Oracle® Communications Session Monitor

Backup and Restore Guide

Release 5.2
F88041-02
October 2024

Oracle Communications Session Monitor Backup and Restore Guide, Release 5.2

F88041-02

Contents

About this Guide

This guide provides guidelines and recommendations for setting up Oracle Communications Session Monitor in a secure configuration. The Oracle Communications Session Monitor product family includes the following products:

- Operations Monitor
- Enterprise Operations Monitor
- Fraud Monitor
- Control Plane Monitor

Documentation Set

Table 1 Documentation Suite for Session Monitor Release 5.2

Document Name	Document Description
Backup and Restore Guide	Provides instructions for backing up and restoring Session Monitor.
Developer Guide	Contains information for using the Session Monitor SAU Extension.
Fraud Monitor User Guide	Contains information for installing and configuring Fraud Monitor to monitor calls and detect fraud.
Installation Guide	Contains information for installing Session Monitor
Mediation Engine Connector User Guide	Contains information for configuring and using the Mediation Engine Connector.
Operations Monitor User Guide	Contains information for monitoring and troubleshooting IMS, VoLTE, and NGN networks using the Operations Monitor.
Release Notes	Contains information about the Session Monitor Release 5.2, including new features.
Security Guide	Contains information for securely configuring Session Monitor.
Upgrade Guide	Contains information for upgrading Session Monitor.

Revision History

This section provides a revision history for this document.

Date	Description
January 2024	Initial release
October 2024	Content updates

1

Session Monitor Backup and Restore

Session Monitor Release 5.2 provides the feature of backing up the Configuration, Database, Block Storage and other essential Files of Session Monitor Servers by providing a Backup and Restore procedure.

Use the Backup and Restore procedure to install Session Monitor Release 5.2, without losing data for the existing pre-5.1 Session Monitor setup. The Backup and Restore procedure can be used to take a backup of your previous Session Monitor setup during the upgrade to Release 5.1, and restore it if the upgrade fails.

> **Note:**
>
> The Backup procedure is not available for Probes. The supported nodes are Mediation Engine, Fraud Monitor and Mediation Engine Connector.

Session Monitor Release 5.2 has been tested for Backup and Restore from specific prior releases. Verify that your current installed release is listed in the table below.

Backup from	Restore on
4.4	5.1
5.0	5.1
5.1	5.1

2
Backup and Restore Strategies

There are two approaches you can follow to create the backup and restore.

- **Strategy 1**: Set up a backup location which could either be a shared drive or a remote server, that has free space greater than (at least 10%) of your original Oracle Communications Session Monitor Server. Take the backup on the shared drive or a remote server, upgrade or reinstall the original Session Monitorserver, restore the data from the backup location to the upgraded or reinstalled Session Monitor server, as required.
- **Strategy 2**: Create a target machine with Oracle Communications Session Monitor Release 5.1 installed, and use this new machine to copy the data from the original Session Monitor server. In this case, you do not need to restore the data again. Hence this step consumes less time. The newly created target machine will be referred to as the **Remote Server** in the subsequent sections later in this Guide.

If you choose Strategy-1, then follow the steps as mentioned in Creating a Backup section to create a backup, and then the Restore Procedure for Backup Created Using Strategy 1 section to restore the backup.

If you choose Strategy-2, then first follow the steps as mentioned in Session Monitor Release 5.2 Installation Guide to install Session Monitor Release 5.2 on the new server.

Next you can follow the steps mentioned in the section Creating a Backup to use the target machine as the backup location directly, and see the steps mentioned in Restore Procedure for Backup Created Using Strategy 2 to restore the backup.

Some steps and sections, in the Backup and Restore procedure, are applicable only to specific nodes and those are mentioned in the respective step or section. The rest of the sections are applicable for all the nodes namely Mediation Engine, Mediation Engine Connector and Fraud Monitor.

> **Note:**
>
> Session Monitor services must be stopped during the Backup and Restore procedure. Hence traffic is not processed by the Session Monitor during this time.

First decide on the strategy, go through and understand the complete Backup and Restore procedure thoroughly before starting with the actual backup and restore.

Follow the steps exactly as mentioned in this guide for creating the backups and restoring the same.

Consult Oracle support for any clarifications, before going ahead with the backup and restore procedure.

> **Note:**
>
> DO NOT delete the data from the backup location, until the Backup and Restore procedure is complete and it is verified that new Session Monitor Release 5.2 is working.

3

Creating a Backup of Session Monitor

This section describes the complete procedure to take a backup of the Session Monitor. Session Monitor supports taking a backup of the Block Storage, MySQL Database, Redis Database, and essential Session Monitor files.

The following are the steps covered for creating a Back up of the Session Monitor:

Prerequisites for Taking a Backup of Session Monitor

This section describes the prerequisites required for the Session Monitor backup procedure. Before starting with Backup procedure ensure that the following tasks are complete:

- For taking a Backup of Session Monitor Releases 4.4 and 5.0, ensure that the Session Monitor Server is on MySQL version 5.7.35 or higher. If it is not, upgrade to the latest GA release of 5.7 (5.7.35 or higher). For more information, see the section Upgrading MySQL section.
- For systems where External Authentication is enabled, it is recommended to temporarily disable External Authentication until the Restore procedure is completed, the Apache Web Server is reverted to NGINX. If the **Admin** user has been set up for External Authentication, set a local password for the **Admin** user while disabling External Authentication.

Disabling External Authentication on the Mediation Engine

For disabling External Authentication on the Mediation Engine, perform the following:

1. Log in to the Mediation Enginewith the configured credentials.
2. Disable External authentication in **admin** > **Settings** > **System Settings**.
3. Click **Update**
4. Log out from Meditation Engine.

Disabling External Authentication on the Mediation Engine Connector

For disabling External Authentication on the Mediation Engine Connector, perform the following:

1. Log in to Mediation Engine Connector with the configured credentials.

2. Navigate to **admin** > **Settings** > **External Authentication**.
3. Disable **External authentication**.
4. Click **Save**
5. Log out from Mediation Engine Connector.

Before Taking the Backup

- The Time zone and system time of both - the source and the destination machine, used for backup and restore, must be same.

Preparing Session Monitor for Taking the Backup

This section describes the steps required for preparing the Session Monitor for taking the backup.

1. Disconnect all probes: both standalone and SBC probes, connected to the Mediation Engine Connector so that the Mediation Engine does not receive any traffic. For the Fraud Monitor, disconnect the Mediation Engines connected to Fraud Monitor. For the Mediation Engine Connector, disconnect the Mediation Engine connected to the Mediation Engine Connector.
2. Run this command to stop the **pldclean** service. (Applicable to the Mediation Engine only):

   ```
   sed -i "s/^12/#&/" /opt/oracle/ocsm/usr/share/pld/configs/me/pld-me.cron.d
   sed -i "s/^42/#&/" /opt/oracle/ocsm/usr/share/pld/configs/me/pld-me.cron.d
   systemctl restart crond.service
   ```

   This helps prevent any data loss caused by the **pldclean** service (the **pldclean** service deletes data regularly based on the retention configured in the PSA.)
3. From the **PSA Page**, create Historical System Diagnostics with the **Create savepoint** and **Include mysqldump** check boxes enabled.
4. Download a copy of the Diagnostics created in the above step, to your Backup location. For more information, see System Diagnostics' section in the Session Monitor Release 5.2 Installation Guide.

   > **Note:**
   >
   > Creating the Savepoint is applicable only for the Mediation Engine. Also, selecting the **Create savepoint** and **Include mysqldump** check boxes is mandatory for taking a backup.

5. Go through each page of the PSA and take screenshots of the page. This data is needed during the Session Monitor Fresh Installation phase of the Restore procedure, (if required).
6. Run the following commands to stop all Session Monitor services:

   ```
   source /opt/oracle/ocsm/ocsm_env.sh
   pld-systemctl stop
   ```

   This helps in preventing the Session Monitor services from writing any new data during the Backup process.

> **Note:**
>
> Capture any additional configurations from the UI before stopping the Session Monitor services as needed.

