



Gestione dei gruppi UNIX in ENEA-GRID

NOTA TECNICA	
ENEAGRID/CRESCO:	NT201102
NOME FILE:	NT201102-GruppiUNIXeSSHD-v1_2.odt
DATA:	14/03/11
STATO:	Versione rivista N. 3
AUTORE/I:	G. Ponti

Abstract:

Questa nota tecnica ha come obiettivo quello di descrivere il modo in cui vengono gestiti i gruppi UNIX in ENEA-GRID e come sono utilizzati per la condivisione di file e directory tra utenti.

Log Documento

VERSIONE	DATA	COMMENTO	AUTORE/I
1.0	07/02/11	Versione iniziale	G. Ponti
1.1	15/02/11	Restrizione accessi al login di SSH	G. Ponti
1.2	14/03/11	Procedura per richiesta e utilizzo di directory condivise su GPFS	G. Ponti

INDICE

Introduzione.....	5
Sezione 1 – Update dei gruppi.....	6
Sezione 1.1 – Assunzioni preliminari.....	6
Sezione 1.2 – Script di update.....	6
Algoritmo group_upg.....	6
Sezione 2 - Condivisione directory tra utenti.....	7
Sezione 2.1 – Lato admin.....	7
Sezione 2.2 - Lato user.....	8
Sezione 5 – Controllo degli accessi usando i gruppi UNIX.....	9
Sezione 4 – Modifiche per gli strumenti esistenti.....	10
Sezione 4.1 - Note delle modifiche da apportare a WARC.....	10
Sezione 4.2 - Note delle modifiche da apportare a FARO.....	11
Appendice A – Richiedere e utilizzare directory condivise.....	12
Appendice B – Script group_upg.sh.....	13
Appendice C – Script retrieveGroups.sh.....	17
Appendice D – Script retrieveBusyGIDs.sh.....	19
Bibliografia.....	21

Introduzione

Allo stato attuale, in ENEA-GRID non è presente un meccanismo per la gestione dei gruppi UNIX. In questa nota tecnica verranno descritte le scelte effettuate e le modifiche da apportare al sistema per permettere la gestione dei gruppi. In particolare, verranno analizzati due problemi principali:

- Come propagare l'update dei gruppi UNIX creati in AFS sulle diverse macchine in ENEA-GRID, garantendo consistenza anche in caso di piattaforme differenti.
- Com'è possibile utilizzare i gruppi UNIX per condividere directory tra utenti che appartengono ad uno stesso gruppo
- Come utilizzare i gruppi UNIX per limitare l'accesso via SSH

Sezione 1 – Update dei gruppi

Sezione 1.1 - Assunzioni preliminari

Prima di procedere con la descrizione delle procedure per la gestione dei gruppi utente, è opportuno illustrare alcune scelte di base e conseguenti interventi da effettuare su AFS. Per prima cosa, si farà riferimento alla directory **newgrp** all'interno di **/afs/enea.it/common/etc/**. Questa directory, in analogia con quanto già avviene per gli utenti, conterrà una serie di files del tipo **group_GRPNAME**, dove 'GRPNAME' sarà il nome del gruppo che è stato creato. Ognuno di questi file sarà composto da una sola riga, che servirà per identificare il gruppo secondo lo standard dei sistemi UNIX, ad eccezione del campo relativo alla password di gruppo che, per mantenere la compatibilità con le scelte fatte già in altri contesti, conterrà una 'X' (maiuscola)¹. Questi files verranno editati qualora si chiedesse di creare o di modificare un gruppo.

Sezione 1.2 - Script di update

Lo script di upgrade dei gruppi ha il compito di allineare le informazioni sui gruppi creati e/o modificati in ENEA-GRID con i gruppi presenti localmente su ogni singola macchina. Il suo nome è **group_upg.sh** e si trova nel path **/afs/enea.it/common/**. L'algoritmo utilizzato nello script implementa il seguente pseudo-codice (lo script è invece riportato in Appendice B):

Algoritmo group_upg

1. A : insieme di group-lines in '/etc/group'
2. B : insieme di group-lines in '/afs/enea.it/common/etc/newgrp/group_*
3. \forall grpB \in B DO
 - o IF \exists grpA \in A | groupName(grpA) == groupName(grpB)
THEN
 A = A \ {grpA} \cup {grpB} // sostituisce 'grpA' con 'grpB' in A
ELSE
 A = A \cup {grpB} // aggiunge 'grpB' in A

¹ Lo script non viene eseguito su sistemi che usano il carattere 'X' per indicare la presenza di password di gruppo cifrate.

