Managing & Troubleshooting Cluster - 360 degrees
This views/content in this slides are those of the author and do not necessarily reflect that of Oracle Corporation and/or its affiliates/subsidiaries. The material in this document is for informational purposes only and is published with no guarantee or warranty, express or implied..

This material should not be reproduced or used without the authors' written permission.
Know your presenter

Syed Jaffer Hussain
Database Support Manager
**Over 20 years IT experience**
**14+ years** as an Oracle DBA
Oracle **ACE Director**
Oracle 10g Certified Master(OCM)
Oracle 10g RAC Certified Expert
OCP v8i,9i,10g & 11g
**ITIL v3 Foundation** Certified
Oracle Database 12c beta tester

Twitter: @sjaffarhussain
http://jaffardba.blogspot.com
Know your presenter

Technologist of the Year, DBA 2011

Know your presenter

Co-authored

Oracle 11g R1/R2 Real Application Clusters Essentials
Design, implement, and support complex Oracle 11g RAC environments for real-world deployments

Syed Jaffer Hussain, Tanio Farooq, Riaz Shamsudeen, and Kai Yu

Expert Oracle RAC 12c
Apress

ORACLE Certified Professional
ORACLE Certified 10g Certified Master
ORACLE Certified Expert
10g RAC Administrator

ORACLE ACE Director

ITIL

Presented by: Syed Jaffer Hussain
AIOUG – SANGAM13
Slide # 5
A famous personality in Oracle community once compared an Oracle DBA with a Pilot!!!
Foreword

Take off
Foreword

Safe Take off

Safe landing
Foreword

Successful database startup
Foreword

Successful database startup

Successful database shutdown
Foreword

Successful database startup

Successful database shutdown
Flight Cockpit
Flight Cockpit
Tools & Utilities

Cluvfy
OSWatcher
RDA
RACcheck
ProWatcher
Cluster Health Monitor (CHM)
Systemstate
Hanganalysis
OraTop
Diagcollection.sh
Debugging
Tools & Utilities

- Cluvfy
- OSWatcher
- RDA
- ProWatcher
- RACcheck
- Systemstate
- Cluster Health Monitor (CHM)
- Hanganalysis
- Diagcollection.sh
- OraTop
- Debugging
Foreword

Any deployment (installation/upgrade) is a single shot
Foreword

Where as
Foreword

Administration and Troubleshooting is never ending
What is covered

- Key enhancements in Clusterware
- Oracle Clusterware software stack
- Clusterware start-up sequence
- Cluster logs & directory tree structure
- Analyzing Cluster logs
- Cluster logs rotation/retention policy
- Troubleshooting Cluster start-up common failures
- Debugging/Tracing Clusterware components
- Tools & Utilities – how to pick the right one
- References
- Q & A
Key new features at a glance

Oracle Grid Infrastructure

- Clusterware and ASM binaries are installed together in a single home directory: Grid Home (GI)
Key new features at a glance

- OCR can also be stored in ASM diskgroup
- Upto five (05) copies of OCR files
Oracle Local Registry – OLR

- Independent OLR copy for each node
- Not shared between nodes
- Stores local node configuration details required by OHASD
- Configured upon installation/upgrade
- Facilitates the CRS startup process when OCR/VD stored in ASM
- ocrcheck -local
- Located under $GRID_HOME/cdata/hostname/hostname.olr
- $GRID_HOME/bin/ocrconfig -local -manualbakup/restore
Key new features at a glance

- Voting Disk (files) can also be stored in ASM diskgroup
- VD copies can’t resides in multiple ASM diskgroups
Key new features at a glance

Clusterized cluster-aware commands

- `crsctl start cluster --all` -- starts cluster on all nodes
- `crsctl stop cluster --all` -- stops cluster on all nodes
- `crsctl check cluster --all` -- verify cluster health on all nodes
Key new features at a glance

- Complete redesign of Cluster daemon in 11gR2, ohasd introduction
- Replaces RACG layer with Agents
- New Services
  - Grid Plug and Play (GPnP)
  - Cluster Time Synchronization Service (CTSS)
  - Grid Name Service
- Cluster can be started in exclusive mode for maintenance purpose
  