7. If the Backup location selected is on a Remote Server (either the Remote Server selected as part of Strategy-1 OR the newly created target machine as part of Strategy-2), generate an SSH Key to authorize the Remote Server for passwordless SSH logins by executing below steps on the current Session Monitor Server (which is going to be backed up):

> **Note:**
>
> If the file authorized_keys is already present under the `/root/.ssh/` folder in the Remote Server, rename that file as authorized_keys_orig using this command:
>
> ```
> mv /root/.ssh/authorized_keys /root/.ssh/authorized_keys_orig
> ```

   a. Log in to the CLI of the current Session Monitor Server as the **root** user.

   b. Type **ssh-keygen** and keep pressing the **Enter** key until the SSH Key is generated.

   For example:

   ```
   [root@localhost ~]# ssh-keygen

   Generating public/private rsa key pair.
   Enter file in which to save the key (/root/.ssh/id_rsa):
   Enter passphrase (empty for no passphrase):
   Enter same passphrase again:
   Your identification has been saved in /root/.ssh/id_rsa.
   Your public key has been saved in /root/.ssh/id_rsa.pub.
   The key fingerprint is:
   SHA256:ez/o897z0lOwGElieJUYlMA25W8gd071IKtuzUqga1g root@localhost
   ```

   c. Run the following command to copy the SSH Key generated, onto the Remote Server:

   ```
   ssh-copy-id -i ~/.ssh/id_rsa.pub <User>@<Remote_Server_IP>
   ```

   For example:

   ```
   [root@localhost ~]# ssh-copy-id -i ~/.ssh/id_rsa.pub root@10.11.12.13

   /usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/
   root/.ssh/id_rsa.pub"
   The authenticity of host '10.11.12.13 (10.11.12.13)' can't be
   established.
   ECDSA key fingerprint is
   SHA256:sEAQZxN2alX76X1rPZcVRKARGczMIZaa+Z4CNTQuTd8.
   Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
   /usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s),
   ```

```
to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you
are prompted now it is to install the new keys
root@10.11.12.13's password:

Number of key(s) added: 1
```

8. Run the following commands to install rsync and parallel on the current Session Monitor Server:

```
yum install rsync
yum install parallel
```

Taking a Backup of the Essential Session Monitor Files

This section describes the procedure for taking a backup of the essential files and folders of Session Monitor. **Instructions in this section are applicable only if the node type is Mediation Engine**.

The following files and folders are backed up as a part of the backup script backupAndRestoreOtherFiles.sh:

- All local.conf files from directory /opt/oracle/ocsm/etc/iptego/
- Savepoints Directory - /opt/oracle/ocsm/var/vsi/savepoints/
- Traces Directory - /opt/oracle/ocsm/var/vsi/dumps/
- Saved Calls Directory - /opt/oracle/ocsm/var/vsi/saved/
- Uploaded Apps Directory - /opt/oracle/ocsm/var/vsi/pscripts/upload/
- cdr, mdr, tdr Directory - /opt/oracle/ocsm/var/vsi/ftp/
- CSV Exports Directory - /opt/oracle/ocsm/var/vsi/exports/
- Packet Inspector Search Result Directory - /opt/oracle/ocsm/var/vsi/pint_results/
- Version Files - /opt/oracle/ocsm/etc/iptego/version and /opt/oracle/ocsm/etc/iptego/version.history

1. Get the backupAndRestoreOtherFiles.sh script file present in the Session Monitor installation software RPM .zip file.
2. Copy the backupAndRestoreOtherFiles.sh script file to the `/root/` directory of your Session Monitor Server.
3. Run this command to provide the necessary permissions for the script:

```
chmod +x backupAndRestoreOtherFiles.sh
```

4. Execute the script to begin taking backup of the essential Session Monitor Files:

```
./backupAndRestoreOtherFiles.sh -a backup
```

You can choose the location to copy the backup on the target machine (Remote Server or Shared Drive). When prompted, the script asks you for IP/path or Path to store the backup based on the input.

> **Note:**
>
> Once the script has run successfully, you can manually copy any other additional folder or files which are considered to be important, from the source machine to the target machine.

Example for Remote Server:

(For more information on setting up passwordless login to remote server, see Generate an SSH Key to authorize the Remote Server for passwordless SSH logins).

```
[root@localhost  ~]# ./backupAndRestoreOtherFiles.sh -a backup

Starting Backup of Essential OCSM Files...

Where do you want to copy the Essential OCSM Files Backup ?
     1. Remote Server (e.g. SAN, newly created OCSM as part of Strategy-2
etc.)
     2. Mounted Disk (e.g. NFS/NAS/DAS etc.)
Your Input is:
1
Remote Server's IP:
10.184.19.114
Remote Server's User (used to ssh):
root
Remote Server's Path to Store Backup:
/root/ocsmBackup/ocsmFilesBackup/

Copying in progress ! Please wait until finished...
```

Example of a Mounted Disk:

```
[root@localhost ~]# ./backupAndRestoreOtherFiles.sh -a backup

Starting Backup of Essential OCSM Files...

Where do you want to copy the Essential OCSM Files Backup ?
     1. Remote Server (e.g. SAN, newly created OCSM as part of Strategy-2
etc.)
     2. Mounted Disk (e.g. NFS/NAS/DAS etc.))
Your Input is:
2
Backup Path of Mounted Disk:
/mnt/oracle/ocsmBackup/ocsmFilesBackup/

Copying in progress ! Please wait until finished...
```

> **Note:**
>
> During the backup process, If copying of any backup files are interrupted due to any network connection issue, system restarts, etc. Re-run the script to resume copying.

Block Storage Backup

This section describes the procedure for taking the backup of Session Monitor's Block Storage.

IMPORTANT: This section is applicable only if the node type is Mediation Engine.

It can take hours to complete the backup of Block Storage, depending on the size of Block Storage and the network bandwidth between the source and the target machine. In our testing in lab, it took ~1 day 21 hours to complete the backup procedure for Block Storage of size ~19 TB with an average network speed of ~123MB/sec.

- Take a call on the task - if you want to copy the Block Storage data directly on the Target Machine (where Release 5.2 RPM will be installed) as part of Strategy-2.
- First copy the Block Storage data onto some temporary network drive and then copy the Block Storage data from the temporary network drive to the Target Machine as part of Stragegy-1. If you select Strategy - 2, the total time taken to complete the procedure will be double.

1. Check the space availability on both Source and Target Machines.
   a. Run this command to check the block storage size of your Session Monitor Server (Source Machine).

      ```
      du -sh /opt/oracle/ocsm/var/vsi/storage/
      ```

   b. Run the following command on the backup location of the Target Machine (Remote Server or Shared Drive) to get the available space:

      ```
      df -kh --output=avail /path/to/copy/backup
      ```

   c. Compare the output of the two commands, and ensure that the available space in the Target Machine is **greater** than the block storage size of the Source Machine.
2. Get the backupAndRestoreBlockStorage.sh script file present in the Session Monitor installation software RPM .zip file.
   - Copy the file backupAndRestoreBlockStorage.sh to the `/root/` directory of your Session Monitor Server.
3. If the backup location is selected on a Remote Server (it could be either the Remote Server selected as part of Strategy-1 or the newly created Target machine as part of Strategy-2), make sure that you have enabled passwordless login by transferring SSH key to Remote Server. For more information, see Generate an SSH Key to authorize the Remote Server for passwordless SSH logins.
4. Execute the backupAndRestoreBlockStorage.sh script.
   a. Run the following command to provide the necessary permissions for the script:

      ```
      chmod +x backupAndRestoreBlockStorage.sh
      ```

   b. To begin the backup of block storage, execute the script:

      ```
      ./backupAndRestoreBlockStorage.sh
      ```

      Select the location to copy the backup (Remote Server or Shared Drive) when prompted. The script prompts for the IP address and the path to store the backup based on your input.

> **Note:**
>
> If the Remote Server is the newly created Target Machine as part of Strategy 2, the block storage data must be directly copied to the location `/opt/oracle/ocsm/var/vsi/storage/`. Make sure Session Monitor services are stopped on the Target Machine before copying the block storage data.

