

White Paper on Green Plum Multinode Installation

Introduction

Insight from Big Data is essential to business today. Predictive analytics of high volumes of data can make the difference between a profit and loss, save lives.

Big data is an evolving term that describes any voluminous amount of structured, semi-structured and unstructured data that has the potential to be mined for information. Although big data doesn't refer to any specific quantity, the term is often used when speaking about petabytes and exabyte of data.

Relational database management systems and desktop statistics and visualization packages often have difficulty handling big data. The work instead requires "massively parallel software running on tens, hundreds, or even thousands of servers, and Greenplum Database is an example of one such database management tool.

It is a purpose-built, dedicated analytic data warehouse designed to extract value from the data. It is basically an extensible relational database platform that uses a shared-nothing, massively parallel processing (MPP) architecture. Built In this architecture, data is partitioned across multiple segment servers, and each segment owns and manages a distinct portion of the overall data.

Installing this database tool is an exhaustive procedure and quite number of technical issues arise during the same. This white paper serves to give a step by step procedural description to the multi-node installation of this database tool. It discusses the probable issues and the solutions which might be encountered during the course of installation, thereby making it student friendly and uncomplicated.

Greenplum Multinode Installation

Green-Plum Architecture usually consists of a master node and several other segments nodes. Multi-node Installation is a technique of installing each of these components on separate hosts where in every host involved in the installation must be configured accordingly to be compatible with the overall architecture.

Issues regarding the network configurations, synchronising problems, using the right compatible versions of the Operating Systems are few of the standard concerns faced during the multimode installation.

Hence it is necessary to have an exhaustive installation guide aiding the people to make the installation procedure as simple as possible.

Glossary

- `greenplum_path.sh` — This file contains the environment variables for Greenplum Database.
- `GPDB-LICENSE.txt` — Greenplum license agreement.
- `bin` — This directory contains the Greenplum Database management utilities. This directory also contains the PostgreSQL client and server programs, most of which are also used in Greenplum Database.
- `demo` — This directory contains the Greenplum demonstration programs.
- `docs` — The Greenplum Database documentation (PDF files).
- `etc` — Sample configuration file for OpenSSL.
- `ext` — Bundled programs (such as Python) used by some Greenplum Database utilities.
- `include` — The C header files for Greenplum Database.
- `lib` — Greenplum Database and PostgreSQL library files.
- `sbin` — Supporting/Internal scripts and programs.
- `share` — Shared files for Greenplum Database.

The Installation Procedure

1. Meeting the System Requirements.

Below listed are the minimum recommended specifications for servers intended to support Greenplum Database in a production environment.

Operating System: The preferred Operating System is **CentOS version 6.6**. The same Operating System must be used on the master as well as across all the segments involved during installation.

Problem face with different Operating Systems: Generally any of the following Operating Systems are recommended.

- SUSE Linux SLES 10.2 or higher
- CentOS 5.0 or higher
- RedHat Enterprise Linux 5.0 or higher

The RedHat OS when tried, required a number of additional yum repositories to be installed and since a free subscription wasn't able this OS had to be abandoned.

After experimenting with a number of versions of the CentOS ,version 6.6 was the considered the most preferable OS providing the most compatible platform to install all the required repositories aiding in the installation procedure.