L'algoritmo è compatibile con la gestione dei gruppi fatta dai sistemi Linux, AIX, IRIX e Solaris [1, 2, 3].

È interessante notare che, a differenza di quanto accade per gli utenti, ogni linea che caratterizza un gruppo può variare abbastanza frequentemente poiché questa presenta, oltre ad altre informazioni specifiche del gruppo quali, e.g., il nome, la password e il GID, anche la lista degli utenti che in quel momento sono membri del gruppo. La chiamata a tale script viene effettuata all'interno dello script **passwd_upg** presente nella directory **/afs/enea.it/common/etc/**, il quale ha il compito di aggiornare le informazioni degli utenti.

Sezione 2 - Condivisione directory tra utenti

Sezione 2.1 - Lato admin

I gruppi UNIX possono anche essere utilizzati per limitare l'accesso a directory presenti nel file system soltanto ad alcuni utenti specifici. Tale aspetto può essere utile in molte situazioni pratiche, quali, e.g., la creazione di cartelle condivise.

Nel seguito verranno descritti e commentati i passi da fare per creare delle directory da condividere tra utenti appartenenti ad uno stesso gruppo. A tale scopo, supponiamo di voler creare un gruppo UNIX **project1** che contenga utenti che partecipano ad un particolare progetto. Per far condividere una directory del file system a tali utenti è opportuno creare una nuova directory (e.g., **/home/prog1**) e renderla di proprietà del gruppo appena creato:

```
# groupadd project1
# mkdir /home/prog1
# chown -R root.project1 /home/prog1
```

l'ultimo dei tre comandi precedenti imposta la directory **/home/prog1** come di proprietà dell'utente **root** e del gruppo **project1** appena creato. A questo punto, bisogna assegnare i permessi alla directory nel seguente modo:

```
# chmod 2770 /home/prog1
```

oppure, usando la modalità simbolica (laddove supportato):

```
# chmod u=rwx,g=rws /home/prog1
```

Il precedente comando assegna alla directory i permessi **770**, vale a dire tutti i permessi al proprietario e al gruppo e nessuno gli altri utenti, e in più assegna al gruppo il bit di **SETGID**. Questo particolare permesso, se applicato ad una directory, fa in modo che tutti i

nuovi file e le sottodirectory in essa creati abbiano come gruppo proprietario quello della directory che li contiene (anziché il gruppo principale dell'utente che li crea). Il bit di SETGID è visualizzato mediante il simbolo **s** oppure **S** al posto del simbolo relativo al permesso di esecuzione del file (se è presente **S** maiuscolo vuol dire che è attivo anche il permesso di esecuzione).

A questo punto, basta solo aggiungere gli utenti specifici al gruppo utilizzando il seguente comando:

```
# gpasswd -a USR_NAME GRP_NAME
```

dove **USR_NAME** e **GRP_NAME** sono, rispettivamente, il nome dell'utente e il nome del gruppo al quale si vuole aggiungere l'utente. Nel nostro esempio, se vogliamo aggiungere un utente generico **user1** al gruppo **project1**, il comando sarà

```
# gpasswd -a user1 project1
```

È opportuno fare una piccola osservazione ai fini della condivisione collaborativa di file. I file e/o le directory creati dentro una directory con il bit SETGID settato posso essere modificati solo dal proprietario se, come nella maggior parte dei casi, la **umask** è impostata sul valore di default **0022**. Ovviamente, il proprietario potrà cambiare questa impostazione non appena riterrà opportuno oppure settarla nel proprio profilo (vedere Sezione 2.2).

Sezione 2.2 - Lato user

Dal lato utente, una volta che questi è stato assegnato ad un gruppo di lavoro, può creare file e directory all'interno dell'area condivisa. Di default, però, solo lui potrà modificare i file creati; questo perché la sua **umask** è impostata di default a **0022** (nella maggior parte dei casi). La **umask** viene generalmente impostata al login dell'utente in maniera globale per tutti gli utenti o localmente per ogni singolo utente inserendo un'apposita riga nel file **/etc/profile** (globale) oppure **~/.profile** (locale), rispettivamente. Umask serve per dire ai comandi e ai programmi avviati dopo il suo settaggio quali sono i permessi da negare in fase di creazione di nuovi file e/o directory. Nell'esempio, impostando una maschera con valore **0022**, vengono negati agli utenti del gruppo e a tutti gli altri i permessi di scrittura (identificati dal valore '2').

