



## Legal Notice

For information about legal notices, trademarks, disclaimers, warranties, export and other use restrictions, U.S. Government rights, patent policy, and FIPS compliance, see <https://www.microfocus.com/en-us/legal>.

© Copyright 2008 - 2023 Open Text

The only warranties for products and services of Micro Focus and its affiliates and licensors ("Micro Focus") are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Micro Focus shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

# About This Guide

The information in this guide is organized as follows:

- ♦ [Part I, “Embedded Database Maintenance,” on page 9](#)
- ♦ [Part II, “External Database Maintenance,” on page 49](#)
- ♦ [Part III, “Database Management - Best Practices, Tips, Troubleshooting,” on page 67](#)

## Audience

This guide is intended for ZENworks administrators.

## Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the *comment on this topic* link at the bottom of each page of the online documentation.

## Additional Documentation

ZENworks is supported by other documentation (in both PDF and HTML formats) that you can use to learn about and implement the product. For additional documentation, see the [ZENworks documentation website \(http://www.novell.com/documentation/zenworks/\)](http://www.novell.com/documentation/zenworks/).



---

# Contents

<b>About This Guide</b>	<b>3</b>
<b>Part I Embedded Database Maintenance</b>	<b>9</b>
<b>1 Retrieving and Storing the Credentials of the Embedded PostgreSQL Database</b>	<b>11</b>
<b>2 Changing the Ports Used by the Embedded PostgreSQL Database</b>	<b>13</b>
<b>3 Configuring PostgreSQL</b>	<b>15</b>
<b>4 Migrating the OEM PostgreSQL Database to an External PostgreSQL Database</b>	<b>17</b>
<b>5 Moving the Data from an Embedded PostgreSQL Database to an External PostgreSQL Database</b>	<b>19</b>
Preparing to Move the Data . . . . .	19
Important Commands and Locations . . . . .	19
Moving the Data from an Embedded PostgreSQL to an External PostgreSQL . . . . .	20
Step 1: Backup the Database on the Embedded Primary Server . . . . .	20
Step 2: Restore the Database on an External Server . . . . .	21
<b>6 Moving an Embedded PostgreSQL Database from One Primary Server to another Primary Server</b>	<b>25</b>
Important Commands and Locations . . . . .	25
Moving an Embedded PostgreSQL Database from One Primary Server to another Primary Server . . . . .	26
Step 1: Backup the Database . . . . .	26
Step 2: Restoring the Database . . . . .	27
<b>7 Backing up an Embedded PostgreSQL Database on One Primary Server and Restoring it on another Primary Server</b>	<b>33</b>
Important Commands and Locations . . . . .	33
Backing up the Database on the Embedded Primary Server . . . . .	34
Restoring the Database on another Server . . . . .	35
<b>8 Backing Up and Restoring the PostgreSQL Database</b>	<b>39</b>
Important Commands and Locations . . . . .	39
Backing Up and Vacuuming the PostgreSQL Database . . . . .	39
Using the zman command to Take a Backup of the Database . . . . .	41
Using the pg_dump Command to Take a Backup of the Database . . . . .	42
Restoring the PostgreSQL Database . . . . .	42
Restoring the Database Backup Taken Using the ZMAN Command . . . . .	42

<b>9 Connecting to the Embedded PostgreSQL Database</b>	<b>45</b>
On Linux / Appliance. ....	45
<b>10 Cleaning Database and Generating Internal Statistics</b>	<b>47</b>
<b>Part II External Database Maintenance</b>	<b>49</b>
<b>11 Configuring the ZENworks Server to Point to the New MS SQL Database Containing Data Moved from Another MS SQL Database</b>	<b>51</b>
Preparing to Move the Data. ....	51
Configuring the ZENworks Server to Point to the New MS SQL Database. ....	52
<b>12 Configuring the ZENworks Server to Point to the New Oracle Database Containing Data Moved from Another Oracle Database</b>	<b>53</b>
Preparing to Move the Data. ....	53
Configuring the ZENworks Server to Point to the New Oracle Database. ....	54
<b>13 Configuring PostgreSQL</b>	<b>55</b>
<b>14 Creating External PostgreSQL Database Schema</b>	<b>57</b>
<b>15 Configuring a ZENworks Server using an Existing Schema</b>	<b>59</b>
<b>16 Backing Up and Restoring the PostgreSQL Database</b>	<b>61</b>
<b>17 Migrating from md5 to scram authentication in PostgreSQL</b>	<b>63</b>
MD5. ....	63
SCRAM-SHA-256. ....	63
<b>18 Database Activities</b>	<b>65</b>
<b>Part III Database Management - Best Practices, Tips, Troubleshooting</b>	<b>67</b>
<b>19 Database Ports</b>	<b>69</b>
<b>20 Database Tips</b>	<b>71</b>
Changing the ZENworks Database User Password. ....	71
Maximum Pool Size. ....	72
Verifying Invalid Objects and Indexes in the Oracle Database. ....	72
Microsoft SQL NTLM V2 Support. ....	75
<b>21 Troubleshooting Database Migration</b>	<b>77</b>
After database migration ZEUS is unable to contact the new database server. ....	77

During Appliance deployment, configuring External PostgreSQL using the New Schema option fails to create Audit database .....	78
Troubleshooting a Java Heap Space Exception .....	78
Troubleshooting an Oracle Database Crash .....	79
Troubleshooting an Oracle Tablespace Issue .....	79
Troubleshooting the Database Migration Failure Issue .....	79
Troubleshooting the Database Migration by Using An Existing User Schema .....	80
Troubleshooting the ORA-01652: unable to extend temp segment by 128 in tablespace TEMP. ....	80
Troubleshooting the ORA-01400: cannot insert NULL. ....	81
Troubleshooting the ORA-12516, TNS: listener could not find available handler with matching protocol stack .....	81
Troubleshooting the database migration failure, duplication of serial numbers in the zDevice table. ....	81
Configure action fails with exception ZEN configure action container is not responding .....	82
<b>22 Customizing JDBC URL for ZENworks</b>	<b>85</b>
<b>23 Using DBeaver to Export SQL Query Results</b>	<b>87</b>





# Embedded Database Maintenance

- ♦ Chapter 1, “Retrieving and Storing the Credentials of the Embedded PostgreSQL Database,” on page 11
- ♦ Chapter 2, “Changing the Ports Used by the Embedded PostgreSQL Database,” on page 13
- ♦ Chapter 3, “Configuring PostgreSQL,” on page 15
- ♦ Chapter 4, “Migrating the OEM PostgreSQL Database to an External PostgreSQL Database,” on page 17
- ♦ Chapter 5, “Moving the Data from an Embedded PostgreSQL Database to an External PostgreSQL Database,” on page 19
- ♦ Chapter 6, “Moving an Embedded PostgreSQL Database from One Primary Server to another Primary Server,” on page 25
- ♦ Chapter 7, “Backing up an Embedded PostgreSQL Database on One Primary Server and Restoring it on another Primary Server,” on page 33
- ♦ Chapter 8, “Backing Up and Restoring the PostgreSQL Database,” on page 39
- ♦ Chapter 9, “Connecting to the Embedded PostgreSQL Database,” on page 45
- ♦ Chapter 10, “Cleaning Database and Generating Internal Statistics,” on page 47



# 1 Retrieving and Storing the Credentials of the Embedded PostgreSQL Database

If you have installed ZENworks with the embedded PostgreSQL database that is bundled with ZENworks, we recommend that you store the credentials of the database for future use.

