Version : **2020.01**

Dernière mise-à-jour : 2020/01/30 03:28

SO201 - Gestion des Utilisateurs et des Groupes

La bonne gestion des utilisateurs passe par une bonne stratégie de groupes. En effet, chaque utilisateur est affecté à un groupe **principal** mais il peut aussi être membre de jusqu'à quinze groupes secondaires.

Comme dans d'autres systèmes d'exploitation, sous Solaris il est préférable de donner les droits d'accès aux groupes et non aux utilisateurs individuels.

Groupes

Pour lister les groupes existants sur le système, saisissez la commande suivante :

```
# cat /etc/group
root::0:
other::1:root
bin::2:root,daemon
sys::3:root,bin,adm
adm::4:root,daemon
uucp::5:root
mail::6:root
tty::7:root,adm
lp::8:root,adm
nuucp::9:root
staff::10:
daemon::12:root
sysadmin::14:
smmsp::25:
```

```
gdm::50:
webservd::80:
postgres::90:
unknown::96:
nobody::60001:
noaccess::60002:
nogroup::65534:
```

Important - Notez que la valeur du GID de root est de **0**.

Chaque ligne est constituée de 4 champs :

- Le nom **unique** du groupe,
- Historiquement, le mot de passe du groupe. Cette pratique est obsolète sous Solaris,
- Le GID. Une valeur unique utilisée pour déterminer les droits d'accès aux fichiers et aux répertoires,
- La liste des membres ayant le groupe comme groupe **secondaire**.

Les conventions Solaris pour les numéros de groupes sont les suivantes :

GID	**Description**
0 à 99	Groupes système prédéfinis
100 à 60000	Groupes ordinaires
60001 et 65534	Groupe nobody et group nogroup
60002	Groupe noaccess

Afin de vérifier le fichier **/etc/group** pour des erreurs éventuelles, saisissez la commande suivante :

```
# grpck
#
```

Dans le cas où vos fichiers ne comportent pas d'erreurs, vous vous retrouverez retourné au prompt.

Utilisateurs

Pour lister les comptes utilisateur existants sur le système, saisissez la commande suivante :

```
# cat /etc/passwd
root:x:0:0:Super-User:/:/sbin/sh
daemon:x:1:1::/:
bin:x:2:2::/usr/bin:
sys:x:3:3::/:
adm:x:4:4:Admin:/var/adm:
lp:x:71:8:Line Printer Admin:/usr/spool/lp:
uucp:x:5:5:uucp Admin:/usr/lib/uucp:
nuucp:x:9:9:uucp Admin:/var/spool/uucppublic:/usr/lib/uucp/uucico
smmsp:x:25:25:SendMail Message Submission Program:/:
listen:x:37:4:Network Admin:/usr/net/nls:
gdm:x:50:50:GDM Reserved UID:/:
webservd:x:80:80:WebServer Reserved UID:/:
postgres:x:90:90:PostgreSQL Reserved UID:/:/usr/bin/pfksh
svctag:x:95:12:Service Tag UID:/:
unknown:x:96:96:Unknown Remote UID:/:
nobody:x:60001:60001:NFS Anonymous Access User:/:
noaccess:x:60002:60002:No Access User:/:
nobody4:x:65534:65534:SunOS 4.x NFS Anonymous Access User:/:
```

Important - Notez que la valeur de l'UID de root est de **0**.

Chaque ligne est constituée de 7 champs :

- Le nom d'utilisateur
- Le mot de passe. Une valeur de **x** dans ce champs indique que le système utilise le fichier **/etc/shadow** pour stocker les mots de passe.

- L'UID. Une valeur unique qui est utilisée pour déterminer les droits aux fichiers et aux répertoires.
- Le GID. Une valeur indiquant le groupe **principal** de l'utilisateur
- Le nom complet. Ce champs optionnel est aussi appelé **GECOS**
- Le répertoire personnel de l'utilisateur
- Le shell de l'utilisateur.

