Enstore Demo

10th international dCache users workshop
April 11-12, 2016, Hosted by PIC
Barcelona, Catalunya

Dmitry Litvintsev

Intent

- Goal of this demo is to perform installation and configuration of Enstore system on 4 computers provided to us by PIC
- Basically we will just follow Enstore Administration guide:
  - http://www-ccf.fnal.gov/enstore/Enstore_Administrator_Guide.pdf

Preliminaries

- We have 4 hosts:
  - ensws01, ensws02, moverws01, moverws02
- moverws01 has rsh access to SL8500 ACSLS control host (Automated Cartridge System Library Software).
- moverws01 and moverws02 are connected to drives in SL8500 library (one drive per node).
- Media is LTO4
- Chimera NFS server is dccore01-pps

System Layout

- ensws01 will be our “main” node, running databases, clerks, configuration, web-server and Inquisitor.
- ensws02 will run Library Managers, “null” movers and host database backup.
- moverws01 will run Media Changer and mover
- moverws02 will run mover

Distribute Enstore Repo file

```
cat enstore.repo
[enstore]
name=Enstore
baseurl=ftp://ssasrv1.fnal.gov/en/sl6x/$basesearch
enabled=1
gpgcheck=0


for host in ensws01 ensws02 moverws01 moverws02;
do
scp enstore.repo root@${h}:/etc/yum.repos.d;
done
```

Install Postgresql

```
for h in ensws01 ensws02 moverws01 moverws02
do
  ssh root@${h} "yum install -y
http://yum.postgresql.org/9.4/redhat/rhel-6-x86_64/pgdg-redhat94-9.4-1.noarch.rpm"
  ssh root@${h} "yum install -y postgresql94  postgresql94-libs postgresql94-server
postgresql94-contrib"
done

setup alternatives on ensws01 (this step is needed only b/c we run postgres as user
enstore)

VER="9.4"
for PROG in "initdb" "pg_ctl" "pg_config" "postmaster" "psql"; do
rm -f /var/lib/alternatives/pgsql-${PROG};
/usr/sbin/alternatives --install /usr/bin/${PROG} pgsql-${PROG} /etc/alternatives/pgsql-
${PROG} 920 --slave /etc/alternatives/pgsql-${PROG} pgsql-${PROG} /usr/pgsql-${VER}/bin/
${PROG};
rm -f /var/lib/alternatives/pgsql-${PROG}man;
/usr/sbin/alternatives --install /usr/share/man/man1/${PROG}.1 pgsql-${PROG}man /etc/
alternatives/pgsql-${PROG}man 920 --slave /etc/alternatives/pgsql-${PROG}man pgsql-$
{PROG}man /usr/pgsql-${VER}/share/man/man1/${PROG}.1;
done

(above gets initdb, pg_config, postmaster, psql into /usr/bin)
```

Install Enstore and dependencies

```
export HOSTS="ensws01 ensws02 moverws01 moverws02"

for h in ${HOSTS};
do
ssh root@${h} “yum -y install mt-st tcl tk enstore”;
done

#enstore installs into /opt/enstore

#on ensws01 install:

yum -y install ncompress gnuplot ImageMagick httpd
```

Enstore environment

```
# copy .bashrc from distribution

for h in ${HOSTS};
do
ssh root@${h} "cp /opt/enstore/external_distr/.bashrc ~enstore”;
done
```

ensws01

```
# web pages will go here:

mkdir -p  /var/www/html/enstore/web-pages
chown -R enstore:enstore /var/www/html/enstore
# setup files
su enstore
cd ~enstore
mkdir –p site_specific/config
cp /opt/enstore/site_specific/config/setup-enstore \
site_specific/config
cp /opt/enstore/etc/enstore_configuration_template \
site_specific/config/enstore.conf
```

Configuration

```
# define enstore environment
# edit ~enstore/site_specific/setup-enstore and
# define:
export ENSSH=/usr/bin/ssh
export ENSCP=/usr/bin/scp
export ENSTORE_CONFIG_HOST=ensws01
export ENSTORE_CONFIG_PORT=7500
export ENSTORE_CONFIG_FILE=/home/enstore/\ site_specific/
config/enstore.conf  ENSTORE_MAIL=youmail@example.com
export FARMLETS_DIR=/usr/local/etc/farmlets
# now copy ~enstore/site_specific/setup-
enstore to other 3 hosts
```