- 1 Retrieve the credentials of the embedded database by entering one of the following commands at the server prompt:

```
zman database-get-credentials
```

or

```
zman dgc
```

The credentials are displayed on the console.

For more information about `zman`, view the `zman` man page (`man zman`) on the server or see `zman(1)` in the [ZENworks Command Line Utilities Reference](#).

- 2 Copy the credentials and save them in a file.

To retrieve and store the credentials of Remote PostgreSQL, Oracle, or Microsoft SQL Server databases, refer to their documentation.



## 2 Changing the Ports Used by the Embedded PostgreSQL Database

PostgreSQL uses port 5432 by default. However, ZENworks uses 54327 port for PostgreSQL. You can change the port on which the database runs.

- 1 Before executing the following steps, stop the ZENworks Services on all Primary servers by running the following configure action:

```
microfocus-zenworks-configure -c Start
```

After running the command, under Action, select Stop.

- 2 In the `postgresql.conf` file, specify the new port number on which the server listens to.

The `postgresql.conf` file `/var/opt/microfocus/pgsql/data`.

- 3 In the `zdm.xml` and `zenaudit.xml` files on all the Primary Servers, specify the new port number in the following entry:

```
<entry key="Port">54327</entry>
```

By default, the entry lists the default port number, 54327.

The `zdm.xml` and `zenaudit.xml` files are located in `/etc/opt/microfocus/zenworks/datamodel`.

- 4 Start the ZENworks Services on all Primary servers by running the following configure action:

```
microfocus-zenworks-configure -c Start
```

- 5 Run `microfocus-zenworks-configure -c GenerateOSPPProperties`



# 3 Configuring PostgreSQL

Each Primary Server on PostgreSQL can consume up to 500 database connections. If you want to add more than one Primary Server, you should change the property values in the postgresql.conf file.

For the embedded database, the postgresql.conf file is available at the following location:

```
/var/opt/microfocus/pgsql/data
```

Use the following formula to calculate the property values.

`max_connections` = Number of primary servers \* 500

`max_prepared_transactions` = Number of primary servers \* 500

After changing the property values in `postgresql.conf`, restart the ZENworks and PostgreSQL service.

To restart the PostgreSQL service:

Run the `systemctl restart zenpostgresql.service` command.





# 4 Migrating the OEM PostgreSQL Database to an External PostgreSQL Database

If you are using the OEM PostgreSQL database in ZENworks 2017 Update x, then ensure that you migrate to an external PostgreSQL database before upgrading your zone to ZENworks 2020.

Using the database migration script, you can migrate the database. The script file can be downloaded from the following locations:

[Download the OEM PostgreSQL to External PostgreSQL Migration script](#)

You can download the script file, and then perform the following steps to migrate to an external PostgreSQL database:

1. Take a backup of the database.
2. Stop all services in ZENworks Primary Server.
3. On the database server, download and extract the `migrate_postgres.sh.zip` file.
4. Download and install the PostgreSQL version supported by ZENworks.

If you are planning to upgrade to ZENworks 24.2, ensure that you install PostgreSQL 12.14.

5. Change the permission for the script by running the `chmod 755 migrate_script.sh` command.
6. The script log will be saved in the `/tmp/migrate_postgres.log` location.
7. Run the following script:

```
./migrate_postgres.sh -p <PostgreSQL_Installed_Directory> --database  
<zenworks_db_name> --password <superuser password>
```

Example: `./migrate_postgres.sh -p /usr/pgsql-11/ --database  
zenworks_database --password novell`

---

**NOTE:** ♦ To see all the available options, run the script without any argument. (Example: `./migrate_postgres.sh`)

- ♦ After migrating to the external PostgreSQL, ensure that you manually clean the OEM PostgreSQL installation.
-



# 5 Moving the Data from an Embedded PostgreSQL Database to an External PostgreSQL Database

ZENworks allows you move the data from an embedded PostgreSQL to an external PostgreSQL database:

- ♦ [“Preparing to Move the Data” on page 19](#)
- ♦ [“Important Commands and Locations” on page 19](#)
- ♦ [“Moving the Data from an Embedded PostgreSQL to an External PostgreSQL” on page 20](#)

## Preparing to Move the Data

Before moving the data from an embedded PostgreSQL database to an external PostgreSQL database, perform the following:

- ♦ Ensure that ZENworks is installed with an embedded PostgreSQL database on a Linux device.
- ♦ Install the external PostgreSQL database. For more information on how to install an external PostgreSQL database, see [Installing PostgreSQL](#) in the [Database Migration from Microsoft SQL to PostgreSQL](#).

## Important Commands and Locations

This section provides some of the command and locations that will be used quite frequently while moving the data. Ensure that you keep the following information handy while initiating the data migration:

- ♦ The ZENworks database and Audit database names on the embedded PostgreSQL database are available in `zdm.xml` and `zenaudit.xml` files. These files are available in the following location:

Check the **Database** key to get the database name.

```
/etc/opt/microfocus/zenworks/datamodel/
```

Example:

```
<entry key="Database">zenworks</entry>
```

```
<entry key="Database">zenworksaudit</entry>
```

- ♦ The database details can be retrieved by running the following commands:
  - ♦ `zman dgc`: Retrieves the database credentials used to connect to the embedded database.

- ♦ `zman dgca`: Retrieves the database credentials used to connect to the audit embedded database.
  - ♦ `zman dgcs`: Retrieves the database credentials of super user that is used to connect to the embedded database.
  - ♦ `zman dgcam`: Retrieves the database credentials of the Antimalware database.
- 
- ♦ **IMPORTANT:** The user that should be used in the following steps (unless explicitly mentioned) should be any user (administrator) who has privileges to control and monitor ZENworks services.
- 

## Moving the Data from an Embedded PostgreSQL to an External PostgreSQL

Moving the data from an embedded PostgreSQL server to an external PostgreSQL server is a two step process:

- ♦ [“Step 1: Backup the Database on the Embedded Primary Server” on page 20](#)
- ♦ [“Step 2: Restore the Database on an External Server” on page 21](#)

### Step 1: Backup the Database on the Embedded Primary Server

---

**NOTE:** ZENworks no longer supports Windows Server as a Primary Server from version 24.2 onwards. For more information, see [End of Support Windows Primary Server](#).