  . /crsctl start res -t -init ---- list all Clusterware daemon resources
Oracle 11gR2 Clusterware Software Stacks

1. Cluster Ready Services daemon (CRSD)

The upper stack
Oracle 11gR2 Clusterware Software Stack

1. Cluster Ready Services daemon (CRS)
2. Oracle High Availability Services daemon (OHASD)

The lower stack
Oracle 11gR2 Clusterware Software Stack

1. Cluster Ready Services daemon (CRS)
   - CRS
   - cssdagent
   - CTSS
   - ONS
   - orarootagent
   - CSS
   - ASM
   - EVM
   - oraagent

2. Oracle High Availability Services daemon (OHASD)

The upper stack
Oracle 11gR2 Clusterware Software Stack

1. Cluster Ready Services Stack (CRS)
   - CRS
   - cssdagent
   - CTSS
   - ONS
   - oraagent
   - orarootagent

2. Oracle High Availability Service Stack (OHASD)
   - appagent
   - GPNPD
   - mDNS
   - scriptagent
   - oragen
   - orasmond
   - orarootagent

The lower stack
Oracle High Availability Services daemon (ohasd)

**HPUX**

```
h1:3:respawn:/sbin/init.d/init.ohasd run >/dev/null 2>&1 </dev/null
```

**Linux**

```
h1:3:respawn:/etc/init.d/init.ohasd run >/dev/null 2>&1 </dev/null
```

**Oracle/RedHat Linux 6**

```
/etc/init, /etc/init.d/init.ohasd run
```
Clusterware Startup Sequence
Clusterware Startup Sequence

Presented by: Syed Jaffer Hussain

AIOUG – SANGAM13

Slide # 32
Clusterware Startup Sequence

Presented by: Syed Jaffer Hussain
Clusterware Startup Sequence
Clusterware Startup Sequence

Level 0
- cssdmonitor
- OHASD oragent

Level 1
- mDNSD
- GIPCD
- EVMD

Level 2
- CRSD
- CTSSD
- Diskmon

Level 3
- CRSD orarootagent

Level 4
- Network sources
- SCANIP
- Node VIP
- ACF Registry
- GNSVIP

Process on the High Availability Stack:
- GPNPD
- CTSSD

Process on the Cluster Ready Service Stack:
- Services

Resource managed by Cluster Ready Service
- eONS
- ONS
- GNS
- GSD
Clusterware Startup Sequence

Oracle High Availability Service Daemon (ohasd)

Cluster Synchronization Service Daemon (cssd)

Event Manager Daemon (evmd)

Cluster Ready Service Daemon (crsd)

GPnP / OLR

Voting disk

OCR
GRID_HOME/log/host_name/

alert<hostname>.log

CRSD == crsd.log

CSSD == ocssd.log

OHASD == ohsad.log

EVMD == evmd.log

Operating System Logs
Cluster logs & directory structure

GRID_HOME/log/host_name/alterrac1.log

- Writes Clusterware stack all important alert messages
- Posts Cluster stack start/stop messages
- nodes eviction messages
- OLR events
- Voting and OCR disk related messages
- Active nodes list
- Preferably the first log file to review upon any cluster issues
Cluster logs & directory structure

GRID_HOME/log/host_name/alterrac1.log

[ohasd(10937)]CRS-1301:Oracle High Availability Service started on node rac1.
[cssd(19712)]CRS-1713:CSSD daemon is started in exclusive mode
[ohasd(19506)]CRS-2765:Resource 'ora.cssdmonitor' has failed on server ‘rac1’

[cssd(19951)]CRS-1605:CSSD voting file is online: /dev/rdsk/c0t13d2; details in /u00/app/11.2.0/grid_1/log/rac1/cssd/ocssd.log.

[cssd(7945)]CRS-1714:Unable to discover any voting files, retrying discovery in 15 seconds; Details at (：CSSNM00070:) in /u00/app/11.2.0/grid/log/usdbt01/cssd/ocssd.log

[/u00/app/11.2.0/grid_1/bin/oraagent.bin(19914)]CRS-5815:Agent '/u00/app/11.2.0/grid_1/bin/oraagent_oracle' could not find any base
type entry points for type 'ora.daemon.type'. Details at (：CRSAGF00108:) {0:9:2} in /u00/app/11.2.0/grid_1/log/usdbp01/agent/ohasd/oraagent_oracle/oraagent_oracle.log.