Les conventions Solaris pour les numéros de comptes sont les suivantes :

UID	**Description**
0 à 99	Comptes système prédéfinis
100 à 60000	Comptes ordinaires
60001 et 65534	Compte nobody et compte nogroup
60002	Compte noaccess

Pour consulter le fichier **/etc/shadow**, saisissez la commande suivante :

```
# cat /etc/shadow
root:U8RdXOcuBi502:6445::::::
daemon:NP:6445::::::
bin:NP:6445::::::
sys:NP:6445::::::
adm:NP:6445::::::
lp:NP:6445::::::
uucp:NP:6445::::::
nuucp:NP:6445::::::
smmsp:NP:6445::::::
listen:*LK*:::::::
gdm:*LK*:::::::
webservd:*LK*:::::::
postgres:NP:::::::
svctag:*LK*:6445::::::
unknown:*LK*:::::::
nobody:*LK*:6445::::::
noaccess:*LK*:6445::::::
```

```
nobody4:*LK*:6445::::::
```

Chaque ligne est constituée de 9 champs :

- Le nom de l'utilisateur. Ce champs est utilisé pour faire le lien avec le fichier **/etc/passwd**,
- Le mot de passe **crypté** de l'utilisateur. Le cryptage est à sens **unique**. Ce champ peut aussi contenir une des trois valeurs suivantes :
  - **NP** - Un mot de passe n'est pas possible,
  - ***LK*** - Le compte est verrouillé,
  - **vide** - aucun mot de passe sera demandé pour l'utilisateur concerné,
- Le nombre de jours entre le **01/01/1970** et le dernier changement du mot de passe,
- Le nombre de jours minimum exigé entre deux changements,
- Le nombre de jours maximum exigé entre deux changements,
- Le nombre de jours avant la date de modification forcée que l'utilisateur recevra un avertissement,
- Le nombre de jours d'inactivité qui provoquera la désactivation du compte,
- Le nombre de jours après lequel le mot de passe doit être changé,
- La date d'expiration du compte exprimée en nombre de jours depuis le **01/01/1970**,
- Le dernier champs est inutilisé.

Afin de vérifier le fichier **/etc/passwd** pour des erreurs éventuelles, saisissez la commande suivante :

```
# pwck
#
```

Dans le cas où il est nécessaire de régénérer le fichier **/etc/shadow**, il convient d'utiliser la commande suivante :

- **pwconv**
  - permet de régénérer le fichier **/etc/shadow** à partir du fichier **/etc/passwd** et éventuellement du fichier **/etc/shadow** existant

Commandes

Les commandes associées à la gestion des groupes sont :

- **groupadd**

    - utilisée pour créer un groupe
- **groupdel**
    - utilisée pour supprimer un groupe
- **groupmod**
    - utilisée pour modifier un groupe existant
- **newgrp**
    - utilisée pour modifier le groupe de l'utilisateur qui l'invoque

Les commandes associées à la gestion des utilisateurs sont :

- **useradd**
    - utilisée pour ajouter un utilisateur
- **userdel**
    - utilisée pour supprimer un utilisateur
- **usermod**
    - utilisée pour modifier un utilisateur
- **passwd**
    - utilsée pour créer ou modifier le mot de passe d'un utilisateur

La commande **useradd** possède des valeurs pas défaut. Pour visualiser ces valeurs, saisissez la commande suivante :

```
# useradd -D
group=other,1  project=default,3  basedir=/home
skel=/etc/skel  shell=/bin/sh  inactive=0
expire=  auths=  profiles=  roles=  limitpriv=
defaultpriv=  lock_after_retries=
```

Pour modifier une de ces valeurs, il convient d'utiliser de nouveau l'option -D combinée avec l'option à modifier suivie par la nouvelle valeur, par exemple :

```
# useradd -D -b /export/home
group=other,1  project=default,3  basedir=/export/home
skel=/etc/skel  shell=/bin/sh  inactive=0
expire=  auths=  profiles=  roles=  limitpriv=
```

```
defaultpriv=  lock_after_retries=
```

Les options à utiliser sont détaillées dans le manuel de la commande.

Vous noterez que la commande useradd utilise le contenu du répertoire **/etc/skel/** comme **fichiers prototypes**.