Example for Remote Server:

```
[root@localhost ~]# ./backupAndRestoreBlockStorage.sh

Starting Backup of Block Storage...

Where do you want to copy the Backup ?
     1. Remote Server (e.g. SAN, newly created OCSM as part of
Strategy-2 etc.)
     2. Mounted Disk (e.g. NFS/NAS/DAS etc.)
Your Input is:
1
Remote Server's IP:
10.184.19.114
Remote Server's User (used to ssh):
root
Remote Server's Path to Store Backup:
/root/ocsmBackup/blockStorageBackup/ OR </opt/oracle/ocsm/var/vsi/
storage/> (if Remote server is Target machine as part of Strategy-2)

Block storage backup copying in progress ! Please wait until finished...
```

Example for Mounted Disk:

```
[root@kvm248-109-vm9 ~]# ./backupAndRestoreBlockStorage.sh

Starting Backup of Block Storage...

Where do you want to copy the Backup ?
     1. Remote Server (e.g. SAN, newly created OCSM as part of
Strategy-2 etc.)
     2. Mounted Disk (e.g. NFS/NAS/DAS etc.)
Your Input is:
2
Backup Path of Mounted Disk:
/mnt/oracle/ocsmBackup/blockStorageBackup/

Block storage backup copying in progress ! Please wait until finished...
```

> **Note:**
>
> During the backup process, if copying of any backup files is interrupted due to any network connection issue, system restarts, and so on, run the script again to resume copying.

MySQL Backup (From Version 4.4 & 5.0)

This section guides you with the procedure required for taking backup of Session Monitor's MySQL Data. This section is applicable only for taking MySQL backup from Session Monitor with version 4.4 and 5.0.

1. **Installing MySQL shell utility**

   a. Download the latest mysql80-community-release rpm file for Oracle Linux 7 from https://dev.mysql.com/downloads/repo/yum/ and copy it to the current Session Monitor Server.

   > **Note:**
   >
   > 'mysql80-communityrelease-el7-7.noarch.rpm' has been used here.

   b. Run these commands to install mysql80-community-release rpm:

   ```
   sudo yum remove mysql-community-release
   rpm -ivh <mysql80-community-release-xxxxx>.rpm
   ```

   For example:

   ```
   rpm -ivh mysql80-community-release-el7-7.noarch.rpm
   ```

   c. Enable 'mysql80-community-source' from repo by running the following command:

   ```
   sed -i '/mysql80-community-source/{ n; n; n; s/enabled=0/
   enabled=1/g }' /etc/yum.repos.d/mysql-community-source.repo
   ```

   d. Verify yum search returns success by running the following command:

   ```
   yum search mysql-shell
   ```

   Sample Output:

   ```
   .
   .
   Repository ol7_UEKR4_archive is listed more than once in the
   configuration
   Repository ol7_UEKR5_archive is listed more than once in the
   configuration
   Repository ol7_kvm_utils is listed more than once in the configuration
   ================================================================================
   ==================== N/S matched: mysql-shell
   ================================================================================
   ==============================
   mysql-shell.x86_64 : Command line shell and scripting environment for
   MySQL
   .
   .
   ```

e. Run the following command to install mysql-shell:

```
yum install mysql-shell
```

f. mysql-shell is installed. Confirm by typing `mysqlsh --no-defaults`, and the mysql-shell console must open. By default it opens JS mode.

Output:

```
[root@localhost ~]# mysqlsh --no-defaults
MySQL Shell 8.0.32

Copyright (c) 2016, 2023, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates.
Other names may be trademarks of their respective owners.

Type '\help' or '\?' for help; '\quit' to exit.
MySQL  JS >
```

> **Note:**
>
> Type `"\quit"` to exit from the mysql shell.

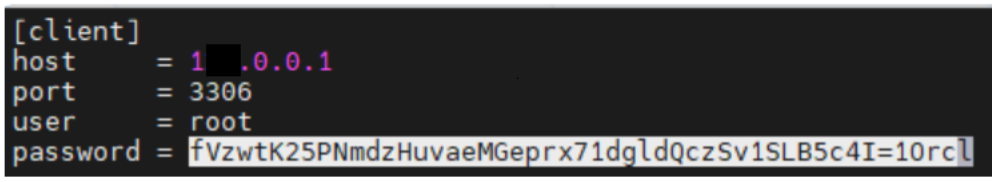
2. Create a temporary directory structure in `/root/` or any other location of the current Session Monitor Server where the space is available.

For example:

```
mkdir /root/mysqlBackup/
```

3. Copy the password from `/root/.my.cnf` and keep it handy.

Figure 3-1 copy password



Starting the MySQL Backup Procedure

Run the commands in sequence to start the MySQL Backup procedure:

1. Type `mysqlsh --no-defaults` command. The JS prompt is opened.
2. Run the following commands one by one at the JS prompt.
   a. Connect to a MySQL instance by typing the below command:

```
\connect root@localhost:3306
```

b. When prompted for the password, paste the password copied from /root/.my.cnf in MySQL Backup (From Version 4.4 & 5.0).

c. Run this command to begin backup:

```
util.dumpSchemas(["<DATABASE>"],"<PATH_TO_MYSQL_BACKUP_DIRECTORY>/
<DATABASE>.dump",{threads:88})


For Mediation Engine, use DATABASE = vsp
For Mediation Engine Connector, use DATABASE = pldmaster
For Fraud Monitor, use DATABASE = fdp
```

Example For Mediation Engine:

```
util.dumpSchemas(["vsp"] "/root/mysqlBackup/vsp.dump",{threads:88})
```

> **Note:**
>
> The total time to take a dump of all schemas depends on the size of database as well as the number of CPUs in the source machine. In our testing in lab, for a VSP database of size 1.1 TB and 96 CPUs in source machine, it took 6 minutes to complete the dump. Refer to the official MySql Shell Utility document for more details: MySQL Shell 8.0.

d. Once the backup is successful, a message is displayed as similar to the below sample:

```
105% (9.97M rows / ~9.47M rows), 26.82K rows/s, 559.12 KB/s
uncompressed, 17.04 KB/s compressed
Dump duration: 00:00:52s
Total duration: 00:00:54s
Schemas dumped: 1
Tables dumped: 63
Uncompressed data size: 260.70 MB
Compressed data size: 7.81 MB
Compression ratio: 33.4
Rows written: 9970295
Bytes written: 7.81 MB
Average uncompressed throughput: 4.93 MB/s
Average compressed throughput: 147.71 KB/s
```

e. Type this command to exit from mysql-shell:

```
\quit
```

3. Run this command to verify a <DATABASE>.dump directory is created under the directory created from MySQL Backup (From Version 4.4 & 5.0):

```
ls -lh /root/mysqlBackup/
```

For example:

```
Example For Mediation Engine,
[root@localhost ~]# ls -lh /root/mysqlBackup/
total 1.2M
drwxr-x---. 2 root root 1.2M Aug 19 13:27 vsp.dump
```

Change the Character Set Encoding of MySQL Backup

Change the character set encoding of the MySQL backup from utf8 to utf8mb4. This is required as the default character set in MySql 8.0 is utf8mb4.

1. Get the convertUTF8ToUTF8mb4.sh script file present in the Session Monitor installation software rpm .zip file
2. Copy the convertUTF8ToUTF8mb4.sh script to the same folder where the <DATABASE>.dump directory was created.

   For example

   ```
   [root@localhost ~]# mv convertUTF8ToUTF8mb4.sh /root/mysqlBackup/
   [root@localhost ~]# ls -lrt /root/mysqlBackup/
   total 144
   -rwxr-xr-x. 1 root root    472 Mar 10 04:51 convertUTF8ToUTF8mb4.sh
   drwxr-x---. 2 root root 143360 Mar 23 08:46 vsp.dump
   ```

3. Navigate to the MySQL Backup directory (For example, `cd /root/mysqlBackup/`) and run this command to provide the necessary permissions:

   ```
   chmod +x convertUTF8ToUTF8mb4.sh
   ```

4. Execute the script:

   ```
   ./convertUTF8ToUTF8mb4.sh
   ```

Taking MySQL Dump of the Blocks Table

Take MySQL dump of blocks table. **This step is required only if the node type is Mediation Engine.**

1. Move to the MySQL Backup directory (For example, `cd /root/mysqlBackup/`) and take a MySQL dump of the Blocks table. Run this command:

   ```
   mysqldump --skip-add-drop-table --skip-add-locks --no-create-info --
   replace vsp blocks > blocks_replace.sql
   ```

2. Verify the `blocks_replace.sql` is created under the MySQL Backup directory. For example:

   ```
   [root@localhost ~]# ls -lrt /root/mysqlBackup/
   total 144
   -rwxr-xr-x. 1 root root    472 Mar 10 04:51 convertUTF8ToUTF8mb4.sh
   drwxr-x---. 2 root root 143360 Mar 23 08:46 vsp.dump
   -rwxr-xr-x. 2 root root 143360 Mar 23 08:46 blocks_replace.sql
   ```

> **Note:**
>
> This step helps to fix any discrepancy caused by difference in the number of blocks between the Source Machine and the Target Machine.