---

1. Before executing the following steps, ensure that you have reviewed the information documented in the section [Important Commands and Locations](#).
2. Stop the ZENworks Services on all the Primary Servers by running the following configure action:

```
microfocus-zenworks-configure -c Start
```

After running the command, under Action, select Stop.

3. On the Primary Server on which the Embedded Database is installed, Start the **Micro Focus ZENworks Embedded Datastore** service using the following steps:

**On Linux:** Run the following command to start the Micro Focus ZENworks Embedded Datastore – PostgreSQL service.

- ♦ **On SLES 12 and later:** `systemctl start zenpostgresql.service`

4. Login to the Primary Server that has the database role.
5. Initiate the backup by setting the following variables:
  - ♦ `PGPORT`: <existing source DB port configured>  
By default, ZENworks uses 54327
  - ♦ `PGDATA`: <embedded\_database\_location>
  - ♦ `PGUSER`: <existing super user for ZENworks postgres DB> The default user for ZENworks is zenpostgres.

- ♦ PGDATABASE: <existing database name> ZENworks default is postgres
- ♦ PGPASSWORD: <password for the above super user PGUSER>

**Example 5-1** Example for Linux Primary Server:

Open the terminal and set the following variables based on the existing configuration.

- ♦ export LD\_LIBRARY\_PATH=/opt/microfocus/zenworks/share/pgsql/lib;
- ♦ export PATH=\$PATH:/opt/microfocus/zenworks/share/pgsql/bin;

**On SLES 15:** export LD\_LIBRARY\_PATH=/opt/microfocus/zenworks/share/pgsql/lib:/opt/microfocus/zenworks/share/pgsql/stackbuilder/lib; export LD\_LIBRARY\_PATH

On SLES 15, the *no version information available* error message can be ignored.

- ♦ export PGPORT=54327;
- ♦ export PGDATA=/var/opt/microfocus/pgsql/data;
- ♦ export PGUSER=<super user retrieved from zman dgcs>;
- ♦ export PGDATABASE=postgres;
- ♦ export PGPASSWORD=<password retrieved from zman dgcs>;

6. Run the following commands to back up both the databases (ZENworks and Audit):

♦ **On Linux:**

- ♦ cd /opt/microfocus/zenworks/share/pgsql/bin/
- ♦ ./pg\_dumpall > "<path to sql file>"  
Example: ./pg\_dumpall > /tmp/postgres\_dump.sql
- ♦ docker exec -e PGUSER=<super user retrieved from zman dgcs> -e PGDATABASE=postgres -e PGPASSWORD=<password retrieved from zman dgcs> zenpostgres pg\_dumpall > /tmp/dump.sql  
Example: docker exec -e PGUSER=zenpostgres -e PGDATABASE=postgres -e PGPASSWORD=Zw0#43cf4525d0f5df6395bbbec9c zenpostgres pg\_dumpall > /tmp/dump.sql

---

**NOTE:** You can ignore the no version information available error message.

---

7. On the Primary Server on which the Embedded Database is installed (source database), Stop the **Micro Focus ZENworks Embedded Datastore** service using the following steps:

**On Linux:** Run the following command to start the Micro Focus ZENworks Embedded Datastore – PostgreSQL service.

- ♦ **On SLES 12:** systemctl stop zenpostgresql.service

## Step 2: Restore the Database on an External Server

---

**NOTE:** ZENworks no longer supports Windows Server as a Primary Server from version 24.2 onwards. For more information, see [End of Support Windows Primary Server](#).

---

Perform the following steps on an external server on which you want to move the PostgreSQL database. After installing PostgreSQL on an external server, ensure that the PostgreSQL database service is running.

To restore the database on an external database server, perform the following steps:

1. Set the following environment variables:
  - ♦ PGPORT: <target DB port>
  - ♦ PGUSER: <target super user for postgres DB>
  - ♦ PGDATABASE: <target initial connection database name> default is postgres
  - ♦ PGPASSWORD: <password for the above super user i.e. PGUSER>

*Example 5-2 Example for Linux Primary Server:*

Open the terminal and set the following variables based on the existing configuration.

- ♦ export LD\_LIBRARY\_PATH=<postgres installation path>/lib;
- ♦ export PATH=\$PATH:<postgres installation path>/bin;
- ♦ export PGPORT=<Port on which PostgreSQL is configured>;
- ♦ export PGUSER=<Postgres super user>;
- ♦ export PGDATABASE=postgres;
- ♦ export PGPASSWORD=<PostgreSQL super user password>;

2. Login as PostgreSQL super user and manually create the database using the following query for the databases that should be restored:

```
create database <DB name> with template=template0 encoding='UTF8';
```

For example:

- ♦ **ZENworks Database:** create database zenworks with template=template0 encoding='UTF8';
- ♦ **Audit Database:** create database zenworksaudit with template=template0 encoding='UTF8';

3. Copy the postgres\_dump.sql file that was created in [Step 1: Backup the Database on the Embedded Primary Server](#) to the target device.
4. Restore the database from the backup taken in [Step 1: Backup the Database on the Embedded Primary Server](#), by running the following commands:

- ♦ In the Terminal, run the following commands as PostgreSQL user:
  - ♦ cd <postgres installation path>/bin
  - ♦ ./psql -v ON\_ERROR\_STOP=0 -f /tmp/postgres\_dump.sql -o /tmp/postgres\_dump\_log.txt -L /tmp/postgres\_dump\_log.log > /tmp/postgres\_dump\_log.err 2>&1

5. After Restoring the database, log into the ZENworks database, and run the below query to delete the database role for the device that has the embedded PostgreSQL database installed by running the following query on the external database server:

```
delete from zZenServerRoles where Roles = 'Database';  
commit;
```

6. Update the following changes in `zdm.xml` and `zenaudit.xml` for all the Primary Servers in the zone:
  - ♦ Delete the `<entry key="Embedded">true</entry>`.
  - ♦ Update the value of the Port key with the port on which the external PostgreSQL database is configured.
 

```
<entry key="Port"><PORT-of-target-DB-device></entry>
```
7. Update the server key value in the `zdm.xml` and `zenaudit.xml` files as shown below:
  - ♦ In the target database server, add the loopback IP address.
 

```
<entry key="Server"><target_database_server-IP></entry>
```
  - ♦ In all the other Primary Server, add the IP of the target database server.
 

```
<entry key="Server"><target_database_server-IP></entry>
```
8. In the new database server, run the following configure actions: `microfocus-zenworks-configure -c GenerateOSPPProperties`
9. Run `microfocus-zenworks-configure -c`  
`GenerateContentDatasourceConfigureAction` on all the content servers.
10. After updating the changes, restart all the ZENworks services by running the `microfocus-zenworks-configure -c Start` configure action.
11. Ensure that the **Micro Focus ZENworks Embedded Datastore** service is disabled and not running on the Primary Server that hosted the Embedded PostgreSQL database.
12. The ZENworks Server now points to new database.  
 Ensure that you can successfully log into ZCC. After logging into ZCC, go to the Diagnostics page and in the ZENworks Databases panel, verify the Host IP addresses and also verify that the Audit database is connected.