[cssd(19951)]CRS-1601:CSSD Reconfiguration complete. Active nodes are rac1,rac2.
[cssd(3726)]CRS-1625:Node rac2, number 2, was manually shut down
[cssd(3726)]CRS-1612:Network communication with node rac2 (2) missing for 50% of timeout interval. Removal of this node from cluster in 14.145 seconds
Cluster logs & directory structure

GRID_HOME/log/host_name/crsd.log

- Cluster Ready Services daemon (CRSD) maintains this log
- Any cluster resources Start/stop/failure occurrence are written
- Review the log when you have resources failures, unable to start etc

- 2013-04-05 02:05:05.379: [ CRSPE][46] {13:24208:51894} Resource ora.prddb.db has been updated in the PE data model:6000000004f82610
- 2013-04-05 02:08:15.371: [ CRSPE][46] {0:33:627} State change received from rac1 for ora.prddb.db
- 2013-04-05 02:08:22.698: [ CRSPE][46] {0:33:628} CRS-2672: Attempting to start 'ora.prddb.prddb_srv.svc' on 'rac1'
- 2013-04-23 17:32:46.340: [ OCRRAW][1]proprioini: all disks are not OCR/OLR formatted
Cluster logs & directory structure

GRID_HOME/log/host_name/ocssd.log

- Cluster Synchronization daemon (CSSD) maintains this log
- Busiest log file
- Records node inter-communication messages
- Heart beat missing and node eviction messages

2013-04-23 16:14:14.713: [CSSD][6](:CSSNM00070:)clssnmCompleteInitVFDiscovery: Voting file not found. Retrying discovery in 15 seconds

- 2013-04-17 11:33:23.286: [GIPCHALO][7] gipchaLowerProcessNode: bootstrap node considered dead because of idle connection time 600014 ms, node 60000000019645b0 { host 'rac2', haName 'CSS_crs', srcLuid 8c78ad11-53582d91, dstLuid 2f6e4604-5e451051 numInf 1, contigSeq 6712447, lastAck 6697196, lastValidAck 6712447, sendSeq [6697202 : 6697202], createTime 2417887052, sentRegister 1, localMonitor 1, flags 0x808 }

- 2013-04-20 15:13:18.953: [CSSD][54]clssnmSendingThread: sending status msg to all nodes
- 2013-04-20 15:13:18.954: [CSSD][54]clssnmSendingThread: sent 4 status msgs to all nodes

- 2013-04-12 17:35:55.351: [CSSD][49]clssnmvReadDskHeartbeat: Reading DHBs to get the latest info for node rac1, 17 LATSvalid 0 uniqueness 1348227938

[cssd(7335)]CRS-1612:Network communication with node rac2 (02) missing for 50% of timeout interval. Removal of this node from cluster in 14.397 seconds 2013-03-15 17:02:44.964
[cssd(7335)]CRS-1611:Network communication with node rac2 (02) missing for 75% of timeout interval. Removal of this node from cluster in 7.317 seconds 2013-03-15 17:02:50.024
[cssd(7335)]CRS-1610:Network communication with node rac2 (02) missing for 90% of timeout interval. Removal of this node from cluster in
Cluster logs & directory structure

GRID_HOME/log/host_name/ohasd.log

- Oracle High Availability Service (OHASD) maintains this log
- High Availability service messages are written
- Review the log when you have issues whilst running root.sh/rootupgrd.sh
- If the service unable to start or becomes unhealthy due to OLR problems
- Loads default debugging levels

2013-04-17 11:32:47.125: [ default][1] Initializing OLR
2013-04-17 11:32:47.255: [ OCRRAW][1]proprioo: for disk 0 (/u00/app/12.1.0/grid_1/cdata/rac2.olr), id match (1), total id sets, need recover (0), my votes (0), total votes (0), commit_lsn (3118), lsn (3118)
2013-04-17 11:32:47.368: [ default][1] Loading debug levels...
2013-04-17 11:32:47.803: [ clsdmt][13]Creating PID [6401] file for home /u00/app/12.1.0/grid_1/host usdbp10 bin ohasd to /u00/app/12.1.0/grid_1/ohasd/init/
Cluster logs & directory structure