Copy the MySQL Backup Directory to the Target Machine

Check the space availability on both Source and Target Machines, and copy the MySQL Backup directory to the Target Machine (Remote Server or Shared Drive).

1. Run this command to check the MySQL backup directory size on the current Session Monitor Server (Source Machine) by running the following command.

   ```
   du -sh <path to mysql backup folder>
   ```

2. Run the following command on the backup location of the Target Machine (Remote Server or Shared Drive) to get the available space:

   ```
   df -kh --output=avail <path to copy backup>
   ```

3. Compare the outputs of the above commands, and make sure the available space (in Step #2) is greater than the MySQL backup size directory size (in Step #1).
4. Copy the MySQL Backup directory to the Target Machine (Remote Server or Shared Drive).
   a. For the Remote Server:

      Copy the MySQL Backup directory containing the <DATABASE>.dump directory and the blocks_replace.sql file by running the following scp command:

      ```
      scp -r <PATH_TO_MYSQL_BACKUP_DIRECTORY>
      <User>@<Target_Machine_IP>:<path to copy backup>
      ```

      For example:

      ```
      scp -r /root/mysqlBackup  root@1.2.3.4:/root/ocsmBackup/
      ```

      Here, the Remote Server is either the Remote Server selected as part of Strategy-1 OR the newly created Target machine as part of Strategy-2.

      (or)

   b. For the Shared Drive:

      Transfer the MySQL Backup directory containing <DATABASE>.dump directory and `blocks_replace.sql` file by running the following command:

      ```
      cp -r <PATH_TO_MYSQL_BACKUP_DIRECTORY> <path to copy backup>
      ```

      For example:

      ```
      cp -r /root/mysqlBackup  /mnt/oracle/ocsmBackup/
      ```

> **Note:**
>
> The time to copy the MySQL backup folder depends on the size of directory as well as the network bandwidth between the source and the target machine.

MySQL Backup (From Version 5.1)

This section provides information on the procedure required for taking backup of Session Monitor's MySQL Data. This section is applicable only for taking MySQL backup from Session Monitor with version 5.1 and restoring it on Session Monitor with version 5.1. This difference is required as here the MySQL shell utility is to be installed from MySql8.x package instead of MySQL 5.x package.

1. Installing MySQL shell utility:
   a. Go to the folder where MySQL RPM files are present. (The MySQL 8 Commercial Package was downloaded from the Oracle software delivery during Session Monitor rpm installation.). cd mysql-8.X/8.X.XX/. For example: cd /root/mysql-8.0/8.0.32/.
   b. Run the following command to install the MySQL shell rpm:

      ```
      yum install mysql-shell-commercial-X.X.XX-X.X.XXX.x86_64.rpm
      ```

      For example:

      ```
      yum install mysql-shell-commercial-8.0.32-1.1.el8.x86_64.rpm
      ```

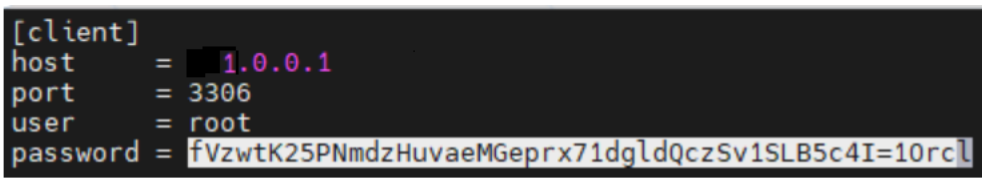
> **Note:**
>
> The mysql-shell rpm is available as a part of the MySQL package itself. So use the same.

2. Create a temporary directory structure under /root/ or any other location on your Session Monitor Server where space is available.

   For example: `mkdir /root/mysqlBackup/`.

3. Copy the password from /root/.my.cnf and keep it handy.

Figure 3-2 Copy Password



4. Type `mysqlsh --no-defaults` command, This opens the JS prompt. Run this commands in sequence at the JS prompt:

a. Run this command to connect to a MySQL instance:

```
\connect root@localhost:3306
```

b. When prompted for password. paste the password as copied from /root/.my.cnf earlier in Step 3.

c. Run this command to begin backup:

```
util.dumpSchemas(["<DATABASE>"],"<PATH_TO_MYSQL_BACKUP_DIRECTORY>/
<DATABASE>.dump",{threads:88})
```

For Mediation Engine, use DATABASE = vsp

For Mediation Engine Connector, use DATABASE = pldmaster

For Fraud Monitor, use DATABASE = fdp

For example, in Mediation Engine:

util.dumpSchemas(["vsp"],"/root/mysqlBackup/vsp.dump",{threads:88})

Once the backup is successful, a message is displayed similar to the below sample:

```
105% (9.97M rows / ~9.47M rows), 26.82K rows/s, 559.12 KB/s
uncompressed, 17.04 KB/s compressed
Dump duration: 00:00:52s
Total duration: 00:00:54s
Schemas dumped: 1
Tables dumped: 63
Uncompressed data size: 260.70 MB
Compressed data size: 7.81 MB
Compression ratio: 33.4
Rows written: 9970295
Bytes written: 7.81 MB
Average uncompressed throughput: 4.93 MB/s
Average compressed throughput: 147.71 KB/s
```

d. To exit from mysql-shell run this command:

```
\quit
```

5. Verify a <DATABASE>.dump directory has been created under the directory created from Step 2. Run this command: .

```
ls -lh /root/mysqlBackup/
```

Example For Mediation Engine:

```
[root@localhost ~]# ls -lh /root/mysqlBackup/
total 1.2M
drwxr-x---. 2 root root 1.2M Aug 19 13:27 vsp.dump
```

Taking MySQL Dump of the Blocks Table

Take MySQL dump of blocks table. This step is required only if the node type is Mediation Engine.

1. Move to the MySQL Backup directory (For example, `cd /root/mysqlBackup/`) and take a MySQL dump of the Blocks table. Run this command:

   ```
   mysqldump --skip-add-drop-table --skip-add-locks --no-create-info --
   replace vsp blocks > blocks_replace.sql
   ```

2. Verify the `blocks_replace.sql` is created under the MySQL Backup directory. For example:

   ```
   [root@localhost ~]# ls -lrt /root/mysqlBackup/
   total 144
   drwxr-x---. 2 root root 143360 Mar 23 08:46 vsp.dump
   -rwxr-xr-x. 2 root root 143360 Mar 23 08:46 blocks_replace.sql
   ```

> **Note:**
>
> This step helps to fix any discrepancy caused by difference in the number of blocks between the Source Machine and the Target Machine.

Copy the MySQL Backup Directory to the Target Machine

Check the space availability on both Source and Target Machines, and copy the MySQL Backup directory to the Target Machine (Remote Server or Shared Drive).

1. Run this command to check the MySQL backup directory size on the current Session Monitor Server (Source Machine) by running the following command.

   ```
   du -sh <path to mysql backup folder>
   ```

2. Run the following command on the backup location of the Target Machine (Remote Server or Shared Drive) to get the available space:

   ```
   df -kh --output=avail <path to copy backup>
   ```

3. Compare the outputs of the above commands, and make sure the available space (in Step #2) is greater than the MySQL backup size directory size (in Step #1).
4. Copy the MySQL Backup directory to the Target Machine (Remote Server or Shared Drive).
   a. For the Remote Server:

      Copy the MySQL Backup directory containing the <DATABASE>.dump directory and the blocks_replace.sql file by running the following scp command:

      ```
      scp -r <PATH_TO_MYSQL_BACKUP_DIRECTORY>
      <User>@<Target_Machine_IP>:<path to copy backup>
      ```

For example:

```
scp -r /root/mysqlBackup  root@1.2.3.4:/root/ocsmBackup/
```

Here, the Remote Server is either the Remote Server selected as part of Strategy-1 OR the newly created Target machine as part of Strategy-2.

