Une clé privée est également qualifiée de clé *secrète*. La clé publique est également qualifiée de clé *partagée* ou *commune*.

2. Entrée (nom de principal) dans un fichier keytab. Voir également fichier keytab.

3. Dans Kerberos, clé de chiffrement dont il existe trois types :

   - *Clé privée* : clé de chiffrement partagée par un principal et le KDC, et distribuée en dehors des limites du système. Voir également Clé privée.
   - *Clé de service* : clé remplissant la même fonction que la clé privée, mais utilisée par des serveurs et des services. Voir également clé de service.
   - *Clé de session* : clé de chiffrement temporaire utilisée entre deux principaux, avec une durée de vie limitée à la durée d'une seule session de connexion. Voir également Clé de session.

Utilisateur privilégié Droits un utilisateur disposant des droits d'utilisateur standard au-delà de la sur un système informatique. Voir également Utilisateurs de confiance.

Utilisateurs de confiance Les utilisateurs que vous avez choisis peuvent effectuer des tâches administratives à un certain niveau de confiance. En général, les administrateurs créent des connexions pour les utilisateurs de confiance d'abord puis et attribuent des droits administratifs qui correspondent au niveau de confiance et de capacité de l'utilisateur. Ces utilisateurs aident à configurer et tenir à jour le système. Egalement appelés *utilisateurs privilégiés*.

variante Historiquement, la *variante de sécurité* et la *variante d'authentification* désignaient la même chose, lorsqu'une variante indiquait un type d'authentification (AUTH_UNIX, AUTH_DES, AUTH_KERB). RPCSEC_GSS est également une variante de sécurité, même s'il permet d'assurer des services d'intégrité et de confidentialité, en plus de l'authentification.

Variante de sécurité Voir variante.

VPN Réseau privé virtuel assurant une communication sécurisée en utilisant les mécanismes de chiffrement et de mise en tunnel pour connecter les utilisateurs via un réseau public.

Index

Nombres et symboles

A

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

X

Z