# 6 Moving an Embedded PostgreSQL Database from One Primary Server to another Primary Server

ZENworks enables you to move the embedded database from one Primary Server to another Primary Server or Appliance. Perform the steps provided in the following section to move the database from one Primary Server to another Primary Server.

- ♦ [“Important Commands and Locations” on page 25](#)
- ♦ [“Moving an Embedded PostgreSQL Database from One Primary Server to another Primary Server” on page 26](#)

---

**NOTE:** ZENworks no longer supports Windows Server as a Primary Server from version 24.2 onwards. For more information, see [End of Support Windows Primary Server](#).

---

## Important Commands and Locations

This section provides some of the command and locations that will be used quite frequently while moving the data. Ensure that you keep the following information handy while initiating the data migration:

- ♦ The ZENworks database and Audit database names on the embedded PostgreSQL database are available in `zdm.xml` and `zenaudit.xml` files. These files are available in the following location:

Check the **Database** key to get the database name.

```
/etc/opt/microfocus/zenworks/datamodel
```

Example:

```
<entry key="Database">zenworks</entry>
```

```
<entry key="Database">zenworksaudit</entry>
```

- ♦ The database details can be retrieved by running the following commands:
  - ♦ `zman dgc`: Retrieves the database credentials used to connect to the embedded database.
  - ♦ `zman dgca`: Retrieves the database credentials used to connect to the audit embedded database.
  - ♦ `zman dgcs`: Retrieves the database credentials of super user that is used to connect to the embedded database.
  - ♦ `zman dgcam`: Retrieves the database credentials of the Antimalware database.

- 
- ♦ **IMPORTANT:** The user that should be used in the following steps (unless explicitly mentioned) should be any user (administrator) who has privileges to control and monitor ZENworks services.
- 
- ♦ To connect to the database, see [Connecting to the Embedded PostgreSQL Database](#).

## Moving an Embedded PostgreSQL Database from One Primary Server to another Primary Server

Moving the data is a two step process:

- ♦ “[Step 1: Backup the Database](#)” on page 26
- ♦ “[Step 2: Restoring the Database](#)” on page 27

---

**IMPORTANT:** While moving the database, we refer the source database server as PSDB1 and the destination database server as PSDB2.

---

---

**NOTE:** ZENworks no longer supports Windows Server as a Primary Server from version 24.2 onwards. For more information, see [End of Support Windows Primary Server](#).

---

### Step 1: Backup the Database

#### Backup the Database on the Embedded Primary Server (PSDB1)

1. Before executing the following steps, ensure that you have reviewed the information documented in the section [Important Commands and Locations](#).

Ensure that you have noted the database username, password of the source Primary Server by running the `zman dgcs` command. Also, copy the target Primary Server’s GUID details, which is required while restoring the database.

2. Stop the ZENworks Services on all the Primary Servers by running the following configure action:

```
microfocus-zenworks-configure -c Start
```

After running the command, under Action, select Stop.

3. Log in to the Primary Server that has the database role.
4. On the Primary Server on which the Embedded Database is installed (PSDB1), start the Micro Focus ZENworks Embedded Datastore service using the following command on SLES 12 and later:

```
systemctl start zenpostgresql.service
```

5. Run the following commands to back up the databases (ZENworks and Audit):

```
docker exec -e PGUSER=<super user retrieved from zman dgcs> -e  
PGDATABASE=postgres -e PGPASSWORD=<password retrieved from zman dgcs>  
zenpostgres pg_dumpall > /tmp/dump.sql
```

```
Example: docker exec -e PGUSER=zenpostgres -e PGDATABASE=postgres -e
PGPASSWORD=Zw0#3dde8547adb388306de141850 zenpostgres pg_dumpall > /tmp/
dump.sql
```

---

**NOTE:** You can ignore the no version information available error message.

---

6. Start the Micro Focus ZENworks Embedded Datastore service using the following command:

- ♦ **On SLES 12 and later:** `systemctl start zenpostgresql.service`

---

**NOTE:** Keep the `/tmp/dump.sql` in a safe place as it has the database backup.

---

## Step 2: Restoring the Database

Depending on the target database platform, refer to the following section to restore the database:

### Restore the Database on a Primary Server or Appliance Server (PSDB2)

On a Linux Primary Server or an Appliance Server, perform the following steps:

- 1 Before executing the following steps, ensure that you have reviewed the information documented in the section [Important Commands and Locations](#).

Ensure that you have noted the database username, password of the source Primary Server by running the `zman dgcs` command.

- 2 Stop all the ZENworks services on all Primary Servers by running the following configure action:

```
microfocus-zenworks-configure -c Start
```

- 3 Load `zenpostgres` docker by running the following commands:

- ♦ `docker load -i /opt/microfocus/zenworks/docker-images/zenpostgres.tar.gz`
- ♦ `docker run --name tmp_postgres -d zenpostgres`
- ♦ `docker cp /tmp/dump.sql tmp_postgres:/var/opt/microfocus/pgsql/`

---

**NOTE:** The `/tmp/dump.sql` file is created in the backup step 5 of the [“Backup the Database on the Embedded Primary Server \(PSDB1\)” on page 26](#)

---

- 4 Get into the temporary postgres container by running the following command:

```
docker exec -it tmp_postgres bash
```

- 5 To check whether the database server is ready to accept connections, run the following commands:

- ♦ `cd /var/opt/microfocus/pgsql/data`
- ♦ `cat postmaster.log`

After executing the commands, the last line should say “Database system is ready to accept connections”. If you did not see this message, wait for some time, and then execute the “`cat postmaster.log`” command again.

- 6 Perform the following steps to prepare the default database, for restore.

Run the `psql -W postgres` command. When prompted for a password, enter *novell* as the password.

**6a** To change the default password, run the following SQL query:

```
ALTER USER zenpostgres with PASSWORD '<Password>';
```

Replace the `<Password>` in the above SQL query with the password you noted down in step 1 of the [“Backup the Database on the Embedded Primary Server \(PSDB1\)”](#) on page 26.