(or)

b. For the Shared Drive:

Transfer the MySQL Backup directory containing <DATABASE>.dump directory and `blocks_replace.sql` file by running the following command:

```
cp -r <PATH_TO_MYSQL_BACKUP_DIRECTORY> <path to copy backup>
```

For example:

```
cp -r /root/mysqlBackup  /mnt/oracle/ocsmBackup/
```

> **Note:**
>
> The time to copy the MySQL backup folder depends on the size of directory as well as the network bandwidth between the source and the target machine.

Redis Backup

This section describes the procedure for taking the backup of Redis data from Fraud Monitor. **This section is applicable only if the node type is Fraud Monitor**.

1. Login as root to the console of current Fraud Monitor Server.
2. Open the redis-cli by running following commands:

```
source /opt/oracle/ocsm/ocsm_env.sh
redis-cli
```

3. From redis-cli, take a dump of the latest Redis dataset by typing following command:

```
127.0.0.1:6379> SAVE
```

This creates a dump.rdb file in the Redis directory (`/opt/oracle/ocsm/var/lib/redis/`).

4. Go to `/opt/oracle/ocsm/var/lib/redis/` directory, and verify that the dump.rdb file has been created.

```
[root@localhost ~]# cd /opt/oracle/ocsm/var/lib/redis/
[root@localhost redis]# ls -lrt
total 4
-rw-r--r--. 1 redis redis 4090 Mar 26 23:08 dump.rdb
```

5. Transfer the dump.rdb file to the backup location of the Target Machine (Remote Server or Shared Drive):

   a. For Remote Server:

      Transfer the dump.rdb file by running the following scp command:

      ```
      scp -r /opt/oracle/ocsm/var/lib/redis/dump.rdb
      <User>@<Target_Machine_IP>:<path to backup location>
      ```

      For example:

      ```
      scp -r /opt/oracle/ocsm/var/lib/redis/dump.rdb  root@1.2.3.4:/root/
      ocsmBackup/redisBackup/
      ```

      Here, the Remote Server is either the Remote Server selected as part of Strategy-1 OR the newly created Target machine as part of Strategy-2.

      Or

   b. For the Shared Drive:

      Transfer the dump.rdb file by running the following cp command:

      ```
      cp -r /opt/oracle/ocsm/var/lib/redis/dump.rdb <path to backup location>
      ```

      For example:

      ```
      cp -r /opt/oracle/ocsm/var/lib/redis/dump.rdb   /mnt/oracle/ocsmBackup/
      redisBackup/
      ```

Backup is now complete. If the Backup procedure was used as part of Upgrading Session Monitor continue with the Upgrade Guide for further instructions; else proceed with the instructions in the Post Backup Tasks section.

Post Backup Tasks

This section describes the things to be taken care of after the backup procedure is complete which enables you to reuse current Session Monitor again.

1. Start Session Monitor services by running the following command:

   ```
   source /opt/oracle/ocsm/ocsm_env.sh
   pld-systemctl start
   ```

2. Start **pldclean** by running the following command (This is applicable only for Mediation Engine):

   ```
   sed -i "/^#12/s/^#//g" /opt/oracle/ocsm/usr/share/pld/configs/me/pld-
   me.cron.d
   sed -i "/^#42/s/^#//g" /opt/oracle/ocsm/usr/share/pld/configs/me/pld-
   me.cron.d
   systemctl restart crond.service
   ```

4

Restoring Backup

Session Monitor supports restoring the backups of Block Storage, MySQL Database, Redis Database, and essential Session Monitor Files.

This section describes the procedure for restoring the Session Monitor's backup taken using the instructions and information provided in the section Creating a Backup of Session Monitor.

Prerequisites for Restoring Backup

This section describes the prerequisites required for the Session Monitor restore procedure.

- Review the complete Restore procedure thoroughly before starting with the actual restore.
- The Time zone and system time of the machine used for restoring must be same as that of machine from which backup is taken.
- Some steps and sections are applicable only to specific Nodes. Lookout for such Notes in the respective step or section.

> **Note:**
>
> For the Mediation Engine Connector, if the previously connected Mediation Engine IP address requires any change to the post-restore procedure, then it is recommended that you do a fresh installation rather than restore. The reason is, that all panels and other configuration added, are stored in the database based on the previous ME IP. So, if the Mediation Engine IP address has changed, this can flood many errors in system and existing data will not be useful.

Preparing for the Restore

If the backup was created as part of Strategy-1, upgrade the current Session Monitor server with release 4.x or 5.0 to Release 5.1. For more information, see the Session Monitor Release 5.1 Upgrade Guide.

If the upgrade is successful, then there is no need to restore anything. However, if there is an issue in the upgrade procedure, and the existing data in the machine is lost, (for any reason), follow the instructions given here to restore data from the remote disk or the shared drive.

If the backup was created as part of Strategy-2, then a freshly installed Session Monitor server already has all the necessary block storage data. So, in this case, block storage restore procedure is not required.

Restore Procedure for Backup Created Using Strategy 1

This section describes the necessary steps required for preparing the Session Monitor for restore in case the upgrade to Release 5.1 fails and the original data is lost as part of Strategy-1.

1. Re-install Session Monitor version 5.1. Perform a fresh installation of Session Monitor Release 5.1. For more information, see the Session Monitor Release 5.1 Installation Guide.
2. Configure the newly installed Session Monitor with the same node type (Mediation Engine/ Fraud Monitor/Mediation Engine Connector) as that of original Session Monitor Server. Configure the node with the same PSA configuration as present in the original Session Monitor server. Refer to the data or screenshots taken during the Preparing Session Monitor for Taking the Backup step.
3. After Node configuration, log in to the application to verify if the node has been successfully installed.
4. Change the password and log out.

> **Note:**
>
> Do not perform any other actions on the system until the Restore procedure, as specified in next sections is complete.

5. Run this CLI commands to stop all Session Monitor services:

```
source /opt/oracle/ocsm/ocsm_env.sh
pld-systemctl stop
```

6. Stop the crond.service by using the command:

```
systemctl stop crond.service
```

Generate SSH Key

If your backup is located on a Remote Server (as we are following restore for backup created using Strategy-1, so here the Remote Server refers to the SAN type etc. and NOT the target Session Monitor machine), generate SSH Key to authorize Session Monitor Server for password less SSH logins by executing below steps on newly installed Session Monitor Server.

Note: If both the files authorized_keys and authorized_keys_orig are present under the `/root/.ssh/` folder in the Remote Server, delete the file authorized_keys using command: `rm /root/.ssh/authorized_keys`

1. Log in to the CLI of Session Monitor Server as root user.
2. Type 'ssh-keygen' and keep pressing the **Enter** key until the SSH Key is generated.

   For example:

```
[root@ocsm-server~]# ssh-keygen

Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
...
```

3. Copy the generated SSH Key to your Remote Server by running the following command:

```
ssh-copy-id -i ~/.ssh/id_rsa.pub <User>@<Remote_Server_IP>
```

> **Note:**
>
> If the ssh-copy-id command is not present in the new server, please install it using `yum install openssh-clients`.

For example:

```
[root@ocsm-server ~]# ssh-copy-id -i ~/.ssh/id_rsa.pub root@10.11.12.13

/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/
id_rsa.pub"
The authenticity of host '10.11.12.13 (10.11.12.13)' can't be established.
ECDSA key fingerprint is
SHA256:sEAQZxN2alX76X1rPZcVRKARGczMIZaa+Z4CNTQuTd8.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to
filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are
prompted now it is to install the new keys
root@10.11.12.13's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'root@10.11.12.13'"
and check to make sure that only the key(s) you wanted were added.
```

Install rsync and parallel

Install rsync and parallel on your Session Monitor Server.

- Run this command:

```
yum install rsync
yum install parallel
```

Restoring Essential Session Monitor Files

This section describes the procedure for restoring the backup of essential files and folders of Session Monitor. **This section is applicable only if the node type is Mediation Engine**.