Se un utente vuole verificare il valore della sua 'umask' nella shell attuale, può digitare il comando:

```
# umask
```

il quale restituisce il valore in notazione ottale. Se si vuole lo stesso risultato ma in notazione simbolica, si può aggiungere l'opzione **-S** (i.e., **umask -S**); tuttavia, quest'ultima modalità non è presente in tutte le shell (come accade, e.g., per csh).

Per cambiare la propria 'umask', l'utente può digitare il comando seguito da una maschera che rappresenta i diritti da negare. Nel nostro caso, se un utente vuole creare dei file e permettere anche ad altri utenti del gruppo di modificarli, allora dovrà scrivere, prima della creazione del file, il seguente comando:

```
# umask 0007
```

il quale nega agli altri utenti qualsiasi permesso sul file, mentre assegna tutti i diritti agli utenti del gruppo.

In Appendice A è presente uno schema che riassume e sintetizza le operazioni da effettuare per richiedere e per utilizzare directory condivise in ENEA-GRID.

Sezione 5 – Controllo degli accessi usando i gruppi UNIX

I gruppi UNIX possono anche essere utilizzati per permettere o negare l'accesso ad ENEA-GRID via SSH ad alcuni gruppi di utenti. Di default, SSH permette l'accesso a tutti gli utenti e a tutti i gruppi. Tuttavia, è possibile definire delle restrizioni da applicare ai singoli utenti e/o a gruppi di utenti. Per fare questo, bisogna intervenire sul file di configurazione di SSH `/etc/sshd_config` e utilizzare le variabili **AllowGroups** e **DenyGroups** seguite dall'elenco dei gruppi ai quali si vuole limitare l'accesso. Ad esempio, supponiamo di inserire nel file di configurazione di SSH le seguenti righe:

```
AllowGroups proj*
```

```
DenyGroups proj2,proj3
```

In base alla prima riga, si vuole permettere l'accesso a tutti gli utenti appartenenti ai gruppi i cui nomi iniziano con la stringa “**proj**” seguita da una sequenza arbitraria di caratteri. La seconda riga, però, limita ulteriormente l'accesso escludendo dalla lista precedente gli utenti che appartengono ai gruppi **proj2** e **proj3**. Da notare che le politiche definite da questi due comandi vengono applicate anche se il gruppo in oggetto non è quello primario dell'utente che sta effettuando l'accesso.

È importante capire come vengono risolti i conflitti, specie quando nel file `/etc/sshd_config` compaiono anche le direttive **AllowUsers** e **DenyUsers**. Nello specifico, se nel file troviamo **AllowGroups** o **DenyGroups** che negano l'accesso ad alcuni gruppi di utenti, l'accesso rimane negato anche se nel file è presente un'altra direttiva che ne permette l'accesso.

In Appendice C è illustrato uno script di utilità che effettua una ricognizione su tutti i nodi in ENEA-GRID e restituisce la lista dei gruppi presenti e, per ogni gruppo, la lista dei **GID** (i.e., *Group ID*) che sono attualmente utilizzati per quel gruppo. Inoltre, in Appendice D

è presente uno script che produce una lista ordinata dei GID già utilizzati in ENEA-GRID, evidenziando anche i possibili range di GID ancora utilizzabili.

Sezione 4 – Modifiche per gli strumenti esistenti

In questa sezione verranno elencate le modifiche da effettuare su alcuni strumenti esistenti che supportano amministratori e utenti in ENEA-GRID.

Sezione 4.1 - Note delle modifiche da apportare a WARC

Di seguito verranno riportate un elenco delle funzionalità da aggiungere/modificare in WARC.