Example: `ALTER USER zenpostgres with PASSWORD 'Zw0#3dde8547adb388306de141850';`

**6b** Drop the following default databases and users:

```
drop database zenworks;  
drop database zenworksaudit;  
drop user zenadmin;  
drop user zenauditadmin;  
drop database zenamdatabase;  
drop user zenamadmin;
```

---

**NOTE:** Ignore errors related to missing database or users if any.

---

**7** Exit the Postgres interactive terminal by executing the `exit` command.

**8** To restore the database, run the following commands:

- ♦ `export PGPASSWORD= <password retrieved from zman dgcs>`
- ♦ `psql -w postgres -v ON_ERROR_STOP=0 -f /var/opt/microfocus/pgsql/dump.sql -o /var/opt/microfocus/pgsql/dump.log -L /var/opt/microfocus/pgsql/dump.log >/var/opt/microfocus/pgsql/dump.err 2>&1`

---

**NOTE:** Execution of the above command may take time depending on the amount of data in the database.

---

**9** Once the restore is completed successfully, exit the docker container by running the `exit` command.

**10** To copy the database folder from the container to the host, run the following commands:

- ♦ `rm -rf /var/opt/microfocus/pgsql`
- ♦ `docker cp tmp_postgres:/var/opt/microfocus/pgsql/ /var/opt/microfocus/`
- ♦ `rm -f /var/opt/microfocus/pgsql/data/postmaster.pid /var/opt/microfocus/pgsql/data/postmaster.log`

**11** To clean up the temporary container, run the following commands:

- ♦ `docker kill tmp_postgres`
- ♦ `docker container rm tmp_postgres`

**12** To change the ownership of the data directory, run the following command:

```
chown -R 1026:1026 /var/opt/microfocus/pgsql/data
```

**13** In the `postgresql.conf` file, under `/var/opt/microfocus/pgsql/data/`, uncomment the `listen_addresses` and `port` entry keys, and update the values as shown below:

```
listen_addresses = '*'
Port= 54327
max_connections = Number of primary servers * 500
```

**14** Edit the `pg_hba.conf` file as shown below:

The `pg_hba.conf` file is available in the following location:

`/var/opt/microfocus/pgsql/data`

- ♦ In the `pg_hba.conf` file, update the content as shown below:

```
# TYPE DATABASE USER ADDRESS METHOD
# "local" is for Unix domain socket connections only
local all all md5
# IPv4 local connections:
host all all 127.0.0.1/32 md5
# IPv6 local connections:
host all all ::1/128 md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
local replication all md5
host replication all 127.0.0.1/32 md5
host replication all ::1/128 md5
host all all 0.0.0.0/0 md5
```

**15** In the local primary server, change the server key in `zdm.xml` and `zenaudit.xml`, available in `/etc/opt/microfocus/zenworks/datamodel` folder with the local host address `Jdbc_Url`:

- ♦ `<entry key="Server">Jdbc_Url</entry>`
- ♦ `<entry key="Embedded">true</entry>`

**16** If you have configured antimalware in the zone, modify the `database.server.address` property in `/etc/opt/microfocus/zenworks/antimalware/amedatasource.properties` file to have the server IP as `127.0.0.1`

**17** To create the Postgres service, create the file in `/usr/lib/systemd/system/zenpostgresql.service` with the following contents:

```
[Unit]
Description=PostgreSQL Daemon for ZENworks Databases
After=syslog.target network.target docker.service

[Service]
Type=simple
ExecStart=/opt/microfocus/zenworks/lib/systemd/system/zenpostgresql-systemd
Restart=on-failure
RestartSec=20
StartLimitBurst=5
StartLimitInterval=120

[Install]
WantedBy=multi-user.target
```

- 18 To reset the file/folder permissions, run the following command:  
`permissions.sh`
- 19 Start the zenpostgres service by running the following commands:  
`systemctl enable zenpostgresql.service`  
`systemctl start zenpostgresql.service`
- 20 Run the following configure actions to update OSP/Content Configurations:  
`microfocus-zenworks-configure -c GenerateOSPProperties`  
`microfocus-zenworks-configure -c RegenerateOSPAuthCfgXml`  
`microfocus-zenworks-configure -c`  
`GenerateContentDataSourceConfigureAction`
- 21 (Conditional) If Antimalware is configured, run the following commands:  
`microfocus-zenworks-configure -c ZENServicesFirewallConfigureAction -`  
`Dservice=antimalware -Doperation=ADD`  
`microfocus-zenworks-configure -c AntimalwareServiceConfigureAction -`  
`Dam.service.action=start`
- 22 On the Primary Server (PSDB2), start the ZENworks services by running the following command:  
`microfocus-zenworks-configure -c Start`
- 23 Assign the database role to the destination database server (PSDB2) by running the following configure action:  
`microfocus-zenworks-configure -c MigrateServerRoleConfigureAction -`  
`DsourceGuid=<GUID of PSDB1> -DdestGuid=<GUID of PSDB2> -Drole=Database`
- 24 To setup Automatic Maintenance, run the following configure action:  
`microfocus-zenworks-configure -c`  
`AutomaticPostgresMaintenanceConfigureAction`
- 25 (Conditional) If you have enabled Kafka, then run `zman server-role-kafka-recreate-connectors -f true` command to recreate the Kafka connectors.  
  
For more information, see Server Commands in the [ZENworks Command Line Utilities](#) guide.
- 26 If you have the firewall enabled on the server, ensure that port 54327 is open.

---

**NOTE:** With this, we complete the Restoring of the Database to the Target Primary Server (PSDB2). Continue further to update other Primary Servers in the zone.

---

- 27 Do the following on all the other primary servers in the zone:
  - 27a Change the server key in `zdm.xml` and `zenaudit.xml`, available in `/etc/opt/microfocus/zenworks/datamodel` folder with the IP address of PSDB2:  
`<entry key="Server">IP OF PSDB2</entry>`
  - 27b If you have configured Antimalware in the zone, modify the `database.server.address` property in `/etc/opt/microfocus/zenworks/antimalware/amedatasource.properties` file to have the server IP as IP of PSDB2.
  - 27c To update OSP/Content Configurations, run the following configure actions:  
`microfocus-zenworks-configure -c GenerateOSPProperties`  
`microfocus-zenworks-configure -c RegenerateOSPAuthCfgXml`

```
microfocus-zenworks-configure -c  
GenerateContentDatasourceConfigureAction
```

**27d** (Conditional) If Antimalware is configured, run the following:

```
microfocus-zenworks-configure -c ZENServicesFirewallConfigureAction  
-Dservice=antimalware -Doperation=ADD  
  
microfocus-zenworks-configure -c AntimalwareServiceConfigureAction  
-Dam.service.action=start
```

**27e** Run the following commands:

- ♦ `systemctl stop zenpostgresql.service`
- ♦ `systemctl disable zenpostgresql.service`

---

**NOTE:** This step is applicable only on PSDB1.

---

**27f** Start all the zenworks services by running the following command:

```
microfocus-zenworks-configure -c Start
```





# 7 Backing up an Embedded PostgreSQL Database on One Primary Server and Restoring it on another Primary Server

ZENworks enables you to backup the embedded database from one Primary Server and restore it in another Primary Server. This procedure can be followed if you want to replace the existing Primary Server with a new server or if you want to move from a Linux to an Appliance server.