$GRID_HOME/log/<hostname>/

- alert_<hostname>.log
- /ohasd
  - ohasd.log
- /crsd
  - crsd.log
- /cssd
  - cssd.log
- /evmd
  - evmd.log
- /client
  - ocrcheck.log, olsnodes.log, ocrdump.log etc
- /ctssd
  - Crssd.log
- /ract
- /ctssd
  - ctsd.log
- /diskmon
- /cvu
- /admin
- /agent
- /ohasd
- /crsd

ohasd.01, 02.. 10
crsd.01, 02.. 10
cssd.01, 02.. 10

Presented by: Syed Jaffer Hussain
AIOUG – SANGAM13
Slide # 43
Operating System logs

HPUX - /var/adm/syslog/syslog.log

AIX - /bin/errpt –a

Linux - /var/log/messages

Windows - Refer .TXT log files under Application/System log using Windows Event Viewer

Solaris - /var/adm/messages
Managing Clusterware log files manually is not recommended...
They are governed and managed automatically......
Cluster logs rotation/retention policy

- Most Clusterware log files follow the 10x10 rule as part of automatic rotation/retention policy and governed automatically.

- 10 copies of `cssd.log` files with 50M retained and rotated subsequently.

- `ohasd`, `evmd`, `crsd` etc logs also retain 10 copies with 10M size.

- The policy doesn’t apply to the `alterHOSTNAME.log` file.
Troubleshooting Cluster start-up failures

$GRID_HOME/bin/crsctl check crs

CRS-4638: Oracle High Availability Services is online
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
Troubleshooting Cluster start-up failures

```
$GRID_HOME/bin/crsctl check crs
$GRID_HOME/bin/crsctl check cluster
```

CRS-4639: Could not contact Oracle High Availability Services

CRS-4124: Oracle High Availability Services startup failed
CRS-4000: Command Check failed, or completed with errors

OR

CRS-4535: Cannot communicate with Cluster Ready Services
CRS-4530: Communications failure contacting Cluster Synchronization Services daemon
CRS-4534: Cannot communicate with Event Manager
Troubleshooting Cluster start-up failures

Oracle High Availability Service (ohasd) start-up failures – common causes
CRS-4639: Could not contact Oracle High Availability Services
CRS-4124: Oracle High Availability Services startup failed
CRS-4000: Command Check failed, or completed with errors

1. Verify whether the cluster auto start-up is configured or not?
   • crsctl config has
   • /var/opt/oracle/scls_scr/hostname/root or /etc/oracle/scls_scr/hostname/root
   • Verify OS run level
   • Check whether ohasd daemon process is up or not: ps -ef |grep ohasd

2. Verify ohasd auto-start pointer in the /etc/init and /etc/inittab files?
   • h1:3:respawn:/sbin/init.d/init.ohasd run >/dev/null 2>&1 </dev/null

3. Verify the OLR availability, corruption and accessibility on the local node
   • Review the ohasd.log file for more details

4. Verify whether the ohasd agents are up or not – for unhealthy cluster
   • ps -ef |grep oraagent|orarootagent|cssdagent|cssdmonitor
   • Review the ohasd.log file

5. Verify Grid Infrastructure location permission
   • Compare with a good node location
Troubleshooting Cluster start-up failures

Oracle High Availability Service (ohasd) start-up failures – troubleshooting common causes

1. Enable Cluster auto start-up
   - crsctl enable has|crs
   - crsctl start crs/cluster

2. Put the following line in the respective OS files
   - h1:3:respawn:/sbin/init.d/init.ohasd run >/dev/null 2>&1 </dev/null

3. Restore or set permissions on the OLR
   - Restore from the recent OLR backup, if the file is corrupted
   - Reset appropriate permission on the local node for the file

4. Verify whether the ohasd agents are up or not – for unhealthy cluster status
   - Set permission if needed
   - If binaries are corrupted, restore them from a latest backup

5. Reset permissions or restore from the recent backups

6. Additionally, remove/rename the files from the /var/tmp/.oracle, /usr/tmp/.oracle or /tmp/.oracle locations
Cluster Synchronization Service (cssd) start-up failures – common causes

