Oracle RAC with gpfs

And Oracle DATAGUARD

Part 1

IBM Systems and Technology Group
IBM Oracle Center
september 2011

ioc@fr.ibm.com
I. Table of Contents

II. INFORMATION................................................................................................................................. 3

CHANGE HISTORY ............................................................................................................................... 3

III. NOTICE ........................................................................................................................................ 4

IV. IBM POWER SYSTEM SETUP FOR ORACLE 11GR2 RAC INSTALLATION .................... 5

1 INTRODUCTION .................................................................................................................................. 5

2 POWER SYSTEM SETUP AND LOGICAL PARTITIONS CONFIGURATION ....................... 6

2.1 THE PHYSICAL ADAPTER RESOURCES ............................................................................... 7

2.2 VIRTUAL CDS AND VIRTUAL OPTICAL DRIVES .............................................................. 9

2.3 GLOBAL SETUP ...................................................................................................................... 13

2.4 THE NODES PARTITION CONFIGURATION ....................................................................... 13

3 AIX CONFIGURATION FOR ORACLE RAC .................................................................................. 15

3.1 TCP/IP FOR NODE INTERCONNECT NETWORK .............................................................. 15

3.2 NETWORK TIME PROTOCOL (NTP) .................................................................................... 17

3.3 DOMAIN NAME SERVER (DNS) ............................................................................................ 18

4 PREPARATION FOR RAC ............................................................................................................ 20

4.1 USER’S CREATION FOR RAC ............................................................................................. 20

4.2 CREATE REQUIRED DIRECTORIES .................................................................................. 25

4.3 ASSIGNED CAPABILITIES TO USERS .............................................................................. 26

5 PREPARE DISKS ............................................................................................................................ 27

5.1 PREPARE DISKS FOR GPFS .................................................................................................. 27

6 USING CLUSTER VERIFICATION UTILITY ................................................................................ 31

6.1 CHECK OPERATING SYSTEM PRE-REQUISITES ............................................................. 38

6.2 CHECKING NETWORK INTERFACES AND VLAN ......................................................... 42

6.3 ORACLE 11GR2 GRID INFRASTRUCTURE INSTALLATION .............................................. 44

6.4 PREPARE THE ENVIRONMENT ........................................................................................... 44

6.5 INSTALL THE CLUSTERWARE ENVIRONMENT ................................................................. 46
II. INFORMATION

Authors

François Martin [IBM] – IT Specialist
Claire NOIRAULT – Senior Specialist

Email address

ioc@fr.ibm.com

Abstract

This document is designed to help reader to implement Oracle 11gR2 Grid Infrastructure on IBM Power System running AIX 7

Version 1.0.0

Contacts

IBM Oracle Center (Montpellier- France)

Change history

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Editor</th>
<th>Reviewers</th>
<th>Editing description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>03/31/2011</td>
<td>NOIRAULT</td>
<td>Claire</td>
<td></td>
</tr>
</tbody>
</table>

The demo described in this document was implemented in October 2010.

We applied the Oracle & IBM recommendations that we have defined for hundred of customers, based on our experience of architecture, IBM Power systems, AIX, PowerVM Virtualization and Oracle clusterware and database components.
III. Notice

This document is based on our experiences. This is not an official (IBM) documentation. This document will be constantly updated and we’re open to any add-on or feedback from your own experiences!

This document is presented “As-Is” and IBM does not assume responsibility for the statements expressed herein.

All statements regarding IBM’s future direction and intent are subject to change or withdrawal without notice, represent goals and objectives only. Contact your local IBM office or IBM authorized reseller for the full text of a specific Statement of General Direction.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally available systems. Some measurements quoted in this document may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

The information contained in this paper resulted from:
- Oracle and IBM documentations
- Experiences done in the IBM Oracle Center

Information in this document concerning non-IBM products was obtained from the suppliers of these products, published announcement material or other publicly available sources. IBM has not tested these products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of these products.

Your comments are important for us, and we thank the ones who send us their feedback about how this document did help them in their implementation. We want our technical papers to be as helpful as possible.
IV. IBM Power System setup for Oracle 11gR2 RAC installation

1 Introduction

This part of the manual will guide you through the configuration and infrastructure setup of the IBM Power System, prior to the Oracle 11gR2 grid installation. It describes the step by step scenario to prepare the environment for the AIX logical partitions. These setup must be done before the Oracle installation.

This document is written for Oracle RAC cluster configuration, and you can use it for a Oracle Single Instance installation. Many parts are related to the AIX OS and PowerVM virtualization features and can be implemented for both RAC and non RAC environments. Only the disk clustering and node interconnect network AIX are compulsory for RAC environment.

The document explains the installation on a virtual environment, including VIO Servers and usage of PowerVM virtualization features. It can also be used for the setup of a physical configuration.

It contains details about the main concepts and the features of IBM PowerVM Virtualization which result in a flexible architecture for the Oracle implementation.

The System environment is defined for a typical configuration and it can be customized depending on the customer requirements. You can contact the authors of this document for architecture design and discussion.

One of the key points is to show and explain you how the PowerVM Virtualization features can be combined with Oracle RAC, and help you to optimize the usage of the resources.

PowerVM virtualization features are integrated from many years in the Power System, you may use some of the main features and tools in the following steps.
2 Power system setup and logical partitions configuration

The Oracle RDBMS software is certified according the AIX version, so there is no hardware dependency and the following configuration can be implemented on any of the IBM Power systems for both Power6 and Power7 processor based systems.

Let's introduce the environment given as example.

Two Power systems are configured, each will host one node of the Oracle RAC cluster.

An Hardware Management Console (HMC) is connected to the Service processor of both systems in order to manage the server resources and the Logical partitions (LPARs). It could have been two different HMCs.

The configuration example contains two Power7 processor based systems. It could have been one Power6 processor based system and one Power7 processor based system or two Power6 processor based systems.

The Oracle RAC environment is fully virtualized and one Virtual I/O Server partition has been created and installed on each server (VIOS version 2.1). For high availability, it would have been better to install two Virtual I/O Servers partitions on each server.
2.1 The physical adapter resources

Ethernet and Fiber Channel are assigned to the VIOS partition, and provide network and Storage access to LAN and SAN, so, all the node partitions are fully virtualized and do not use any I/O physical resource.

There is no restriction to assign physical resources to the RAC nodes partitions and mixed them with virtual adapters, and use redundant path mechanisms such as Etherchannel, or heterogeneous Multipathing (MPIO/SDDPCM, ...).

Nodes LPARs are defined with Virtual Ethernet adapters and are connected to the Virtual LANs defined in the Power Hypervisor Virtual Switch.
Nodes LPARs get access to the SAN disks Logical Units (LUNs) via VIOS and a WWPN.

```bash
{demo1:root}/ # lsdev | grep Fibre
fcs0       Available C5-T1       Virtual Fibre Channel Client Adapter
fcs1       Available C6-T1       Virtual Fibre Channel Client Adapter
sfwcomm0   Available C5-T1-01-FF Fibre Channel Storage Framework Comm
sfwcomm1   Available C6-T1-01-FF Fibre Channel Storage Framework Comm
```

Other LPARs can be hosted on the servers for other purpose and workload consolidation.
2.2 Virtual CDs and Virtual Optical Drives

You can create a virtual optical drive on the VIO Server and map it to your client partition (node demo1 and demo2) using the Virtual SCSI protocol. So, a cd/dvd drive is assigned to your client partition after running `cfgmgr` command (well known configuration command in AIX).

Virtual CDs can be created backed on files. They could be defined as blank cds or burned as iso files and containing several files from a directory. These CDs are saved in directory `/var/vio/VMLibrary`.

These virtual CDs can be loaded in the Virtual optical drives and then can be used from client partitions.

Blanks cds are usually defined as writable and can be loaded to one client drive at a time for backup purpose.

In our example, one Virtual iso CD has been burned with Oracle binaries and it is available as read only in the Virtual Media repository of VIOS, so this is the way you will get access from the nodes to the Oracle distribution for the installation and also GPFS filesets.

How to share Oracle distrib across nodes for installation?

See the following VIO Server commands examples to understand process and purpose:
2.2.1 Virtual Media Repository commands

You can create a virtual optical device and map it to one of your nodes using the following command on VIOS server:

```bash
# mkdev -fbo -vadapter vhost -dev node#cd
```

Don’t forget to delete it using using the following command.

```bash
# rmdev -dev node#cd
```

2.2.2 Command examples

```bash
$ mksp -f coderepossp hdisk22
$ mksp -fb fbpool -sp coderepossp -size 20G
$ mkrep -sp coderepossp -size 20G
$ lsrep
```

Virtual Media Repository Created

```bash
$ mkisofs -R -o /var/vio/VMLibrary/OracleCDs.iso /home/padmin/WS_11gR2_Distrib
```

2.2.3 Virtual Optical Drive commands

On the VIOS Partition, use as an example the following commands to create virtual cd device and map it to the virtual client partition.

To create a physical device.

```bash
$ mkdev -fbo -vadapter vhost0 -dev nodeld
```

To create and burn the iso CD containing the files from the `/home/padmin/...` directory, run the following command as root (execute `oem_setup_env` first).

```bash
$ echo "mkisofs -R -o /var/vio/VMLibrary/OracleCDs.iso /home/padmin/WS_11gR2_Distrib" | oem_setup_env
```

To create a blank cd with 5GB size:

```bash
$ mkvopt -name aixmksysb -size 5G
```

You can create a virtual optical device and map it to one of your nodes using the following command on VIOS server:

```bash
# mkdev -fbo -vadapter vhost -dev node#cd
```

Don’t forget to delete it using using the following command.

```bash
# rmdev -dev node#cd
```
To list the available virtual drives and display the loaded CDs:

```
$ lsvopt
VTD          Media                  Size(mb)
nodel1cd     No Media               n/a
node2cd      scripts.iso           1
node3cd      scripts.iso           1
```

To list the virtual CDs:

```
$ lsrep
Size(mb) Free(mb) Parent Pool        Parent Size       Parent Free
20270     2289 coderepossp           139904           119552

Name              File Size Optical Access
OracleCDs.iso      12860 None       ro
aixmksysb          5120 node1cd    rw
scripts.iso        1 node2cd       ro
scripts.iso        1 node3cd       ro
```

To load a virtual cd in the virtual drive:

```
$ loadopt -disk aixmksysb -vtd node1cd
```

```
$ lsrep
Size(mb) Free(mb) Parent Pool        Parent Size       Parent Free
20270     2289 coderepossp           139904           119552

Name              File Size Optical Access
OracleCDs.iso      12860 None       ro
aixmksysb          5120 node1cd    rw
scripts.iso        1 node2cd       ro
scripts.iso        1 node3cd       ro
```

To unload a virtual drive:

```
$ unloadopt -vtd node1cd
```

```
$ lsrep
Size(mb) Free(mb) Parent Pool        Parent Size       Parent Free
20269     7497 coderepossp           139904           119552

Name              File Size Optical Access
OracleCDs.iso      12771 None       rw
scripts.iso        1 node2cd       ro
scripts.iso        1 node3cd       ro
```

To delete a virtual cd:

```
$rmvopt -name aixopt1
```

```
```

To delete a virtual optical device:

```
$rmdev -dev node1cd
```

```
```
Another way for not simultaneous sharing: 1:N drive

Node n
Node n+1
Node n+2

"any partition -
but only one vscli mapping active at a time"

VIOS1

VIOS2

2.2.4 Example of AIX system backup using the Virtual cd and VIOS technique

On the AIX client run **cfgmgr** command to configure the virtual drive after VIO Server configuration.

```
$ cfgmgr
```

Create a system backup image with command **mksysb** in UDF format to your virtual DVD-RAM from the system rootvg

```
$ smitty sysbackup
```
2.3 Global setup

In this example, our cluster will be made of two RAC nodes and a third node for disaster Recovery.

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Logical partition name</th>
<th>IP address</th>
</tr>
</thead>
<tbody>
<tr>
<td>JSC_VIOS_lpm</td>
<td>JSC_VIOS_LPM_SANCERRE</td>
<td>10.16.25.50</td>
</tr>
<tr>
<td>demo1</td>
<td>JSC_demo_node1_FM</td>
<td>10.3.25.121</td>
</tr>
<tr>
<td>demo2</td>
<td>JSC_demo_node2_FM</td>
<td>10.3.25.122</td>
</tr>
<tr>
<td>demo3</td>
<td>JSC_demo_node3_FM</td>
<td>10.3.25.123</td>
</tr>
</tbody>
</table>

2.4 The nodes partition configuration

The partition nodes are defined as micro partitions with uncapped parameters. In our example, the nodes demo1 and demo2 are identical.

```
{lparstat -i}
```

Node Name : demo1
Partition Name : JSC_demo_node1_FM
Partition Number : 66
Type : Shared-SMT-4
Mode : Uncapped
Entitled Capacity : 0.20
Partition Group-ID : 32834
Shared Pool ID : 0
Online Virtual CPUs : 2
Maximum Virtual CPUs : 4
Minimum Virtual CPUs : 1
Online Memory : 6144 MB
Maximum Memory : 8192 MB
Minimum Memory : 1024 MB
Variable Capacity Weight : 128
Minimum Capacity : 0.10
Maximum Capacity : 0.40
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity Increment</td>
<td>0.01</td>
</tr>
<tr>
<td>Maximum Physical CPUs in system</td>
<td>64</td>
</tr>
<tr>
<td>Active Physical CPUs in system</td>
<td>48</td>
</tr>
<tr>
<td>Active CPUs in Pool</td>
<td>48</td>
</tr>
<tr>
<td>Shared Physical CPUs in system</td>
<td>48</td>
</tr>
<tr>
<td>Maximum Capacity of Pool</td>
<td>4800</td>
</tr>
<tr>
<td>Entitled Capacity of Pool</td>
<td>3010</td>
</tr>
<tr>
<td>Unallocated Capacity</td>
<td>0.00</td>
</tr>
<tr>
<td>Physical CPU Percentage</td>
<td>10.00%</td>
</tr>
<tr>
<td>Unallocated Weight</td>
<td>0</td>
</tr>
<tr>
<td>Memory Mode</td>
<td>Dedicated</td>
</tr>
<tr>
<td>Total I/O Memory Entitlement</td>
<td>-</td>
</tr>
<tr>
<td>Variable Memory Capacity Weight</td>
<td>-</td>
</tr>
<tr>
<td>Memory Pool ID</td>
<td>-</td>
</tr>
<tr>
<td>Physical Memory in the Pool</td>
<td>-</td>
</tr>
<tr>
<td>Hypervisor Page Size</td>
<td>-</td>
</tr>
<tr>
<td>Unallocated Variable Memory Capacity Weight</td>
<td>-</td>
</tr>
<tr>
<td>Unallocated I/O Memory entitlement</td>
<td>-</td>
</tr>
<tr>
<td>Memory Group ID of LPAR</td>
<td>-</td>
</tr>
<tr>
<td>Desired Virtual CPUs</td>
<td>2</td>
</tr>
<tr>
<td>Desired Memory</td>
<td>6144 MB</td>
</tr>
<tr>
<td>Desired Variable Capacity Weight</td>
<td>128</td>
</tr>
<tr>
<td>Desired Capacity</td>
<td>0.20</td>
</tr>
<tr>
<td>Target Memory Expansion Factor</td>
<td>-</td>
</tr>
<tr>
<td>Target Memory Expansion Size</td>
<td>-</td>
</tr>
<tr>
<td>Power Saving Mode</td>
<td>Dynamic Power Savings (Favor Power)</td>
</tr>
</tbody>
</table>
3 AIX configuration for Oracle RAC

The following step will guide you through the prerequisite configuration of your node partitions for Oracle RAC/CRS configuration.

