Oracle® Communications

Diameter Signaling Router

DSR APIGW Disaster Recovery Guide

Release 8.5.1

F51306-01

December 2021

ORACLE

Oracle Communications Diameter Signaling Router, DSR APIGW Disaster Recovery Guide

CAUTION: Use only the DR procedures included in the Disaster Recovery Kit.

Before recovering any system, please access My Oracle Support (MOS) (https://support.oracle.com) and review any Technical Service Bulletins (TSBs) that relate to this DR procedure

My Oracle Support (MOS) (https://support.oracle.com) is your initial point of contact for all product support and training needs. A representative at Customer Access Support (CAS) can assist you with MOS registration.

Call the CAS main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at http://www.oracle.com/us/support/contact/index.html.

See more information on MOS in the Appendix section.

Table of Contents

List of Procedures

List of Tables

1. Introduction

1.1 Purpose and Scope

This document is a guide to describe procedures used to execute disaster recovery for DSR API Gateway. This includes recovery of partial or a complete loss of one or more DSR APIGW servers. The audience for this document includes GPS groups such as Software Engineering, Product Verification, Documentation, and Customer Service including Software Operations and First Office Application. This document can also be executed by Oracle customers, as long as Oracle Customer Service personnel are involved and/or consulted. This document provides step-by-step instructions to execute disaster recovery for DSR APIGW. Executing this procedure also involves referring to and executing procedures in existing support documents.

Note: Please note that failures can happen from the host or Infrastructure level too. Different infrastructures have different approaches to recover VMs which is not covered in this document. For example, VMWare has a vMotion feature which can migrate VM from one host to another. Any such Infrastructure/Hypervisor related migrations/disaster recovery scenarios are out of scope of this document. This document covers the DR scenarios within the DSR application.

1.2 References

[1] DSR API Gateway Installation Guide
[2] DSR / SDS NOAM Failover User's Guide

1.3 Acronyms

Table 1: Acronyms

Acronym	Definition
BIOS	Basic Input Output System
CD	Compact Disk
DSR	Diameter Signaling Router
ESXi	Elastic Sky X Integrated
FABR	Full Address Based Resolution
iDIH	Integrated Diameter Intelligence Hub
IPFE	IP Front End
IWF	Inter Working Function
NAPD	Network Architecture Planning Diagram
NOAM	Network Operations, Administration & Maintenance
OS	Operating System
OVA	Open Virtualization Appliance
PDRA	Policy Diameter Routing Agent
PCA	Policy and Charging Application
RBAR	Range Based Address Resolution
SAN	Storage Area Network
SFTP	Secure File Transfer Protocol
SNMP	Simple Network Management Protocol
SOAM	Systems Operations, Administration & Maintenance
TPD	Tekelec Platform Distribution
VM	Virtual Machine
vSTP	Virtual Signaling Transfer Point

1.4 Terminology

Table 2: Terminology

Base software	Base software includes deploying the VM image.
Failed server	A failed server in disaster recovery context refers to a VM that has suffered partial or complete software failure to the extent that it cannot restart or be returned to normal operation and requires intrusive activities to re-install the software.
Software Centric	The business practice of delivering an Oracle software product, while relying upon the customer to procure the requisite hardware components. Oracle provides the hardware specifications, but does not provide the hardware or hardware firmware, and is not responsible for hardware installation, configuration, or maintenance.
Enablement	The business practice of providing support services (hardware, software, documentation, etc) that enable a 3rd party entity to install, configuration, and maintain Oracle products for Oracle customers.

1.5 General Description

The DSR APIGW disaster recovery procedure falls into following categories:

Disaster Recovery - Backup and Restore using management client (ndb_mgm) and ndb_restore	MySQL NDB Cluster
Disaster Recovery - Backup and Restore using manual approach	MySQL NDB Cluster
Recovery with Application servers lost [*Recovery Scenario 1: Admin is up and running, App server(s) lost*]	• All Application servers failed
Recovery of Admin server [*Recovery Scenario 2: App servers are up and running, Admin server lost*]	• Admin server failed
Recovery of Admin and lost Application servers [*Recovery Scenario 3: At least one App server is up, Admin and App server(s) lost*]	• Admin server failed • One App server intact
Recover of both Admin and Application servers *[Recovery Scenario 4: Admin and App servers lost*]	• Both Admin and App server failed

2. Procedure Overview

This section lists the materials required to perform disaster recovery procedures and a general overview (disaster recovery strategy) of the procedure executed.