CRS-4530: Communications failure contacting Cluster Synchronization Services daemon:

1. Verify the following:
   - GPnP profile accessibility
   - Voting disk files accessibility
   - Check the underlying network (private network) for any connectivity issues

2. Verify whether the daemon status on the OS
   - ps -ef |grep ocssd.bin
Troubleshooting Cluster start-up failures

Cluster Synchronization Service (cssd) start-up failures – troubleshooting common causes

1. Review the ocssd.log file to diagnose the issue:
   - Review the ocssd.log file if the daemon is able to access the GPnP profile
   - Run, crsctl query css votedisk to verify whether the voting disk files are accessible
   - If Voting disk permissions are lost, reset them
   - Resolve underlying network issues for any heart-beat issues and bring up the interconnect resource:
     ./crsctl start res ora.cluster_interconnect.haip -init

2. Start the process manually
   - Try to start the daemon process manually if it is not up or unhealthy: ./crsctl start res ora.cssd -init
Troubleshooting Cluster start-up failures

Cluster Ready Service (crsd) start-up failures – common causes

CRS-4535: Cannot communicate with Cluster Ready Services:

1. Verify the following:
   - Oracle Cluster Registry (OCR) accessibility
     ./ocrcheck
   - Look for any Grid Home ownership and permission changes
   - Check for the OCR mirror copy issues
   - Verify and validate underlying network (private network)

2. Verify whether the daemon status on the OS
   - ps -ef |grep crsd.bin
   - crsctl stat res -t -init, look for ora.crsd status

3. Verify crsd agents
   - ps -ef |grep oraagent|orarootagent
Troubleshooting Cluster start-up failures

Cluster Ready Service (crsd) start-up failures – troubleshooting common causes

1. Take the following action:
   - Review the crsd.log file
   - Take appropriate steps to resolve ownership, privilege issues on the OCR files.
   - Compare with a good node, and restore the directory
   - ./ocrcheck
   - Verify and validate underlying network (private network)

2. Verify whether the daemon status on the OS
   - Restart the process manually
     ./crsctl start res ora.crsd -init

3. Verify the following:
   - ./crs_stat -t

4. Ensure sufficient free space available under the $GRID_HOME to avoid cluster unhealthy issues.
### Troubleshooting other clusterware process

```
crsctl stat res -t -init
```

<table>
<thead>
<tr>
<th>Name</th>
<th>Target</th>
<th>State</th>
<th>Server</th>
<th>State details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster Resources</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ora.asm</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>rac1</td>
<td>Started,STABLE</td>
</tr>
<tr>
<td>ora.cluster_interconnect.haip</td>
<td>ONLINE</td>
<td>OFFLINE</td>
<td>rac1</td>
<td>STABLE</td>
</tr>
<tr>
<td>ora.crsd</td>
<td>ONLINE</td>
<td>OFFLINE</td>
<td>rac1</td>
<td>STABLE</td>
</tr>
<tr>
<td>ora.cssd</td>
<td>ONLINE</td>
<td>OFFLINE</td>
<td>rac1</td>
<td>STABLE</td>
</tr>
<tr>
<td>ora.cssdmonitor</td>
<td>ONLINE</td>
<td>UNKNOWN</td>
<td>rac1</td>
<td>STABLE</td>
</tr>
<tr>
<td>ora.ctssd</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>rac1</td>
<td>ACTIVE:0,STABLE</td>
</tr>
<tr>
<td>ora.diskmon</td>
<td>OFFLINE</td>
<td>OFFLINE</td>
<td></td>
<td>STABLE</td>
</tr>
<tr>
<td>ora.drivers.acfs</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>rac1</td>
<td>STABLE</td>
</tr>
<tr>
<td>ora.evmd</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>rac1</td>
<td>STABLE</td>
</tr>
<tr>
<td>ora.gipcd</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>rac1</td>
<td>STABLE</td>
</tr>
<tr>
<td>ora.gpnpd</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>rac1</td>
<td>STABLE</td>
</tr>
<tr>
<td>ora.mdnsd</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>rac1</td>
<td>STABLE</td>
</tr>
<tr>
<td>ora.storage</td>
<td>ONLINE</td>
<td>ONLINE</td>
<td>rac1</td>
<td>STABLE</td>
</tr>
</tbody>
</table>
Troubleshooting Cluster start-up failures