3.1 TCP/IP for node interconnect network

An administration network has been previously defined for your nodes (i.e: 10.3.25.X ip address. You have now to configure additional networks for the RAC interconnect and add Virtual IP addresses for VIP on the administration network. Also an additional network is defined for GPFS.

In this example, the IP addresses of your nodes will be configured using the following name convention:

20.20.20.n is used for Virtual IP address (VIP)

30.30.30.n is used for RAC interconnect

40.40.40.n is used for GPFS

All en1, en2 and en3 interfaces are defined using virtual adapters ent1, ent2 and ent3 on different virtual LANs.

1 Use smitty chinet command from the AIX CLI to configure network 20.20.20.n on en1 interface, 30.30.30.n on en2 interface and 40.40.40.n on en3 interface or the following command from the AIX CLI of the nodes.

```bash
{demo1:root}/ # rmdev -dl 'en1'
{demo1:root}/ # rmdev -dl 'en2'
{demo1:root}/ # rmdev -dl 'en3'
{demo1:root}/ # mkdev -c if -s EN -t en -a netaddr='20.20.20.121' \
   -a netmask='255.255.255.0' -w 'en1' -a state='up' -a arp='on'
{demo1:root}/ # mkdev -c if -s EN -t en -a netaddr='30.30.30.121' \
   -a netmask='255.255.255.0' -w 'en2' -a state='up' -a arp='on'
{demo1:root}/ # mkdev -c if -s EN -t en -a netaddr='40.40.40.121' \
   -a netmask='255.255.255.0' -w 'en3' -a state='up' -a arp='on'
```

2 Edit the /etc/hosts file for your node and add the corresponding addresses of the other nodes of your cluster. Add a line to the /etc/hosts file to reference a DNS which will provide the SCAN adress.

```bash
{demo1:root}/ # cat /etc/hosts
127.0.0.1 loopback localhost  # loopback (lo0) name/address
::1 loopback localhost  # IPv6 loopback (lo0) name/address
#
# node public on en0
# en0
10.3.25.121 demo1
10.3.25.122 demo2
10.3.25.123 demo3
#
10.3.25.131 demo1-vip
10.3.25.132 demo2-vip
#
# en1
40.40.40.121 demo1-gpfs
40.40.40.122 demo2-gpfs
40.40.40.123 demo3-gpfs
#
# en2
```
1. Repeat the steps 1 and 2 on other nodes to configure IP addresses on all enx interfaces.
3.2 Network Time Protocol (NTP)

NTP is a protocol designed to synchronize the clocks of computers over a network. It is formalized by RFCs released by the IETF.

The ntpd daemon maintains the system time of day in synchronism with Internet standard time servers by exchanging messages with one or more configured servers at designated poll intervals.

Under ordinary conditions, ntpd adjusts the clock in small steps so that the timescale is effectively continuous and without discontinuities. Under conditions of extreme network congestion, the roundtrip delay jitter can exceed three seconds and the synchronization distance, which is equal to one-half the roundtrip delay plus error budget terms, can become very large. The ntpd algorithms discard sample offsets exceeding 128 ms, unless the interval during which no sample offset is less than 128 ms exceeds 900s. The first sample after that, no matter what the offset, steps the clock to the indicated time. In practice this reduces the false alarm rate where the clock is stepped in error to a vanishingly low incidence.

As the result of this behavior, once the clock has been set, it very rarely stays more than 128 ms, even under extreme cases of network path congestion. Sometimes, in particular when ntpd is first started, the error might exceed 128 ms. With RAC, this behavior is unacceptable. If the -x option is included on the command line, the clock will never be stepped and only slew corrections will be used.

The -x option sets the threshold to 600 s, which is well within the accuracy window to set the clock manually.

To configure the daemon NTP, proceed as follow:

1. Edit the file /etc/ntp.conf and add the following lines

   ```
   # Broadcast client, no authentication.
   #
   #broadcastclient
   server OEMGrid
driftfile /etc/ntp.drift
   #tracefile /etc/ntp.trace
   authenticate no
   OPTIONS="-x"
   ```

2. Run `/usr/sbin/xntpd -x` to start the NTP daemon.
   Don’t forget the `x` option.

3. To start `ntpd` daemon automatically at reboot with -x (slewing option), update the file `/etc/rc.tcpip` and uncomment the following line and add the -x options.

   ```
   -
   # Start up Network Time Protocol (NTP) daemon
   startsrc /usr/sbin/xntpd "$src_running" "-x"
   -
   # lssrc -a | grep ntp
   ```

To start `ntpd` daemon manually with -x (slewing option), enter:

```
# startsrc -s xntpd -a "-x"
```
3.3 Domain Name Server (DNS)

A DNS server has been setup on a partition of the administration network. The purpose is to resolve the cluster name.

1. Edit `/etc/resolv.conf` file and add the following lines.

   ```
   domain jsc.com
   nameserver 10.3.25.203
   ```

2. Edit the `/etc/netsvc.conf` file and add the following line to specify the name resolution order.

   ```
   hosts=local,bind
   ```

3.3.1 SCAN – Single Client Name

Scan (Single Client Access Name) is a single point of access for all applications connecting to an Oracle 11gR2 RAC Cluster and allows consistent connections without the need to know how many nodes are in the cluster. Virtual IPs are still used internally and can still be used for connections, but the initial connection to the cluster is made via a scan. Connections to any database in a cluster will be made via the scan.

The SCAN is a virtual IP name, similar to the names used for virtual IP addresses, such as demo1-vip. However, unlike a virtual IP, the SCAN is associated with the entire cluster, rather than an individual node, and associated with multiple IP addresses, not just one address.

The SCAN works by being able to resolve to multiple IP addresses reflecting multiple listeners in the cluster handling public client connections. When a client submits a request, the SCAN listener listening on a SCAN IP address and the SCAN port is contracted on a client's behalf. Because all services on the cluster are registered with the SCAN listener, the SCAN listener replies with the address of the local listener on the least-loaded node where the service is currently being offered. Finally, the client establishes connection to the service through the listener on the node where service is offered. All of these actions take place transparently to the client without any explicit configuration required in the client.

The SCAN should be configured so that it is resolvable either by using Grid Naming Service (GNS) within the cluster, or by using Domain Name Service (DNS) resolution. For high availability and scalability, Oracle recommends that you configure the SCAN name so that it resolves to three IP addresses. At a minimum, the SCAN must resolve to at least one address. If you specify a GNS domain, then the SCAN name defaults to `clustername-scan.GNS_domain`. Otherwise, it defaults to `clustername-scan.current_domain`. For example, if you start Oracle grid infrastructure installation from the server node1, the cluster name is mycluster, and the GNS domain is grid.example.com, then the SCAN Name is mycluster-scan.grid.example.com.

The GNS VIP address is the ip address of the server that will host the GNS. You need to make sure this one is available for use.

In our case, we decided to no use the GNS Domain and the DCHP server in order to control the IP addresses allocation.

1. On node1, as root user:

   ```
   [demo1:root]# cat /etc/hosts | grep clusterha
   20.20.20.121  clusterhaappsVIP1
   20.20.20.122  clusterhaappsVIP2
   ```

   ```
   [demo1:root]# nslookup clusterha
   Server:    10.3.25.203
   Address:   10.3.25.203#53

   Name:  clusterha.jsc.com
   Address: 10.3.25.171
   ```
Name: clusterha.jsc.com
Address: 10.3.25.172
Name: clusterha.jsc.com
Address: 10.3.25.173

(demo1:root)/ # ping clusterha
PING clusterha.jsc.com: (10.3.25.172): 56 data bytes
64 bytes from 10.3.25.172: icmp_seq=0 ttl=255 time=0 ms
64 bytes from 10.3.25.172: icmp_seq=1 ttl=255 time=0 ms
64 bytes from 10.3.25.172: icmp_seq=2 ttl=255 time=0 ms
^C
---clusterha.jsc.com PING Statistics---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms

2 Repeat step 1 commands on demo2 node as root user !!
4 Preparation for RAC

4.1 User’s creation for RAC

For the ease of administration we’ll implement a different user for each installation.

User grid will be responsible for GRID_HOME installation.

User oracle will be responsible for ORACLE_HOME installation.

1 On each node, connect as root and create the unix groups required with the following command:

On demo1, as root user:

```
{demo1:root}/ # /usr/bin/mkgroup -'A' id='1000' adms='root' oinstall
{demo1:root}/ # /usr/bin/mkgroup -'A' id='1100' adms='root' asmadmin
{demo1:root}/ # /usr/bin/mkgroup -'A' id='1200' adms='root' dba
{demo1:root}/ # /usr/bin/mkgroup -'A' id='1201' adms='root' oper
{demo1:root}/ # /usr/bin/mkgroup -'A' id='1300' adms='root' asmdba
{demo1:root}/ # /usr/bin/mkgroup -'A' id='1301' adms='root' asmoper
{demo1:root}/ # /usr/bin/mkgroup -'A' id='1600' adms='root' orauser
```

On demo2, as root user:

```
{demo2:root}/ # /usr/bin/mkgroup -'A' id='1000' adms='root' oinstall
{demo2:root}/ # /usr/bin/mkgroup -'A' id='1100' adms='root' asmadmin
{demo2:root}/ # /usr/bin/mkgroup -'A' id='1200' adms='root' dba
{demo2:root}/ # /usr/bin/mkgroup -'A' id='1201' adms='root' oper
{demo2:root}/ # /usr/bin/mkgroup -'A' id='1300' adms='root' asmdba
{demo2:root}/ # /usr/bin/mkgroup -'A' id='1301' adms='root' asmoper
{demo2:root}/ # /usr/bin/mkgroup -'A' id='1600' adms='root' orauser
```

2 On each node, connect as root and create the users grid and oracle required and assign required unix groups using the following commands:

On demo1, as root user:

```
{demo1:root}/ # /usr/bin/mkuser id='1100' pgrp='oinstall'
groups='dba,asmadmin,oper,asmdba,asmoper' admgrroups='system' home='/home/grid' grid
{demo1:root}/ # /usr/bin/mkuser id='1101' pgrp='oinstall' groups='dba,asmdba'
admgroups='system' home='/home/oracle' oracle
{demo1:root}/ #
```

On demo2, as root user:

```
{demo2:root}/ # /usr/bin/mkuser id='1100' pgrp='oinstall'
groups='dba,asmadmin,oper,asmdba,asmoper' admgrroups='system' home='/home/grid' grid
{demo2:root}/ #
{demo2:root}/ # /usr/bin/mkuser id='1101' pgrp='oinstall' groups='dba,asmdba'
admgroups='system' home='/home/oracle' oracle
{demo2:root}/ #
```

Be sure that all the groups and user numbers are identical through the nodes.

3 Setup user password for grid and oracle users.

Connect as root and setup password for users grid and oracle with the following command on each node.

On demo1, as root user:

```
{demo1:root}/ # passwd grid
Changing password for "grid"
grid's New password:
Enter the new password again:
```
On demo1, as root user:

```
{demo1:root}/ # passwd oracle
Changing password for "oracle"
oracle's New password:
Enter the new password again:
{demo1:root}/ #
```

On demo2, as root user:

```
{demo2:root}/ # passwd grid
Changing password for "grid"
grid's New password:
Enter the new password again:
{demo2:root}/ # passwd oracle
Changing password for "oracle"
oracle's New password:
Enter the new password again:
{demo2:root}/ #
```

4 Validate user password for grid and oracle users

Before to install Oracle RAC, you must validate users password on each node.

On demo1 for user grid:

```
{demo1:root}/ # telnet node1
Trying...
Connected to node1.
Escape character is '^]'.

AIX Version 6
login: grid
grid's Password: 
[compat]: 3004-610 You are required to change your password.
Please choose a new one.

grid's New password:
Enter the new password again: 
******************************************************************************

1 unsuccessful login attempt since last login.
Last unsuccessful login: Mon Jun 28 12:18:28 GMT+02:00 2010 on /dev/pts/1

$ exit
Connection closed.
{demo1:root}/ #
```

Repeat for user oracle on node1.

Then exit, and validate user password for grid and oracle users node2.
Setup ssh for grid and oracle users

To setup ssh, you can process the old way, wait for the Oracle Universal Installer (OUI) to do it for you, or do it right now using provided Oracle shell script to setup!

| Usage | .sshUserSetup.sh -user <user name> [ -hosts "<space separated hostlist>" ] [ -hostfile <absolute path of cluster configuration file> ] [ -advanced ] [ -verify ] [ -exverify ] [ -logfile <desired absolute path of logfile> ] [ -confirm ] [ -shared ] [ -help ] [ -usePassphrase ] [ -noPromptPassphrase ] |

To solve SSH setup issue (issue is tracked with Oracle bug 9044791) while installing with Oracle Universal Installer, use the following workaround.

On all nodes (demo1 and demo2), as root user, apply workaround or use the following script as root user from node1.

| for node in demo1 demo2 |
| do echo $node |
| ssh $node cp /usr/bin/sshkeygen /usr/local/bin/ssh-keygen |
| ssh $node cp /usr/bin/ssh /usr/local/bin/ssh |
| done |

Then from demo1, as root user, execute the script sshUserSetup.sh to setup SSH for grid user.

| {demo1:root}/cd/Grid_Infrastructure_11gR2/grid/sshsetup # ls |
| sshUserSetup.sh |
| {demo1:root}/cd/Grid_Infrastructure_11gR2/grid/sshsetup # ./sshUserSetup.sh -hosts "demo1 demo2" -user grid -advanced -noPromptPassphrase |

The output of this script is also logged into /tmp/sshUserSetup_2010-07-05-00-38-31.log

Hosts are demo1 demo2
user is grid
Platform: AIX
Checking if the remote hosts are reachable
Remote host reachability check succeeded.
The following hosts are reachable: demo1 demo2.
The following hosts are not reachable: .
All hosts are reachable. Proceeding further...
The script will setup SSH connectivity from the host demo1 to all the remote hosts. After the script is executed, the user can use SSH to run commands on the remote hosts or copy files between this host demo1 and the remote hosts without being prompted for passwords or confirmations.