- Abilitare la gestione dei gruppi UNIX (creazione, modifica, aggiunta di utenti, etc.). Gli interventi grafici da effettuare sono i seguenti:
 - Area “*Admin tools*” - Aggiungere una sezione “**UNIX Groups**” e inserire le seguenti funzioni: *creazione, verifica, rimozione e aggiornamento* di un gruppo UNIX. In particolare:
 - *Create* – permette di creare un nuovo gruppo UNIX. Riceve il nome del nuovo gruppo e il GID.
 - *Verify* – fornisce informazioni sui gruppi UNIX già presenti. Riceve il nome del gruppo (tra quelli esistenti) e restituisce un resoconto con i dettagli del gruppo selezionato.
 - *Delete* – permette di rimuovere un gruppo UNIX. Riceve il nome del gruppo da rimuovere.
 - *Update* – permette di aggiornare un gruppo UNIX esistente. Riceve il nome del gruppo da aggiornare e permette di cambiare il nome e il GID del gruppo.

ENEA

WELCOME TO ARC!

Log		Whoami
<div style="text-align: center; font-weight: bold; font-size: 14px; margin-bottom: 10px;">Admin tools</div> <ul style="list-style-type: none"> • Add Organization Unit • List/Remove Organization Unit • Create Group • Verify Group • Delete Group • Update Group • Add Group Admin • List Group Admins • Remove Group Admin • Create Host in K5 DB • Delete Host in K5 DB • Info Host • Check Fileservers <div style="border: 1px dashed red; padding: 5px; margin-top: 10px; text-align: center;"> <p style="font-weight: bold; color: red; font-size: 14px;">UNIX Groups</p> <ul style="list-style-type: none"> - Create - Verify - Delete - Update </div>	<div style="text-align: center; font-weight: bold; font-size: 14px; margin-bottom: 10px;">User tools</div> <ul style="list-style-type: none"> • Create • Verify • Delete • Change group • Delete (Force) • Change group • Add Remote User Volume <div style="border: 1px dashed red; padding: 5px; margin-top: 10px; text-align: center;"> <p style="color: red; font-size: 12px;">- Add user to UNIX group</p> </div> <p style="font-size: 10px; margin-top: 10px;">Info:</p> <ul style="list-style-type: none"> • See info • Update info • Search • List all users • List users in a group • List users in O.U. <p style="font-size: 10px; margin-top: 10px;">Account/Password:</p> <ul style="list-style-type: none"> • Change password • Check Account Expiration Date • Set Account Expiration Date • Password Policy • Expire Account • Unlock Account <p style="font-size: 10px; margin-top: 10px;">Disk quota:</p> <ul style="list-style-type: none"> • Show Quota • Set Quota • Set Remote Quota 	<div style="text-align: center; font-weight: bold; font-size: 14px; margin-bottom: 10px;">Project tools</div> <ul style="list-style-type: none"> • Declare Space • UnDeclare Space • Update Space • See Info • List All Projects • Create Project/Subproject • Delete Project/Subproject • Create Remote RO Volume • Delete Remote RO Volume • Change Ownership of html Folder • Create PTS Group • Delete PTS Group • List PTS Groups • Add a User to a PTS Group • Remove a User from a PTS Group • List Users in a PTS Group • Release Project Volume • Disable R/O Replica • Enable R/O Replica • Manage administrator: <ul style="list-style-type: none"> • Add project administrator • Remove project administrator • List project administrators • Project quota: <ul style="list-style-type: none"> • Show project quota • Set project quota

fig. 1: Interventi da eseguire su WARC per la gestione dei gruppi UNIX (in rosso)

- Tutte le operazioni fatte da WARC sui gruppi devono essere riportate nel file `/afs/enea.it/common/etc/newgrp/group_GRPNAME`, dove **GRPNAME** è il nome del gruppo oggetto di tali operazioni

Sezione 4.2 - Note delle modifiche da apportare a FARO

Di seguito verranno riportate un elenco delle funzionalità da aggiungere/modificare in FARO.

- Quando un utente fa un'operazione e/o lancia un programma, aggiungere un'opzione che setta la sua 'umask'. In particolare, basterà aggiungere un box dove l'utente inserisce con il mouse un segno di spunta per scegliere una delle due modalità con cui intende operare. All'atto pratico, FARO deve controllare lo stato di questo box e settare propriamente la 'umask' per quella sessione utente

Appendice A – Richiedere e utilizzare directory condivise

Di seguito sono riportati i passi per la richiesta di creazione di directory condivise su GPFS e le azioni da intraprendere per l'utilizzo delle stesse.