The following files and folders are restored as a part to below script:

- All local.conf files from directory /opt/oracle/ocsm/etc/iptego/
- Savepoints Directory - /opt/oracle/ocsm/var/vsi/savepoints/
- Traces Directory - /opt/oracle/ocsm/var/vsi/dumps/
- Saved Calls Directory - /opt/oracle/ocsm/var/vsi/saved/

- Uploaded Apps Directory - /opt/oracle/ocsm/var/vsi/pscripts/upload/
- cdr, mdr, tdr Directory - /opt/oracle/ocsm/var/vsi/ftp/
- CSV Exports Directory - /opt/oracle/ocsm/var/vsi/exports/
- Packet Inspector Search Result Directory - /opt/oracle/ocsm/var/vsi/pint_results/
- Version Files - /opt/oracle/ocsm/etc/iptego/version and /opt/oracle/ocsm/etc/iptego/version.history

1. Get the `backupAndRestoreOtherFiles.sh` script present in the Session Monitor installation software rpm .zip file.
2. Copy the script to the /root/ directory of your newly installed Session Monitor Server.
3. If you are planning to restore the backup from a Remote Server, make sure you have enabled passwordless login by transferring SSH key from Session Monitor Server to Remote Server. For more information, see Generate SSH Key.
4. Run this command to provide the necessary permissions for the script:

   ```
   chmod +x backupAndRestoreOtherFiles.sh
   ```

5. To begin restoring essential Session Monitor files, execute the script:

   ```
   ./backupAndRestoreOtherFiles.sh -a restore
   ```

   This will download all the backup files from Remote Server or Shared Drive on to your new Session Monitor Server. You can choose the location of your backup (Remote Server or Shared Drive) when prompted. The script will ask for IP address and the Path of the backup based on your input.

   Example for Remote Server:

   ```
   [root@ocsm-server ~]# ./backupAndRestoreOtherFiles.sh -a restore

   Starting Restore of Essential OCSM Files...

   Where is your Essential OCSM Files Backup Data present ?
        1. Remote Server
        2. Mounted Disk
   Your Input is:
   1
   Remote Server's IP:
   10.184.19.114
   Remote Server's User (used to ssh):
   root
   Backup Directory Location of Remote Server:
   /root/ocsmBackup/ocsmFilesBackup/

   Restoring backup in progress ! Please wait until finished...
   ```

   Example for mounted disk

   ```
   [root@ocsm-server ~]# ./backupAndRestoreOtherFiles.sh -a restore

   Starting Restore of Essential OCSM Files...
   ```

```
Where is your Essential OCSM Files Backup Data present ?
     1. Remote Server
     2. Mounted Disk
Your Input is:
2
Backup Path of Mounted Disk:
/mnt/oracle/ocsmBackup/ocsmFilesBackup/

Restoring backup in progress ! Please wait until finished...
```

> **Note:**
>
> During the restore process, If copying of any backup files are interrupted due to any network connection issue, system restarts, etc. Re-run the script to resume copying.

Restoring Block Storage

This section describes the procedure for restoring the backup of Session Monitor's Block Storage. **This section is applicable only if node type is Mediation Engine**.

1. If you are planning to restore the backup from a Remote Server, make sure you have enabled passwordless login by transferring SSH key from Remote Server to Session Monitor Server. For more information, see Generate SSH Key
2. Transfer the Block Storage Backup from the Remote Server or Shared Drive. This will download all the Block Storage Backup files from Remote Server or Shared Drive on to your new Session Monitor Server.

   - For Remote Server:

     Transfer the Block Storage Backup by running the following rsync command in the newly installed Session Monitor Server:

     ```
     rsync -avh
     <User>@<Remote_Server_IP>:<path_to_MySQL_backup_directory> /opt/oracle/
     ocsm/var/vsi/storage/
     ```

     For example:

     ```
     rsync -avh root@10.184.19.114:/root/ocsmBackup/blockStorageBackup/ /opt/
     oracle/ocsm/var/vsi/storage/
     ```

     or

   - For the Shared Drive:

     Transfer the Block Storage Backup by running the following rsync command in the newly installed Session Monitor Server:

     ```
     rsync -avh <path_to_block_backup_directory> /opt/oracle/ocsm/var/vsi/
     storage/
     ```

For example:

```
rsync -avh /mnt/oracle/ocsmBackup/blockStorageBackup/  /opt/oracle/
ocsm/var/vsi/storage/
```

> **Note:**
>
> During the restore process, If copying of block storage files are interrupted due to any network connection issue, system restarts, etc. Re-run the same above rsync command to resume the copying.

> **Note:**
>
> It can take significant time to restore the block storage files depending on the block storage data and the network bandwidth.

3. Give necessary permission for blocks file by running the following commands:

```
chown -R ocsm:ocsm /opt/oracle/ocsm/var/vsi/storage/*
```

Restoring MySQL

This section guides you with the procedure required for restoring the backup of Session Monitor's MySQL Data.

1. Copy the MySQL Backup directory containing <DATABASE>.dump directory and blocks_replace.sql file from Remote Server or Shared Drive on to your new Session Monitor Server. This downloads all the MySQL Backup files from Remote Server or Shared Drive on to your new Session Monitor Server.

   For the Remote Server, run the following scp command in the newly installed Session Monitor Server:

   ```
   scp -r <User>@<Remote_Server_IP>:<path_to_MySQL_backup_directory>
   <path_to_copy_backup>
   ```

   For example:

   ```
   scp -r root@10.184.19.114:/root/ocsmBackup/mysqlBackup/ /root/
   ```

   For Shared Drive, run the following cp command in the newly installed Session Monitor Server:

   ```
   cp -r <path_to_MySQL_backup_directory> <path_to_copy_backup>
   ```

   For example:

   ```
   cp -r /mnt/oracle/ocsmBackup/mysqlBackup/  /root/
   ```

2. **This step is required only if the node type is Mediation Engine** .

a. Take a mysql dump of blocks table of newly installed Session Monitor Server by running the following command.

```
mysqldump vsp blocks > blocks_dumps.sql
```

b. Copy this blocks_dumps.sql file to MySQL backup directory transferred in STEP 1.

Installing mysql-shell Utility

Install the mysql-shell utility on the newly installed Session Monitor Server.

1. Navigate to the folder where MySQL RPMs are present (MySQL 8 Commercial Package was downloaded from the Oracle software delivery during the 5.2 Session Monitor RPM installation):

   ```
   cd mysql-8.X/8.X.XX/
   ```

   For example:

   ```
   cd /root/mysql-8.0/8.0.34/
   ```

2. Install the mysql shell rpm by running the following command:

   ```
   yum install mysql-shell-commercial-X.X.XX-X.X.XXX.x86_64.rpm
   ```

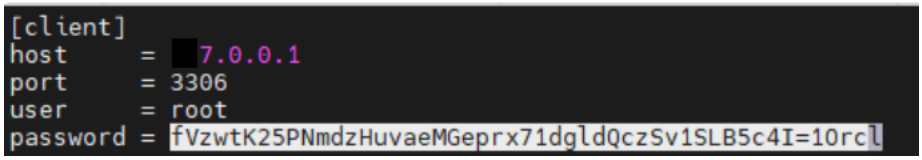
   For example:

   ```
   yum install mysql-shell-commercial-8.0.34-1.1.el8.x86_64.rpm
   ```

   > **Note:**
   >
   > The mysql-shell rpm used here is based on MySQL 8.0.34.

3. Copy the password from /root/.my.cnf and keep it handy.

Figure 4-1 Copy Password



Restoring MySQL Backup

Run the mysqlsh shell utility at the CLI prompt.