NOTE 1: As part of the setup procedure, this script will use ssh and scp to copy files between the local host and the remote hosts. Since the script does not store passwords, you may be prompted for the passwords during the execution of the script whenever ssh or scp is invoked.

NOTE 2: AS PER SSH REQUIREMENTS, THIS SCRIPT WILL SECURE THE USER HOME DIRECTORY AND THE .ssh DIRECTORY BY REVOKE GROUP AND WORLD WRITE PRIVILEGES TO THESE DIRECTORIES.

Do you want to continue and let the script make the above mentioned changes (yes/no)?

The user chose yes
User chose to skip passphrase related questions.
Creating .ssh directory on local host, if not present already
Creating authorized_keys file on local host
Changing permissions on authorized_keys to 644 on local host
Creating known_hosts file on local host
Changing permissions on known_hosts to 644 on local host
Creating config file on local host
Removing old private/public keys on local host
Running SSH keygen on local host with empty passphrase
Generating public/private rsa key pair.
Your identification has been saved in //ssh/id_rsa.
Your public key has been saved in //ssh/id_rsa.pub.
The key fingerprint is:
The key's randomart image is:

```
E
 . . .
 = S . .
 o = . .
 B . . o
 o O . .
 o0+=...
```

Creating .ssh directory and setting permissions on remote host demo1
The script would also be revoking write permissions for group and others on the home directory for grid. This is an SSH requirement.
The script would create /home/grid/.ssh/config file on remote host demo1. If a config file exists already at /home/grid/.ssh/config, it would be backed up to /home/grid/.ssh/config.backup.
The user may be prompted for a password here since the script would be running SSH on host demo1.
Done with creating .ssh directory and setting permissions on remote host demo1.
Creating .ssh directory and setting permissions on remote host demo2
The script would also be revoking write permissions for group and others on the home directory for grid. This is an SSH requirement.
The script would create /home/grid/.ssh/config file on remote host demo2. If a config file exists already at /home/grid/.ssh/config, it would be backed up to /home/grid/.ssh/config.backup.
The user may be prompted for a password here since the script would be running SSH on host demo2.
Done with creating .ssh directory and setting permissions on remote host demo2.
Copying local host public key to the remote host demo1
The user may be prompted for a password or passphrase here since the script would be using SCP for host demo1.
Done copying local host public key to the remote host demo1
Copying local host public key to the remote host demo2
The user may be prompted for a password or passphrase here since the script would be using SCP for host demo2.
Done copying local host public key to the remote host demo2
Creating keys on remote host demo1 if they do not exist already. This is required to setup SSH on host demo1.
Creating keys on remote host demo2 if they do not exist already. This is required to setup SSH on host demo2.
Updating authorized_keys file on remote host demo1
Updating known_hosts file on remote host demo1
Updating authorized_keys file on remote host demo2
Updating known_hosts file on remote host demo2
SSH setup is complete.

-----------------------------------------------

Verifying SSH setup
-----------------------------------------------
The script will now run the date command on the remote nodes using ssh to verify if ssh is setup correctly. If the setup is correctly setup, there should be no output other than the date and SSH should not ask for passwords. If you see any output other than date or are prompted for the password, ssh is not setup correctly and you will need to resolve the issue and set up ssh again.
The possible causes for failure could be:
1. The server settings in /etc/ssh/sshd_config file do not allow ssh for user grid.
2. /home/grid or /home/grid/.ssh on the remote host may not be owned by grid.
3. User may not have passed -shared option for shared remote users or may be passing the -shared option for non-shared remote users.
6. If there is output in addition to the date, but no password is asked, it may be a security alert shown as part of company policy. Append the additional text to the <OMS HOME>/sysman/prov/resources/ignoreMessages.txt file.

---

--demo1:--
Running /usr/bin/ssh -x -l grid demo1 date to verify SSH connectivity has been setup from local host to demo1.
IF YOU SEE ANY OTHER OUTPUT BESIDES THE OUTPUT OF THE DATE COMMAND OR IF YOU ARE PROMPTED FOR A PASSWORD HERE, IT MEANS SSH SETUP HAS NOT BEEN SUCCESSFUL. Please note that being prompted for a passphrase may be OK but being prompted for a password is ERROR.
Mon Jul 11 17:26:33 CEST 2011

--demo2:--
Running /usr/bin/ssh -x -l grid demo2 date to verify SSH connectivity has been setup from local host to demo2.
IF YOU SEE ANY OTHER OUTPUT BESIDES THE OUTPUT OF THE DATE COMMAND OR IF YOU ARE PROMPTED FOR A PASSWORD HERE, IT MEANS SSH SETUP HAS NOT BEEN SUCCESSFUL. Please note that being prompted for a passphrase may be OK but being prompted for a password is ERROR.
Mon Jul 11 17:26:33 CEST 2011

Verifying SSH connectivity has been setup from demo1 to demo1
IF YOU SEE ANY OTHER OUTPUT BESIDES THE OUTPUT OF THE DATE COMMAND OR IF YOU ARE PROMPTED FOR A PASSWORD HERE, IT MEANS SSH SETUP HAS NOT BEEN SUCCESSFUL.
Mon Jul 11 17:26:34 CEST 2011

Verifying SSH connectivity has been setup from demo1 to demo2
IF YOU SEE ANY OTHER OUTPUT BESIDES THE OUTPUT OF THE DATE COMMAND OR IF YOU ARE PROMPTED FOR A PASSWORD HERE, IT MEANS SSH SETUP HAS NOT BEEN SUCCESSFUL.
Mon Jul 11 17:26:35 CEST 2011

--Verification from complete-
SSH verification complete.

{demo1:root}/tmp #

---

3 Then from demo1, as root user, execute the script sshUserSetup.sh to setup SSH for oracle user.

{nodel:root}/cd/Grid_Infrastructure_11gR2/grid/sshsetup # sshUserSetup.sh -hosts "demo1 demo2" -user oracle -advanced -noPromptPassphrase
4.2 Create Required Directories

On each node, connect as root and create the directories as follow:

1. On demo1, as root user:

```
{demo1:root}# /usr/bin/mkdir -p /oracle/app/11.2.0/grid
{demo1:root}# /usr/bin/chown -R grid:oinstall /oracle
{demo1:root}# /usr/bin/mkdir /oracle/app/oracle
{demo1:root}# /usr/bin/chown oracle:oinstall /oracle
{demo1:root}# /usr/bin/chmod -R 775 /oracle
```

2. On demo2, as root user:

```
{demo2:root}# /usr/bin/mkdir -p /oracle/app/11.2.0/grid
{demo2:root}# /usr/bin/chown -R grid:oinstall /oracle
{demo2:root}# /usr/bin/mkdir /oracle/app/oracle
{demo2:root}# /usr/bin/chown oracle:oinstall /oracle
{demo2:root}# /usr/bin/chmod -R 775 /oracle
```
4.3 Assigned Capabilities to Users

On each node, connect as root and create the directories as follow:

1. On demo1, as root user:

   ```
   [demo1:root]/ # /usr/sbin/lsuser -a capabilities grid
grid capabilities=CAP_NUMA_ATTACH,CAP_BYPASS_RAC_VMM,CAP_PROPAGATE
   [demo1:root]/ # /usr/sbin/lsuser -a capabilities oracle
   grid capabilities=CAP_NUMA_ATTACH,CAP_BYPASS_RAC_VMM,CAP_PROPAGATE,oracle capabilities=CAP_NUMA_ATTACH,CAP_BYPASS_RAC_VMM,CAP_PROPAGATE
   ```

2. On demo2, as root user:

   ```
   [demo2:root]/ # /usr/sbin/lsuser -a capabilities grid
   grid capabilities=CAP_NUMA_ATTACH,CAP_BYPASS_RAC_VMM,CAP_PROPAGATE
   [demo2:root]/ # /usr/sbin/lsuser -a capabilities oracle
   grid capabilities=CAP_NUMA_ATTACH,CAP_BYPASS_RAC_VMM,CAP_PROPAGATE,oracle capabilities=CAP_NUMA_ATTACH,CAP_BYPASS_RAC_VMM,CAP_PROPAGATE
   ```
5 Prepare Disks

Before to install Oracle Clusterware, you must choose between a clustered filesystem (GPFS) or ASM. In this demo, we choose clustered filesystem (GPFS).

5.1 Prepare Disks for GPFS

5.1.1 Hardware and software requirements

1 Verify that the SAN is attached to each node in the cluster and that you have enough disks to contain the file system.

<table>
<thead>
<tr>
<th>{demo1:root}/</th>
<th># lspv</th>
</tr>
</thead>
<tbody>
<tr>
<td>hdisk1</td>
<td>none</td>
</tr>
<tr>
<td>hdisk2</td>
<td>00c5fcefba3476c2</td>
</tr>
<tr>
<td>hdisk3</td>
<td>none</td>
</tr>
<tr>
<td>hdisk4</td>
<td>none</td>
</tr>
<tr>
<td>hdisk5</td>
<td>none</td>
</tr>
<tr>
<td>hdisk6</td>
<td>none</td>
</tr>
<tr>
<td>hdisk7</td>
<td>none</td>
</tr>
<tr>
<td>hdisk8</td>
<td>00c5fcef76355529</td>
</tr>
<tr>
<td>hdisk9</td>
<td>none</td>
</tr>
</tbody>
</table>

2 Install the GPFS fileset.

The installation process includes the GPFS licensed program on your system.

<table>
<thead>
<tr>
<th>{demo1:root}/</th>
<th># installp -aXY -d /oracle/gpfs gpfs</th>
</tr>
</thead>
<tbody>
<tr>
<td>{demo1:root}/</td>
<td># lslpp -l gpfs*</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fileset</th>
<th>Level</th>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>gpfs.base</td>
<td>3.4.0.4</td>
<td>COMMITTED</td>
<td>GPFS File Manager</td>
</tr>
<tr>
<td>gpfs.msg.en_US</td>
<td>3.4.0.0</td>
<td>COMMITTED</td>
<td>GPFS Server Messages - U.S. English</td>
</tr>
</tbody>
</table>

3 Verify the IP addresses.

It is necessary to have a dedicated high speed network supporting the IP protocol since GPFS passes a large amount of data between its daemons. GPFS communications require invariant static IP addresses for each specific GPFS node.

40.40.40.121 demo1-gpfs
40.40.40.122 demo2-gpfs
40.40.40.123 demo3-gpfs

In the event of a disk failure in which GPFS™ can no longer read or write to the disk, GPFS will discontinue use of the disk until it returns to an available state. You can guard against loss of data availability from disk failure by:

- Utilizing hardware data replication as provided by a Redundant Array of Independent Disks (RAID) device.
- Reduced recovery time using Persistent Reserve.
At the disk level, consider preparing disks for use with your file system by specifying failure groups that are associated with each disk. With this configuration, information is not vulnerable to a single point of failure.

When a node has access to file systems, it obtains disk leases that allow it to submit I/O. However, when a node fails, that node cannot obtain or renew a disk lease. When GPFS selects another node to perform recovery for the failing node, it first waits until the disk lease for the failing node expires. This allows for the completion of previously submitted I/O and provides for a consistent file system metadata log. Waiting for the disk lease to expire also avoids data corruption in the subsequent recovery step.

4 Ensure that the PATH environment variable on each node includes /usr/lpp/mmfs/bin.

```
{demo2:root}/ # echo $PATH
/usr/bin:/etc:/usr/sbin:/usr/ucb:/bin:/usr/bin/X11:/sbin:/usr/local/bin:/usr/lpp/mmfs/bin
```

5.1.2 Create a cluster GPFS

The mmcrnssd command is used to create cluster-wide names for NSDs used by GPFS.

1 Update file /rhosts on each node before to use mmcrcluster command to create the cluster.

```
{demo1:root}/ # cat /.rhosts
demo1 root
demo2 root
demo3 root
{demo1:root}/ #
```

2 Create a file that contains all of the nodes in your GPFS cluster.

```
{demo1:root}/ # echo 40.40.40.121:quorum-manager >/tmp/nodelist
{demo1:root}/ # echo 40.40.40.122:quorum >/tmp/nodelist
{demo1:root}/ # echo 40.40.40.123:quorum >>/tmp/nodelist
```

3 Create the cluster and use, for example, the node demo1-gpfs as the primary GPFS cluster configuration server node to store the GPFS configuration data.

```
{demo1:root}/ # mmcrcluster -N /tmp/nodelist -p demo1-gpfs
{demo1:root}/ #
```

4 Register the license and modify some attributes.

```
{demo1:root}/ # nodelist=demo1-gpfs,demo2-gpfs;export nodelist
{demo1:root}/ # mmchlicense server --accept -N $nodelist
{demo1:root}/ # mmchconfig prefetchThreads=500 -N $nodelist
{demo1:root}/ # no -r -o ipqmaxlen=512
```

5 Start the script mmdevdiscover to provide a list of available disk devices that appear in the node's local /dev file system.

```
{demo1:root}/ #/usr/lpp/mmfs/bin/mmdevdiscover
```
6: Starts the GPFS subsystem on all the nodes.

```bash
{demo1:root}/ # mmastartup -a
```

7: Create the pools for the disks.

```bash
{demo1:root}/ # echo "/dev/hdisk1:::disk1:poolA" >>/tmp/diskpool
{demo1:root}/ # echo "/dev/hdisk2:::disk2:poolA" >>/tmp/diskpool
{demo1:root}/ # echo "/dev/hdisk3:::disk3:poolA" >>/tmp/diskpool
{demo1:root}/ # echo "/dev/hdisk4:::disk4:poolB" >>/tmp/diskpool
{demo1:root}/ # echo "/dev/hdisk5:::disk5:poolB" >>/tmp/diskpool
{demo1:root}/ # echo "/dev/hdisk6:::disk6:poolB" >>/tmp/diskpool
{demo1:root}/ # echo "/dev/hdisk7:::disk7:poolC" >>/tmp/diskpool
{demo1:root}/ # echo "/dev/hdisk9:::disk9:poolC" >>/tmp/diskpool
{demo1:root}/ # echo "/dev/hdisk10:::disk10:poolC" >>/tmp/diskpool
```

7: Create cluster-wide names for NSDs used by GPFS with mmmcrnsd command.

```bash
{demo1:root}/ # mmmcrnsd -F /tmp/diskpool -v no
```

5.1.3 Create filesystem

On AIX nodes, the file /etc/filesystems contains lists for all GPFS file systems that exist in the cluster.

1: Create a GPFS file system with the mmcrfs command.