2.1 Required Materials

The following items are needed for disaster recovery:

1. A hardcopy of this document (E76332) and hardcopies of all documents in the reference list
2. Hardcopy of all NAPD performed at the initial installation and network configuration of this customer's site. If the NAPD cannot be found, escalate this issue within My Oracle Support (MOS) until the NAPD documents can be located.
3. DSR APIGW recent backup files: electronic backup file (preferred) or hardcopy of all DSR APIGW configuration and provisioning data.
4. Latest Network Interface data; XSI interface lost
5. The ocsgdr.praperties file to fill-in the parameter details
6. **recoverAdminServer.py** script to recover Admin server
7. ***recoverAppServers.py*** script to recover Application server

2.2 Procedure Preparation

Disaster recovery procedure execution is dependent on the failure conditions in the network. The severity of the failure determines the recovery scenario for the network. Use Table 3: Recovery Scenarios below to evaluate the correct recovery scenario and follow the procedure(s) listed to restore operations.

Note: A failed server in disaster recovery context refers to a server that has suffered partial or complete software failure to the extent that it cannot restart or be returned to normal operation and requires intrusive activities to re-deploy base software.

Table 3: Recovery Scenarios

Recovery Scenario	Failure Condition	Section
1	• All database servers failed.	Section **Error! Reference source not found.**
2	• At least one database server is intact and available.	Section **Error! Reference source not found.**
3	• Admin is up and running, App server(s) lost	Section Recovery Scenario 1: Admin is up and running, App server(s) lost
4	• App servers are up and running, Admin server lost	Section Recovery Scenario 2: App servers are up and running, Admin server lost
5	• At least one App server is up, Admin and App server(s) lost	Section Recovery Scenario 3: At least one App server is up, Admin and App server(s) lost
6	• Admin and App servers lost	Section Recovery Scenario 4: Admin and App servers lost

3. DSR APIGW Database Disaster Recovery Procedure

Call My Oracle Support (MOS) prior to executing this procedure to ensure that the proper recovery planning is performed.

Before disaster recovery, users must properly evaluate the outage scenario. This check ensures that the correct procedures are executed for the recovery.

****** WARNING *******

Note: *Disaster recovery is an exercise that requires collaboration of multiple groups and is expected to be coordinated by the ORACLE SUPPORT prime. Based on ORACLE SUPPORT's assessment of Disaster, it may be necessary to deviate from the documented process.*

3.1 Recovering and Restoring System Configuration

Disaster recovery requires configuring the system as it was before the disaster and restoration of operational information.


!!WARNING!!

Whenever there is need to restore the backup for database servers in any of below Recovery Scenarios, the backup directory may not be there in the system as system will be DRed.

3.1.1 Disaster Recovery - Backup and Restore using management client (ndb_mgm) and ndb_restore

See Disaster Recovery for the complete procedure.

3.1.2 Disaster Recovery - Backup and Restore using manual approach

- On regular basis it is suggested to take the MySQL dumps (data backup) as shown:

  ```
  mysqldump -h <Ipaddress of SQL Node1> -u <mysql username> -p<Password> --
  databases gatekeeper > gatekeeper_data.sql
  ```

  Example :

  ```
  mysqldump -h 10.75.217.94 -u mysqluser -pMyNewPass4! --databases gatekeeper >
  gatekeeper_data.sql
  ```

- Rebuild the mysql ndb cluster (make sure ip's of mysql cluster node VMs are not changed) as per steps mentioned in Section 6.1 in *Install and Configure MySQL NDB Cluster of DSR API Gateway Installation Guide* and use the backup taken manually to restore the database.
- SSH to SQL Node1 VM as root.
- Enter the command: `mysql -u <new mysql user> -p gatekeeper < backedupschemafile`

  Example: `mysql -u mysqluser -p gatekeeper < backup_gatekeeper.sql`

  Note: The mysqluser used in the above example needs to be replaced with appropriate new user created for mysql.