Pour consulter la liste des fichiers dans **/etc/skel**, saisissez la commande suivante :

```
# ls -la /etc/skel
total 20
drwxr-xr-x   2 root     sys          512 Nov 29 13:15 .
drwxr-xr-x  86 root     sys         4608 Jan 14 11:19 ..
-rw-r--r--   1 root     other        144 Jan 11  2013 .profile
-rw-r--r--   1 root     sys          136 Jan 11  2013 local.cshrc
-rw-r--r--   1 root     sys          157 Jan 11  2013 local.login
-rw-r--r--   1 root     sys          174 Jan 11  2013 local.profile
```

Pour connaître l'UID, le GID et l'appartenance aux groupes d'un utilisateur, il convient d'utiliser la commande **id**. Saisissez la commande suivante :

```
# id root
uid=0(root) gid=0(root)
```

La commande id peut être utilisée avec l'option **-a**. Dans ce cas, les groupes secondaires éventuels de l'utilisateur sont également détaillés :

```
# id -a root
uid=0(root) gid=0(root) groups=1(other),2(bin),3(sys),4(adm),5(uucp),6(mail),7(tty),8(lp),9(nuucp),12(daemon)
```

Pour seulement connaître les groupes d'un utilisateur, il convient d'utiliser la commande **groups**. Saisissez la commande suivante :

```
# groups root
root other bin sys adm uucp mail tty lp nuucp daemon
```

L'activité de connexion d'un compte est géré par la configuration du fichier **/etc/default/login** :

```
# cat /etc/default/login
#ident  "@(#)login.dfl  1.14    04/06/25 SMI"
#
# Copyright 2004 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.

# Set the TZ environment variable of the shell.
#
#TIMEZONE=EST5EDT

# ULIMIT sets the file size limit for the login.  Units are disk blocks.
# The default of zero means no limit.
#
#ULIMIT=0

# If CONSOLE is set, root can only login on that device.
# Comment this line out to allow remote login by root.
#
CONSOLE=/dev/console

# PASSREQ determines if login requires a password.
#
PASSREQ=YES

# ALTSHELL determines if the SHELL environment variable should be set
#
ALTSHELL=YES

# PATH sets the initial shell PATH variable
#
#PATH=/usr/bin:

# SUPATH sets the initial shell PATH variable for root
#
```

```
#SUPATH=/usr/sbin:/usr/bin

# TIMEOUT sets the number of seconds (between 0 and 900) to wait before
# abandoning a login session.
#
#TIMEOUT=300

# UMASK sets the initial shell file creation mode mask.  See umask(1).
#
#UMASK=022

# SYSLOG determines whether the syslog(3) LOG_AUTH facility should be used
# to log all root logins at level LOG_NOTICE and multiple failed login
# attempts at LOG_CRIT.
#
SYSLOG=YES

# SLEEPTIME controls the number of seconds that the command should
# wait before printing the "login incorrect" message when a
# bad password is provided.  The range is limited from
# 0 to 5 seconds.
#
#SLEEPTIME=4

# DISABLETIME  If present, and greater than zero, the number of seconds
# login will wait after RETRIES failed attempts or the PAM framework returns
# PAM_ABORT. Default is 20. Minimum is 0. No maximum is imposed.
#
#DISABLETIME=20

# RETRIES determines the number of failed logins that will be
# allowed before login exits. Default is 5 and maximum is 15.
# If account locking is configured (user_attr(4)/policy.conf(4))
# for a local user's account (passwd(4)/shadow(4)), that account
```

```
# will be locked if failed logins equals or exceeds RETRIES.
#
#RETRIES=5
#
# The SYSLOG_FAILED_LOGINS variable is used to determine how many failed
# login attempts will be allowed by the system before a failed login
# message is logged, using the syslog(3) LOG_NOTICE facility.  For example,
# if the variable is set to 0, login will log -all- failed login attempts.
#
#SYSLOG_FAILED_LOGINS=5
```

Dans ce fichier, nous constatons plusieurs directives importantes :

Directive	**Description**
CONSOLE=/dev/console	**AUCUNE** connexion root via le réseau
PASSREQ=YES	Les logins nécessitent un mot de passe
TIMEOUT=300	Une connexion est abandonnée au bout de 5 minutes d'inactivité
SYSLOG=YES	Les connexions de root et les connexions ratées sont consignées par syslogd
SLEEPTIME=4	Implique un délai de 4 secondes entre tentatives de connexion
SYSLOG_FAILED_LOGINS=5	Les connexions ratées sont consignées après la sixième tentative

Important - A l'aide d'internet et le manuel en ligne, trouvez la signification de la directive ALTSHELL=YES. Ensuite éditez votre fichier pour activer la directive SLEEPTIME et TIMEOUT puis modifiez la directive SYSLOG_FAILED_LOGINS=5 à SYSLOG_FAILED_LOGINS=1.