- Type `mysqlsh --no-defaults` command at the CLI prompt, it opens the JS prompt. Then run the below commands one by one in the prompt:
  1. Connect to a MySQL instance by typing the below command:

     ```
     \connect root@localhost:3306
     ```

2. When prompted for a password, paste the password as copied from `/root/.my.cnf` in Copy Password.
3. Prepare for restore by typing the below command one by one:

```
\sql set GLOBAL local_infile=1;

\sql ALTER INSTANCE DISABLE INNODB REDO_LOG;

\sql drop database <DATABASE>;
```

For Mediation Engine, use DATABASE = vsp

For Mediation Engine Connector, use DATABASE = pldmaster

For Fraud Monitor, use DATABASE = fdp

4. Run the below command to begin restore:

```
util.loadDump("/<PATH_TO_MYSQL_BACKUP_DIRECTORY>/<DATABASE>.dump",
{threads: 88, ignoreVersion: true})
```

Example For Mediation Engine:

```
 util.loadDump("/root/mysqlBackup/vsp.dump", {threads:
88,ignoreVersion:true})
```

Once the restore is successful, you will see a message similar to the below sample:

```
Target is MySQL 8.0.28-commercial. Dump was produced from MySQL 5.7.35-
enterprise-commercial-advanced-log
WARNING: Destination MySQL version is newer than the one where the dump
was created. Loading dumps from different major MySQL versions is not
fully supported and may not work. The 'ignoreVersion' option is
enabled, so loading anyway.
NOTE: Load progress file detected. Load will be resumed from where it
was left, assuming no external updates were made.
You may enable the 'resetProgress' option to discard progress for this
MySQL instance and force it to be completely reloaded.
Scanning metadata - done
Executing common preamble SQL
Executing DDL - done
Executing view DDL - done
Starting data load
88 thds loading \ 100% (260.70 MB / 260.70 MB), 8.44 MB/s, 544 / 63
tables and partitions done
Recreating indexes - done
Executing common postamble SQL
NOTE: The redo log is currently disabled, which causes MySQL to not be
crash safe! Do not forget to enable it again before putting this
instance in production.
7796 chunks (9.97M rows, 260.70 MB) for 63 tables in 1 schemas were
loaded in 1 min 40 sec (avg throughput 7.83 MB/s)
0 warnings were reported during the load.
```

5. Enable REDO_LOG by typing the below command:

```
\sql ALTER INSTANCE ENABLE INNODB REDO_LOG;
```

6. Exit from mysql-shell by typing the below command:

```
\quit
```

Execute the MySQL Delta Script

Execute the MySQL Delta script (Not Applicable for same version restore, that is, Backup & Restore from 5.1 to 5.1)

MySQL table changes part of the newer Session Monitor version is not present in the older Session Monitor version backup files. This is resolved by MySQL Delta script

1. Get the MySQLDeltaUpgrade.sh script file present in the Session Monitor installation software RPM ZIP file and Copy it to the /root/ directory of your newly installed Session Monitor Server.
2. Provide necessary permissions for the script by running the following command:

```
chmod +x MySQLDeltaUpgrade.sh
```

3. Execute the MySQL Delta script by running the following command:

```
./MySQLDeltaUpgrade.sh
```

Restore the blocks Table

Restore the blocks table. **This step is required only if the node type is Mediation Engine**.

1. Go inside the MySQL Backup folder (for example, /root/mysqlBackup)
2. Execute the following commands to restore the blocks table which fixes any discrepancy caused by difference in number of blocks between backup and restore machines.

```
mysql vsp < blocks_dumps.sql
mysql vsp < blocks_replace.sql
```

Restoring Redis

This section describes the procedure for restoring the backup of Redis data from Fraud Monitor. **This section is applicable for Fraud Monitor only**.

1. Log in as root to the console of your Fraud Monitor Server.
2. Stop the valkey service by running the following command:

```
systemctl stop pld-valkey
```

3. Transfer the Redis Backup file to newly installed Fraud Monitor. This will download all the Backup files from the remote server or shared drive on to your new Session Monitor server.

For the Remote Server, transfer the Redis Backup by running the following scp command in the newly installed Fraud Monitor Server:

```
scp -r <User>@<Remote_Server_IP>:<path_to_Redis_backup_directory>/* /opt/
oracle/ocsm/var/lib/valkey/
```

For example,

```
scp -r root@10.184.19.114:/root/ocsmBackup/redisBackup/*  /opt/oracle/
ocsm/var/lib/valkey/
```

or

For Shared Drive, Transfer the Redis Backup by running the following cp command in the newly installed Fraud Monitor Server:

```
cp -r <path_to_Redis_backup_directory>/* /opt/oracle/ocsm/var/lib/valkey/
```

For example:

```
cp -r /mnt/oracle/ocsmBackup/redisBackup/*  /opt/oracle/ocsm/var/lib/
valkey/
```

4. Provide the necessary permission for the added dump.rdb file by running the following commands:

```
chown valkey:valkey /opt/oracle/ocsm/var/lib/valkey/dump.rdb
chown 644 /opt/oracle/ocsm/var/lib/valkey/dump.rdb
```

5. Start the valkey service by running the following command:

```
systemctl start pld-valkey
```

valkey restore is now complete.

Restore Procedure for Backup Created Using Strategy 2

This section describes the steps required for preparing Session Monitor for restoring data as part of Strategy-2. As per Strategy-2, the freshly installed Session Monitor server already has all the necessary block storage data. So, in this case, block storage restore procedure would not be required.

1. Before starting restore, stop all Session Monitor services by running the following commands on CLI:

```
source /opt/oracle/ocsm/ocsm_env.sh
pld-systemctl stop
```

> **Note:**
>
> **Do not perform any other actions on the system until the Restore procedure is complete**.

2. Run the command to stop the cron service:

   ```
   systemctl stop crond.service
   ```

3. Start the crond.service again after the restore.

Restoring Essential Session Monitor Files

This section describes the procedure for restoring the backup of essential files and folders of Session Monitor. **This section is applicable only if node type is Mediation Engine**.

Your essential Session Monitor files backup is present in your new Session Monitor Server in the Backup Path provided during the section: Taking a Backup of the Essential Session Monitor Files. The following files and folders are restored as a part to the `backupAndRestoreOtherFiles.sh` script:

- All local.conf files from directory/opt/oracle/ocsm/etc/iptego/
- Savepoints Directory - /opt/oracle/ocsm/var/vsi/savepoints/
- Traces Directory - /opt/oracle/ocsm/var/vsi/dumps/
- Saved Calls Directory - /opt/oracle/ocsm/var/vsi/saved/
- Uploaded Apps Directory - /opt/oracle/ocsm/var/vsi/pscripts/upload/
- cdr, mdr, tdr Directory - /opt/oracle/ocsm/var/vsi/ftp/
- CSV Exports Directory - /opt/oracle/ocsm/var/vsi/exports/
- Packet Inspector Search Result Directory - /opt/oracle/ocsm/var/vsi/pint_results/
- Version Files - /opt/oracle/ocsm/etc/iptego/version and /opt/oracle/ocsm/etc/iptego/version.history

1. Get the backupAndRestoreOtherFiles.sh script present in the Session Monitor installation software rpm .zip file.
2. Copy the script to the /root/ directory of your newly installed Session Monitor Server.
3. Run this command to provide necessary permissions for the script:

   ```
   chmod +x backupAndRestoreOtherFiles.sh
   ```

4. To begin restoring essential Session Monitor files, execute the script:

   ```
   ./backupAndRestoreOtherFiles.sh -a restore
   ```

5. When prompted, provide input as 2 and provide the path of backup directory. Doing this restores all essential Session Monitor backup files from the Backup location on your new Session Monitor Server to respective directories.

   For example:

   ```
   [root@ocsm-server ~]# ./backupAndRestoreOtherFiles.sh -a restore
   ```

```
Starting Restore of Essential OCSM Files...

Where is your Essential OCSM Files Backup Data present ?
     1. Remote Server
     2. Mounted Disk
Your Input is:
2
Backup Path of Mounted Disk:
/root/ocsmBackup/ocsmFilesBackup/

Restoring backup in progress ! Please wait until finished...
```

Restoring MySQL

This section provides instructions on the procedure required for restoring the backup of Session Monitor's MySQL Data. .

MySQL backup will be present in your new Session Monitor Server in the Backup Path provided in the MySQL Backup (From Version 4.4 & 5.0) or MySQL Backup (From Version 5.1)procedure.

1. **This step is required only if the node type is Mediation Engine**
   a. . Run this command to take a MySQL dump of the blocks table of the newly installed Session Monitor Server:

      ```
      mysqldump vsp blocks > blocks_dumps.sql
      ```

   b. Copy this blocks_dumps.sql file to your existing MySQL backup directory.
2. Install the mysql-shell utility on the newly installed Session Monitor Server.
   a. Go to the folder where MySQL RPMs are present. (MySQL 8 Commercial Package was downloaded from the Oracle software delivery during 5.1 Session Monitor rpm installation):

      ```
      cd mysql-8.X/8.X.XX/
      ```

      For example:

      ```
      cd /root/mysql-8.0/8.0.32/
      ```

   b. Install the mysql shell rpm by running the following command:

      ```
      yum install mysql-shell-commercial-X.X.XX-X.X.XXX.x86_64.rpm
      ```

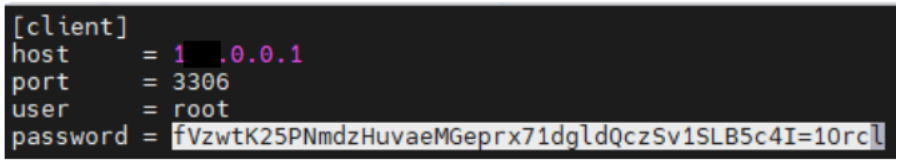
      For example:

      ```
      yum install mysql-shell-commercial-8.0.32-1.1.el8.x86_64.rpm
      ```

      > **Note:**
      >
      > The mysql-shell rpm used here is based on the MySQL 8.0.32.

c. Copy the password from /root/.my.cnf and keep it handy.