- ♦ [“Important Commands and Locations” on page 33](#)
- ♦ [“Backing up the Database on the Embedded Primary Server” on page 34](#)
- ♦ [“Restoring the Database on another Server” on page 35](#)

## Important Commands and Locations

This section provides some of the command and locations that will be used quite frequently while moving the data. Ensure that you keep the following information handy while initiating the data migration:

- ♦ The ZENworks database and Audit database names on the embedded PostgreSQL database are available in `zdm.xml` and `zenaudit.xml` files. These files are available in the following location:

Check the **Database** key to get the database name.

**On Linux:** `/etc/opt/microfocus/zenworks/datamodel/`

Example:

```
<entry key="Database">zenworks</entry>
```

```
<entry key="Database">zenworksaudit</entry>
```

- ♦ The database details can be retrieved by running the following commands:
  - ♦ `zman dgcs`: Retrieves the database credentials used to connect to the embedded database.
  - ♦ `zman dgca`: Retrieves the database credentials used to connect to the audit embedded database.
  - ♦ `zman dgcs`: Retrieves the database credentials of super user that is used to connect to the embedded database.
  - ♦ `zman dgcam`: Retrieves the database credentials of the Antimalware database.

# Backing up the Database on the Embedded Primary Server

1. Run the following commands and note down the ZENworks database credentials of the source machine:
  - zman dgv
  - zman dgca
  - zman dgcs
  - zman dgcam
2. Stop the ZENworks Services on all the Primary Servers by running the following configure action:  
`microfocus-zenworks-configure -c Start`  
After running the command, under **Action**, select **Stop**.
3. On the Primary Server on which the Embedded Database is installed, start the **Micro Focus ZENworks Embedded Datastore** service using the following steps:  
Run the following command to start the Micro Focus ZENworks Embedded Datastore – PostgreSQL service.
  - ♦ **On SLES 12 and later:** `systemctl start zenpostgresql.service`
4. Login to the Primary Server that has the database role.
5. Initiate the backup by setting the following variables:
  - ♦ PGPORT: <existing source DB port configured>  
By default, ZENworks uses 54327
  - ♦ PGDATA: <embedded\_database\_location>
  - ♦ PGUSER: <existing super user for ZENworks postgres DB> The default user for ZENworks is zenpostgres.
  - ♦ PGDATABASE: <existing database name> ZENworks default is postgres
  - ♦ PGPASSWORD: <password for the above super user PGUSER>
6. Run the following commands to back up both the databases (ZENworks and Audit):
  - ♦ `./pg_dumpall > "<path to sql file>"`  
Example: `./pg_dumpall > /tmp/postgres_dump.sql`
  - ♦ `docker exec -e PGUSER=<super user retrieved from zman dgcs> -e PGDATABASE=postgres -e PGPASSWORD=<password retrieved from zman dgcs> zenpostgres /opt/microfocus/zenworks/share/pgsql/bin/pg_dumpall > /tmp/dump.sql`  
Example: `docker exec -e PGUSER=zenpostgres -e PGDATABASE=postgres -e PGPASSWORD=Zw0#43cf4525d0f5df6395bbbec9c zenpostgres /opt/microfocus/zenworks/share/pgsql/bin/pg_dumpall > /tmp/dump.sql`

---

**NOTE:** You can ignore the no version information available error message.

---

# Restoring the Database on another Server

1. Stop the ZENworks Services on the Primary Servers where the database is to be restored, by running the following configure action:

```
microfocus-zenworks-configure -c Start
```

After running the command, under Daemons, deselect **Micro Focus ZENworks Embedded PostgreSQL**, under **Action**, select **Stop**.

2. **On the Primary Server:**

- a. Switch the user to **zenpostgres** by running the command `su zenpostgres`

- b. Enter zenpostgres container prompt by running below command:

```
docker exec -it -e PGUSER=<super user retrieved from zman dgcs> -e  
PGDATABASE=postgres -e PGPASSWORD=<password retrieved from zman  
dgcs> zenpostgres bash
```

- c. `cd /opt/microfocus/zenworks/share/pgsql/bin`

3. Start PSQL connection by running `./psql`.

4. Close the database dangling connections by running the following command:

```
SELECT pg_terminate_backend(pid) FROM pg_stat_activity WHERE pid =  
pg_backend_pid();
```

5. Drop the following databases and users:

- ♦ `DROP DATABASE ZENWORKS;`
- ♦ `DROP DATABASE ZENWORKSAUDIT;`
- ♦ `DROP USER ZENAUDITADMIN;`
- ♦ `DROP USER ZENADMIN;`

6. Create the database using the following query for the databases that should be restored:

```
create database <DB name> with template=template0 encoding='UTF8';
```

For example:

- ♦ **ZENworks Database:** `create database zenworks with template=template0 encoding='UTF8';`
- ♦ **Audit Database:** `create database zenworksaudit with template=template0 encoding='UTF8';`

7. Run following command to change the password for user 'zenpostgres':

```
ALTER ROLE zenpostgres WITH PASSWORD '<password retrieved from zman  
dgcs of the source db machine>';
```

8. Exit PSQL.

9. Set the following variables again:

- ♦ `PGPORT: <existing source DB port configured>`  
By default, ZENworks uses 54327
- ♦ `PGDATA: <embedded_database_location>`
- ♦ `PGUSER: <existing super user for ZENworks postgres DB>` The default user for ZENworks is zenpostgres.

- ♦ PGDATABASE: <existing database name> ZENworks default is postgres
- ♦ PGPASSWORD: <password for the above super user PGUSER>

**Example 7-1** Example for Linux Primary Server:

Open the terminal and set the following variables based on the existing configuration.

- ♦ export LD\_LIBRARY\_PATH=/opt/microfocus/zenworks/share/pgsql/lib\$LD\_LIBRARY\_PATH;
- ♦ export PATH=\$PATH:/opt/microfocus/zenworks/share/pgsql/bin;
- ♦ export PGPORT=54327;
- ♦ export PGDATA=/var/opt/microfocus/pgsql/data;
- ♦ export PGUSER=<super user retrieved from zman dgcs of the source database device>;
- ♦ export PGDATABASE=postgres;
- ♦ export PGPASSWORD=<password retrieved from zman dgcs of the source database device>;

10. Restore the database by running the following commands:

- ♦ **On Linux:**

In the Terminal, run the following commands as PostgreSQL user:

- ♦ Run the following command to restore the database:

```
./psql -w postgres -v ON_ERROR_STOP=0 -f /var/opt/microfocus/pgsql/data/<dump_file> -o /var/opt/microfocus/pgsql/data/<log_file> -L /var/opt/microfocus/pgsql/data/<log_file> > /var/opt/microfocus/pgsql/data/<error_log_location> 2>&1
```

- ♦ **-f <filename>:** Uses the file filename as the source of SQL commands.
- ♦ **-L <filename>:** Writes all query output into the file, in addition to the normal output destination.
- ♦ **-o <filename>:** Logs all query output into the file.