2. Setting the GreenPlum Recommended OS parameters.

- Set the following parameters in the /etc/sysctl.conf file and reboot:
xfs_mount_options = rw,noatime,inode64,allocsize=16m
sysctl.kernel.shmmax = 500000000
sysctl.kernel.shmmni = 4096
sysctl.kernel.shmall = 4000000000
sysctl.kernel.sem = 250 512000 100 2048
sysctl.kernel.sysrq = 1
sysctl.kernel.core_uses_pid = 1
sysctl.kernel.msgmnb = 65536
sysctl.kernel.msgmax = 65536
sysctl.kernel.msgmni = 2048
sysctl.net.ipv4.tcp_syncookies = 1
sysctl.net.ipv4.ip_forward = 0
sysctl.net.ipv4.conf.default.accept_source_route = 0
sysctl.net.ipv4.tcp_tw_recycle = 1
sysctl.net.ipv4.tcp_max_syn_backlog = 4096
sysctl.net.ipv4.conf.all.arp_filter = 1
sysctl.net.ipv4.ip_local_port_range = 1025 65535
sysctl.net.core.netdev_max_backlog = 10000
sysctl.vm.overcommit_memory = 2
- Set the following parameters in the /etc/security/limits.conf file:
soft nfile 65536
hard nfile 65536
soft nproc 131072
hard nproc 131072
- XFS is the preferred file system on Linux platforms for data storage. Greenplum recommends the following xfs mount options:
rw,noatime,inode64,allocsize=16m
- The Linux disk I/O scheduler for disk access supports different policies, such as CFQ, AS, and deadline.
Greenplum recommends the following scheduler option: deadline
To specify a scheduler, run the following:
echo schedulename > /sys/block/devname/queue/scheduler
For example: # echo deadline > /sys/block/sda/queue/scheduler
- Each disk device file should have a read-ahead (blockdev) value of 16384.
To verify the read-ahead value of a disk device:
/sbin/blockdev --getra devname
For example: # /sbin/blockdev --getra /dev/sda

To set blockdev (read-ahead) on a device:

```
# /sbin/blockdev --setra bytes devname
```

For example: **# /sbin/blockdev --setra 16385 /dev/sda**

- Edit the /etc/hosts file and make sure that it includes the host names and all interface address names for every machine participating in your Greenplum Database system.

An example of the file is as depicted below:

```
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4  
:::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
```

```
10.10.1.77 mdw
```

```
10.10.2.77 mdw-1
```

```
10.10.1.69 sdw1
```

```
10.10.2.69 sdw1-1
```

```
10.10.1.81 sdw2
```

```
10.10.2.81 sdw2-1
```

Note: So in the above file, we have a single master running on 10.10.1.77 with two interfaces- mdw and mdw-1.

Similarly we have two segments, each of them with two interfaces. The first segment runs on 10.10.1.69 and the second segment on 10.10.1.81.

4. Running the GreenPlum Installer(To install GreenPlum libraries on master host)

- Download or copy the installer file to the machine that will be the Greenplum Database master host.
Recommended version: greenplum-db-4.3.3.1-build-1-RHEL5-x86_64.zip
- Unzip the installer file where PLATFORM is either RHEL5-i386 (RedHat 32-bit), RHEL5-x86_64 (RedHat 64-bit), SOL-x86_64 (Solaris 64-bit) or SuSE10-x86_64 (SuSe Linux 64 bit).
unzip greenplum-db-4.3.3.1-build-1-RHEL5-x86_64.zip
- Launch the installer using bash.
/bin/bash greenplum-db-4.3.3.1-build-1-RHEL5-x86_64.bin
- The installer will prompt you to accept the Greenplum Database license agreement. Type yes to accept the license agreement.
The installer will prompt you to provide an installation path. Press ENTER to accept the default install path (/usr/local/ greenplum-db-4.3.3.1) or enter an absolute path to an install location. You must have write permissions to the location you specify. Preferably choose the default install path.
- The installer will install the Greenplum software and create a greenplum-db symbolic link one directory level above your version-specific Greenplum installation directory. The symbolic link is used to facilitate patch maintenance and upgrades between versions. The installed location is referred to as \$GPHOME.

5. Installing and Configuring GreenPlum on all the hosts.

When a Greenplum Database system is first initialized, the system contains one predefined superuser role (also referred to as the system user), gpadmin. This is the user who owns and administers the Greenplum Database.

To install and configure Greenplum Database on all specified hosts.