Figure 4-2 Copy Password



Running the mysqlsh Shell Utility

Run the mysqlsh shell utility at the CLI prompt.

1. Type `mysqlsh --no-defaults` command on CLI. This opens the JS prompt.
2. Run the following commands in sequence at the CLI prompt:
3. Connect to a MySQL instance by typing the below command:

```
\connect root@localhost:3306
```

4. When prompted for a password, paste the password copied from /root/.my.cnf inCopy Password.
5. Prepare for restore by typing the below command one by one:

```
\sql set GLOBAL local_infile=1;
\sql ALTER INSTANCE DISABLE INNODB REDO_LOG;
\sql drop database <DATABASE>;
```

For Mediation Engine, use DATABASE = vsp

For Mediation Engine Connector, use DATABASE = pldmaster

For Fraud Monitor, use DATABASE = fdp

6. Run the below command to begin the restore process:

```
util.loadDump("/<PATH_TO_MYSQL_BACKUP_DIRECTORY>/<DATABASE>.dump",
{threads: 88, ignoreVersion: true})
```

For example, for ME:

```
util.loadDump("/root/mysqlBackup/vsp.dump", {threads: 88, ignoreVersion:
true})
```

Once the restore is successful, you will see a message similar to the below sample:

```
Target is MySQL 8.0.28-commercial. Dump was produced from MySQL 5.7.35-
enterprise-commercial-advanced-log
WARNING: Destination MySQL version is newer than the one where the dump
was created. Loading dumps from different major MySQL versions is not
fully supported and may not work. The 'ignoreVersion' option is enabled,
so loading anyway.
NOTE: Load progress file detected. Load will be resumed from where it was
```

```
left, assuming no external updates were made.
You may enable the 'resetProgress' option to discard progress for this
MySQL instance and force it to be completely reloaded.
Scanning metadata - done
Executing common preamble SQL
Executing DDL - done
Executing view DDL - done
Starting data load
88 thds loading \ 100% (260.70 MB / 260.70 MB), 8.44 MB/s, 544 / 63 tables
and partitions done
Recreating indexes - done
Executing common postamble SQL
NOTE: The redo log is currently disabled, which causes MySQL to not be
crash safe! Do not forget to enable it again before putting this instance
in production.
7796 chunks (9.97M rows, 260.70 MB) for 63 tables in 1 schemas were loaded
in 1 min 40 sec (avg throughput 7.83 MB/s)
0 warnings were reported during the load.
```

7. Enable REDO_LOG by typing the below command:

```
\sql ALTER INSTANCE ENABLE INNODB REDO_LOG;
```

8. Exit from mysql-shell by typing the below command:

```
\quit
```

Execute the MySQL Delta Script

Execute the MySQL Delta script (Not Applicable for same version restore, i.e. Backup and Restore from 5.1 to 5.1)

MySQL table changes part of the newer Session Monitorversion is not present in the older Session Monitor version backups. This is resolved by MySQL Delta script.

1. Get the MySQLDeltaUpgrade.sh script file present in the Session Monitor installation software RPM .zip file
2. Copy the MySQLDeltaUpgrade.sh script to the /root/ directory of your newly installed Session Monitor Server.
3. Provide necessary permissions for the script by running the following command:

```
chmod +x MySQLDeltaUpgrade.sh
```

4. Execute the MySQL Delta script by running the following command:

```
./MySQLDeltaUpgrade.sh
```

Restoring Blocks Tables

Execute the commands in this section to restore the Blocks table, which fixes any discrepancy caused by difference in number of blocks between the backup and restore machines.

This step is required only if the node type is Mediation Engine

1. Go inside the MySQL Backup folder (e.g. /root/mysqlBackup).

2. Execute the following commands:

```
mysql vsp < blocks_dumps.sql
mysql vsp < blocks_replace.sql
```

Restoring Redis

This section describes the procedure for restoring the backup of Redis data from Fraud Monitor. **This section is applicable for Fraud Monitor only**.

Your Redis backup will be present in your new Session Monitor server in the Backup Path provided during the Redis Backup procedure. For more information, see Redis Backup.

1. Log in as the root to the console of your Fraud Monitor Server.
2. Stop the valkey service by running the following command:

```
systemctl stop pld-valkey
```

3. Transfer the Redis Backup file from the backup location by running the following cp command in the newly installed Fraud Monitor Server.

```
cp -r <path_to_Redis_backup_directory>/* /opt/oracle/ocsm/var/lib/valkey/
```

For example:

```
cp -r /root/ocsmBackup/redisBackup/*  /opt/oracle/ocsm/var/lib/valkey/
```

4. Provide the necessary permission for the dump.rdb file by running the following commands:

```
chown valkey:valkey /opt/oracle/ocsm/var/lib/valkey/dump.rdb
chown 644 /opt/oracle/ocsm/var/lib/valkey/dump.rdb
```

5. Start the valkey service by running the following command:

```
systemctl start pld-valkey
```

valkey restore is now complete.

Post Restore Tasks

This section describes the things to be taken care after restore procedure is complete which enables you to start using your Session Monitor.

After restoring the backup, complete the following steps:

1. Start Session Monitor services by running the following command:

```
source /opt/oracle/ocsm/ocsm_env.sh
pld-systemctl start
```

2. Run the command to restart the crond services:

   ```
   systemctl start crond.service
   ```

3. Certificate Exchange: Before logging into the system, you will need to exchange certificates between respective nodes as required.
4. Configuring connection between nodes:

   Post restore, you will need to re-establish all the connections between all nodes such as:

   - Mediation Engine-Probe
   - Mediation Engine-Fraud Monitor
   - and Mediation Engine-Mediation Engine Connector
   - **For Mediation Engine and Standalone Probe Machine**, Delete older Mediation Engine details from the **Probe** and then add the new **Mediation Engine** details.
   - For Mediation Engine and Embedded Probe Machine, delete the Probe info from Mediation Engine from admin→Settings→Probe section to re-establish connection
5. **Multi VSP**: Post the restore, multi-vsp will be disabled by default. You will need to enable multi-vsp again as per your requirement.
6. **SELinux** : SELinux state won't be restored. Post restore, you can Enable or Disable SELinux again as per your requirement. SELinux policy modules have changed with Session Monitor 5.1, See Enabling SELinux for information.
7. **Changing nginx to httpd**: httpd changes won't be restored. Post restore procedure Apache Web Server will be reverted back to NGINX. You will need to change back from nginx to httpd again as per your requirement.
8. **External Authentication**: For External Authentication enabled Machines, re-enable 'External Authentication' from Settings as these changes will not be restored. Post restore, it is mandatory to copy the new pld.conf template from /opt/oracle/ocsm/etc/httpd/conf.d/pld.conf to /etc/httpd/conf.d/ folder and configure External Authentication details again. This ensures new fixes and any changes in the pld.conf template to be applied on the system.
9. **Retention**: For systems with retention configured in Mediation Engine, If backup and restore procedure took more than 24 hours, it is recommend to adjust retention by adding x days (n+x) and then readjust retention back to your existing one after n days. x = Number of days taken for backup and restore, n = Existing Retention Configured.

   For example, A system has Calls Retention set to 10 days. The backup and restore procedure took 2 Days. Once Restore is completed, in Mediation Engine we need to set Calls Retention as 12 Days (10 + 2). Then after 10 Days you need to set Retention back to 10 Days.
10. **System Diagnostics**: Keep safely in the new Session Monitor Server the backup copy of the System Diagnostics taken during backup process which will be required in future diagnostics.
11. You can now rename the /root/.ssh/authorized_keys_orig file in the Remote Server to /root/.ssh/authorized_keys

    The Restore procedure is now complete.