All the errors will be redirected to <error\_log\_location>.

---

**NOTE:** Only /var/opt/microfocus/pgsql/data is accessible by zenpostgres. Hence, keep dump and logs in the same path.

---

For example: ./psql -w postgres -v ON\_ERROR\_STOP=0 -f /var/opt/microfocus/pgsql/data/dump.sql -o /var/opt/microfocus/pgsql/data/postgres\_log.txt -L /var/opt/microfocus/pgsql/data/postgres\_log.log > /var/opt/microfocus/pgsql/data/postgres\_log.err 2>&1

---

**NOTE:**

- ♦ The database schema already exists in the zenpostgres image. Hence, ERROR about existing schema and invalid locale should be ignored.
  - ♦ You can ignore the no version information available error message.
-

- ♦ `./psql -v ON_ERROR_STOP=0 -f <dump_location> -o <log_location> -L <log_location> > <error_log_location> 2>&1`
  - ♦ **-f <filename>**: Uses the file filename as the source of SQL commands. Ensure that the PostgreSQL user has access to the dump being imported.
  - ♦ **-L <filename>**: Writes all query output into the file, in addition to the normal output destination
  - ♦ **-o <filename>**: Logs all query output into the file

All the errors will be redirected to <error\_log\_location>.

```
./psql -v ON_ERROR_STOP=0 -f /tmp/postgres_dump.sql -o /tmp/
postgres_log.txt -L /tmp/postgres_log.log > /tmp/
postgres_log.err 2>&1
```



# 8 Backing Up and Restoring the PostgreSQL Database

- ♦ [“Important Commands and Locations” on page 39](#)
- ♦ [“Backing Up and Vacuuming the PostgreSQL Database” on page 39](#)
- ♦ [“Using the zman command to Take a Backup of the Database” on page 41](#)
- ♦ [“Using the pg\\_dump Command to Take a Backup of the Database” on page 42](#)
- ♦ [“Restoring the PostgreSQL Database” on page 42](#)

## Important Commands and Locations

This section provides some of the command and locations that will be used quite frequently while backing-up the data. Ensure that you keep the following information handy while initiating the data migration:

- ♦ The ZENworks database and Audit database names on the embedded PostgreSQL database are available in `zdm.xml` and `zenaudit.xml` files. These files are available in the following location:

Check the **Database** key to get the database name.

```
/etc/opt/microfocus/zenworks/datamodel/
```

Example:

```
<entry key="Database">zenworks</entry>
```

```
<entry key="Database">zenworksaudit</entry>
```

- ♦ The database details can be retrieved by running the following commands:
  - ♦ `zman dgc`: Retrieves the database credentials used to connect to the embedded database.
  - ♦ `zman dgca`: Retrieves the database credentials used to connect to the audit embedded database.
  - ♦ `zman dgcs`: Retrieves the database credentials of super user that is used to connect to the embedded database.

## Backing Up and Vacuuming the PostgreSQL Database

By default, ZENworks automatically backs up and vacuums the PostgreSQL database every week. By using the `AutomaticPostgresMaintenanceConfigureAction`, you can schedule the automatic backup and vacuum of the database. The default database backup and vacuum schedule is set to Sunday midnight. However, you can use the `pgsql-maintenance.xml` file to customize the schedule.

---

**NOTE:** ♦The AutomaticPostgresMaintenanceConfigureAction takes backup only the Embedded PostgreSQL database.

- ♦ In ZENworks 2020 Update 2 and earlier, pgsqlDBbackupConf.xml was used to take a backup of the PostgreSQL database. While upgrading from Update 2 to the latest version, pgsqlDBbackupConf.xml is replaced with pgsql-maintenance.xml. If customer has enabled backup only for audit, vacuum will also be done only for the audit database. If the backup is done for both databases, the vacuum will also be done for both databases.
- ♦ Database backup directory for Linux should be under /var/opt/microfocus/zenworks
- ♦ If you have configured your zone to use custom JDBC URL for connecting to database, automatic database maintenance actions will not be performed.

---

After successfully running the configure action, the backup (PGDUMP) is saved in the following location:

`/var/opt/microfocus/zenworks/common/ZCMDB_pg_dump/currentBackupFolder`

`/var/opt/microfocus/zenworks/common/AuditDB_pg_dump/currentBackupFolder`

Based on requirements, you can customize the `pgsql-maintenance.xml` file and then run the `microfocus-zenworks-configure -c AutomaticPostgresMaintenanceConfigureAction` command.

The `pgsql-maintenance.xml` file is located in the following location:

`/etc/opt/microfocus/zenworks/pgsql`

---

**NOTE:** By default, only ZENworks database is backed up. If required, you can enable the backup of audit database by setting the `AuditEnabled` parameter to true in the `pgsql-maintenance.xml` file.

---

Following is an example of the `pgsql-maintenance.xml` file:



```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<PostgresMaintenanceInputsType xmlns:ns2="http://www.novell.com/ZENworks/
v1.0/Backup" xmlns:ns3="http://www.novell.com/ZENworks/v1.0">
  <ns2:PostgresBackupInputs>
    <ns2:backupDir>/var/opt/microfocus/zenworks</ns2:backupDir>
    <ns2:backupZen>true</ns2:backupZen>
    <ns2:backupAudit>>false</ns2:backupAudit>
  </ns2:PostgresBackupInputs>
  <ns2:PostgresVacuumInputs>
    <ns2:vacuumZen>true</ns2:vacuumZen>
    <ns2:vacuumAudit>>false</ns2:vacuumAudit>
  </ns2:PostgresVacuumInputs>
  <ns3:Schedule>
    <ns3:DailySchedule>
      <ns3:DaysOfWeek>
        <ns3:Sunday>true</ns3:Sunday>
        <ns3:Monday>>false</ns3:Monday>
        <ns3:Tuesday>>false</ns3:Tuesday>
        <ns3:Wednesday>>false</ns3:Wednesday>
        <ns3:Thursday>>false</ns3:Thursday>
        <ns3:Friday>>false</ns3:Friday>
        <ns3:Saturday>>false</ns3:Saturday>
      </ns3:DaysOfWeek>
      <ns3:StartTime Hour="23" Minute="55" Second="0"/>
    </ns3:DailySchedule>
  </ns3:Schedule>
</PostgresMaintenanceInputsType>
```

---

**IMPORTANT:** In case you have your own mechanism of taking backup of the databases, and do not want ZENworks to take a backup, you can run the following command:

```
microfocus-zenworks-configure -c
AutomaticPostgresMaintenanceConfigureAction -
Dpgsql.delete.maintenance.queue=yes
```

---

## Using the zman command to Take a Backup of the Database

To back up an embedded database, run the zman db command on the server that has the database role.

