Problem - "id laci" command was not showing many groups of which laci is member of.

steps to debug -

1)
divekar@mcclogin $ id nscd
uid=28(nscd) gid=28(nscd) groups=28(nscd)
divekar@mcclogin $ grep nscd /etc/passwd
nscd:x:28:28:NSCD Daemon:/:/sbin/nologin
divekar@mcclogin $ ps -ef | grep nscd
divekar 2622 2137 0 12:17 pts/16 00:00:00 grep nscd
nscd 8423 1 0 Feb22 ? 00:00:50 /usr/sbin/nscd

service nscd restart
Check "id laci"

2) Flush out nscd cache
grep enable-cache /etc/nscd.conf | grep -v "^#"
nscd -i group
nscd -i passwd
nscd -i hosts
Check "id laci"

3) Stop nscd and restart from command line -
service nscd stop
nscd -d -d -d
Check "id laci"

4) other techniques using strace -
Can you please stop nscd, run
strace -o /tmp/nscd.strace /usr/sbin/nscd -d -d -d > /tmp/nscd.log 2>&1 &
/usr/sbin/nscd -i password
/usr/sbin/nscd -i group
and now run the strace -o /tmp/id.log1 /usr/bin/id (or whatever results in the
failure to look up username)
and then strace -o /tmp/id.log2 /usr/bin/id username

Problem - This problem is related to RHEL-4 to RHEL-5 Migration. Some of the user or application related shell scripts which use certain options of commands like sort, cut, grep, find etc. might face issues after migrating to RHEL-5.

Workaround Solution –
I was able to resolve that issue by adding one line in the script – "export _POSIX2_VERSION=199209"

Some theory –
The gnu utilities normally conform to the version of POSIX that is standard for your system. To cause them to conform to a different version of POSIX, define the _POSIX2_VERSION environment variable to a value of the form yyyymm specifying the year and month the standard was adopted. Two values are currently supported for _POSIX2_VERSION: '199209' stands for POSIX 1003.2-1992, and '200112' stands for POSIX 1003.1-2001. For example, if you have a newer system but are running software that assumes an older version of POSIX and uses 'sort +1' or 'tail +10', you can work around any compatibility problems by setting '_POSIX2_VERSION=199209' in your environment.

Problem - This problem is related to upgraded ImageMagick RPM. One of our major application (elog application running on physics-elog), broke yesterday after Taylor cron job ran today at 2 AM. This happened possibly because ImageMagick rpm got upgraded by Taylor.

This is how "convert" command fails –

```
divekar@mcclogin $ convert tiger.ps tiger.jpg
Error: /undefinedfilename in (72x72)
Operand stack:

Execution stack:
   %interp_exit   .runexec2   --nostringval--   --nostringval--   --nostringval--
2   %stopped_push   --nostringval--   --nostringval--   --nostringval--   false
1   %stopped_push
Dictionary stack:
   --dict:1154/1684(ro)(G)--   --dict:0/20(G)--   --dict:70/200(L)--
Current allocation mode is local
Last OS error: 2
GPL Ghostscript 8.70: Unrecoverable error, exit code 1
Error: /undefinedfilename in (72x72)
Operand stack:

Execution stack:
   %interp_exit   .runexec2   --nostringval--   --nostringval--   --nostringval--
2   %stopped_push   --nostringval--   --nostringval--   --nostringval--   false
1   %stopped_push
Dictionary stack:
   --dict:1154/1684(ro)(G)--   --dict:0/20(G)--   --dict:70/200(L)--
Current allocation mode is local
Last OS error: 2
GPL Ghostscript 8.70: Unrecoverable error, exit code 1
convert: no decode delegate for this image format `tiger.ps'.
convert: Postscript delegate failed `tiger.ps'.
convert: missing an image filename `tiger.jpg'.
```

Solution - # yum downgrade ImageMagick
I was able to fix the problem by rolling back ImageMagick RPM.

