SLAC perfSONAR-PS Customisations

Background

SLAC install has some differences

- all files kept in AFS
- centralised configuration
- mount of disk partition for data
- move of mysql database storage to separate partition
- use of apache redirect for calls

AFS

- symlinked `/opt/perfsonar` to `/afs/slac.stanford.edu/package/perfSONAR-PS/dist/`
- config files kept in `/afs/slac.stanford.edu/package/perfSONAR-PS/etc/`
    - each file is named to appropriate service
- customisations on init.d files to reflect above
    - install file located at `/afs/slac.stanford.edu/package/perfSONAR-PS/etc/init.d/`
    - will delete services from install and put in new ones with pointers to new config files
    - add's chkconfig stuff

Partitions

- largest disk partition mounted at `/var/lib/perfsonar` for data
- mysql
    - `mv /var/lib/mysql /var/lib/perfsonar/`
    - `sudo ln -sf /var/lib/perfsonar/mysql/ mysql`

Log Rotate

```
/etc/logrotate.d/perfsonar
/var/log/perfsonar/perfsonar_services.log {
    missingok
    notifempty
    sharedscripts
    postrotate
    endscript
}
```

Apache Redirects

all perfsonar hosts are

hostname
psnr-lat01.slac.stanford.edu
psnr-bw01.slac.stanford.edu

which run the following dns alias'

alias	purpose
psnr-ls.slac.stanford.edu	lookup service
psnr-pinger.slac.stanford.edu	pinger service
psnr-buoy.slac.stanford.edu	perfsonar buoy service
owamp.slac.stanford.edu	owamp service
bwctl.slac.stanford.edu	bwctl service

psnr-snmpma.slac.stanford.edu	snmp ma service
psnr-traceroute.slac.stanford.edu	traceroute service
ndt.slac.stanford.edu and netspeed.slac.stanford.edu	ndt service
npad.slac.stanford.edu	npad service

in addition, for http services (such as the web service communication and the perfAdmin web pages), we shall use an apache redirect such that

outside address	redirect address
perfsonar.slac.stanford.edu:80/perfSONAR_PS/services/hLS	psnr-ls.slac.stanford.edu:9995/perfSONAR_PS/services/hLS
perfsonar.slac.stanford.edu:80/perfSONAR_PS/services/pinger/ma	psnr-pinger.slac.stanford.edu:8075/perfSONAR_PS/services/pinger/ma
perfsonar.slac.stanford.edu:80/perfSONAR_PS/services/pinger/mp	psnr-pinger.slac.stanford.edu:8075/perfSONAR_PS/services/pinger/mp
perfsonar.slac.stanford.edu:80/perfSONAR_PS/services/pSB	psnr-buoy.slac.stanford.edu:8085/perfSONAR_PS/services/pSB
perfsonar.slac.stanford.edu:80/perfSONAR_PS/service/snmpma	psnr-snmpma.slac.stanford.edu:9990/perfSONAR_PS/services/SNMPMA

Database

moved to /var/lib/perfsonar/mysql

```
$ sudo cp -rpv /var/lib/mysql /var/lib/perfsonar/
$ sudo chown -R mysql:perfsonar /var/lib/perfsonar/mysql/
```

/etc/my.cnf

```
[mysqld]
datadir=/var/lib/perfsonar/mysql
socket=/var/lib/perfsonar/mysql/mysql.sock
user=mysql
# Default to using old password format for compatibility with mysql 3.x
# clients (those using the mysqlclient10 compatibility package).
old_passwords=1
tmpdir=/tmp/

# Disabling symbolic-links is recommended to prevent assorted security risks;
# to do so, uncomment this line:
# symbolic-links=0

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```

Initialise the DB

```
$ sudo /usr/bin/mysql_install_db --user=mysql
Installing MySQL system tables...
100512 16:36:15 [Warning] option 'max_join_size': unsigned value 18446744073709551615 adjusted to 4294967295
100512 16:36:15 [Warning] option 'max_join_size': unsigned value 18446744073709551615 adjusted to 4294967295
OK
Filling help tables...
100512 16:36:15 [Warning] option 'max_join_size': unsigned value 18446744073709551615 adjusted to 4294967295
100512 16:36:15 [Warning] option 'max_join_size': unsigned value 18446744073709551615 adjusted to 4294967295
OK

To start mysqld at boot time you have to copy
support-files/mysql.server to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
To do so, start the server, then issue the following commands:
/usr/bin/mysqladmin -u root password 'new-password'
/usr/bin/mysqladmin -u root -h psnr-bw01 password 'new-password'

Alternatively you can run:
/usr/bin/mysql_secure_installation

which will also give you the option of removing the test
databases and anonymous user created by default.  This is
strongly recommended for production servers.

See the manual for more instructions.

You can start the MySQL daemon with:
cd /usr ; /usr/bin/mysqld_safe &

You can test the MySQL daemon with mysql-test-run.pl
cd mysql-test ; perl mysql-test-run.pl

Please report any problems with the /usr/bin/mysqlbug script!

The latest information about MySQL is available on the web at
http://www.mysql.com
Support MySQL by buying support/licenses at http://shop.mysql.com
```

```
$ sudo /etc/init.d/mysqld start
Starting MySQL:  [  OK  ]
```

```
$ sudo /usr/bin/mysql_secure_installation



NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!


In order to log into MySQL to secure it, we'll need the current
password for the root user.  If you've just installed MySQL, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MySQL
root user without the proper authorisation.

Set root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
 ... Success!


By default, a MySQL installation has an anonymous user, allowing anyone
to log into MySQL without having to have a user account created for
them.  This is intended only for testing, and to make the installation
go a bit smoother.  You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] Y
 ... Success!

Normally, root should only be allowed to connect from 'localhost'.  This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] Y
 ... Success!

By default, MySQL comes with a database named 'test' that anyone can
access.  This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] Y
 - Dropping test database...
 ... Success!
 - Removing privileges on test database...
 ... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] Y
 ... Success!

Cleaning up...



All done!  If you've completed all of the above steps, your MySQL
installation should now be secure.

Thanks for using MySQL!
```