The block size and replication factors chosen affect file system performance. A maximum of 256 file systems can be mounted in a GPFS cluster at one time, including remote file systems.

```bash
{demo1:root}/ # mmcrfs gpfs1 "disk1;disk2;disk3" -B 512K -M 2 -R 2 -v no -T /gpfs1
{demo1:root}/ # mmcrfs gpfs2 "disk4;disk5;disk6" -B 512K -M 2 -R 2 -v no -T /gpfs2
{demo1:root}/ # mmcrfs gpfs3 "disk7;disk9;disk10" -B 512K -M 2 -R 2 -v no -T /gpfs3
```

2: Use the mmmlsfs command to list the attributes of a file system.

```bash
{demo1:root}/ # mmmlsfs
```

3: Mounts GPFS file systems on the nodes in the cluster.

Use the command mmmount to unmounts GPFS™ file systems on one or more nodes in the cluster.

```bash
{demo1:root}/ # mmmount gpfs1
{demo1:root}/ # mmmount gpfs2
{demo1:root}/ # mmmount gpfs3
{demo1:root}/ # lfs
```

<table>
<thead>
<tr>
<th>Name</th>
<th>Nodename</th>
<th>Mount Pt</th>
<th>VFS</th>
<th>Size</th>
<th>Options</th>
<th>Auto Accounting</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/hd4</td>
<td></td>
<td>/</td>
<td>jfs2</td>
<td>524288</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>/dev/hd1</td>
<td></td>
<td>/home</td>
<td>jfs2</td>
<td>131072</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>/dev/hd2</td>
<td></td>
<td>/usr</td>
<td>jfs2</td>
<td>5242880</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>/dev/hd9var</td>
<td></td>
<td>/var</td>
<td>jfs2</td>
<td>1310720</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>/dev/hd3</td>
<td></td>
<td>/tmp</td>
<td>jfs2</td>
<td>2097152</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>/dev/hd11admin</td>
<td></td>
<td>/admin</td>
<td>jfs2</td>
<td>262144</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>/proc</td>
<td></td>
<td>/proc</td>
<td>procfs</td>
<td>--</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>/dev/hd10opt</td>
<td></td>
<td>/opt</td>
<td>jfs2</td>
<td>917504</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>/dev/livedump</td>
<td></td>
<td>/var/adm/ras/livedump</td>
<td>jfs2</td>
<td>524288</td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>/dev/oracle</td>
<td></td>
<td>/oracle</td>
<td>jfs2</td>
<td>41025536</td>
<td>rw</td>
<td>no</td>
</tr>
<tr>
<td>/dev/gpfs1</td>
<td></td>
<td>/gpfs1</td>
<td>mmfs</td>
<td>--</td>
<td></td>
<td></td>
</tr>
<tr>
<td>/dev/gpfs2</td>
<td></td>
<td>/gpfs2</td>
<td>mmfs</td>
<td>--</td>
<td></td>
<td></td>
</tr>
<tr>
<td>/dev/gpfs3</td>
<td></td>
<td>/gpfs3</td>
<td>mmfs</td>
<td>--</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4: The mmchconfig command allows you to set a cluster-wide option to automatically start GPFS anytime a node is booted.

```bash
mmchconfig autoload=yes
```
5.1.4 Test the GPFS configuration

To control the configuration, issue the following commands:

1. Use `mmgetstate` to display the quorum, the number of nodes up, and the total number of nodes for the GPFS cluster.

   ```bash
   [demo1:root]# mmgetstate -a -L
   ```

2. Use `mmlsconfig` to display the current configuration data for a GPFS™ cluster, issue the following command:

   ```bash
   [demo1:root]# mmlsconfig
   ```

3. To verify the GPFS cluster configuration, examine the following files:
   The data pertaining to the node are stored in `/var/mmfs/gen/mmfsNodeData`.
   
   ```bash
   [demo1:root]# cat /var/mmfs/gen/mmfsNodeData
   %home%%:20_MEMBER_NODE::1:1:demo1-gpfs:40.40.40.121:demo1-gpfs :manager:::demol-gpfs:demol-gpfs:1207:3.4.0.4:AIX:Q:::::server
   ```

   The file `/var/mmfs/gen/mmsdrfs` contains a local copy of the mmsdrfs file found on the primary and secondary GPFS cluster configuration server nodes.

   The file `/var/mmfs/gen/mmfs.cfg` contains GPFS daemon startup parameters.
6 Using Cluster Verification Utility

Before to install the grid infrastructure, run the script `runcluvfy.sh` to check the configuration.

You must execute the program `cluvfy` on each node, as `grid` user.

On `demo1`, as grid user

```
{nodel:root}/cd/Grid_Infrastructure_11gR2/grid # runcluvfy.sh stage -pre crsinst -n demo1,demo2 -fixup -verbose
```

Performing pre-checks for cluster services setup

Checking node reachability...

Check: Node reachability from node "demo1"

<table>
<thead>
<tr>
<th>Destination Node</th>
<th>Reachable?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo2</td>
<td>yes</td>
</tr>
<tr>
<td>demo1</td>
<td>yes</td>
</tr>
</tbody>
</table>

Result: Node reachability check passed from node "demo1"

Checking user equivalence...

Check: User equivalence for user "grid"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>passed</td>
<td></td>
</tr>
<tr>
<td>demo2</td>
<td>passed</td>
<td></td>
</tr>
</tbody>
</table>

Result: User equivalence check passed for user "grid"

Checking node connectivity...

Checking hosts config file...

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>passed</td>
<td></td>
</tr>
<tr>
<td>demo2</td>
<td>passed</td>
<td></td>
</tr>
</tbody>
</table>

Verification of the hosts config file successful

<table>
<thead>
<tr>
<th>Name</th>
<th>MTU</th>
<th>IP Address</th>
<th>Subnet</th>
<th>Gateway</th>
<th>Def. Gateway</th>
<th>HW Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>en1</td>
<td>1500</td>
<td>20.20.20.121</td>
<td>20.20.20.0</td>
<td>20.20.20.121</td>
<td>10.3.25.254</td>
<td>12:E5:29:EF:77:03</td>
</tr>
<tr>
<td>en0</td>
<td>1500</td>
<td>10.3.25.121</td>
<td>10.3.25.0</td>
<td>10.3.25.121</td>
<td>10.3.25.254</td>
<td>12:E5:29:EF:77:02</td>
</tr>
<tr>
<td>en3</td>
<td>1500</td>
<td>40.40.40.121</td>
<td>40.40.40.0</td>
<td>40.40.40.121</td>
<td>10.3.25.254</td>
<td>12:E5:29:EF:77:07</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>MTU</th>
<th>IP Address</th>
<th>Subnet</th>
<th>Gateway</th>
<th>Def. Gateway</th>
<th>HW Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>en0</td>
<td>1500</td>
<td>10.3.25.122</td>
<td>10.3.25.0</td>
<td>10.3.25.122</td>
<td>10.3.25.254</td>
<td>12:E5:2F:2A:16:02</td>
</tr>
</tbody>
</table>
Check: Node connectivity of subnet "20.20.20.0"

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Connected?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1:en1</td>
<td>demo2:en1</td>
<td>yes</td>
</tr>
</tbody>
</table>

Result: Node connectivity passed for subnet "20.20.20.0" with node(s) demo1,demo2

Check: TCP connectivity of subnet "20.20.20.0"

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Connected?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1:20.20.20.121</td>
<td>demo2:20.20.20.122</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: TCP connectivity check passed for subnet "20.20.20.0"

Check: Node connectivity of subnet "10.3.25.0"

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Connected?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1:en0</td>
<td>demo2:en0</td>
<td>yes</td>
</tr>
</tbody>
</table>

Result: Node connectivity passed for subnet "10.3.25.0" with node(s) demo1,demo2

Check: TCP connectivity of subnet "10.3.25.0"

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Connected?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1:10.3.25.121</td>
<td>demo2:10.3.25.122</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: TCP connectivity check passed for subnet "10.3.25.0"

Check: Node connectivity of subnet "30.30.30.0"

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Connected?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1:en2</td>
<td>demo2:en2</td>
<td>yes</td>
</tr>
</tbody>
</table>

Result: Node connectivity passed for subnet "30.30.30.0" with node(s) demo1,demo2

Check: TCP connectivity of subnet "30.30.30.0"

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Connected?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1:30.30.30.121</td>
<td>demo2:30.30.30.122</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: TCP connectivity check passed for subnet "30.30.30.0"

Check: Node connectivity of subnet "40.40.40.0"

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Connected?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1:en3</td>
<td>demo2:en3</td>
<td>yes</td>
</tr>
</tbody>
</table>

Result: Node connectivity passed for subnet "40.40.40.0" with node(s) demo1,demo2

Check: TCP connectivity of subnet "40.40.40.0"

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Connected?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1:40.40.40.121</td>
<td>demo2:40.40.40.122</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: TCP connectivity check passed for subnet "40.40.40.0"

Interfaces found on subnet "20.20.20.0" that are likely candidates for VIP are:
demo1 en1:20.20.20.121
demo2 en1:20.20.20.122

Interfaces found on subnet "10.3.25.0" that are likely candidates for VIP are:
Interfaces found on subnet "30.30.30.0" that are likely candidates for VIP are:

demo1 en0:10.3.25.121
demo2 en0:10.3.25.122

Interfaces found on subnet "40.40.40.0" that are likely candidates for VIP are:

demo1 en3:40.40.40.121
demo2 en3:40.40.40.122

WARNING:
Could not find a suitable set of interfaces for the private interconnect

Result: Node connectivity check passed

Check: Total memory

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>2GB (2097152.0KB)</td>
<td>1.5GB (1572864.0KB)</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>2GB (2097152.0KB)</td>
<td>1.5GB (1572864.0KB)</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Total memory check passed

Check: Available memory

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>461.8789MB (472964.0KB)</td>
<td>50MB (51200.0KB)</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>725.6406MB (743056.0KB)</td>
<td>50MB (51200.0KB)</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Available memory check passed

Check: Swap space

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>512MB (524288.0KB)</td>
<td>3GB (3145728.0KB)</td>
<td>failed</td>
</tr>
<tr>
<td>demo2</td>
<td>512MB (524288.0KB)</td>
<td>3GB (3145728.0KB)</td>
<td>failed</td>
</tr>
</tbody>
</table>

Result: Swap space check failed

Check: Free disk space for "demo1:/tmp/

<table>
<thead>
<tr>
<th>Path</th>
<th>Node Name</th>
<th>Mount point</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>/tmp/</td>
<td>demo1</td>
<td>/tmp</td>
<td>1114.7852MB</td>
<td>1GB</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Free disk space check failed for "demo1:/tmp/

Check: Free disk space for "demo2:/tmp/

<table>
<thead>
<tr>
<th>Path</th>
<th>Node Name</th>
<th>Mount point</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>/tmp/</td>
<td>demo2</td>
<td>/tmp</td>
<td>1114.7852MB</td>
<td>1GB</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Free disk space check failed for "demo2:/tmp/

Check: User existence for "grid"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>exists(1100)</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>exists(1100)</td>
<td>passed</td>
</tr>
</tbody>
</table>

Checking for multiple users with UID value 1100
Result: Check for multiple users with UID value 1100 passed

Result: User existence check passed for "grid"

Check: Group existence for "oinstall"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>exists</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>exists</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Group existence check passed for "oinstall"

Check: Group existence for "dba"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
</table>

WARNING:
Could not find a suitable set of interfaces for the private interconnect
demo1 exists passed

Result: Group existence check passed for "dba"

Check: Membership of user "grid" in group "oinstall" [as Primary]

<table>
<thead>
<tr>
<th>Node Name</th>
<th>User Exists</th>
<th>Group Exists</th>
<th>User in Group</th>
<th>Primary</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Membership check for user "grid" in group "oinstall" [as Primary] passed

Check: Run level

<table>
<thead>
<tr>
<th>Node Name</th>
<th>run level</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Run level check passed

Check: Hard limits for "maximum open file descriptors"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Type</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>hard</td>
<td>922372036854775807 65536</td>
<td>passed</td>
<td></td>
</tr>
<tr>
<td>demo2</td>
<td>hard</td>
<td>922372036854775807 65536</td>
<td>passed</td>
<td></td>
</tr>
</tbody>
</table>

Result: Hard limits check passed for "maximum open file descriptors"

Check: Soft limits for "maximum open file descriptors"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Type</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>soft</td>
<td>922372036854775807 1024</td>
<td>passed</td>
<td></td>
</tr>
<tr>
<td>demo2</td>
<td>soft</td>
<td>922372036854775807 1024</td>
<td>passed</td>
<td></td>
</tr>
</tbody>
</table>

Result: Soft limits check passed for "maximum open file descriptors"

Check: Hard limits for "maximum user processes"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Type</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>hard</td>
<td>16384</td>
<td>16384</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>hard</td>
<td>16384</td>
<td>16384</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Hard limits check passed for "maximum user processes"

Check: Soft limits for "maximum user processes"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Type</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>soft</td>
<td>16384</td>
<td>2047</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>soft</td>
<td>16384</td>
<td>2047</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Soft limits check passed for "maximum user processes"

Check: System architecture

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>powerpc</td>
<td>powerpc</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>powerpc</td>
<td>powerpc</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: System architecture check passed

Check: Kernel version

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>7.1-6.1.0.0</td>
<td>6.1-6100.02.01</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>7.1-6.1.0.0</td>
<td>6.1-6100.02.01</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Kernel version check passed

Check: Kernel parameter for "SEM_NSEMS_MAX"
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Configured</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>32768</td>
<td>256</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>32768</td>
<td>256</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Kernel parameter check passed for "SEM_NSEMS_MAX"

Check: Kernel parameter for "SEM_VALUE_MAX"
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Configured</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>32767</td>
<td>100</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>32767</td>
<td>100</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Kernel parameter check passed for "SEM_VALUE_MAX"

Check: Kernel parameter for "ncargs"
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Configured</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>256</td>
<td>128</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>256</td>
<td>128</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Kernel parameter check passed for "ncargs"

Check: Package existence for "gpfs.base-3.2.1.8"
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>gpfs.base-3.4.0.4-0</td>
<td>gpfs.base-3.2.1.8</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>gpfs.base-3.4.0.4-0</td>
<td>gpfs.base-3.2.1.8</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Package existence check passed for "gpfs.base-3.2.1.8"

Check: Package existence for "bos.adt.base-..."
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>bos.adt.base-7.1.0.15-0</td>
<td>bos.adt.base-...</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>bos.adt.base-7.1.0.15-0</td>
<td>bos.adt.base-...</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Package existence check passed for "bos.adt.base-..."

Check: Package existence for "bos.adt.lib-..."
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>bos.adt.lib-7.1.0.0-0</td>
<td>bos.adt.lib-...</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>bos.adt.lib-7.1.0.0-0</td>
<td>bos.adt.lib-...</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Package existence check passed for "bos.adt.lib-..."

Check: Package existence for "bos.adt.libm-..."
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>bos.adt.libm-7.1.0.15-0</td>
<td>bos.adt.libm-...</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>bos.adt.libm-7.1.0.15-0</td>
<td>bos.adt.libm-...</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Package existence check passed for "bos.adt.libm-..."