Edit configuration file (1)

```
# straightforward but tedious, editing
~enstore/site_specific/config/enstore.conf

system_name="EnstoreWS”
enstore_head_node="ensws01.pic.es”
monitoring_and_web=enstore_head_node
enstore_DB=enstore_head_node
enstore_backup="ensws02.pic.es”
enstore_library="ensws02.pic.es”
enstore_pnfs="dccore01-pps.pic.es”
media_changer_host="moverws01.pic.es"
html_dir = "/var/www/html/enstore/web-pages"
log_dir = "/var/log/enstore"
backup_dir =  "/home/enstore/backups"
inventory_dir = "/home/enstore/db-inventory"
httpd_dir = "/var/www/html/enstore”
ratekeeper_dir = "/home/enstore/ratekeeper"
db_basedir = "/home/enstore/databases”
```

Edit configuration file (2)

```
# we need connection to chimera to be able
# to mark files bad when they appear in
# t_location_info_trash table

configdict['namespace']    = {
system_name : {'dbname' : 'chimera',
               'dbhost' :  enstore_pnfs,
               'dbport' :  5432,
               'dbuser' : 'srmdcache' },
}
```

Edit configuration file (3)

```
# allow these subnets to connect to config server:

configdict['domains'] = {
  'invalid_domains’: [],
  'valid_domains’  :  ['192.168.20','193.109.174','193.109.172'],
}
```

Edit configuration file (4)

```
# configure crons
 configdict['crons'] = {
       'web_node'        : monitoring_and_web,
       'html_dir'        : html_dir,
       'log_dir'         : log_dir
       'tmp_dir'         : "/tmp",
       'email'           : ”litvinse@fnal.gov", # put here operation email address
       'developer_email' : "moibenko@fnal.gov", # put here developers email
       'backup_node'     : enstore_backup,
       'backup_dir'      : backup_dir,
     # Only on the following nodes are the ecron outputs monitored
    'monitored_nodes' : [
                enstore_head_node,
                enstore_DB,
                enstore_backup,
                enstore_library,
                enstore_pnfs,
                ],
'farmlets_dir'    : "/usr/local/etc/farmlets",
'url_dir'         : "http://ensws01.fnal.gov/enstore/", # make it up
'enstore_name'    : "EnstoreWS", # make it up
'spool_dir'       : "/home/enstore/MigrationSpool", #make_failed_copies
'backup2tape_dir' : "/pnfs/pic.es/data/enstore-ws/backup", # replace with name of backup
directory in name space
'copy_ran_file_log_dir'   : "/home/enstore/copy_ran_file",
}
```

Edit configuration file (5)

```
# configuration for presentation of system by inquisitor on web \
page(s)

configdict['www_server'] = { 'system_tag' : "EnstoreWS: Enstore for
the Enstore Workshop", # make up the name
                    'media' : {
                               '1' : 'Robot at PIC',
                               },
                    'inq_ignore' : 1 }
```

Edit configuration file (6)

```
# configure Library Manager

configdict['LTO4.library_manager'] = {
    'host':library_manager_host,
    'port':8546,
    'logname':'LTO4LM',
    'norestart':'INQ',
    'max_encp_retries':3,
    'max_file_size':(5000L*GB) - 1,
    'min_file_size':300*MB,
    'max_suspect_movers':3,
    'max_requests':15000,
    'restrict_access_in_bound': True,
    'CleanTapeVolumeFamily': 'CLEAN.CleanTapeFileFamily.noWrapper',
    'suspect_volume_expiration_time':3600*24,
    'legal_encp_version':legal_encp_version,
    'blank_error_increment': blank_error_increment,
    'mover_port': 7020,
    'encp_port':7021,
    'use_threads':False,
    'use_raw_input': 1,
}
```

Edit configuration file (7)

```
# STK media changer
configdict['SL8500.media_changer'] = {
    'host':media_changer_host,
    'port':7508,
    'logname':'SL8500G1MC',
    'type':'STK_MediaLoader',
    'norestart':'INQ',
    'max_work':6,
    'acls_host':'acsls',      # ACSLS controller host
    'acls_uname':'acsss',     # ACSLS user name
    'DriveCleanTime':{'LTO3':[60,1],
                      'LTO4':[60,1],
                      'T10000T2':[60,1],
                },
    'tape_library':"GCC StreamLine 8500",
}
```

Configure Movers (1)