4. DSR APIGW Admin and Application Disaster Recovery Procedure

4.1 Recovery Scenario 1: Admin is up and running, App server(s) lost

Procedure 1. Recovery Scenario 1: Admin is up and running, App server(s) lost

Step #	Procedure	Description
The intent of this procedure is to recover when Admin is up and running and the application servers are lost. Check off (√) each step as it is completed. Boxes have been provided for this purpose under each step number. If this procedure fails, contact My Oracle Support (MOS), and ask for assistance.		
1. ☐	**VMWare/Openstack**: Create lost App VMs	Create the Application VMs, which has to be recovered, with same IP addresses. Refer to the following procedures from reference [1]: For VMWare based deployments: 1. Create DSR APIGW Admin/Application VMs (VMWare) For KVM/Openstack based deployments: 1. Create DSR APIGW Admin/Application VMs (Openstack)
2. ☐	**Admin Server:** Edit properties file	1. Login to **Admin** server 2. Navigate to ***/u02/app/oracle/scripts/*** `$ cd /u02/app/oracle/scripts/` 3. Edit the file **osgdr.properties**. Add respective property values in the file. Feed in file with all the lost App servers data. Refer to Appendix B for parameter details.
3. ☐	**Admin Server:** Execute App VM recovery script	From Admin server, execute the script as follows: 1. Login to **Admin** server 2. Navigate to ***/u02/app/oracle/scripts*** 3. Execute ***recoverAppServers.py*** to recover Application server.

4.2 Recovery Scenario 2: App servers are up and running, Admin server lost

Procedure 2. Recovery Scenario 2: App servers are up and running, Admin server lost

Step #	Procedure	Description
The intent of this procedure is to recover when application servers are up and running and the Admin server is lost. Check off (√) each step as it is completed. Boxes have been provided for this purpose under each step number. If this procedure fails, contact My Oracle Support (MOS), and ask for assistance.		
1. ☐	**Openstack Controller:** Create lost Admin server	Create the Admin server with same IP addresses. Refer to the following procedures from reference [1]: For VMWare based deployments: 1. Create DSR APIGW Admin/Application VMs (VMWare) For KVM/Openstack based deployments: 1. Create DSR APIGW Admin/Application VMs (Openstack)
2. ☐	**Openstack GUI:** Copy the .pem file (key-pair) used to create the VMs to Admin server in any location.	1. Login to Openstack controller console 2. Copy the pem file from the opentack controller to the Admin server in any location. `$ scp -i /root/dsr-keypair.pem /root/ dsr-keypair.pem admusr@<aminserverip>:/u02` **Note**: PEM certificates are frequently used for web servers as they can easily be translated into readable data using a simple text editor. Generally when a PEM encoded file is opened in a text editor, it contains very distinct headers and footers. Refer to **Error! Reference source not found.** for creating a PEM f ile.
3. ☐	**Admin Server:** Edit properties file	1. Login to **Admin** server 2. Navigate to ***/u02/app/oracle/scripts/*** `$ cd /u02/app/oracle/scripts/` 3. Edit the file **osgdr.properties**. Add respective property values in the file. Feed in **osgdr.properties** file with the lost Admin server data and back up server details. Refer to Appendix B for parameter details.
4. ☐	**Admin Server:** Execute Admin server recovery script	From Admin server, execute the script as follows: 1. Login to **Admin** server 2. Navigate to ***/u02/app/oracle/scripts*** 3. Execute ***recoverAdminServer.py*** to recover Admin server.