Check: Package existence for "bos.perf.libperfstat-6.1.2.1"
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>bos.perf.libperfstat-7.1.0.15-0</td>
<td>bos.perf.libperfstat-6.1.2.1</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>bos.perf.libperfstat-7.1.0.15-0</td>
<td>bos.perf.libperfstat-6.1.2.1</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Package existence check passed for "bos.perf.libperfstat-6.1.2.1"

Check: Package existence for "bos.perf.perfstat-..."
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>bos.perf.perfstat-7.1.0.15-0</td>
<td>bos.perf.perfstat-...</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>bos.perf.perfstat-7.1.0.15-0</td>
<td>bos.perf.perfstat-...</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Package existence check passed for "bos.perf.perfstat-..."

Check: Package existence for "bos.perf.proctools-..."
<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>bos.perf.proctools-7.1.0.15-0</td>
<td>bos.perf.proctools-...</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>bos.perf.proctools-7.1.0.15-0</td>
<td>bos.perf.proctools-...</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Package existence check passed for "bos.perf.proctools-..."
Check: Package existence for "rsct.basic.rte-..."

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>rsct.basic.rte-3.1.0.4-0</td>
<td>rsct.basic.rte-...</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>rsct.basic.rte-3.1.0.4-0</td>
<td>rsct.basic.rte-...</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Package existence check passed for "rsct.basic.rte-..."

Check: Package existence for "rsct.comapt.clients.rte-...

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>rsct.comapt.clients.rte-3.1.0.0-0</td>
<td>rsct.comapt.clients.rte-...</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>rsct.comapt.clients.rte-3.1.0.0-0</td>
<td>rsct.comapt.clients.rte-...</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Package existence check passed for "rsct.comapt.clients.rte-..."

Check: Package existence for "xlC.aix61.rte-10.1.0.0"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>xlC.aix61.rte-11.1.0.1-0</td>
<td>xlC.aix61.rte-10.1.0.0</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>xlC.aix61.rte-11.1.0.1-0</td>
<td>xlC.aix61.rte-10.1.0.0</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Package existence check passed for "xlC.aix61.rte-10.1.0.0"

Check: Operating system patch for "Patch IZ41855"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Applied</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>missing</td>
<td>Patch IZ41855</td>
<td>failed</td>
</tr>
<tr>
<td>demo2</td>
<td>missing</td>
<td>Patch IZ41855</td>
<td>failed</td>
</tr>
</tbody>
</table>

Result: Operating system patch check failed for "Patch IZ41855"

Check: Operating system patch for "Patch IZ51456"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Applied</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>missing</td>
<td>Patch IZ51456</td>
<td>failed</td>
</tr>
<tr>
<td>demo2</td>
<td>missing</td>
<td>Patch IZ51456</td>
<td>failed</td>
</tr>
</tbody>
</table>

Result: Operating system patch check failed for "Patch IZ51456"

Check: Operating system patch for "Patch IZ52319"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Applied</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>missing</td>
<td>Patch IZ52319</td>
<td>failed</td>
</tr>
<tr>
<td>demo2</td>
<td>missing</td>
<td>Patch IZ52319</td>
<td>failed</td>
</tr>
</tbody>
</table>

Result: Operating system patch check failed for "Patch IZ52319"

Checking for multiple users with UID value 0
Result: Check for multiple users with UID value 0 passed

Check: Current group ID
Result: Current group ID check passed

Checking Core file name pattern consistency...
Core file name pattern consistency check passed.

Checking to make sure user "grid" is not in "root" group

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>does not exist</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>does not exist</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: User "grid" is not part of "root" group. Check passed

Check default user file creation mask

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>022</td>
<td>0022</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>022</td>
<td>0022</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Default user file creation mask check passed

Starting Clock synchronization checks using Network Time Protocol (NTP)...

NTP Configuration file check started...
The NTP configuration file "/etc/ntp.conf" is available on all nodes
NTP Configuration file check passed
Checking daemon liveness...

Check: Liveness for "xntpd"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Running?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>yes</td>
</tr>
<tr>
<td>demo2</td>
<td>yes</td>
</tr>
</tbody>
</table>

Result: Liveness check passed for "xntpd"

Checking NTP daemon command line for slewing option "-x"

Check: NTP daemon command line

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Slewing Option Set?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>yes</td>
</tr>
<tr>
<td>demo2</td>
<td>yes</td>
</tr>
</tbody>
</table>

Result: Clock synchronization check using Network Time Protocol (NTP) passed

Result: User ID < 65535 check passed

Result: Operating system patch check passed for Patch IZ41855, Patch IZ51456 and Patch IZ52319.

We can ignore them. These patches are included in last AIX 6.1 Technical level 5.

We can see “Pre-check for cluster services setup was unsuccessful on all the nodes.”. Let check what is failed:

```
{demo1:root}@/tmp # cat /tmp/cvu_check.txt | grep failed
```

Result: Swap space check failed

<table>
<thead>
<tr>
<th>/tmp/</th>
<th>demo1</th>
<th>/tmp</th>
<th>767.9288MB</th>
<th>1GB</th>
<th>failed</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>missing</td>
<td>Patch IZ51456</td>
<td>failed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>demo2</td>
<td>missing</td>
<td>Patch IZ51456</td>
<td>failed</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Result: Operating system patch check failed for "Patch IZ241855"

| demo1 | missing | Patch IZ52319 | failed |
| demo2 | missing | Patch IZ52319 | failed |

Result: Operating system patch check failed for "Patch IZ241855".

This amount of swap space is not mandatory on AIX! We can ignore it.

About line 3: Result: Swap space check failed

About lines 4: Result: Operating system patch check failed for Patch IZ41855, Patch IZ51456 and Patch IZ52319.

We can ignore them. These patches are included in last AIX 6.1 Technical level 5.
6.1 Check Operating system pre-requisites

We can check requirements in document

*Oracle® Grid Infrastructure*  
*Installation Guide*

*11g Release 2 (11.2) for IBM AIX Based Systems*  
*E10814-03*

http://download.oracle.com/docs/cd/E11882_01/install.112/e10814.pdf

Check Configuration on all nodes using the following shell script:

```bash
#!/bin/ksh
# set -x

(($#<1)) && { print "$0 usage $0 list of node" ; exit 2 ;}

while (($#!=0))
do

print $1

for node in $1
do

    echo ----------------------------------------
    echo $node
    echo ----------------------------------------

    ssh $node date
    ssh $node oslevel -r
    ssh $node /usr/sbin/lsattr -E -l sys0 -a realmem
    ssh $node /usr/sbin/lsps -s
    ssh $node /usr/sbin/lsps -a
    ssh $node /usr/bin/df -k /tmp
    ssh $node /usr/bin/df -k
    ssh $node bootinfo -K
    ssh $node who -r
    ssh $node lslpp -l bos.adt.base bos.adt.lib bos.adt.libm
    bos.perf.libperfstat bos.perf.perfstat bos.perf.proctools
    rsct.basic.rte rsct.compat.clients.rte xIc.aix61.rte
    #ssh $node instfix -i -k "IZ41855 IZ51456 IZ52319"
    ssh $node instfix -i -k "IZ71987"
    ssh $node instfix -i -k "IZ41855"
    ssh $node instfix -i -k "IZ51456"
    ssh $node instfix -i -k "IZ52319"
    ssh $node instfix -i -k "IZ30726"
    ssh $node instfix -i -k "IZ5274"
    ssh $node lslpp -l cluster.es.* rsct.hacmp.rte
    rsct.compat.basic.hacmp.rte rsct.compat.clients.hacmp.rte
    ssh $node ioo -o aio_maxreqs
    ssh $node lsattr -E -l sys0 -a maxproc
    ssh $node lsattr -E -l sys0 -a pre520tune
    ssh $node lsattr -E -l sys0 -a ncargs
    ssh $node lsattr -E -l sys0 -a ncargs
    for param in ipqmaxlen rfc1323 sb_max tcp_recvspace tcp_sendspace
    udp_recvspace udp_sendspace
    do
    sssh $node no -a | grep $param
    done
    for param in minperm maxperm maxclient lru_file_repage strict_maxclient
    strict_maxperm
```
do
    ssh $node vmo -a | grep $param
done

done

ssh $node schedo -a | grep vpm_xvcpus
echo
done

shift
done

---

<table>
<thead>
<tr>
<th>Path: /etc/objrepos</th>
<th>Fileset</th>
<th>Level</th>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bos.adt.base</td>
<td>6.1.5.0</td>
<td>COMMITTED</td>
<td>Base Application Development Toolkit</td>
<td></td>
</tr>
<tr>
<td>bos.adt.lib</td>
<td>6.1.2.0</td>
<td>COMMITTED</td>
<td>Base Application Development Libraries</td>
<td></td>
</tr>
<tr>
<td>bos.adt.libm</td>
<td>6.1.5.0</td>
<td>COMMITTED</td>
<td>Base Application Development Math Library</td>
<td></td>
</tr>
<tr>
<td>bos.perf.libperfstat</td>
<td>6.1.5.0</td>
<td>COMMITTED</td>
<td>Performance Statistics Library Interface</td>
<td></td>
</tr>
<tr>
<td>bos.perf.perfstat</td>
<td>6.1.5.0</td>
<td>COMMITTED</td>
<td>Performance Statistics Interface</td>
<td></td>
</tr>
<tr>
<td>bos.perf.proctools</td>
<td>6.1.5.0</td>
<td>COMMITTED</td>
<td>Proc Filesystem Tools</td>
<td></td>
</tr>
<tr>
<td>rsct.basic.rte</td>
<td>2.5.5.0</td>
<td>APPLIED</td>
<td>RSCT Basic Function</td>
<td></td>
</tr>
<tr>
<td>rsct.compat.clients.rte</td>
<td>2.5.5.0</td>
<td>APPLIED</td>
<td>RSCT Event Management Client Function</td>
<td></td>
</tr>
<tr>
<td>xlC.aix61.rte</td>
<td>10.1.0.3</td>
<td>COMMITTED</td>
<td>XL C/C++ Runtime for AIX 6.1</td>
<td></td>
</tr>
</tbody>
</table>

---

There was no data for IZ71987 in the fix database.
There was no data for IZ41855 in the fix database.
There was no data for IZ52319 in the fix database.
There was no data for IZ30726 in the fix database.

There was no data for IZ5274 in the fix database.

lslpp: 0504-132 Fileset cluster.es.* not installed.
lslpp: 0504-132 Fileset rsct hacmp.rte not installed.
lslpp: 0504-132 Fileset rsct.compat.basic.hacmp.rte not installed.
lslpp: 0504-132 Fileset rsct.compat.clients.hacmp.rte not installed.
aio_maxregs = 65536
maxproc 16384 Maximum number of PROCESSES allowed per user True
pre520tune disable Pre-520 tuning compatibility mode True
ncargs 256 ARG/ENV list size in 4K byte blocks True
ncargs 256 ARG/ENV list size in 4K byte blocks True

ipqmaxlen = 512
rfc1323 = 1
sb_max = 1310720
tcp_recvspace = 65536
tcp_sendspace = 65536
udp_recvspace = 655360
udp_sendspace = 65536

minperm = 30039
minperm% = 3
maxperm = 901287
vpm_xvcpus = 0

-------------------------------------------------------
Libraries
-------------------------------------------------------
rsct.basic.rte 2.5.5.0 APPLIED RSCT Basic Function
rsct.compat.clients.rte 2.5.5.0 APPLIED RSCT Event Management Client
bos.adt.lib 6.1.2.0 COMMITTED Base Application Development
bos.adt.libm 6.1.5.0 COMMITTED Base Application Development Toolkit
bos.perf.libperfstat 6.1.5.0 COMMITTED Performance Statistics Library
bos.perf.perfstat 6.1.5.0 COMMITTED Performance Statistics Interface
bos.perf.proctools 6.1.5.0 COMMITTED Proc Filesystem Tools
rsct.basic.rte 2.5.5.0 APPLIED RSCT Basic Function
rsct.compat.clients.rte 2.5.5.0 APPLIED RSCT Event Management Client Function
xlC.aix61.rte 10.1.0.3 COMMITTED XL C/C++ Runtime for AIX 6.1

Path: /usr/lib/objrepos
bos.adt.base 6.1.5.0 COMMITTED Base Application Development Toolkit
bos.adt.lib 6.1.2.0 COMMITTED Base Application Development Libraries
bos.adt.libm 6.1.5.0 COMMITTED Base Application Development Math Library
bos.perf.libperfstat 6.1.5.0 COMMITTED Performance Statistics Library Interface
bos.perf.perfstat 6.1.5.0 COMMITTED Performance Statistics Interface
bos.perf.proctools 6.1.5.0 COMMITTED Proc Filesystem Tools
rsct.basic.rte 2.5.5.0 APPLIED RSCT Basic Function
rsct.compat.clients.rte 2.5.5.0 APPLIED RSCT Event Management Client Function
xlC.aix61.rte 10.1.0.3 COMMITTED XL C/C++ Runtime for AIX 6.1
Path: /etc/objrepos
bos.adt.base        6.1.5.0 COMMITTED Base Application Development Toolkit
bos.perf.libperfstat 6.1.5.0 COMMITTED Performance Statistics Library Interface
bos.perf.perfstat    6.1.5.0 COMMITTED Performance Statistics Interface
rsct.basic.rte      2.5.5.0 APPLIED RSCT Basic Function

There was no data for IZ71987 in the fix database.
There was no data for IZ41855 in the fix database.
There was no data for IZ51456 in the fix database.
There was no data for IZ52319 in the fix database.
There was no data for IZ30726 in the fix database.
There was no data for Z55274 in the fix database.
lslpp: 0504-132 Fileset cluster.es.* not installed.
lslpp: 0504-132 Fileset rsct.hacmp.rte not installed.
lslpp: 0504-132 Fileset rsct.compat.basic.hacmp.rte not installed.
lslpp: 0504-132 Fileset rsct.compat.clients.hacmp.rte not installed.
aio_maxreqs = 65536
maxuproc 16384 Maximum number of PROCESSES allowed per user True
pre520tune disable Pre-520 tuning compatibility mode True
ncargs 256 ARG/ENV list size in 4K byte blocks True
ncargs 256 ARG/ENV list size in 4K byte blocks True
ipqmaxlen = 512
rfc1323 = 1
sb_max = 1310720
tcp_recvspace = 65536
tcp_sendspace = 65536
udp_recvspace = 655360
udp_sendspace = 65536
minperm = 29946
minperm% = 3
maxperm = 898419
vpm_xvcpus = 0
6.2 Checking Network Interfaces and VLAN

Let list network interfaces on all nodes! each node must have the same number of interfaces, and each interface must reference the same IP adress.

en0: 10.3.25.x
➤ Will be used as first Public Network Interface for node hostname, and node VIP

en1: 20.20.20.x
➤ Will be used as second Public Network Interface for application VIP purpose

en2: 30.30.30.x
➤ Will be used as Private Network for Oracle Clusterware Interconnect, and Oracle RAC cach fusion

en3: 40.40.40.x
➤ Will be used as Private Network Interface for IBM GPFS Interconnect