For ensuring that Taylor won't upgrade ImageMagick Added following in /etc/taylor.opts file –

taylor_time=manual

Appended ImageMagick* to the exclude line in /etc/yum.conf

```
divekar@physics-elog $ cat /etc/yum.conf
[main]
http_caching=none
exclude=filesystem ImageMagick*
cachedir=/var/cache/yum
keepcache=0
debuglevel=2
logfile=/var/log/yum.log
distroverpkg=redhat-release
tolerant=1
exactarch=1
obsoletes=1
gpgcheck=1
plugins=1

# Note: yum-RHN-plugin doesn't honor this.
metadata_expire=1h

# Default.
installonly_limit = 8

# PUT YOUR REPOS HERE OR IN separate files named file.repo
# in /etc/yum.repos.d
```

Problem – From some OPIs printing not working for remote printers.

Solution –
This is a RHEL-4 to RHEL-5 upgrade issue.
We recently migrated from RHEL-4 to RHEL-5. After the migration we are seeing a weird issue related to printer configuration. "lostat -a" shows certain printers as printer@hostname - [root@lcls-opi01 ~]# lpstat -a | grep physics-lcl physics-lclslog accepting requests since Wed 29 Feb 2012 03:28:16 PM PST physics-lclslog@mccfs5 accepting requests since Wed 29 Feb 2012 11:35:27 AM PST All our scripts are hardcoded for printer physics-lclslog and not for physics-lclslog@mccfs5.

Workaround – Create local print queues by using following script -

```
cat /etc/printcap | grep ":rp=" | grep mccfs5 | awk -F':' '{print $3}'  | awk -F'rp=' '{print $2}' | awk  -F'@' '{print $1}' | while read a; do  lpadmin -p $a -E -v  ipp://mccfs5:631/printers/$a  -m raw;  done
```

Permanent solution (most probably !) –
service cups stop

>/var/cache/cups/remote.cache   <<<- I think this cache file was causing problem.

>/var/cache/cups/job.cache

service cups start

Problem – Servers hang after taylor upgrade or after mirror disks are created using linuxdd command.

Solution – This problem is associated with using File System Labels in certain configuration files.

RHEL-5 Grub related issues and solutions –

Important files -

/boot/grub/grub.conf
/etc/fstab
/boot/grub/device.map

Our standard mechanism for root disk mirroring is dd. We also have a linux-mirror script which uses dump command.
If following files contain references to disk partition LABELs then we can face bizarre issues due to disk mirroring –
/boot/grub/grub.conf
/etc/fstab

/etc/grub.conf would look like following (without LABEL entries) –
[root@mcclogin ~]# cat /etc/grub.conf
grub.conf generated by anaconda
#
Note that you do not have to rerun grub after making changes to this file
NOTICE: You do not have a /boot partition. This means that
all kernel and initrd paths are relative to /, eg.
root (hd0,0)
kernel /boot/vmlinuz-version ro root=/dev/sda1
initrd /boot/initrd-version.img
#boot=/dev/sda
default=7
timeout=5
splashimage=(hd0,0)/boot/grub/splash.xpm.gz
hiddenmenu
title Red Hat Enterprise Linux Client (2.6.18-308.el5)
root (hd0,0)

kernel /boot/vmlinuz-2.6.18-308.el5 ro root=/dev/sda1
initrd /boot/initrd-2.6.18-308.el5.img
title Red Hat Enterprise Linux Client (2.6.18-308.el5PAE)
root (hd0,0)
kernel /boot/vmlinuz-2.6.18-308.el5PAE ro root=/dev/sda1
initrd /boot/initrd-2.6.18-308.el5PAE.img

/etc/fstab would look like following (without LABEL entries) –
[root@mcclogin ~]# cat /etc/fstab
/dev/sda1 / ext3 defaults 1 1
/dev/sda8 /scratch ext3 defaults 1 2
/dev/sda7 /scswork ext3 defaults 1 2
/dev/sda6 /usr/vice/cache ext3 defaults 1 2
/dev/sda5 /tmp ext3 defaults 1 2
/dev/sda3 /var ext3 defaults 1 2
mccfs2:/export/mccfs/u1 /mccfs2/u1 nfs defaults 0 0
tmpfs /dev/shm tmpfs defaults 0 0
devpts /dev/pts devpts gid=5,mode=620 0 0
sysfs /sys sysfs defaults 0 0
proc /proc proc nosuid 0 0
/dev/sda2 swap swap defaults 0 0

Other important file which can cause unbootable mirror disk –

/boot/grub/device.map
Before mirroring the disk, ensure that this file contains entries for all the disks currently installed in the server.