- **Richiesta di creazione di un gruppo e di una relativa directory condivisa**
 1. *Aprire un ticket* utilizzando il sito all'indirizzo <https://gridticket.enea.it/>
 2. Specificare il *nome del gruppo* da creare e la *lista degli utenti* che fanno parte del gruppo (nome, cognome e username in ENEA-GRID)
 3. Indicare il *nome della directory* radice del progetto.
- **Utilizzare la directory condivisa**
 1. Effettuare il login in ENEA-GRID
 2. Posizionarsi dentro la directory condivisa, la cui radice sarà `/gpor_proj/PROG_ROOT`, dove 'PROG_ROOT' è il nome della directory specificata in fase di creazione
 3. Scegliere la modalità di utilizzo dell'area progettuale condivisa secondo una delle due specificate di seguito:
 - *Gli altri utenti del gruppo possono solo leggere ed eseguire i file creati dall'utente corrente nella directory condivisa.* Prima di eseguire qualsiasi operazione, digitare nel proprio terminale il comando **'umask 0027'** (per maggiori dettagli consultare la Sezione 2.2)
 - *Gli utenti del gruppo possono anche modificare i file creati dall'utente corrente nella directory condivisa.* Prima di eseguire qualsiasi operazione, digitare nel proprio terminale il comando **'umask 0007'** (per maggiori dettagli consultare la Sezione 2.2).

Si noti che il cambio della umask riguarda la sessione in corso e interessa solo i file creati e/o modificati in quella sessione. Al successivo login, quindi, tutti i valori saranno ripristinati al profilo di default dell'utente.
 4. Procedere con il proprio lavoro.

Appendice B – Script *group_upg.sh*

```
#!/bin/sh

# script per la gestione dei gruppi
# aggiornamento di /etc/group
# G. Ponti 04/02/2011

echo "--- Group update script starting ---"
printf "Looking for temp dir..."
tmpdir=NOOK
if [ -d /tmp ]; then
    freek=`df -Pk /tmp | tail -1 | awk '{print $4}'`
    if [ "$freek" -gt 10000 ]; then
        tmpdir=OK
    fi
fi
if [ "$tmpdir" != "OK" ]; then
    exit
fi
printf " DONE\n"
printf "Looking for AFS dir..."
AFS=NOOK
if [ -d /afs ]; then
    afs=`df -Pk /afs | grep AFS`
    if [ "$afs" != "" ]; then
        AFS=OK
    fi
fi
if [ "$AFS" != "OK" ]; then
    exit
fi
```

```
printf " DONE\n"
#####
#file dei gruppi sul nodo
grpLocal=/etc/group
export grpLocal

#file dei gruppi su AFS
grpAFS=/tmp/group_afs.tmp.$$
export grpAFS

#file dei gruppi AFS in locale
grpAFSlocal=/tmp/group_afs_local.tmp.$$
export diffFile

#file finale
output=/tmp/group_upg.tmp.$$
export output

# directory per i salvataggi periodici
if [ ! -x /tmp/etc ] ; then
    mkdir /tmp/etc 1> /dev/null 2>&1
fi
#####
# faccio il backup del file /etc/group
# se e' la prima volta che si esegue lo script
if [ ! -f /etc/group.bak-group_upg ]; then
    capitalX=`grep ":X:" /etc/group`
    if [ "$capitalX" = "" ]; then
        cp /etc/group /etc/group.bak-group_upg
    else
        exit
    fi
fi
```

```
fi

# trovo i gruppi AFS cambiati rispetto al file dei gruppi locali
echo "Retrieving AFS groups"

find /afs/enea.it/common/etc/newgrp/ -name "group_*" -exec cat {} >>
$grpAFS \;

# prendo i gruppi AFS in locale e vedo se ci sono cambi
echo "Retrieving local groups"

cat $grpLocal | grep ":X:" >> $grpAFSlocal
diff $grpAFSlocal $grpAFS 1> /dev/null 2>&1

# se non ci sono cambi, allora non faccio nulla
if [ $? != 0 ]; then
    printf "Updating groups..."
    # trovo i gruppi non AFS locali
    cat $grpLocal | grep -v ":X:" >> $output
    # aggiungo i gruppi AFS cambiati all'output
    cat $grpAFS >> $output

    # faccio backup del file locale appendendo la data
    cp $grpLocal /tmp$grpLocal.`date +%Y%m%d%H%m`

    # Sostituzione del file group locale con quello generato qui
    cp $output $grpLocal
    # Cancellazione del file di output temporaneo
    rm $output
    printf " DONE\n"
fi

# Rimozione di tutti i file /tmp/etc/passwd* dei mesi precedenti
ls /tmp$grpLocal* | grep -v `date +%Y%m` | while read f_del
do
    rm $f_del
done
```



```
# Cancellazione dei file temporanei  
rm $grpAFS  
rm $grpAFSlocal  
echo "Script executed correctly"  
exit
```