Le fichier qui recense les valeurs par défaut de la gestion des mots de passe est **/etc/default/passwd** :

```
# cat /etc/default/passwd
#ident  "@(#)passwd.dfl 1.7     04/04/22 SMI"
#
```

```
# Copyright 2004 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
MAXWEEKS=
MINWEEKS=
PASSLENGTH=6

# NAMECHECK enables/disables login name checking.
# The default is to do login name checking.
# Specifying a value of "NO" will disable login name checking.
#
#NAMECHECK=NO

# HISTORY sets the number of prior password changes to keep and
# check for a user when changing passwords.  Setting the HISTORY
# value to zero (0), or removing/commenting out the flag will
# cause all users' prior password history to be discarded at the
# next password change by any user.  No password history will
# be checked if the flag is not present or has zero value.
# The maximum value of HISTORY is 26.
#
# This flag is only enforced for user accounts defined in the
# local passwd(4)/shadow(4) files.
#
#HISTORY=0
#
# Password complexity tunables.  The values listed are the defaults
# which are compatible with previous releases of passwd.
# See passwd(1) and pam_authtok_check(5) for use warnings and
# discussion of the use of these options.
#
#MINDIFF=3
#MINALPHA=2
#MINNONALPHA=1
```

```
#MINUPPER=0
#MINLOWER=0
#MAXREPEATS=0
#MINSPECIAL=0
#MINDIGIT=0
#WHITESPACE=YES
#
#
# passwd performs dictionary lookups if DICTIONLIST or DICTIONDBDIR
# is defined. If the password database does not yet exist, it is
# created by passwd. See passwd(1), pam_authtok_check(5) and
# mkdict(1) for more information.
#
#DICTIONLIST=
#DICTIONDBDIR=/var/passwd
```

Dans ce fichier, nous constatons plusieurs directives importantes :

Directive	**Description**
MAXWEEKS	Durée maximale de la validité d'un mot de passe
MINWEEKS	Durée minimale de la validité d'un mot de passe
PASSLENGTH=6	Nombre minimum de caractères dans un mot de passe. Le nombre *maximum* est 8

Important - Consultez le manuel de passwd et trouvez la signification de la directive HISTORY=0. Passez en revue les options de la commande passwd. Notez comment supprimer un mot de passe, comment laisser l'utilisateur choisir son mot de passe lors de sa première connexion, comment verrouiller un compte et comment déverrouiller un compte.

LAB #1 - Gestion des groupes et des utilisateurs

Créez maintenant trois groupes **groupe1**, **groupe2** et **groupe3**. La valeur du GID du groupe **groupe3** doit être de **807** :

```
# groupadd groupe1; groupadd groupe2; groupadd -g 807 groupe3
```

Créez maintenant trois utilisateurs **user1**, **user2** et **user3**. Les trois utilisateurs ont pour groupe principal **groupe1**, **groupe2** et **groupe3** respectivement :

```
# useradd -m -g groupe2 user2; useradd -m -g 807 user3; useradd -m -g groupe1 user1
```

user2 est aussi membre des groupes **group1** et **group3**. **user1** à un GECOS de **tux1** :

```
# usermod -G groupe1,groupe3 user2
# usermod -c "tux1" user1
```

En consultant la fin de votre fichier **/etc/passwd**, vous obtiendrez un résultat similaire à celui-ci:

```
# tail /etc/passwd
webservd:x:80:80:WebServer Reserved UID:/:
postgres:x:90:90:PostgreSQL Reserved UID:/:/usr/bin/pfksh
svctag:x:95:12:Service Tag UID:/:
unknown:x:96:96:Unknown Remote UID:/:
nobody:x:60001:60001:NFS Anonymous Access User:/:
noaccess:x:60002:60002:No Access User:/:
nobody4:x:65534:65534:SunOS 4.x NFS Anonymous Access User:/:
user2:x:100:101::/export/home/user2:/bin/sh
user3:x:101:807::/export/home/user3:/bin/sh
user1:x:102:100:tux1:/export/home/user1:/bin/sh
```