4.3 Recovery Scenario 3: At least one App server is up, Admin and App server(s) lost

Procedure 3. Recovery Scenario 3: At least one App server is up, Admin and App server(s) lost

Step #	Procedure	Description
The intent of this procedure is to recover when Admin and the some of the application servers are lost. Check off (√) each step as it is completed. Boxes have been provided for this purpose under each step number. If this procedure fails, contact My Oracle Support (MOS), and ask for assistance.		
1. ☐	**VMWare/Openstack:** Create lost Admin server	Create the Admin server and the lost App server with same IP addresses. Refer to the following procedures from reference [1]: For VMWare based deployments: 1. Create DSR APIGW Admin/Application VMs (VMWare) For KVM/Openstack based deployments: 1. Create DSR APIGW Admin/Application VMs (Openstack)
2.	**Openstack GUI:** Copy the .pem file (key-pair) used to create the VMs to Admin server in any location.	1. Login to Openstack controller console 2. Copy the pem file from the opentack controller to the Admin server in any location. `$ scp -i /root/dsr-keypair.pem /root/ dsr-keypair.pem admusr@<aminserverip>:/u02` **Note**: PEM certificates are frequently used for web servers as they can easily be translated into readable data using a simple text editor. Generally when a PEM encoded file is opened in a text editor, it contains very distinct headers and footers. Refer to **Error! Reference source not found.** for creating a PEM f ile.
3. ☐	**Admin Server:** Edit properties file	1. Login to **Admin** server 2. Navigate to ***/u02/app/oracle/scripts/*** `$ cd /u02/app/oracle/scripts/` 3. Edit the file **osgdr.properties**. Add respective property values in the file. Feed in **osgdr.properties** file with the lost Admin server data and back up server details. Refer to Appendix B for parameter details.
4. ☐	**Admin Server:** Execute Admin server recovery script	From Admin server, execute the script as follows: 1. Login to **Admin** server 2. Navigate to ***/u02/app/oracle/scripts*** 3. Execute ***recoverAdminServer.py*** to recover Admin server.

Step #	Procedure	Description
5. ☐	**Admin Server:** Execute App VMs recovery script	From Admin server, execute the script as follows: 1. Login to **Admin** server 2. Navigate to ***/u02/app/oracle/scripts*** 3. Execute ***recoverAppServers.py*** to recover Application server.

4.4 Recovery Scenario 4: Admin and App servers lost

Procedure 4. Recovery Scenario 4: Admin and App servers lost

Step #	Procedure	Description
The intent of this procedure is to recover when Admin and the application servers are lost. Check off (√) each step as it is completed. Boxes have been provided for this purpose under each step number. If this procedure fails, contact My Oracle Support (MOS), and ask for assistance.		
1. ☐	**VMWare/Openstack**: Create lost Admin server	Create the Admin server with same IP addresses. Refer to the following procedures from reference [1]: For VMWare based deployments: 2. Create DSR APIGW Admin/Application VMs (VMWare) For KVM/Openstack based deployments: 2. Create DSR APIGW Admin/Application VMs (Openstack)
2.	**Openstack GUI:** Copy the .pem file (key-pair) used to create the VMs to Admin server in any location.	1. Login to Openstack controller console 2. Copy the pem file from the opentack controller to the Admin server in any location. `$ scp -i /root/dsr-keypair.pem  /root/ dsr-keypair.pem admusr@<aminserverip>:/u02` **Note**: PEM certificates are frequently used for web servers as they can easily be translated into readable data using a simple text editor. Generally when a PEM encoded file is opened in a text editor, it contains very distinct headers and footers. Refer to **Error! Reference source not found.** for creating a PEM f ile.
3. ☐	**Admin Server:** Edit properties file	1. Login to **Admin** server 2. Navigate to ***/u02/app/oracle/scripts/*** `$ cd /u02/app/oracle/scripts/` 3. Edit the file **osgdr.properties**. Add respective property values in the file. Feed in **osgdr.properties** file with the lost Admin server data and back up server details. Refer to Appendix B for parameter details.