Troubleshooting other clusterware process

./crsctl start res ora.cluster_interconnect.haip -init

./crsctl start res ora.cssd -init

The following output will be displayed at your screen:

CRS-2679: Attempting to clean 'ora.cssdmonitor' on 'racl'
CRS-2681: Clean of 'ora.cssdmonitor' on 'racl' succeeded
CRS-2672: Attempting to start 'ora.cssdmonitor' on 'racl'
CRS-2676: Start of 'ora.cssdmonitor' on 'racl' succeeded
CRS-2672: Attempting to start 'ora.cssd' on 'racl'
CRS-2676: Start of 'ora.cssd' on 'racl' succeeded
CRS-2672: Attempting to start 'ora.cluster_interconnect.haip' on 'racl'
CRS-2672: Attempting to start 'ora.crsd' on 'racl'
CRS-2676: Start of 'ora.cluster_interconnect.haip' on 'racl' succeeded
CRS-2676: Start of 'ora.crsd' on 'racl' succeeded
Debugging/Tracing Cluster components

- Flexibility to modify the default tracing/logging levels for any Clusterware main and sub-processes

- Range from 1 – 5, 0 value disables the tracing level

  `ohasd.log` file also writes the default trace levels message when Oracle High Availability service daemon starts up on the local node

- `crsctl get log {css|crs|evm} ALL` – lists existing trace levels for the modules
- `crsctl lsmodule` – list the module details
- `crsctl lsmodule {css|crs|evm}`
Debugging/Tracing Cluster components

Default trace levels:

./crsctl get log css all

Get CSSD Module: BCCM  Log Level: 2
Get CSSD Module: CLSF  Log Level: 0
Get CSSD Module: CLSINET  Log Level: 0
Get CSSD Module: CSSD  Log Level: 2
Get CSSD Module: GIPCBCCM  Log Level: 2
Get CSSD Module: GIPCCM  Log Level: 2
Get CSSD Module: GIPCNGM  Log Level: 2
Get CSSD Module: GIPCNM  Log Level: 2
Get CSSD Module: GPNP  Log Level: 1
Get CSSD Module: OLR  Log Level: 0
Get CSSD Module: SKGFD  Log Level: 0

Default modules:

./crsctl lsmodules
Usage:
  crsctl lsmodules {mdns|gpnp|css|crf|crs|ctss|evm|gipc}
  where
    mdns  multicast Domain Name Server
    gpnp  Grid Plug-n-Play Service
    css  Cluster Synchronization Services
    crf  Cluster Health Monitor
    crs  Cluster Ready Services
    ctss  Cluster Time Synchronization Service
    evm  EventManager
    gipc  Grid Interprocess Communications
The following enable various tracing levels:

```
./crsctl set log crs crsmain=3  
./crsctl set log crs crsmain=3,crsevt=4  

./crsctl set log crs all=5  
./crsctl set log res ora.prddb.db:5
```

The following examples explains how to set tracing levels on the OS:

```
export ORA_CRSDEBUG_ALL=1   --sets debugging level 1 to all modules  
export ORA_CRSDDEBUG_CRS=2   --sets debugging level 2 to CRS module
```
The following disable tracing:

./crsctl set log crs crsmain=0
./crsctl set log res ora.prd.db:0
./crsctl set log res ora.crs:0 -init

The following examples explains how to set tracing levels on the OS:

export ORA_CRSDEBUG_ALL=1    --sets debugging level 1 to all modules
export ORA_CRSDDDEBUG_CRS=2   --sets debugging level 2 to CRS module
Diagcollection.pl:

• Located under $GRID_HOME/bin location

• Is a tool that gathers required Clusterware diagnostic information in a bunch of trace files from various resources: CRS logs, trace & core files, OCR data etc.