- Log in to the master host as root:
`$ su -`
- Source the path file from your master host's Greenplum Database installation directory:
`# source /usr/local/greenplum-db/greenplum_path.sh`
- Create a file called hostfile_exkeys that has the machine configured host names and host addresses (interface names) for each host in your Greenplum system (master, standby master and segments). Make sure there are no blank lines or extra spaces. For example, if you have a master, and two segments with two network interfaces per host, your file would look something like:
`mdw
mdw-1
mdw-2
sdw1
sdw1-1
sdw1-2
sdw2
sdw2-1
sdw2-2`
- Run the gpseinstall utility referencing the hostfile_exkeys file you just created. Use the -u and -p options to create the Greenplum system user (gpadmin) on all hosts and set the password for that user on all hosts.
`# gpseinstall -f hostfile_exkeys -u gpadmin -p P@$word`

Problem faced: gpseinstall is just bound to fail if the versions of the OS are different on different segments.

[When run as root, gpseinstall copies the Greenplum Database installation from the current host and installs it on a list of specified hosts, creates the Greenplum system user (gpadmin), sets the system user's password (default is changeme), sets the ownership of the Greenplum Database installation directory, and exchanges ssh keys between all specified host address names (both as root and as the specified system user)]

Error when different OS's were used on the master and the segments.

OS on master CentOS 7, segment has ubuntu 12.04

Two interface cards on each master and server. All listed under all_hosts file

The problem:

When we execute gpseinstall -f hostfileexkeys -u gpadmin -p ****

It said –

```
gpsegininstall:mdw:root-[INFO]:-exchange ssh keys for user apadmin
gpsegininstall:mdw:root-[ERROR]:-failed doing a test read of file: su gpadmin
-c "cat hostfileexkeys &> /dev/null"
gpsegininstall:mdw:root-[ERROR]:-hostfile_exkeys is not accessible as user
gpadmin
gpsegininstall:mdw:root-[CRITICAL]:-early exit from gsegininstall
```

Hence it is highly recommended that the OS versions are the same on the master and the segments.

6. Confirming the Installation

Run the following confirmation steps from your Greenplum master host. If necessary, correct any problems before continuing on to the next task.

- Log in to the master host as gpadmin: `$ su - gpadmin`

Source the path file from Greenplum Database installation directory:

```
# source /usr/local/greenplum-db/greenplum_path.sh
```

- Use the gpssh utility to see if you can login to all hosts without a password prompt, and to confirm that the Greenplum software was installed on all hosts. Use the hostfile_exkeys file you used for installation.

```
$ gpssh -f hostfile_exkeys -e ls -l $GPHOME
```

- If the installation was successful, you should be able to log in to all hosts without a password prompt. All hosts should show that they have the same contents in their installation directories, and that the directories are owned by the gpadmin user.
- If you are prompted for a password, run the following command to redo the ssh key exchange:

```
$ gpssh-exkeys -f hostfile_exkeys
```

7. Creating the Greenplum Database Configuration File.

- Your Greenplum Database configuration file tells the gpinitssystem utility how you want to configure your Greenplum Database system. An example configuration file can be found in `$GPHOME/docs/cli_help/gpconfigs/gpinitssystem_config`.

- **To create a gpinitssystem_config file**

Log in as gadmin.

```
$ su - gadmin
```

Make a copy of the gpinitssystem_config file to use as a starting point.

```
$ cp $GPHOME/docs/cli_help/gpconfigs/gpinitssystem_config  
/home/gadmin/gpconfigs/gpinitssystem_config
```

- The DATA_DIRECTORY parameter is what determines how many segments per host will be created. If your segment hosts have multiple network interfaces, and you used their interface address names in your host file, the number of segments will be evenly spread over the number of available interfaces.