Step #	Procedure	Description
4. ☐	**Admin Server:** Execute Admin server recovery script	From Admin server, execute the script as follows: 1. Login to **Admin** server 2. Navigate to ***/u02/app/oracle/scripts*** 3. Execute ***recoverAdminServer.py*** to recover Admin server.
5. ☐	**Admin Server:** Execute App VMs recovery script	From Admin server, execute the script as follows: 1. Login to **Admin** server 2. Navigate to ***/u02/app/oracle/scripts*** 3. Execute ***recoverAppServers.py*** to recover Application server.

Appendix A. Disaster Recovery

This section provides the Back up and restore using Management client (ndb_mgm) and ndb_restore.

A.1. Backup

The MySQL NDB Cluster allows taking a snapshot of the database while it is active. A backup of all the data will be stored in each data nodes. Management client (ndb_mgm) is used to take the backup of the complete data in MySQL NDB Cluster.
A backup is a snapshot of the database at a given time. The backup consists of three main parts:

- Metadata stored in BACKUP-backup_id.node_id.ctl –

  The names and definitions of all database tables are stored along with a file containing control information and metadata. Each node saves the same table definitions (for all tables in the cluster) to its own version of this file.

- Table records stored in BACKUP-backup_id-0.node_id.data -
  The data is stored in the database tables during backup. A data file containing the table records are saved based on per-fragment basis. Different nodes save different fragments during the backup.
  The file saved by each node starts with a header that states the tables to which the records belong.

- Transaction log stored in BACKUP-backup_id.node_id.log -
  This record shows how and when the data was stored in the database. A log file contains the record of committed transactions. Only the transactions on the tables stored in the backup are also stored in the log. Nodes involved in the backup saves different records as different node hosts different database fragments.

The BackupDataDir parameter configured in the management node determines the location of the backup files.

A.2. Procedure to take Backup

Perform these steps to take backup:

1. Run the management client (ndb_mgm).
2. Run the START BACKUP <backupid> WAIT STARTED command. The backupid is an optional parameter. If the values are not given, then the default available ids are assigned (for example: 1, 2, 3 and so on). The backupid is in the format **YYMMDDHHMM.** Once the backup is completed, the management client displayes the following messages.

```
ndb_mgm> START BACKUP 1902181047 WAIT STARTED
Connected to Management Server at: localhost:1186
Waiting for started, this may take several minutes
Node 3: Backup 1902181047 started from node 49
ndb_mgm> Node 3: Backup 1902181047 started from node 49
completed
StartGCP: 4132 StopGCP: 4139
#Records: 4002083 #LogRecords: 0
Data: 800054076 bytes Log: 0 bytes

ndb_mgm>
```

3. To check the status of the backup in different data nodes, use the following command <node_id> REPORT BACKUPSTATUS

```
ndb_mgm> 1 REPORT BACKUPSTATUS

Node 1: Local backup status: backup 1902181047 started from
node 49
#Records: 267421 #LogRecords: 0
Data: 53482836 bytes Log: 0 bytes

ndb_mgm> 2 REPORT BACKUPSTATUS

Node 2: Local backup status: backup 1902181047 started from
node 49
#Records: 274075 #LogRecords: 0
Data: 54814420 bytes Log: 0 bytes


ndb_mgm> 3 REPORT BACKUPSTATUS

Node 3: Local backup status: backup 1902181047 started from
node 49
#Records: 283234 #LogRecords: 0
Data: 56647268 bytes Log: 0 bytes


ndb_mgm> 4 REPORT BACKUPSTATUS

Node 4: Local backup status: backup 1902181047 started from
node 49
#Records: 706022 #LogRecords: 0
Data: 141204104 bytes Log: 0 bytes
```

```
Once backup is completed the status shows as backup not
started.
1 REPORT BACKUPSTATUS

Node 1: Backup not started
```

Note: To cancel the in-progress backup, perform the following steps:

1. Start the management client (ndb_mgm).
2. Run the following command

```
"ABORT BACKUP backup_id"

ndb_mgm> ABORT BACKUP 1902181047
```

A.3. Restore

Using the ndb_restore program, MySQL NDB Cluster is restored. It is an NDB API program that supports both restoring the schema and data. Perform the restore in the following three steps:

1. Restore the schema.
2. Restore the data with indexes disabled.
3. Rebuild the indexes.

Note: Before using ndb_restore, the cluster has to be running in **single user mode**,

A.4. Restore MySQL NDB Cluster using backup

Note: Assuming the backup id is 1902181047 and backup is stored in the `/var/lib/mysql/dbbackdata/BACKUP` directory.