• Can collect diagnostic information at different layers and homes: Cluster, Oracle RDBMS, Core, Oracle Base etc

• All the information will be then zipped into a few zip files

• Duration required to gather the information is directly proportional to the levels used

• Upload these files to My Oracle Support for issue investigation
Examples:

`./diagcollection.sh --collect --chmos`

`./diagcollection.sh --collect --chmos --incidenttime <timeperiod> --incidentduration 05:00` (five hours report)

Alternative, you can use the following:

`./oclumon dumpnodeview -allnodes -v -last "04:59:59 »/tmp/output.txt`

`./oclumon dumpnodeview -allnodes -v -s "2013-04-24 09:00" -e "2013-04-24 03:15:00`

Output files

- crsData_rac1_20121204_1103.tar.gz
- ocrData_rac1_20121204_1103.tar.gz
- coreData_rac1_20121204_1103.tar.gz
- osData_rac1_20121204_1103.tar.gz
Tools & Utilities - Cluster Health Monitor (CHM)

- Is a tool designed, developed to detect and analyze OS, Cluster resources failures etc.
- Formerly known as Instantaneous Problem Detector for OS(IPD/OS).
- Pre 11gR2 versions, you need to download the tool from OTN.
- With 11gR2, it is the integral part of the software and integrated closely with GI.
- `ora.crf CHM resource introduced | crsctl stat res -t -init`
- Not available on some platforms.
- Can be used on RAC and non-RAC environments.
- Collects OS real-time (every second, 5 sec from 11203) statistics: memory, swap, I/O, net work etc
- Stores real-time monitoring metrics in the CHM repository.
- Historical data can be used to diagnose: node eviction, instance hang, server perf. etc
- Contains two services:
  - **System Monitoring Service** (`osysmond`)
    - runs on every node, monitor and collect OS metrics and send data to
  - **OloggeredCluster Logger Service** (`ologgered`)
    - Stores the information received from the nodes in the repository
    - Runs in one node as master service and standby service on other nodes
- CHM vs OSWatcher:
  - CHM takes less CPU, and less overhead on the node, OSWatcher doesn’t run
  - When the server CPU is heavily used
Tools & Utilities - Cluster Health Monitor (CHM)

- Consumes less than %5 CPU/core, minimal overhead on the server
- Takes 1GB space by default across all nodes.
- Apprx. 0.5GB data per day.
- Data can be kept for 3 days.
- `./oclumon manage -get repsize`
- `oclumon` - a command-line tool, used to manage CHM repository
- Stores in a management repository database with 12c.
• Is a tool captures OS performance metrics and stores in the statistical data in a file
  • vmstat, netstat, top, tracerouts, ps, iostat etc
• Available on MOS.
• On RAC, need to configure, schedule on individual nodes
• Supports most UNIX/LINUX platforms.
• ./startoswbb.sh (default interval/retention, 30 sec/48 hrs)
• ./startoswbb.sh 60 10 (60 seconds interval, 10 hrs data retention)
• ./stoposwbb.sh
• Review the dat file in the /archive directory
• Is a OS top-like utility on Linux Platforms
• Provides nearly real-time database monitoring capabilities for a RAC and Non-RAC dbs for 11.2.0.3 or higher.
• A very light weight utility, consumes 0.20% memory and <1% CPU.
• Download the oratop.zip from MOS, set chmod 755.
• Db init parameters: statistics_level = TYPICAL, timed_statistics = TRUE must be set
• Need to input username/password, connects as system user when no credentials provides.
• Set the following on the OS
  
  ```
  $ ORACLE_UNQNAME=<dbname>
  $ ORACLE_SID=<instance_name1>
  $ ORACLE_HOME=<db_home>
  $ export LD_LIBRARY_PATH=$ORACLE_HOME/lib
  $ export PATH=$ORACLE_HOME/bin:$PATH
  ```
• Download the oratop.zip from MOS, set chmod 755
• Needs a TNS to the source database
• Need privileges on, v_$SESSION, v_$SYSTEMETRIC, v_$INSTANCE, v_$PROCESS, V_$SYSTEM_EVENT etc
Tools & Utilities - oratop

- Live window: lists 5 TOP wait events, top Oracle sessions of i/o, memory, db load,
- Provides database blocking details.
- Press q/Q or Control+C to abort

./oratop -i 5 / as sysdba - interval every 5 seconds
./oratop -i 5 username/password@tns_alias

% db - (values > 99%)
% CU - (load > 2 x cpu counts & host cpu > 99)
HLD - (load > 2 * cpu counts and aas > cpu counts)
IORL - (value > 20ms)
% FR - (value < 1%)
ASW - (value = session counts, USN)
AAS - (value > cpu counts)
DBW - (value > 50%)
EVENT - Active wait event
PGA – potential unusual memory growth
BLOCKER - a blocking session with (wait time > 5 minutes)

---

Header

Databases

Top 5 DB events

Processes
• A RAC configuration auditing utility that audits various important configuration settings:
  Cluster, ASM, Grid Infrastructure etc

• Audits, OS Kernal parameters/Packages, 11.2.0.3 upgrade readiness etc

• Download the raccheck.zip from MOS, chmod to 755.