Here is an example of the required parameters in the gpinitssystem_config file:

```
ARRAY_NAME="EMC Greenplum DW"
```

```
SEG_PREFIX=gpseg
```

```
PORT_BASE=40000
```

```
declare -a DATA_DIRECTORY=(/data1/primary /data1/primary /data1/primary  
/data2/primary /data2/primary /data2/primary)
```

```
MASTER_HOSTNAME=mdw
```

```
MASTER_DIRECTORY=/data/master
```

```
MASTER_PORT=5432
```

```
TRUSTED_SHELL=ssh
```

```
CHECK_POINT_SEGMENT=8
```

```
ENCODING=UNICODE
```

- *(optional) If you want to deploy mirror segments, uncomment and set the mirroring parameters according to your environment.*

Here is an example of the optional mirror parameters in the gpinitssystem_config file:

```
MIRROR_PORT_BASE=50000
```

```
REPLICATION_PORT_BASE=41000
```

```
MIRROR_REPLICATION_PORT_BASE=51000
```

```
declare -a MIRROR_DATA_DIRECTORY=(/data1/mirror /data1/mirror /data1/mirror  
/data2/mirror /data2/mirror /data2/mirror)
```

Note: You can initialize your Greenplum system with primary segments only and deploy mirrors later using the gpaddmirrors utility.

Save and close the file.

8. Run the initialization Utility

To run the initialization utility:

- Run the following command referencing the path and file name of your initialization configuration file (gpinitssystem_config) and host file (hostfile_gpinitssystem).
`$ cd ~`
`$ gpinitssystem -c gpconfigs/gpinitssystem_config -h gpconfigs/hostfile_gpinitssystem`
- For a fully redundant system (with a standby master and a spread mirror configuration) include the -s and -S options.

`$ gpinitssystem -c gpconfigs/gpinitssystem_config -h gpconfigs/hostfile_gpinitssystem -s standby_master_hostname -S`

Problem faced: The above command throws an “unable to locate locale value on the sdw1 error” error if used different versions or entirely different Operating Systems on the master and the segments.

- The utility will verify your setup information and make sure it can connect to each host and access the data directories specified in your configuration. If all of the pre-checks are successful, the utility will prompt you to confirm your configuration.

Continue with Greenplum creation? Yy/Nn

Press y to start the initialization.

The utility will then begin setup and initialization of the master instance and each segment instance in the system. Each segment instance is set up in parallel.

Depending on the number of segments, this process can take a while.

At the end of a successful setup, the utility will start your Greenplum Database system.

You should see: => Greenplum Database instance successfully created.

9. Setting Greenplum Environment variables.

- You must configure your environment on the Greenplum Database master (and standby master).
- A greenplum_path.sh file is provided in your \$GPHOME directory with environment variable settings for Greenplum Database.
- You can source this file in the gpadmin user's startup shell profile (such as .bashrc).
- The Greenplum Database management utilities also require that the MASTER_DATA_DIRECTORY environment variable be set. This should point to the directory created by the gpinitssystem utility in the master data directory location.

- To set up your user environment for Greenplum
- Make sure you are logged in as gpadmin:
\$ su - gpadmin
- Open your profile file (such as .bashrc) in a text editor.
\$ vi ~/.bashrc
- Add lines to this file to source the greenplum_path.sh file and set the MASTER_DATA_DIRECTORY environment variable.

**source /usr/local/greenplum-db/greenplum_path.sh export
MASTER_DATA_DIRECTORY=/data/master/gpseg-1**
- (optional) You may also want to set some client session environment variables such as PGPORT, PGUSER and PGDATABASE for convenience.

export PGPORT=5432

export PGUSER=gpadmin export

PGDATABASE=default_login_database_name
- Save and close the file.
After editing the profile file, source it to make the changes active.
\$ source ~/.bashrc
- If you have a standby master host, copy your environment file to the standby master as well.

\$ cd ~ \$ scp .bashrc standby_hostname:`pwd`

Note: The .bashrc file should not produce any output. If you wish to have a message display to users upon logging in, use the .profile file instead.

Conclusion

This paper gives an overall description of the multimode installation procedure of the GreenPlum database and makes it as simple as possible which otherwise is a pretty cumbersome procedure.

All the unnecessary errors and challenges faced due to lack of understanding of the entire working mechanism and complex architecture is eliminated with the provision of the exact reasoning for every command that is depicted in this paper.