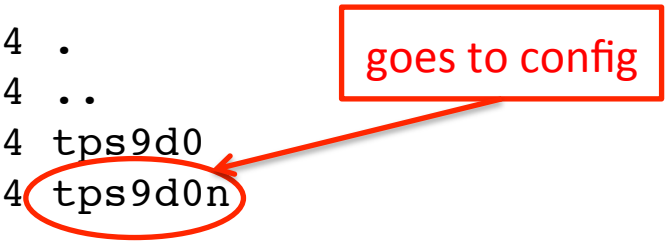
```
# build tape devices

# on moverws01 as a root:

source ~enstore/.bashrc
/etc/rc.local

Making SCSI device files:                                  [  OK  ]

[root@moverws01 ~]# ls -al  /dev/rmt
total 0
drwxr-xr-x  2 root root     80 Mar 25 18:14 .
drwxr-xr-x 20 root root   3880 Mar 25 18:14 ..
crw-rw-rw-  1 root root 9,   0 Mar 25 18:14 tps9d0
crw-rw-rw-  1 root root 9, 128 Mar 25 18:14 tps9d0n
```



Configure Movers (2)

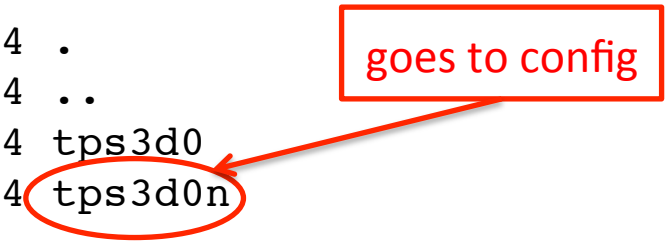
```
# build tape devices

# on moverws02 as a root:

source ~enstore/.bashrc
/etc/rc.local

Making SCSI device files:                                  [   OK   ]

[root@moverws02 ~]# ls -al /dev/rmt
total 0
drwxr-xr-x  2 root root     80 Apr  8 23:54 .
drwxr-xr-x 20 root root   3920 Apr  8 23:54 ..
crw-rw-rw-  1 root root 9,   0 Apr  8 23:54 tps3d0
crw-rw-rw-  1 root root 9, 128 Apr  8 23:54 tps3d0n
```



Configure Movers (3)

```
# determine mc device

# on moverws01:

[root@moverws01 ~]# ls -1 /dev/tape/by-path
pci-0000:05:02.1-fc-0x500104f000a42443-lun-0
pci-0000:05:02.1-fc-0x500104f000a42443-lun-0-nst

# on moverws01 query acsls

rsh -l acsss acsls "echo 'display drive * -f wwn  \r logoff' | bin/cmd_proc -l -q " | egrep "42$"
--------Oracle ACSLS (Automated Cartridge System Library Software) 8.2.0-------
  Copyright 1989, 2012 Oracle and/or its affiliates.  All rights reserved.

 0   3   1    0    50.01.04.f0.00.a4.24.42
```

Look for number that is 1 less

Goes to config

Configure Movers (4)

```
# determine mc device

# on moverws02:

[root@moverws02 ~]# ls -1 /dev/tape/by-path
pci-0000:05:02.1-fc-0x500104f000a424c7-lun-0
pci-0000:05:02.1-fc-0x500104f000a424c7-lun-0-nst

# on moverws01 query acsls

rsh -l acsss acsls "echo 'display drive * -f wwn  \r logoff' | bin/cmd_proc -l -q " | egrep "c6$"
--------Oracle ACSLS (Automated Cartridge System Library Software) 8.2.0-------
  Copyright 1989, 2012 Oracle and/or its affiliates.  All rights reserved.

 0   0   1    1   50.01.04.f0.00.a4.24.c6
```

Look for number that is 1 less

Goes to config

Configure Movers (5)

```
configdict['LTO4_01.mover'] = {
    'host':'moverws01',
    'data_ip':'moverws01',
    'port':7903,
    'logname':'LTO4_01MV',
    'statistics_path':'/tmp/enstore/enstore/LTO4_01.stat',
  . . .
    'check_written_file':lto4_mvr_check_f,
    'check_first_written_file':lto4_mvr_check_1st,
    'min_buffer':lto4_min_buffer,
    'max_buffer':lto4_max_buffer,
    'max_rate': lto4_rate,
    'mount_delay':15,
    'update_interval':lto4_update_interval,
    'library':'LTO4.library_manager',
    'device':'/dev/rmt/tps9d0n',
    'driver':'FTTDriver',
    'mc_device':'0,3,1,0',
    'media_changer':'SL8500.media_changer',
  . . .
    'dismount_delay': lto4_dismount_delay,
    'single_filemark': 1,
}
```