In case of any discrepancies, recreate the /boot/grub/device.map file by issuing following commands –
mv /boot/grub/device.map /boot/grub/device.map.bak
echo | grub --device-map=/boot/grub/device.map
Note – The “echo |” will make the grub command non-interactive.

For a server having two disks (root and root-mirror), the /boot/grub/device.map file would look like following -
[root@mcclogin ~]# cat /boot/grub/device.map
(fd0) /dev/fd0
(hd0) /dev/sda
(hd1) /dev/sdb

How to find out LABEL of a file system ?
By using e2label command.
For example -
e2label /dev/sda1
/ROOT

Small script for converting LABEL entries of /etc/fstab -

```
#!/bin/bash
declare -a MYRA
MYRA=($(/bin/grep LABEL /etc/fstab | /bin/cut -d " " -f 1))

#iecho number of elements ${#MYRA[*]}
#echo number of elements ${#MYRA[@]}
/bin/cp /etc/fstab /etc/fstab.bkup
/bin/cp /etc/fstab /etc/fstab.intermed
for LABELID in ${MYRA[@]}
do
DEVNAME=($(/sbin/blkid -t $LABELID | /bin/cut -d " " -f 1 | /bin/cut -d ":" -f1))
if [ "$DEVNAME" != "" ]
then

echo Replacing $LABELID with $DEVNAME
/bin/cat /etc/fstab.intermed |/bin/sed "s#$LABELID #$DEVNAME #g" > /etc/fstab.intermed2
/bin/cp /etc/fstab.intermed2 /etc/fstab.intermed
rm -rf /etc/fstab.intermed2
else
echo blank
fi
done
/bin/mv /etc/fstab.intermed /etc/fstab
```

Problem – Slow response from NFS Server wain029.

```
After we migrated lcls-archsvr from RHEL-4 to RHEL-5  we started facing NFS
slowness issues for wain029.
On RHEL-5 the rsize and wsize by default get set to 1048576 for NFS server
wein029 (On RHEL-4 they get set to 32768 by default I believe).

I think we have more or less confirmed that this issue can be resolved by
changing rsize and wsize to lower value 32768/65536.
This can be done by your team if by changing the amd.nfs map entries (Or possibly
by editing /etc/amd.conf on lcls-archsrv ? ).

Solution-
SCCS did following –
Change # 1 ) He edited /etc/amd.conf file and changed selectors_on_defaults  to
selectors_in_defaults.  (As per Karl, selectors_on_defaults keyword has become
obsolete).
Change # 2) He did touch  /usr/etc/linux24.amd.flag

$  ypcat amd.nfs | grep -i wain029
```

```
-opts:=rw,intr,nosuid  host==wain029;type:=link;fs:=/g.archiver
host!=wain029;type:=nfs;rhost:=wain029;rfs:=/g.archiver
-opts:=rw,intr,nosuid  host==wain029;type:=link;fs:=/g.archiver
host!=wain029;type:=nfs;rhost:=wain029;rfs:=/g.archiver


[root@lcls-archsrv g.archiver]# cat /proc/mounts | grep wain029

wain029:/g.archiver /a/wain029/g.archiver nfs
rw,nosuid,vers=3,rsize=1048576,wsize=1048576,hard,intr,proto=tcp,timeo=600,retran
s=3,sec=sys,addr=wain029 0 0
```