On demo1, as user root

```
{demo1:root}/tmp # for node in node1 node2
> do
> echo $node
> ssh $node ifconfig -l
> done
node1
en0 en1 en2 en3 lo0
node2
en0 en1 en2 en3 lo0
{demo1:root}/tmp #
```

```
{demo1:root}/tmp # for node in demo1 demo2
> do
> echo $node
> ssh $node lscfg | grep en
> done
node1
Model Implementation: Multiple Processor, PCI bus
* ent3 U9117.MMA.651A260-V17-C7-T1 Virtual I/O Ethernet Adapter (1-lan)
* ent2 U9117.MMA.651A260-V17-C6-T1 Virtual I/O Ethernet Adapter (1-lan)
* ent1 U9117.MMA.651A260-V17-C5-T1 Virtual I/O Ethernet Adapter (1-lan)
* ent0 U9117.MMA.651A260-V17-C2-T1 Virtual I/O Ethernet Adapter (1-lan)
node2
Model Implementation: Multiple Processor, PCI bus
* ent3 U9117.MMA.651A260-V20-C7-T1 Virtual I/O Ethernet Adapter (1-lan)
* ent2 U9117.MMA.651A260-V20-C4-T1 Virtual I/O Ethernet Adapter (1-lan)
* ent1 U9117.MMA.651A260-V20-C3-T1 Virtual I/O Ethernet Adapter (1-lan)
* ent0 U9117.MMA.651A260-V20-C2-T1 Virtual I/O Ethernet Adapter (1-lan)
{demo1:root}/tmp #
```

```
{demo1:root}/tmp # ifconfig -a
en0:
flags=I+IWL+X+T+IOP+IWP+IOM+IOW+IOMD+IOPM+IOMD+IOPMD
inet 10.3.25.101 netmask 255.255.255.128 broadcast 10.3.25.255
tcp_sendmss 512 tcp_recvmss 512

en1:
flags=I+IWL+X+T+IOP+IWP+IOM+IOW+IOMD+IOPM+IOMD+IOPMD
inet 20.20.20.101 netmask 255.255.255.128 broadcast 20.20.20.255
tcp_sendmss 512 tcp_recvmss 512
```
<table>
<thead>
<tr>
<th>Interface</th>
<th>IP Address</th>
<th>Netmask</th>
<th>Broadcast</th>
<th>TCP Send/Receive Space</th>
<th>RFC 1323</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>en3</td>
<td>30.30.30.101</td>
<td>0xffffff</td>
<td>30.30.30.255</td>
<td>262144 262144</td>
<td>1</td>
<td>1e080863, 480</td>
</tr>
<tr>
<td>lo0</td>
<td>40.40.40.101</td>
<td>0xffffff</td>
<td>40.40.40.255</td>
<td>262144 262144</td>
<td>1</td>
<td>e08084b, c0</td>
</tr>
<tr>
<td>lo0</td>
<td>127.0.0.1</td>
<td>0xff000000</td>
<td>127.255.255.255</td>
<td>131072 131072</td>
<td>1</td>
<td>:1%1/0</td>
</tr>
</tbody>
</table>

2 Repeat step 1 to check also on node demo2.
6.3 Oracle 11gR2 grid Infrastructure Installation

6.4 Prepare the environment

Before to install Clusterware, the grid infrastructure, you must prepare the environment. On all nodes (demo1, demo2), as root user, it is mandatory execute the script rootpre.sh.

1 Launch the script rootpre.sh on demo1

```
{demol:root}/ # cd /cd/Grid_Infrastructure_11gR2/grid
{node1:root}/cd/Grid_Infrastructure_11gR2/grid # ls
    doc  response  rootpre.sh  runInstaller  sshsetup  welcome.html
install  rootpre  rpm  runcluvfy.sh  stage
```

```
{demol:root}/cd/Grid_Infrastructure_11gR2/grid # ./rootpre.sh
./rootpre.sh output will be logged in /tmp/rootpre.out_10-06-28.15:20:09
Saving the original files in /etc/ora_save_10-06-28.15:20:09....
Copying new kernel extension to /etc....
Loading the kernel extension from /etc

Oracle Kernel Extension Loader for AIX
Copyright (c) 1998,1999 Oracle Corporation

Successfully loaded /etc/pw-syscall.64bit_kernel with kmid: 0x50949000
Successfully configured /etc/pw-syscall.64bit_kernel with kmid: 0x50949000
The kernel extension was successfully loaded.

Checking if group services should be configured....
Nothing to configure.
```

2 Launch the script rootpre.sh on demo2

3 On all nodes (demo1, demo2), as root user, execute:

```
[root@node1 ~]# slibclean
```

4 On demo1, as root user, execute:

```
{demol:root}/ # for node in demol demo2
> do
>   ssh $node mkdir /oracle/tmp
>   ssh $node chmod 777 /oracle/tmp
> done
{demol:root}/ #
```

5 Connect as grid user, update .profile file on all nodes for grid user and reconnect as grid on demo1.

```
{demol:root}/ # su - grid
$
$ pwd
/home/grid
$ vi .profile
```

```
```
```
```
```
```
```
```
```
if [ -t 0 ]; then
  stty intr ^C
fi

if [ -s "$MAIL" ] # This is at Shell startup. In normal
  then echo "$MAILMSG" # operation, the Shell checks
  fi  # periodically.

$ . ./profile
$ set | grep tmp
TEMP=/oracle/tmp
TEMPDIR=/oracle/tmp
TMP=/oracle/tmp
TMPDIR=/oracle/tmp
$

6 Under VNC Client session, or other graphical interface, execute from ONLY one node (demo1), as root user :

[root@node1 ~]# xhost +
access control disabled, clients can connect from any host
[root@node1 ~]#
6.5 Install the clusterware environment

```bash
{demo1:root}/ # su - grid
$ ls /cd/Grid_Infrastructure_11gR2/grid/
doc  response  rootpre.sh  runInstaller  sshsetup  welcome.html
install  rootpre  rpm  runcluvfy.sh  stage
$

$ /cd/Grid_Infrastructure_11gR2/grid/runInstaller

Your platform requires the root user to perform certain pre-installation OS preparation. The root user should run the shell script 'rootpre.sh' before you proceed with Oracle installation. rootpre.sh can be found at the top level of the CD or the stage area.

Answer 'y' if root has run 'rootpre.sh' so you can proceed with Oracle installation.
Answer 'n' to abort installation and then ask root to run 'rootpre.sh'.

Has 'rootpre.sh' been run by root? [y/n] (n)
y
Starting Oracle Universal Installer...

Checking Temp space: must be greater than 190 MB. Actual 1215 MB Passed
Checking swap space: must be greater than 150 MB. Actual 512 MB Passed
Checking monitor: must be configured to display at least 256 colors. Actual 16777216 Passed
Preparing to launch Oracle Universal Installer from /tmp/OraInstall2010-06-29_04-57-18PM.
Please wait ...
```

Wait until the OUI Installation screen appear.

1. **At the Select Installation Option screen**, select Install and Configure Grid Infrastructure for a Cluster, then **click the Next button**.

2. **At the Select Installation Type Screen** screen, select Advanced Installation, then **click the Next button**.

Typical Install follows the same goal of the Typical install for Database. This is meant for those who want an easy fast install. As much as possible is defaulted with limited interview screens.
3 At the Select Product Languages screen, select the languages in which the Oracle Grid Infrastructure Product will run and then click on the Next button.

In our example, we kept the default language which is English.

4 At the Grid Plug and Play Information screen, modify Cluster Name, Scan Name, Scan port and uncheck the box related to the GNS domain.

Enter a name for your cluster that is unique throughout your entire enterprise network. As you set up the SCAN entries in DNS, with the SCAN name resolving up to three different ip addresses, don’t check Configure GNS.

During installation, listeners are created on nodes for the SCAN IP addresses. Oracle Net Services routes application requests to the least loaded instance providing the service. Because the SCAN addresses resolve to the cluster, rather than to a node address in the cluster, nodes can be added to or removed from the cluster without affecting the SCAN address configuration.
5 Then click on the Next Button and wait for the screen Cluster Node Information.

6 At the Cluster Node Information screen, add an entry for each node within the cluster by clicking on the Add... button.

The node names provided at this step have to be able to resolve their name.

The management of the virtual ip address will be done automatically as long as a working DHCP service is available to serve ip addresses within that network. In our example, there is no DHCP server available, then the IP Addresses resolution must be done through the /etc/hosts file.

If the GNS Configuration box has been checked at the previous step, the resolution of VIPs can be done automatically and you do not need to provide any virtual hostname for each node.

The ssh configuration is already done, so you don’t need to click on the SSH Connectivity button neither have to enter the grid password. The RunInstaller will automatically check it.

7 Enter the details of the second node in the cluster, then click the OK button.
8 Once the configuration is done, click on the Next button and wait for the screen Specify Network Interface Usage.

9 At the Specify Network Interface Usage screen, define which interface will be used as public and/or private network.

Check the public and private networks are specified correctly.
en0 is Public Network (Node hostname and node VIP)
en1 is “Do Not use” Network (Application VIP)
en2 is Private Network (Oracle Cluster Interconnect)
en3 is “Do Not use” Network (IBM GPFS Interconnect)

10 Then click the Next button and wait for the Storage Option Information Screen.

With this new Release (11gR2), it is now possible to store the OCR and Voting Disks inside ASM. You have two possibilities:

- ASM : OCR & Voting Disks will be placed in an ASM Diskgroup.
- Shared File System. Raw devices and block devices are not supported to store the OCR and Voting disks. You have to use a cluster file system like GPFS to use this option.

In our case, we will use GPFS Option to store the OCR and Voting Disks.

11 Select the redundancy and the disks you want to use for OCR Files, then click the Next button.
12 Select the redundancy and the disks you want to use for Voting Disk, then click the Next button.

13 Assign different groups to specific management options and then click on the Next Button.
At the Specify Installation Location screen, specify the Oracle base and Software Location for the Grid Infrastructure. These locations have to exist (read and write) on all nodes that you plan to install the software on.

Oracle Grid Home should be installed parallel to the Oracle Base, so not inside the Oracle Base!

Provide the Oracle Inventory Directory and then click on the Next button.

Wait while the system is being verified thoroughly for prerequisites.
The result can be seen in the next screen shot.

![Screen Shot of Oracle Database 11g Prerequisite Checks](image)

Some of the minimum requirements for installation are not completed. Review and fix the issues listed in the following table, and recheck the system.

<table>
<thead>
<tr>
<th>Checks</th>
<th>Status</th>
<th>Fixable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Swap Size</td>
<td>Failed</td>
<td></td>
</tr>
<tr>
<td>05 Patch</td>
<td>1241855</td>
<td>Failed</td>
</tr>
<tr>
<td>05 Patch</td>
<td>1251456</td>
<td>Failed</td>
</tr>
<tr>
<td>05 Patch</td>
<td>1352319</td>
<td>Failed</td>
</tr>
</tbody>
</table>

This is a prerequisite condition to test whether sufficient total swap space is available on the system. See more details.

Check Failed on Nodes: [node1, node2, node3]
A new feature in this release is the fixup script. Everything that Oracle can change in a script, a script is provided for by the installer. Just run the script and those checks that failed and are Fixable are being fixed. Don’t worry about prerequisite checks before running the installer. Just let the installer do the work for you. It’ll report everything you need to fix before installing. Just push the “Check Again” button as many times as you wish until everything is reported as passed.

Below are an example of the explanation errors we get:
- Swap Size : Don’t expand the swap size in our case!
- AIX APAR’s : ALL APAR’s included in AIX 6.1 TL5, Don’t need to!

17 Check the box Ignore All to ignore the errors and click on the Next button.

18 The next screen summarizes all informations provided in the previous steps. Save Response File by clicking on the Save Response File button.
Then start the installation by clicking on Finish button.
Run these two scripts to finished with the installation of Grid Infrastructure.

The following configuration scripts need to be executed as the "root" user in each cluster node.

**Scripts to be executed:**

<table>
<thead>
<tr>
<th>Number</th>
<th>Script Location</th>
<th>Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/oracle/app/oraInventory/orainstRoot.sh</td>
<td>node1,node2,node3</td>
</tr>
<tr>
<td>2</td>
<td>/oracle/grid/app/11.2.0/grid/root.sh</td>
<td>node1,node2,node3</td>
</tr>
</tbody>
</table>

To execute the configuration scripts:
1. Open a terminal window
2. Log in as "root"
3. Run the scripts in each cluster node
4. Return to this window and click "OK" to continue

Run the script on the local node first. After successful completion, you can run the script in parallel on all the other nodes.

---

**On each node, one after the others (not at the same time), as root user:**

**FIRST on demo1, as root user:**

```bash
{demo1:root}/ # /oracle/app/oraInventory/orainstRoot.sh
Changing permissions of /oracle/app/oraInventory.
Adding read,write,execute permissions for world.
Changing groupname of /oracle/app/oraInventory to oinstall.
The execution of the script is complete.
{demo1:root}/tmp #
```

```bash
{demo2:root}/ # /oracle/app/oraInventory/orainstRoot.sh
Changing permissions of /oracle/app/oraInventory.
Adding read,write permissions for group.
Removing read,write,execute permissions for world.
Changing groupname of /oracle/app/oraInventory to oinstall.
The execution of the script is complete.
{demo2:root}/ #
```

**On each node, one after the others (not at the same time), as root user:**

**FIRST on demo1, as root user:**

```bash
{demo1:root}/ # /oracle/grid/app/11.2.0/grid/root.sh
Running Oracle 11g root.sh script...