1. Enter the Single user mode.
   Find the node id of the [API] from the configuration or SHOW command

```
ndb_mgm> show
Connected to Management Server at: 10.75.213.88:1186
Cluster Configuration
---------------------
[ndbd(NDB)] 4 node(s)
id=1 @10.75.212.250 (mysql-5.7.24 ndb-7.6.8, Nodegroup: 0, *)
id=2 @10.75.213.104 (mysql-5.7.24 ndb-7.6.8, Nodegroup: 0)
id=3 @10.75.212.231 (mysql-5.7.24 ndb-7.6.8, Nodegroup: 1)
id=4 @10.75.213.125 (mysql-5.7.24 ndb-7.6.8, Nodegroup: 1)

[ndb_mgmd(MGM)] 2 node(s)
id=49 @10.75.213.149 (mysql-5.7.24 ndb-7.6.8)
id=50 @10.75.213.88 (mysql-5.7.24 ndb-7.6.8)

[mysqld(API)] 3 node(s)
id=55 @10.75.213.245 (mysql-5.7.24 ndb-7.6.8)
id=56 @10.75.213.165 (mysql-5.7.24 ndb-7.6.8)
id=57 (not connected, accepting connect from any host)

ndb_mgm>
```

2. Enter the Single User Mode using the above node id.

```
ndb_mgm> ENTER SINGLE USER MODE 57
Connected to Management Server at: localhost:1186

Single user mode entered
Access is granted for API node 57 only.
ndb_mgm>
```

3. Restore the schema with indexes disabled, run the following command in any of the Data node. The database tables must be recreated in one of the nodes using --restore_meta(-m) option. This restoration of the metadata on single node is sufficient to restore the metadata information to whole cluster.

   In Data Node 1

```
ndb_restore --ndb-
connectstring=10.75.213.149:1186,10.75.213.88:1186 --nodeid=1 --
backupid=1902181047 --
backup_path=/var/lib/mysql/dbbackdata/BACKUP/BACKUP-1902181047 --
restore_meta --disable-indexes
```

4. Restore the data with indexes disabled in each of the data nodes, execute below commands in each of the data nodes.
   In Data Node 1

```
ndb_restore --ndb-
connectstring=10.75.213.149:1186,10.75.213.88:1186 --nodeid=1 --
backupid=1902181047 --restore_data --
backup_path=/var/lib/mysql/dbbackdata/BACKUP/BACKUP-1902181047 --
disable-indexes
```

   In Data Node 2

```
ndb_restore --ndb-
connectstring=10.75.213.149:1186,10.75.213.88:1186 --nodeid=2 --
backupid=1902181047 --restore_data --
backup_path=/var/lib/mysql/dbbackdata/BACKUP/BACKUP-1902181047 --
disable-indexes
```

In Data Node 3

```
ndb_restore --ndb-
connectstring=10.75.213.149:1186,10.75.213.88:1186 --nodeid=3 --
backupid=1902181047 --restore_data --
backup_path=/var/lib/mysql/dbbackdata/BACKUP/BACKUP-1902181047 --
disable-indexes
```

In Data Node 4

```
ndb_restore --ndb-
connectstring=10.75.213.149:1186,10.75.213.88:1186 --nodeid=4 --
backupid=1902181047 --restore_data --
backup_path=/var/lib/mysql/dbbackdata/BACKUP/BACKUP-1902181047 --
disable-indexes
```

5. Rebuild the indexes in one of the data node, for ex execute the below command in data node 1.

   In Data Node 1

   ```
   ndb_restore --nodeid=1 --backupid=1902181047 --
   backup_path=/var/lib/mysql/dbbackdata/BACKUP/BACKUP-1902181047 --
   rebuild-indexes
   ```

6. Exit the Single User Mode.

   ```
   ndb_mgm> EXIT SINGLE USER MODE
   Exiting single user mode in progress.
   Use ALL STATUS or SHOW to see when single user mode has been
   exited.
   ndb_mgm>
   ```

This completes restoring the data in MySQL NDB cluster.