Configure Movers (6)

```
configdict['LTO4_02.mover'] = {
    'host':'moverws02',
    'data_ip':'moverws02',
    'port':7903,
    'logname':'LTO4_02MV',
    'statistics_path':'/tmp/enstore/enstore/LTO4_02.stat',
  . . .
    'check_written_file':lto4_mvr_check_f,
    'check_first_written_file':lto4_mvr_check_1st,
    'min_buffer':lto4_min_buffer,
    'max_buffer':lto4_max_buffer,
    'max_rate': lto4_rate,
    'mount_delay':15,
    'update_interval':lto4_update_interval,
    'library':'LTO4.library_manager',
    'device':'/dev/rmt/tps3d0n',
    'driver':'FTTDriver',
    'mc_device':'0,0,1,1',
    'media_changer':'SL8500.media_changer',
  . . .
    'dismount_delay': lto4_dismount_delay,
    'single_filemark': 1,
}
```

We are done with Config!

```
# lets start the config server

# on ensws01:
enstore start —-just config

# check that it works:
enstore conf —show
```

Make Farmlets

```
# Farmlets are used to run enstore
# cluster commands (like Estart and Estop)

# on ensws01 as root:

source ~enstore/.bashrc
/opt/enstore/external_distr/make_farmlets.sh

ls /usr/local/etc/farmlets/

dccore01-pps  enstore  enstore-down  ensws01
ensws02  movers  moverws01  moverws02  servers
```

Create web-serer config

```
[root@ensws01 ~]# /opt/enstore/sbin/install_enstore_html
Stopping httpd:                                            [FAILED]
CONF /etc/httpd//conf/httpd.conf
Writing file  /etc/httpd//conf/httpd.conf  Root  /etc/httpd/
exists False /var/www/html/enstore/html/enstore
log directory  /var/log/enstore
CONF /etc/httpd//conf/httpd.conf
Starting httpd:                                            [  OK  ]
[root@ensws01 ~]#

# check that this worked:
```

http://ensws01.pic.es/enstore/

Databases

create databases, on ensws01:

as root
source ~enstore/.bashrc:

```
python $ENSTORE_DIR/sbin/create_database.py enstoredb
python /opt/enstore/sbin/update_database_schema.py enstoredb
psql -h ensws01.pic.es -p 8888 -U enstore enstoredb -f enstoredb_update.sql

python $ENSTORE_DIR/sbin/create_database.py accounting
python /opt/enstore/sbin/update_database_schema.py accounting
psql -h ensws01.pic.es -p 8800 -U enstore accounting -f accounting_update.sql

python $ENSTORE_DIR/sbin/create_database.py drivestat
python /opt/enstore/sbin/update_database_schema.py drivestat
psql -h ensws01.pic.es -p 8802 -U enstore drivestat -f drivestat_update.sql
```

Install crons

```
#install crons:

# as root pm ensws01:

 source ~enstore/.bashrc

 python $ENSTORE_DIR/tools/install_crons.py
```

Start Enstore

```
# either run

   enstore Estart

# or run

   enstore start

# on each node
```

Define tapes

```
# on ensrvws01:

for v in G05501 G05502 G05503 G05504 G05505; do
enstore vol --VOL1OK --bypass-label-check --add ${v} LTO4  none none none LTO4 800GB
done
```

Install encp client

on dccore01-pps

wget ftp://ssasrv1.fnal.gov/en/slf6x/x86_64/encp-ups-opt-dcache-v3.11c-1.x86_64.rpm

rpm –i encp-ups-opt-dcache-v3.11c-1.x86_64.rpm

Create directory tags

```
# on dccore01 (for instance):

mount -v -o vers=3 dccore01-pps:/pnfs /mnt
mkdir /mnt/pic.es/data/enstore-ws
chown enstore:enstore /mnt/pic.es/data/enstore-ws
cd /mnt/pic.es/data/enstore-ws

echo “test” > “.(tag)(file_family)”
echo “2” > “.(tag)(file_family_width)”
echo “cpio_odc” > “.(tag)(file_family_wrapper)”
echo “LTO4” > “.(tag)(library)”
echo “enstore” > “.(tag)(storage_group)”
```

Try write and read file

```
export ENSTORE_CONFIG_HOST=ensws01
export ENSTORE_CONFIG_PORT=7500
# write
cd /mnt/pic.es/data/enstore-ws/test
/opt/encp/encp /etc/fstab fstab
#read
/opt/encp/encp /tmp
```

Observe file in the system

```
# check layers

cd /mnt/pic.es/data/enstore-ws/test
cat “.(use)(1)(fstab)
CDMS146016281400000

cat ".(use)(4)(fstab)"
G05501
0000_000000000_0000001
855
test
/mnt/pic.es/data/enstore-ws/test/fstab

0000B7E20C2EAC0343739C41C37F867A2FF9

CDMS146016281400000
moverws01:/dev/rmt/tps9d0n:1310163670
2109462862
```

That is it Folks!