En regardant la fin de votre fichier **/etc/group**, vous obtiendrez un résultat similaire à celui-ci:

```
# tail /etc/group
gdm::50:
webservd::80:
postgres::90:
unknown::96:
nobody::60001:
noaccess::60002:
nogroup::65534:
groupe1::100:user2
groupe2::101:
groupe3::807:user2
```

Modifiez maintenant le nom du groupe3 en groupe99 :

```
# groupmod -n groupe99 groupe3
```

Consultez la fin du fichier /etc/group :

```
# tail /etc/group
gdm::50:
webservd::80:
postgres::90:
unknown::96:
nobody::60001:
noaccess::60002:
nogroup::65534:
groupe1::100:user2
groupe2::101:
groupe99::807:user2
```

Supprimez maintenant le groupe99 :

```
# groupdel groupe99
```

Consultez la fin du fichier /etc/group :

```
# tail /etc/group
gdm::50:
webservd::80:
postgres::90:
unknown::96:
nobody::60001:
noaccess::60002:
nogroup::65534:
groupe1::100:user2
groupe2::101:
```

Consultez maintenant le fichier /etc/passwd :

```
# tail /etc/passwd
webservd:x:80:80:WebServer Reserved UID:/:
postgres:x:90:90:PostgreSQL Reserved UID:/:/usr/bin/pfksh
svctag:x:95:12:Service Tag UID:/:
unknown:x:96:96:Unknown Remote UID:/:
nobody:x:60001:60001:NFS Anonymous Access User:/:
noaccess:x:60002:60002:No Access User:/:
nobody4:x:65534:65534:SunOS 4.x NFS Anonymous Access User:/:
user2:x:100:101::/export/home/user2:/bin/sh
user3:x:101:807::/export/home/user3:/bin/sh
user1:x:102:100:tux1:/export/home/user1:/bin/sh
```

Important - Notez que l'utilisateur a un groupe principal qui n'existe plus !

Supprimez maintenant l'utilisateur user3 :

```
# userdel user3
```

Le fait de supprimer un utilisateur **sans** l'option **-r** implique que le répertoire personnel de l'utilisateur demeure sur la machine. Saisissez la commande suivante pour vérifier :

```
# ls -l /export/home
total 22
drwx------   2 root     root        8192 Nov 29 13:15 lost+found
drwxr-xr-x   2 user1    groupe1      512 Jan 14 12:48 user1
drwxr-xr-x   2 user2    groupe2      512 Jan 14 12:48 user2
drwxr-xr-x   2 101      807          512 Jan 14 12:48 user3
```

Pour supprimer les fichiers de cet utilisateur, il convient de saisir la commande suivante :

```
# find /export/home -user 101 -exec rm -rf {} \;
# ls -l /export/home
total 20
drwx------   2 root     root        8192 Nov 29 13:15 lost+found
drwxr-xr-x   2 user1    groupe1      512 Jan 14 12:48 user1
drwxr-xr-x   2 user2    groupe2      512 Jan 14 12:48 user2
```

Créez maintenant le mot de passe pour **user1**. Indiquez un mot de passe identique au nom du compte :

```
# passwd user1
New Password: user1
passwd: Password too short - must be at least 6 characters.

Please try again
New Password: ^C
<code>

Saisissez un mot de passe de plus de 6 caractères :
```

```
<code>
# passwd user1
New Password: trainee
passwd: The password must contain at least 1 numeric or special character(s).

Please try again
New Password: ^C
```

Saisissez un mot de passe de plus de 6 caractères et contenant un caractère spécial :

```
# passwd user1
New Password: tra@inee
Re-enter new Password: tra@inee
passwd: password successfully changed for user1
```

Créez maintenant le mot de passe pour **user2**. Indiquez un mot de passe identique au mot de passe d'user1 :

```
# passwd user2
New Password: tra@inee
Re-enter new Password: tra@inee
passwd: password successfully changed for user2
```