Appendix B. OCSG DR Properties file

Table 4: OCSG DR Properties file

Section	Parameter Name	Description
Admin	`servers`	IMI Interface address of Admin Server. `servers = ["AdminServer: xxx.xxx.xxx.xxx "]` **Note**: It is mandatory to follow the name of Admin server as 'AdminServer' This is the DSRAPIGW DB server address where data is backed up. DR procedure will use this data.
Admin	`xmiInterface`	XMI Interface address of Admin Server `xmiInterface = ["AdminServer: xxx.xxx.xxx.xxx "]`
Admin	`backupServer`	Provide the IMI VIP of DSR API GW Database. Admin server should have access to this server using the key/pem file. This is the location in the DSRAPIGW DB server where the data should be backed up. For example, `backupServer = xxx.xxx.xxx.xxx`
Admin	`backupDomain`	Full path including the DSR API GW domain folder name to where the DSR API GW files need to be backed up on backup server. For example, `backupDomain = /var/TKLC/db/filemgmt/backup/services-gatekeeper-domain`
App	`servers`	Add App server name and IP. Add comma seperated entries for multiple servers. For example, servers = `["AppServer1:xxx.xxx.xxx.xxx", "AppServer2:xxx.xxx.xxx.xxx"]` **Note**: It is mandatory to follow the name of App servers as 'AppServer1', 'AppServer2' etc.
App	`xmiInterfaces`	XMI Interface address for all AppServers in [“Ip1”,”Ip2”…] format. For example, `xmiInterfaces = ["AppServer1: xxx.xxx.xxx.xxx ", "AppServer2: xxx.xxx.xxx.xxx "]`
App	`xsiInterfaces`	XSI Interface address for all AppServers in [“Ip1”,”Ip2”…] format.

Section	Parameter Name	Description

		For example,
		`xsiInterfaces = ["AppServer1: xxx.xxx.xxx.xxx ", "AppServer2: xxx.xxx.xxx.xxx "]`
		
		`To add multiple XSIs to each AppServer the format should be,`
		`["AppServer1:XSI1-IP","AppServer2:XSI2","AppServer2:XSI1-IP","AppServer2:XSI2"]`

App	`exteralLoadbalancerIP`	IP used to publish T8 APIs. This IP will be used when displaying T8 API access URLs in Partner and API management Portal. `exteralLoadbalancerIP = xxx.xxx.xxx.xxx`
Servers	`cleanUpBeforeInstall`	If the script failed to execute while running, the server will be in a bad shape for a fresh install. Keeping cleanUpBeforeInstall as "yes" will clean up the server and make it ready for script re-run.
Servers	`ntp`	Provide NTP server IP `ntp = xxx.xxx.xxx.xxx`
Servers	`mtu`	Maximum transmission unit. The script copies multiple files from Admin server to App server. Before copying the MTU has to be set. Recommended value is “9000”. `mtu = 9000`
Servers	`apiroot`	This variable is part of the API creation. <apiroot> is prefixed to the context uri of the APIs exposed. # For example, the API name of Device triggering is "apiroot-dt"
Servers	`dsrMpList`	Provice DSR MP XSI Ip list in format, MP1-XSI-IP:port,MP2-XSI1-IP:port…..
Files	`pemfile`	Provide the .pem file location. `pemfile = /u02/software/ocsg-db-key.pem`
Files	`logfile`	Custom log file for Installation. Change log file name if required. `logfile = ocsg_install.log`
Files	`presentFolder`	The scripts will be present in this location. This property should not be changed `presentFolder = /u02`