Solution –

```
An option named changed in /etc/amd.conf.  i am using the
selectors_in_defaults amd option to allow us to set the rsize
and wsize differently from the defaults.

the default r/wsize for rhel5 is 1MB (the client and server negotiate
the highest possible).

with this amd option, we can tell amd to use a different size
other than the default.

now after your next reboot (which you are doing now), all amd
mounts on this machine will use 32K rsize and wsize.
```

divekar@lcls-archsrv $ cat /etc/amd.conf

```
# ---------------------------------------------------------------------
# Configuration file for amd.
# ---------------------------------------------------------------------
# check amd.conf(5) man page for details about options in this file
#


# GLOBAL OPTIONS SECTION
[ global ]
normalize_hostnames =   no
print_pid =           yes
pid_file =           /var/run/amd.pid
restart_mounts =       yes
auto_dir =           /a
log_file =           syslog
log_options =         all,noinfo
plock =              no
selectors_in_defaults = yes
print_version =       no
map_type =           nis
```

search_path = /etc
browsable_dirs = no
show_statfs_entries = no
fully_qualified_hosts = no
local_domain = slac.stanford.edu
dismount_interval = 1200
cache_duration = 3000

DEFINE AN AMD MOUNT POINT
[/nfs]
map_name = amd.nfs
map_type = nis

divekar@lcls-archsrv $ cat /proc/mounts | grep wain
wain029:/g.archiver /a/wain029/g.archiver nfs
rw,nosuid,vers=3,rsize=32768,wsize=32768,hard,intr,proto=tcp,timeo=600,retrans=3,sec=sys,addr=wain
029 0 0

Problem – Sosreport not working on mccfs5 (RHEL-4 Server).

Workaround –
Found that sosreport rpm was installed on mccfs5. Could not install that rpm due to broken yum configuration etc.

What I did –
Sosreport is python script. Also it requires certain python libraries.

Copied /usr/sbin/sosreport on to mccfs5 from another RHEL-4 Server lcls-uwd.
When tried to run it, got following error -
Error while running sosreport on mccfs5 -
File "/usr/sbin/sosreport", line 31, in ?
import sos.policyredhat
ImportError: No module named sos.policyredhat

How it was fixed -
I was not able to install sosreport rpm on mccfs5 due to yum configuration issues.
1) copied /usr/sbin/sosreport from lcls-uwd.
2) on lcls-uwd, which is RHEL-4 server, sosreport was working.
I created tar file of directory /usr/lib/python2.3/site-packages/sos and copied it over to mccfs5:/usr/lib/python2.3/site-packages and untared it.

cd /usr/lib/python2.3/site-packages
tar -xvf sos.tar

/usr/sbin/sosreport started working.

Problem – system-config-printers broken on OPIs. This is RHEL-5 upgrade issue for OPIs.

Solution – This is happening due to PATH variable. PATH variable for user root on OPIs as well as for server environments needs to be same.

We would face issues while running "system-config-printers" since it invokes a python script. As a workaround edit /root/.bash_profile file and edit PATH entry. It should be like - PATH=/sbin:/bin:/usr/sbin:/usr/bin:$PATH:$HOME/bin

Note - $PATH for user root on OPIs are due to different /etc/profile file. This comes from kickstart.

I copied /etc/profile from mcclogin onto lcls-opi10, that fixed the PATH issue for root on lcls-opi10. Though it needs to be tested for physics account. I will also check with SCCS tomorrow.

On opi's the following code in /etc/profile is setting the weird path – if ["$EUID" = "0"]; then
        pathmunge /sbin
        pathmunge /usr/sbin
        pathmunge /usr/local/sbin
fi

……

```
for i in /etc/profile.d/*.sh ; do
    if [ -r "$i" ]; then
        if [ "${-#*i}" != "$-" ]; then
            . $i
        else
            . $i >/dev/null 2>&1
        fi
    fi
done
```

Problem - Screen grab icon call not working on EOIC OPI.

Solution – This is RHEL-5 upgrade issue. Screen grab icon calls /usr/local/bin/scgrab.tk

This opi was recently upgraded to RHEL-5. In RHEL-5 the path for X11 is different than RHEL-4.

Edited /etc/X11/xorg.conf file. Commented existing line for RgbPath and put new line in it's place.

```
 #     RgbPath      "/usr/X11R6/lib/X11/rgb"
       RgbPath         "/usr/share/X11/rgb"
```

Closed all the icons/apps running and did "Control-Alt-Backspace" to reload X./usr/local/bin/scgrab.tk