Appendice C – Script *retrieveGroups.sh*

```
#!/bin/sh

# Script per il recupero dei gruppi
# presenti in ENEA-GRID
# G. Ponti --- 12/02/2011

export HOSTS=./hostNames
export EG_GROUPS_TMP=./egrid-groups-tmp
export EG_GROUPS=./egrid-groups
export RESULT=./egrid-retrieve-result

lsload -l | grep -v 'HOST_NAME' | grep -v 'unavail' | awk '{print $1}' >
$HOSTS

# recupero la lista dei vari /etc/group in ENEAGRID
for i in `cat $HOSTS`; do
    lsrun -m $i cat /etc/group >> $EG_GROUPS_TMP
done
cat $EG_GROUPS_TMP | awk -F: '{print $1 ":" $3}' | sort > $EG_GROUPS

# creo il file con i groupName e i groupIDs
latestGroupName=""
latestGroupId=""
grpLine=""
count=0
for i in `cat $EG_GROUPS`; do
    grpName=`echo $i | awk -F: '{print $1}'`
    grpId=`echo $i | awk -F: '{print $2}'`
    if [ "$grpName" != "$latestGroupName" ]; then # gruppo non presente, lo
inserisco
        if [ $count -ne 0 ]; then
```

```
    echo $grpLine >> $RESULT
fi

grpLine=$grpName":"$grpId
latestGroupName=$grpName
latestGroupId=$grpId
count=`echo "$count+1" | bc`

else

    if [ "$grpId" != "$latestGroupId" ]; then # gruppo presente, ma groupID
diverso

        grpLine=$grpLine" "$grpId

        latestGroupId=$grpId

    fi

fi

done

# scrivo l'ultima riga, altrimenti la perdo
echo $grpLine >> $RESULT

# rimuovo i file di supporto
rm $EG_GROUPS_TMP
rm $EG_GROUPS
rm $HOSTS
exit
```

Appendice D – Script *retrieveBusyGIDs.sh*

```
#!/bin/sh

# Script per il recupero dei GID
# gia' usati in ENEA-GRID.
# Lo script calcola anche i vari range di GID disponibili.
# G. Ponti --- 18/02/2011

export HOSTS=./hostNames
export EG_GROUPS=./egrid-groups
export EG_GID=./egrid-GID
export RESULT=./egrid-busyGIDs-result

lsload -l | grep -v 'HOST_NAME' | grep -v 'unavail' | awk '{print $1}' >
$HOSTS

# recupero la lista dei vari /etc/group in ENEAGRID
for i in `cat $HOSTS`; do
    lsruncat -m $i cat /etc/group >> $EG_GROUPS
done
cat $EG_GROUPS | awk -F: '{print $3}' | sort -g > $EG_GID

# creo l'elenco dei GID
latestGID=-1
currentGID=0
deltaFreeGID=0
for i in `cat $EG_GID`; do
    latestGID=$currentGID
    currentGID=$i
    deltaFreeGID=`echo "$currentGID-$latestGID-1" | bc`
    if [ $deltaFreeGID -gt 0 ]; then
```

```
printf $currentGID"\t##### Available GIDs : "$deltaFreeGID"\n\n" >>
$RESULT

else

  if [ $deltaFreeGID -eq 0 ]; then

    echo $currentGID >> $RESULT

  fi

fi

done

# rimuovo i file di supporto

rm $EG_GROUPS

rm $EG_GID

rm $HOSTS

exit
```

Bibliografia

- [1] Gruppi in AIX: <http://www.ibm.com/developerworks/aix/library/aix-aixuseradmin/index.html>
- [2] Gruppi in IRIX: <http://www.cs.bgu.ac.il/~arik/usail/man/irix/group.4.html>
- [3] Gruppi in Solaris: <http://publib.boulder.ibm.com/infocenter/db2luw/v8/index.jsp?topic=/com.ibm.db2.udb.doc/start/t0006742.htm>