The following environment variables are set as:
  ORACLE_OWNER= grid
  ORACLE_HOME= /oracle/grid/app/11.2.0/grid

Enter the full pathname of the local bin directory: [/usr/local/bin]:
The file "dbhome" already exists in /usr/local/bin. Overwrite it? (y/n) [n]:
The file "oraenv" already exists in /usr/local/bin. Overwrite it? (y/n) [n]:
The file "oraenv" already exists in /usr/local/bin. Overwrite it? (y/n) [n]:
```
Entries will be added to the /etc/oratab file as needed by Oracle Configuration Assistant when a database is created.

Finished running generic part of root.sh script.

Creating trace directory

Creating OCR keys for user 'root', privgrp 'system'.

Operation successful.

Adding daemon to inittab

CRS-4123: Oracle High Availability Services has been started.

CRS-2672: Attempting to start 'ora.gipcd' on 'nodel'
CRS-2672: Attempting to start 'ora.mdnsd' on 'nodel'
CRS-2676: Start of 'ora.gipcd' on 'nodel' succeeded
CRS-2676: Start of 'ora.mdnsd' on 'nodel' succeeded
CRS-2672: Attempting to start 'ora.gnpd' on 'nodel'
CRS-2676: Start of 'ora.gnpd' on 'nodel' succeeded
CRS-2672: Attempting to start 'ora.cssdmonitor' on 'nodel'
CRS-2676: Start of 'ora.cssdmonitor' on 'nodel' succeeded
CRS-2672: Attempting to start 'ora.cssd' on 'nodel'
CRS-2676: Start of 'ora.cssd' on 'nodel' succeeded
CRS-2672: Attempting to start 'ora.diskmon' on 'nodel'
CRS-2676: Start of 'ora.diskmon' on 'nodel' succeeded
CRS-2672: Attempting to start 'ora.ctssd' on 'nodel'
CRS-2676: Start of 'ora.ctssd' on 'nodel' succeeded

ASM created and started successfully.

DiskGroup OCRVOT_DG created successfully.

clscfg: --install mode specified
Successfully accumulated necessary OCR keys.
Creating OCR keys for user 'root', privgrp 'system'...

Operation successful.

CRS-2672: Attempting to start 'ora.crsd' on 'nodel'
CRS-2676: Start of 'ora.crsd' on 'nodel' succeeded

CRS-4256: Updating the profile

Successful addition of voting disk 4d29385e75cd4f7ebfa0a432938d0313.
Successful addition of voting disk 215481992ded4f6bbf6881afdf19467df.
Successful addition of voting disk a765c830e1c64f92bf8dc0a937fe40daf.
Successfully replaced voting disk group with +OCRVOT_DG.
CRS-4256: Updating the profile
CRS-4266: Voting file(s) successfully replaced

<table>
<thead>
<tr>
<th>STATE</th>
<th>File Universal Id</th>
<th>File Name Disk group</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONLINE</td>
<td>4d293fe775c4d47ebfa0a432938d0313 (/dev/ASM_OCRVOT_disk5)</td>
<td>[OCRVOT_DG]</td>
</tr>
<tr>
<td>ONLINE</td>
<td>2b15481992ded4f6b6f5681af6d19467dfe0a622888023931481848608662141051928514291295246798374066515345408</td>
<td>[OCRVOT_DG]</td>
</tr>
<tr>
<td>ONLINE</td>
<td>a765c830e1c64f292b6c0a3937fe40daf (/dev/ASM_OCRVOT_disk3)</td>
<td>[OCRVOT_DG]</td>
</tr>
</tbody>
</table>

Located 3 voting disk(s).

CRS-2676: Start of 'ora.evmd' on 'node1' succeeded
CRS-2673: Attempting to stop 'ora.ctssd' on 'node1'
CRS-2676: Start of 'ora.cssd' on 'node1' succeeded
CRS-2673: Attempting to stop 'ora.cssdmonitor' on 'node1'
CRS-2677: Stop of 'ora.cssdmonitor' on 'node1' succeeded
CRS-2676: Stop of 'ora.crsd' on 'node1' succeeded
CRS-2677: Stop of 'ora.asm' on 'node1' succeeded
CRS-2677: Stop of 'ora.asm' on 'node1' succeeded
CRS-2676: Stop of 'ora.asm' on 'node1' succeeded
CRS-2677: Stop of 'ora.asm' on 'node1' succeeded
CRS-2672: Attempting to start 'ora.asm' on 'node1'
CRS-2677: Stop of 'ora.asm' on 'node1' succeeded
CRS-2676: Stop of 'ora.asm' on 'node1' succeeded
CRS-2677: Stop of 'ora.asm' on 'node1' succeeded
CRS-2672: Attempting to start 'ora.asm' on 'node1'
CRS-2677: Stop of 'ora.asm' on 'node1' succeeded

CRS-2676: Start of 'ora.diskmon' on 'node1' succeeded
CRS-2672: Attempting to start 'ora.diskmon' on 'node1'
CRS-2676: Start of 'ora.diskmon' on 'node1' succeeded
CRS-2677: Stop of 'ora.diskmon' on 'node1' succeeded
CRS-2672: Attempting to start 'ora.diskmon' on 'node1'
CRS-2677: Stop of 'ora.diskmon' on 'node1' succeeded

CRS-2676: Start of 'ora.tap' on 'node1' succeeded
CRS-2672: Attempting to start 'ora.tap' on 'node1'
CRS-2676: Start of 'ora.tap' on 'node1' succeeded
CRS-2677: Stop of 'ora.tap' on 'node1' succeeded
CRS-2672: Attempting to start 'ora.tap' on 'node1'
CRS-2677: Stop of 'ora.tap' on 'node1' succeeded
CRS-2676: Start of 'ora.tap' on 'node1' succeeded
CRS-2672: Attempting to start 'ora.tap' on 'node1'
CRS-2677: Stop of 'ora.tap' on 'node1' succeeded
CRS-2676: Start of 'ora.tap' on 'node1' succeeded
CRS-2672: Attempting to start 'ora.tap' on 'node1'
CRS-2677: Stop of 'ora.tap' on 'node1' succeeded

Checking swap space: must be greater than 500 MB.  Actual 1024 MB  Passed

The following environment variables are set as:

```
ORACLE_OWNER= grid
ORACLE_HOME= /oracle/grid/app/11.2.0/grid
```

On demo2, as root user:

```
{demo2:root})/ # /oracle/grid/app/11.2.0/grid/root.sh
Running Oracle 11g root.sh script...

The following environment variables are set as:

ORACLE_OWNER= grid
ORACLE_HOME= /oracle/grid/app/11.2.0/grid
```
Once the root.sh is executed on both nodes, click on the OK button and then on the Close button.
22 On `demo2`, as root user, check that the Grid Infrastructure is installed and running. The target must be on line.

```
{demo2:root}@/oracle/app/oraInventory/logs # su - grid
$ id
uid=1100(grid) gid=1000(oinstall)
groups=1100(asmadmin),1200(dba),1201(oper),1300(asmdba),1301(asmoper)
$ cd $ORACLE_HOME/bin
$ pwd
/oracle/grid/app/11.2.0/grid/bin
$ ./cluvfy stage -post crsinst -n all > /tmp/cluvfy-cluster.log
$ cat /tmp/cluvfy-cluster.log
```

Performing post-checks for cluster services setup
Checking node reachability...
Check: Node reachability from node "demo1"

```
Destination Node          Reachable?
-------------------------  ------------
          demo2                     yes
          demo1                     yes
```

Result: Node reachability check passed from node "demo1"

Checking user equivalence...

23 From `demo1`, as grid user, verify the installation by running the cluvfy utility tool. You can have the output displayed on your screen or redirect the output in a file.
Check: User equivalence for user "grid"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: User equivalence check passed for user "grid"

Checking time zone consistency...

Time zone consistency check passed.

Checking Cluster manager integrity...

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>running</td>
</tr>
<tr>
<td>demo2</td>
<td>running</td>
</tr>
</tbody>
</table>

Oracle Cluster Synchronization Services appear to be online.
Cluster manager integrity check passed

Check default user file creation mask

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Available</th>
<th>Required</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>022</td>
<td>0022</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>022</td>
<td>0022</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Default user file creation mask check passed

Checking cluster integrity...

<table>
<thead>
<tr>
<th>Node Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
</tr>
<tr>
<td>demo2</td>
</tr>
</tbody>
</table>

Cluster integrity check passed

Checking OCR integrity...

Checking the absence of a non-clustered configuration...
All nodes free of non-clustered, local-only configurations

Checking OCR config file "/etc/oracle/ocr.loc"...
OCR config file "/etc/oracle/ocr.loc" check successful

Checking OCR location "/gpfs1/storage/ocr1"...
Check for OCR location "/gpfs1/storage/ocr1" successful

Checking OCR location "/gpfs2/storage/ocr2"...
Check for OCR location "/gpfs2/storage/ocr2" successful

Checking OCR location "/gpfs3/storage/ocr3"...
Check for OCR location "/gpfs3/storage/ocr3" successful

WARNING:
This check does not verify the integrity of the OCR contents. Execute 'ocrcheck' as a privileged user to verify the contents of OCR.
OCR integrity check passed

Checking CRS integrity...
The Oracle clusterware is healthy on node "demo1"
The Oracle clusterware is healthy on node "demo2"
CRS integrity check passed

Checking node application existence...

Checking existence of VIP node application

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Required</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>yes</td>
<td>online</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>yes</td>
<td>online</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Check passed.
Checking existence of ONS node application

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Required</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>no</td>
<td>online</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>no</td>
<td>online</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Check passed.

Checking existence of GSD node application

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Required</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>no</td>
<td>does not exist</td>
<td>ignored</td>
</tr>
<tr>
<td>demo2</td>
<td>no</td>
<td>does not exist</td>
<td>ignored</td>
</tr>
</tbody>
</table>

Result: Check ignored.

Checking existence of EONS node application

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Required</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>no</td>
<td>online</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>no</td>
<td>online</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Check passed.

Checking existence of NETWORK node application

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Required</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>no</td>
<td>online</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>no</td>
<td>online</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: Check passed.

Checking Single Client Access Name (SCAN)...

<table>
<thead>
<tr>
<th>SCAN VIP name</th>
<th>Node</th>
<th>Running?</th>
<th>ListenerName</th>
<th>Port</th>
<th>Running?</th>
</tr>
</thead>
<tbody>
<tr>
<td>clusterha</td>
<td>demo2</td>
<td>true</td>
<td>LISTENER</td>
<td>1521</td>
<td>true</td>
</tr>
</tbody>
</table>

Checking name resolution setup for "clusterha"

<table>
<thead>
<tr>
<th>SCAN Name</th>
<th>IP Address</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>clusterha</td>
<td>10.3.25.171</td>
<td>passed</td>
<td></td>
</tr>
<tr>
<td>clusterha</td>
<td>10.3.25.172</td>
<td>passed</td>
<td></td>
</tr>
<tr>
<td>clusterha</td>
<td>10.3.25.173</td>
<td>passed</td>
<td></td>
</tr>
</tbody>
</table>

Verification of SCAN VIP and Listener setup passed

Checking Oracle Cluster Voting Disk configuration...

Oracle Cluster Voting Disk configuration check passed

Checking to make sure user "grid" is not in "root" group

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Status</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>does not exist</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>does not exist</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: User "grid" is not part of "root" group. Check passed

Checking if Clusterware is installed on all nodes...

Check of Clusterware install passed

Checking if CTSS Resource is running on all nodes...

Check: CTSS Resource running on all nodes

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>passed</td>
</tr>
<tr>
<td>demo2</td>
<td>passed</td>
</tr>
</tbody>
</table>

Result: CTSS resource check passed

Querying CTSS for time offset on all nodes...

Result: Query of CTSS for time offset passed

Check CTSS state started...

Check: CTSS state
Node Name                            State
---------------------------------------
demo1                                  Observer
demo2                                  Observer

CTSS is in Observer state. Switching over to clock synchronization checks using NTP

Starting Clock synchronization checks using Network Time Protocol (NTP)...
NTP Configuration file check started...
The NTP configuration file "/etc/ntp.conf" is available on all nodes
NTP Configuration file check passed
Checking daemon liveness...
Check: Liveness for "xntpd"

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Running?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>yes</td>
</tr>
<tr>
<td>demo2</td>
<td>yes</td>
</tr>
</tbody>
</table>

Result: Liveness check passed for "xntpd"

Checking NTP daemon command line for slewing option "-x"
Check: NTP daemon command line

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Slewing Option Set?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>yes</td>
</tr>
<tr>
<td>demo2</td>
<td>yes</td>
</tr>
</tbody>
</table>

Result: NTP daemon slewing option check passed

Checking NTP daemon's boot time configuration, in file "/etc/rc.tcpip", for slewing option "-x"

Check: NTP daemon's boot time configuration

<table>
<thead>
<tr>
<th>Node Name</th>
<th>Slewing Option Set?</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo1</td>
<td>yes</td>
</tr>
<tr>
<td>demo2</td>
<td>yes</td>
</tr>
</tbody>
</table>

Result: NTP daemon's boot time configuration check for slewing option passed

Result: Clock synchronization check using Network Time Protocol (NTP) passed

Oracle Cluster Time Synchronization Services check passed

Post-check for cluster services setup was successful.

Update the .bash_profile for grid user on each node as follow:

```
{demol:root}/tmp # su - grid
$ id
uid=1100(grid) gid=1000(oinstall)
groups=1100(asmadmin),1200(dba),1300(asmdba),1301(asmoper)
$ cat .profile
PATH=/script:/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:
export PATH
if [ -s "$MAIL" ]
  # This is at Shell startup. In normal
  then echo "$MAILMSG"
  # operation, the Shell checks
fi
  # periodically.
export MANPATH=$MANPATH:/usr/local/man
export DSH_NODE_LIST=/tmp/node_list
export DSH_NODE_RSH=/usr/bin/ssh
export DSH_NODE_RCP=/usr/bin/scp
AIXTHREAD_SCOPE=5; export AIXTHREAD_SCOPE
TMP=/oracle/tmp; export TMP
TMPDIR=/oracle/tmp; export TMPDIR
TEMP=/oracle/tmp; export TEMP
```
ORACLE_HOSTNAME=demo1 ; export ORACLE_HOSTNAME
ORA_CRS_HOME=/oracle/app/11.2.0/grid; export ORA_CRS_HOME
ORACLE_BASE=/oracle/app/grid ; export ORACLE_BASE
ORACLE_HOME=$ORA_CRS_HOME ; export ORACLE_HOME
PATH=$ORACLE_HOME/bin:$PATH ; export PATH

set -o vi

if [ -t 0 ]; then
    stty intr ^C
fi

umask 022
DISPLAY=demo1:1; export DISPLAY

$
### 6.5.1 Check Oracle Clusterware resources

Command crsctl is a versatile command to config or display configuration.