• Ability to compare between two outputs.

• All the recommendations/output written to a HTML file.

• The output include overall health check rating – out of 100, bug fixes, patch recommendations etc.

• Upload the .zip file if MOS ask to do so.
Examples:

./raccheck - follow the interactive steps
./raccheck -u -o pre|post
./raccheck -h
./raccheck -s

./raccheck -diff report1 report2

Usage: ./raccheck [-abvhpmsuo:c:rt:]
  -a      All (Perform best practice check and recommended patch check)
  -b      Best Practice check only. No recommended patch check
  -h      Show usage
  -v      Show version
  -p      Patch check only
  -m      exclude checks for Maximum Availability Architecture
  -u      Run raccheck to check pre-upgrade or post-upgrade best practices.-o pre or -o post is mandatory with -u option like ./raccheck -u -o pre
  -f      Run Offline. Checks will be performed on data already
  -o      Argument to an option. if -o is followed by
          v,V,Verbose,VERBOSE or Verbose, it will print checks which
          pass on the screen
          if -o option is not specified, it will print only failures on
          screen. for eg: raccheck -a -o v -r To include High availability best practices also in regular
          healthcheck eg ./racchekck -r(not applicable for exachk) -c Pass specific module or component to check
          best practice
          for. By default it will check for components indentified fr
• HANGANALYZE helps detecting the cause of database hang

• Advised to run the HANGANLYZE when a database suffers from hang, performance degradation, latching issues etc

• Available since 8.1.6, provides cluster-wide analysis from 9i

Examples:

```
SQL> sqlplus " / as sysdba"
SQL> oradebug setmypid
SQL> oradebug unlimit
SQL> oradebug setinst all
SQL> oradebug -g def hanganalyze 3
   -- wait 90 seconds
SQL> oradebug -g def hanganalyze 3
SQL> oradebug tracefile_name
SQL> exit
```
HANGANLYZE Level:

10   Dump all processes
5    Level 4 + Dump all processes involved in wait chains (NLEAF state)
4    Level 3 + Dump leaf nodes (blockers) in wait chains (LEAF, LEAF_NW, IGN_DMP state)
3    Level 2 + Dump only processes thought to be in a hang (IN_HANG state)
     -- recommended
1-2  Only HANGANALYZE output, no process dump at all
Review the trace file:

HANG ANALYZE (section)

CYCLES
list process dependencies for deadlock/hung state

BLOCKER OF MANY SESSIONS:
When a session block too many session, you will have this section, when a session block 10 or more sessions

STATE OF NODES | OPEN CHAINS | OTHER CHAINS
Dumping system state

When a database in a complete hung state and you can’t connect to the database as / as sysdba, when memory leaks are suspected, use the following:

sqlplus – prelim / as sysdba

SQL> oradebug setmypid
SQL> oradebug unlimit;
SQL> oradebug – g all dump systemstate 10|266
Wait for 60 seconds
SQL> oradebug -g all dump systemstate 10|266

Review/upload the trace file to MOS
References

RACcheck - RAC Configuration Audit Tool [ID 1268927.1]
Troubleshoot Grid Infrastructure Startup Issues [ID 1050908.1]
Oracle Clusterware CRSD OCSSD EVMD Log Rotation Policy [ID 557204.1]
CRS Diagnostic Data Gathering: A Summary of Common tools and their Usage [ID 783456.1]
Remote Diagnostic Agent (RDA) 4 - Getting Started [ID 314422.1]
Data Gathering for Troubleshooting Oracle Clusterware (CRS or GI) And Real Application Cluster (RAC) Issues [ID 289690.1]
Q & A

You can write me at sjaffarhussain@gmail.com