Section	Parameter Name	Description
Files	`targetFolder`	The scripts will be copied to this location. This v should not be changed `targetFolder` = */u03*
Files	`targetPath`	Provide the location of the scripts. This property should not be changed `targetPath` = */app/oracle/*
Files	`scripts`	Provide the folder name where scripts need to be stored. This property should not be changed. `scripts` = *scripts*
Files	`extendWizard`	Custom scripts will be present here. This property should not be changed. `extendWizard` = *extend_wizard/*
Files	`SCEFPackage_EAR`	Default EAR file name. This property should not be changed. `SCEFPackage_EAR` = *SCEFHandlers.ear*
Files	`nodemgr`	Node manager service file name. This property should not be changed `nodemgr` = *nodemgr*
Files	`DefaultJar`	Location of ocsg_generic_jar. This property should not be changed `defaultJar` = */usr/TKLC/dsrapigw/ocsg_generic_jar*
Files	`volumeName`	Provide the Volume name, This property should not be changed `volumeName` = *ocsgv*
Files	`volumeSize`	Volume size in GB. Script woll create a new volume of this size. This field should not be changed `volumeSize` = 10
Files	`inventoryLoc`	Inventory log location of OCSG. This property should not be changed `inventoryLoc` = */u02/inventory*
Credentials	`mysqlJdbcServerUrl`	MySQL DB credentials. Provide IMI VIP of the DSR API GW database setup. jdbc:mysql://<db-server-ip>:15616/gatekeeper For Example, mysqlJdbcServerUrl = jdbc:mysql://30.30.30.17:15616/gatekeeper
Credentials	`mysqlUserName`	This property should not be changed. `mysqlUserName` = *awadmin*

Section	Parameter Name	Description
		Note: MySQL password will be the default comcol password. It is present in dsrapigw_default_params.rsp file.
Credentials	weblogicUser	Provide the DSR API GW Admin portal credentials. weblogicUser = *weblogic* weblogicPassword = *tekelec123*
Credentials	weblogicPassword	
Credentials	nodeManagerUser	Provide the Nodemanager credentials which will be used in all Admin and AppServers nodeManagerUser = *nodemanager* nodeManagerPassword = *tekelec123*
Credentials	nodeManagerPassword	
Credentials	operatorUser	A new operator will be crated with this details to access partner relationship management portal. operatorUser = *oracleop3* operatorPassword = *tekelec123*
Credentials	operatorPassword	
Credentials	adminServerUser	Below is the ssh user name in Admin and AppServers adminServerUser = *admusr* appServerUser = *admusr*
Credentials	appServerUser	
Ports	adminListenPort appListenPort appListenPortSSL	These are the default ports opened on IMI network should not be changed, these ports are used only for internal communication adminListenPort = *7001* appListenPort = *8001* appListenPortSSL = *8002*
Ports	adminIMIPorts adminXMIPorts	Ports to be enabled in IP Firewall on Admin server: adminIMIPorts = 7001,5556,7002,9876,8050,3075,9090,7 adminXMIPorts = 9002
Ports	appIMIPorts appXMIPorts appXSIPorts	Ports to be enabled in IP Firewall on AppServers: appIMIPorts = 8001,8002,9876,5556,8050,3075,9090,7 appXMIPorts = 9002 appXSIPorts = 10001,10002

Appendix C. My Oracle Support (MOS)

MOS (https://support.oracle.com) is your initial point of contact for all product support and training needs. A representative at Customer Access Support (CAS) can assist you with MOS registration.

Call the CAS main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at http://www.oracle.com/us/support/contact/index.html. When calling, make the selections in the sequence shown below on the Support telephone menu:

1. Select 2 for New Service Request
2. Select 3 for Hardware, Networking and Solaris Operating System Support
3. Select one of the following options:
   - For Technical issues such as creating a new Service Request (SR), Select 1.
   - For Non-technical issues such as registration or assistance with MOS, Select 2.

You will be connected to a live agent who can assist you with MOS registration and opening a support ticket.

MOS is available 24 hours a day, 7 days a week, and 365 days a year.