```
$ crsctl
Usage: crsctl <command> <object> [<options>]

command:
enable|disable|config|start|stop|relocate|replace|stat|add|delete|modify|getperm|
setperm|check|set|get|unset|debug|lsmodules|query|pin|unpin

For complete usage, use:
crsctl [-h|--help]
For detailed help on each command and object and its options use:
crsctl <command> <object> -h e.g. crsctl relocate resource -h
```

1 On demo1, as grid user, you may use the following commands to display the configuration.

To display the local Resources and Cluster Resources:

```
$ crsctl status resource -t

<table>
<thead>
<tr>
<th>NAME</th>
<th>TARGET</th>
<th>STATE</th>
<th>SERVER</th>
<th>STATE DETAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local Resources</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ora.LISTENER.lsnr</td>
<td>ONLINE</td>
<td>OFFLINE</td>
<td>demo1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.LISTENER_RDBMS.lsnr</td>
<td>ONLINE</td>
<td>OFFLINE</td>
<td>demo1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.asm</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>demo1</td>
<td>Started</td>
</tr>
<tr>
<td></td>
<td>OFFLINE</td>
<td>OFFLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.eons</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>demo1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.qsd</td>
<td>OFFLINE</td>
<td>OFFLINE</td>
<td>demo1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>OFFLINE</td>
<td>OFFLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.net1.network</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>demo1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.ons</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>demo1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>Cluster Resources</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ora.LISTENER_SCAN1.lsnr</td>
<td>1</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.LISTENER_SCAN2.lsnr</td>
<td>1</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.LISTENER_SCAN3.lsnr</td>
<td>1</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.demo1.vip</td>
<td>1</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.demo2.vip</td>
<td>1</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.oc4j</td>
<td>1</td>
<td>OFFLINE</td>
<td></td>
<td>Open</td>
</tr>
<tr>
<td>ora.ora11.db</td>
<td>1</td>
<td>ONLINE</td>
<td>demo1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>OFFLINE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ora.scan1.vip</td>
<td>1</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
<tr>
<td>ora.scan2.vip</td>
<td>1</td>
<td>ONLINE</td>
<td>demo2</td>
<td></td>
</tr>
</tbody>
</table>
```
ora.scan3.vip

To display status of the cluster resources

$ crsctl status resource
$ id
uid=1100(grid) gid=1000(oinstall) groups=1100(asmadmin),1200(dba),1300(asmdba),1
301(asmoper)
$ crsctl status resource
NAME=ora.LISTENER.lsnr
TYPE=ora.listener.type
TARGET=ONLINE , ONLINE
STATE=ONLINE on demo2
NAME=ora.LISTENER_RDBMS.lsnr
TYPE=ora.listener.type
TARGET=ONLINE , ONLINE
STATE=OFFLINE, ONLINE on demo2
NAME=ora.LISTENER_SCAN1.lsnr
TYPE=ora.scan_listener.type
TARGET=ONLINE
STATE=ONLINE on demo2
NAME=ora.LISTENER_SCAN2.lsnr
TYPE=ora.scan_listener.type
TARGET=ONLINE
STATE=ONLINE on demo2
NAME=ora.LISTENER_SCAN3.lsnr
TYPE=ora.scan_listener.type
TARGET=ONLINE
STATE=ONLINE on demo2
NAME=ora.asm
TYPE=ora.asm.type
TARGET=ONLINE , OFFLINE
STATE=ONLINE on demo1, OFFLINE
NAME=ora.demo1.vip
TYPE=ora.cluster_vip_net1.type
TARGET=ONLINE
STATE=INTERMEDIATE on demo2
NAME=ora.demo2.vip
TYPE=ora.cluster_vip_net1.type
TARGET=ONLINE
STATE=ONLINE on demo2
NAME=ora.eons
TYPE=ora.eons.type
TARGET=ONLINE , ONLINE
STATE=ONLINE on demo1, ONLINE on demo2
NAME=ora.gsd
TYPE=ora.gsd.type
TARGET=OFFLINE, OFFLINE
STATE=OFFLINE, OFFLINE
NAME=ora.net1.network
TYPE=ora.network.type
$ id
TARGET=ONLINE , ONLINE
STATE=ONLINE on demo1, ONLINE on demo2
NAME=ora.oc4j
TYPE=ora.oc4j.type
TARGET=OFFLINE
STATE=OFFLINE
NAME=ora.ons
TYPE=ora.ons.type
TARGET=ONLINE, ONLINE
STATE=ONLINE on demo1, ONLINE on demo2

NAME=ora.ora11.db
TYPE=ora.database.type
TARGET=ONLINE, OFFLINE
STATE=ONLINE on demo1, OFFLINE

NAME=ora.scan1.vip
TYPE=ora.scan_vip.type
TARGET=ONLINE
STATE=ONLINE on demo1, ONLINE on demo2

NAME=ora.scan2.vip
TYPE=ora.scan_vip.type
TARGET=ONLINE
STATE=ONLINE on demo1, ONLINE on demo2

NAME=ora.scan3.vip
TYPE=ora.scan_vip.type
TARGET=ONLINE
STATE=ONLINE on demo1, ONLINE on demo2

To display the state of the cluster:

```
$ crs_stat -t
Name Type Target State Host
ora....ER.1snr ora....er.type ONLINE ONLINE demo2
ora....MS.1snr ora....er.type ONLINE ONLINE demo2
ora....N1.1snr ora....er.type ONLINE ONLINE demo2
ora....N2.1snr ora....er.type ONLINE ONLINE demo2
ora....N3.1snr ora....er.type ONLINE ONLINE demo2
ora.asm ora.asm.type ONLINE ONLINE demo1
ora....SM1.asm application ONLINE ONLINE demo1
ora....O1.1snr application ONLINE OFFLINE demo1
ora....O1.1snr application ONLINE OFFLINE demo1
ora.demo1.ons application ONLINE OFFLINE demo1
ora.demo1.vip ora....ti.type ONLINE ONLINE demo2
ora....SM2.asm application OFFLINE OFFLINE demo2
ora....O2.1snr application ONLINE ONLINE demo2
ora....O2.1snr application ONLINE ONLINE demo2
ora.demo2.gsd application OFFLINE OFFLINE demo2
ora.demo2.ons application ONLINE ONLINE demo2
ora.demo2.vip ora....ti.type ONLINE ONLINE demo2
ora.eons ora.eons.type ONLINE ONLINE demo1
ora.gsd ora.gsd.type OFFLINE OFFLINE
ora....network ora....rk.type ONLINE ONLINE demo1
ora.oc4j ora.oc4j.type OFFLINE OFFLINE
ora.ons ora.ons.type ONLINE ONLINE demo1
ora.ora11.db ora....se.type ONLINE ONLINE demo1
ora.scan1.vip ora....ip.type ONLINE ONLINE demo2
ora.scan2.vip ora....ip.type ONLINE ONLINE demo2
ora.scan3.vip ora....ip.type ONLINE ONLINE demo2
```

$ To check registered network interfaces and subnets

```
{demo1:root}@/ # su - grid
$ oifcfg getif
en0  10.3.25.0  global  public
en2  30.30.30.0  global  cluster_interconnect
```

© Copyright 2011
On each node, create following script, named for example crsstat1 in $ORA_CRS_HOME/bin:
Include the following content in the file crsstat1.

```bash
#!/usr/bin/ksh

# Sample 10g CRS resource status query script

# Requirements:
# - $ORA_CRS_HOME should be set in your environment

AWK=/usr/bin/awk    # if not available use /usr/bin/awk

# Table header:
printf "%-45s %-10s %-18s
", "HA Resource", "Target", "State";

BEGIN {
    printf "%-45s %-10s %-18s
", "--------", "-----", "------";
}

# Table body:
#   - The argument, $RSC_KEY, is optional and if passed to the script, will
#     limit the output to HA resources whose names match $RSC_KEY.
#   - $ORA_CRS_HOME is optional and if passed to the script, will
#     limit the output to HA resources whose names match $RSC_KEY.

crsstat $QSTAT | $AWK

# Examples:
# $1~/STATE/ && state == 2 {appstate = $2; state=3;}

# Table footer:
#   - The argument, $RSC_KEY, is optional and if passed to the script, will
#     limit the output to HA resources whose names match $RSC_KEY.
#   - $ORA_CRS_HOME is optional and if passed to the script, will
#     limit the output to HA resources whose names match $RSC_KEY.

# End Shell Script
```

and change attributes to execute the file

Now as grid user from any node, execute the script:

```
# chmod +x crsstat1
#/ crsstat1
```

<table>
<thead>
<tr>
<th>HA Resource</th>
<th>Target</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>ora.LISTENER.lsnr</td>
<td>ONLINE</td>
<td>ONLINE on demo2</td>
</tr>
<tr>
<td>ora.LISTENER_RDMS.lsnr</td>
<td>ONLINE</td>
<td>ONLINE on demo2</td>
</tr>
<tr>
<td>ora.LISTENER_SCAN1.lsnr</td>
<td>ONLINE</td>
<td>ONLINE on demo2</td>
</tr>
<tr>
<td>ora.LISTENER_SCAN2.lsnr</td>
<td>ONLINE</td>
<td>ONLINE on demo2</td>
</tr>
<tr>
<td>ora.LISTENER_SCAN3.lsnr</td>
<td>ONLINE</td>
<td>ONLINE on demo2</td>
</tr>
<tr>
<td>ora.asm</td>
<td>ONLINE</td>
<td>ONLINE on demo1</td>
</tr>
<tr>
<td>ora.demo1.ASM1.asm</td>
<td>ONLINE</td>
<td>ONLINE on demo1</td>
</tr>
<tr>
<td>ora.demo1.LISTENER_DEMO1.lsnr</td>
<td>ONLINE</td>
<td>OFFLINE</td>
</tr>
<tr>
<td>ora.demo1.LISTENER_RDMS_DEMO1.lsnr</td>
<td>ONLINE</td>
<td>OFFLINE</td>
</tr>
<tr>
<td>ora.demo1.gsd</td>
<td>OFFLINE</td>
<td>OFFLINE</td>
</tr>
<tr>
<td>ora.demo1.ons</td>
<td>ONLINE</td>
<td>ONLINE on demo1</td>
</tr>
</tbody>
</table>
**List all VIP resources:**

```
$ ./.crsstat1 | grep vip
ora.demo1.vip       ONLINE  ONLINE on demo2
ora.demo2.vip       ONLINE  ONLINE on demo2
ora.scan1.vip       ONLINE  ONLINE on demo2
ora.scan2.vip       ONLINE  ONLINE on demo2
ora.scan3.vip       ONLINE  ONLINE on demo2
$```

**List all listener resources:**

```
$ ./.crsstat1 | grep lsnr
ora.LISTENER.lsnr    ONLINE  ONLINE on demo2
ora.LISTENER_RDBMS.lsnr ONLINE  ONLINE on demo2
ora.LISTENER_SCAN1.lsnr ONLINE  ONLINE on demo2
ora.LISTENER_SCAN2.lsnr ONLINE  ONLINE on demo2
ora.LISTENER_SCAN3.lsnr ONLINE  ONLINE on demo2
ora.demo1.LISTENER_DEMO1.lsnr ONLINE  OFFLINE
ora.demo1.LISTENER_RDBMS_DEMO1.lsnr ONLINE  OFFLINE
ora.demo2.LISTENER_DEMO2.lsnr ONLINE  ONLINE on demo2
ora.demo2.LISTENER_RDBMS_DEMO2.lsnr ONLINE  ONLINE on demo2
$```

**Check OCR Disks:**

```
## ocrcheck
Status of Oracle Cluster Registry is as follows :
  Version : 3
  Total space (kbytes) : 262120
  Used space (kbytes) : 2536
  Available space (kbytes) : 259584
  ID : 1229607430
  Device/FileName : /gpfs1/storage/ocr1
                  Device/File integrity check succeeded
  Device/FileName : /gpfs2/storage/ocr2
                  Device/File integrity check succeeded
  Device/FileName : /gpfs3/storage/ocr3
                  Device/File integrity check succeeded
                  Device/File not configured
                  Device/File not configured
  Cluster registry integrity check succeeded
  Logical corruption integrity check succeeded
##```

**Check Voting Disks:**

```
$ crsctl query css votedisk
## STATE   File Universal Id
```

---

**IBM Oracle Center**

© Copyright 2011
Command to stop the cluster:

```bash
# crsctl stop

Usage:
   crsctl stop resource {<resName> [...]|-w <filter>]|-all} [-n <server>] [-k <cid>] [-d <did>]
   [-env "env1=val1,env2=val2,..."] [-f] [-i]
   Stop designated resources

where
   resName [...] One or more blank-separated resource names
   -w Resource filter
   -all All resources
   -n Server name
   -k Resource cardinality ID
   -d Resource degree ID
   -env Stop command environment
   -f Force option
   -i Fail if request cannot be processed immediately

   crsctl stop crs [-f]
   Stop OHAS on this server

where
   -f Force option

   crsctl stop cluster [[-all]|[-n <server> [...]]] [-f]
   Stop CRS stack

where
   Default Stop local server
   -all Stop all servers
   -n Stop named servers
   server [...] One or more blank-separated server names
   -f Force option
```

./crsctl stop cluster -all

Command to start the cluster -all

```bash
# crsctl start

Usage:
   crsctl start resource {<resName> [...]|-w <filter>]|-all} [-n <server>] [-k <cid>] [-d <did>]
   [-env "env1=val1,env2=val2,..."] [-f] [-i]
   Start designated resources

where
   resName [...] One or more blank-separated resource names
   -w Resource filter
   -all All resources
   -n Server name
   -k Resource cardinality ID
   -d Resource degree ID
   -env Start command environment
   -f Force option
   -i Fail if request cannot be processed immediately

   crsctl start crs [-excl|-nowait]
   Start OHAS on this server

where
   -excl Start Oracle Clusterware in exclusive mode
   -nowait Do not wait for OHAS to start

   crsctl start cluster [[-all]|[-n <server> [...]]]
   Start CRS stack
```
where
- Default Start local server
- -all Start all servers
- -n Start named servers
- server [...] One or more blank-separated server names

```
$ /crsctl start cluster -all
```

If there is an error, you can display the crs log file.

```
$ cd /oracle/app/11.2.0/grid/log/demo2/crsd
$ ls
crsd.l01  crsd.log  crsd.trc  crsdOUT.log
$ cat crsd.log
```

```text
2011-08-31 11:42:03.262: [GIPCXCPT][1] gipcmodClsaAuthStart: failuring during clsaauthmsg ret clsaretOSD (8), endp 1110a2010 [0000000000000023] { gipcEndpoint : localAddr 'clsc://(ADDRESS=(PROTOCOL=ipc)(KEY=)(GIPCID=0202b236-1b7ff506-23789760))', remoteAddr 'clsc://(ADDRESS=(PROTOCOL=ipc)(KEY=OCSSD_LL_demo2_)(GIPCID=1b7ff506-0202b236-3146204))', numPend 5, numReady 0, numDone 2, numDead 0, numTransfer 0, objFlags 0x3001dc, pidPeer 0, flags 0x2ca712, usrFlags 0x34000 }
```

```text
2011-08-31 11:42:03.262: [GIPCXCPT][1] gipcmodClsaAuthStart: failuring during clsaauthmsg ret clsaretOSD (8), endp 1110a2010 [0000000000000023] { gipcEndpoint : localAddr 'clsc://(ADDRESS=(PROTOCOL=ipc)(KEY=)(GIPCID=89b6971e-b0ad51ca-23789760))', remoteAddr 'clsc://(ADDRESS=(PROTOCOL=ipc)(KEY=OCSSD_LL_demo2_)(GIPCID=b0ad51ca-89b6971e-3146204))', numPend 5, numReady 0, numDone 2, numDead 0, numTransfer 0, objFlags 0x3001dc, pidPeer 0, flags 0x2ca712, usrFlags 0x34000 }
```

```text
# srvctl stop listener -l LISTENER
# srvctl start listener -l LISTENER
```