The backup is created in the following location:

```
/var/opt/microfocus/zenworks/common
```

To configure the backup location and automatic backup schedule, go through the following steps:

1. Go to the backup folder location.
  - ♦ /var/opt/microfocus/zenworks/common

By default, the backup should be available in the *ZCMDB\_pg\_dump* or *AuditDB\_pg\_dump* folder.

2. Go to the following location:

On Linux: `/etc/opt/microfocus/zenworks/pgsql/pgsql-maintenance.xml`

3. Edit the backup location in the `pgsql-maintenance.xml` file.
4. After editing the XML file, run `microfocus-zenworks-configure -c AutomaticPostgresMaintenanceConfigureAction`
5. Restart the PostgreSQL service.

Run the `systemctl restart zenpostgresql.service` command.

6. Run the command `zman db`. The database backup should be created in the location that was updated in the `pgsql-maintenance.xml` file.
7. Modify the schedule in the `pgsql-maintenance.xml` file, and restart the ZENworks PostgreSQL service.

Automatically the backup will be created in the location and schedule specified in the `pgsql-maintenance.xml` file.

## Using the `pg_dump` Command to Take a Backup of the Database

You can use the PostgreSQL `pg_dump` command to do a full online backup of both Embedded and External database.

`export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:`

For more information, see [pg\\_dump](https://www.postgresql.org/docs/11/backup-dump.html) (<https://www.postgresql.org/docs/11/backup-dump.html>).

## Restoring the PostgreSQL Database

Restoring an archived database is performed by using the `pg_restore` command. Using the `pg_restore` command, you can restore the backup taken by both `pg_dump` and `AutomaticPostgresMaintenanceConfigureAction`.

To restore a backup, use the following command:

`pg_restore [connection-option...] [option...] [filename]`

For more information on options, see [pg\\_restore](https://www.postgresql.org/docs/11/app-pgrestore.html) (<https://www.postgresql.org/docs/11/app-pgrestore.html>).

## Restoring the Database Backup Taken Using the ZMAN Command

To restore a backup that was taken using the `zman` command (`zman db`), perform the following:

### On Linux

1. Stop the Services by running the following configure action:

`microfocus-zenworks-configure -c Start`

After running the command, under Action, select Stop.

2. (conditional) If Vertica is enabled in your zone, then stop the following services in ZENworks Appliance, using following command:

- ♦ `systemctl stop zenworks-connect.service`
- ♦ `systemctl stop zenworks-kafka.service`

3. Close the database dangling connections by running the following commands:

```
SELECT pg_terminate_backend(pid) FROM pg_stat_activity WHERE pid =  
pg_backend_pid();
```

4. Change the user by running the `su zenpostgres` command.
5. Set the super user password for `pg_restore` and `dropdb` using the following command:

```
$ PGPASSWORD=<<set_superuser_password>>; export PGPASSWORD
```

6. Set the library path and BIN path as shown below:

- ♦ `LD_LIBRARY_PATH=/opt/microfocus/zenworks/share/pgsql/lib; export LD_LIBRARY_PATH`
- ♦ `PATH=$PATH:/opt/microfocus/zenworks/share/pgsql/bin; export PATH`

7. Drop the existing database, the database name can be `zenworks` or `audit` database.

```
docker exec -it zenpostgres dropdb $ dropdb --host=localhost --  
dbname=<<database_name>> --port=54327 --username=<<superuser_name>>  
<<database_name>>
```

```
$ dropdb --host=localhost --dbname=<<database_name>> --port=54327 --  
username=<<superuser_name>> <<database_name>>
```

8. Restore the database by running the following command:

```
docker exec -it zenpostgres pg_restore $ pg_restore --host=localhost --  
port=54327 --username=<<superuser_name>> --format=directory --  
dbname=<<database_name>> -C <<full path to zenDB_pg_dump>>
```

---

**NOTE:** Ensure that the dump directory is accessible from `zenpostgres` or `postgres` user.

---



# 9 Connecting to the Embedded PostgreSQL Database

To connect to the embedded PostgreSQL database, based on the existing configurations, set the following environment variables:

---

**NOTE:** 1. The parameter values are available in the `zdm.xml` file.

The `zdm.xml` file is available in the following location:

`/etc/opt/microfocus/zenworks/datamodel`

2. Use the `zman dgc` commands to get the user name and password. For more information, see [ZENworks Command Line Utilities Reference](#).

---

- ♦ “On Linux / Appliance” on page 45

## On Linux / Appliance

Open the terminal and execute the following commands:

- ♦ `su zenpostgres`
- ♦ `LD_LIBRARY_PATH=/opt/microfocus/zenworks/share/pgsql/lib; export LD_LIBRARY_PATH`  
**On SLES 15:** `export LD_LIBRARY_PATH=/opt/microfocus/zenworks/share/pgsql/lib:/opt/microfocus/zenworks/share/pgsql/stackbuilder/lib; export LD_LIBRARY_PATH`  
On SLES 15, the *no version information available* error message can be ignored.
- ♦ `PATH=$PATH:/opt/microfocus/zenworks/share/pgsql/bin; export PATH`
- ♦ `PGPORT=<port_number>; export PGPORT`
- ♦ `PGUSER=<user_name>; export PGUSER`
- ♦ `PGDATABASE=<database_name>; export PGDATABASE`
- ♦ `PGPASSWORD=<password>; export PGPASSWORD`
- ♦ `./psql`

Type **quit** to exit from the prompt.



# 10 Cleaning Database and Generating Internal Statistics

PostgreSQL required a strong maintenance mechanism to deal with these DEAD tuples and statistics. VACUUM is the maintenance process which takes care of DEAD tuples along with it analyzes the contents of a tables and collects statistics about the distribution of values in each column of every table.

---

**NOTE:** ZENworks no longer supports Windows Server as a Primary Server from version 24.2 onwards. For more information, see [End of Support Windows Primary Server](#).

---

Perform the following steps to clean the PostgreSQL database:

1. Login to the Primary Server that has the database role.
2. Run the following command to VACUUM on all databases:
  - ♦ `docker exec -it -e PGPORT=<postgresql port> -e PGUSER=<super user retrieved from zman dgcs> -e PGDATABASE=postgres -e PGPASSWORD=<password retrieved from zman dgcs> zenpostgres vacuumdb --analyze --full --skip-locked --verbose --all`

By default, ZENworks uses 54327 as the PostgreSQL port.

---

**NOTE:**

- ♦ Currently, the automatic PostgreSQL backup/vacuum process does not include the Antimalware database. In zones where Antimalware is enabled, you should execute the following command to ensure that the Antimalware database is enabled for backup:

```
docker exec -it -e PGPORT=<postgresql port> -e PGUSER=<super user
retrieved from zman dgcs> -e PGDATABASE=postgres -e
PGPASSWORD=<password retrieved from zman dgcs> zenpostgres vacuumdb --
analyze --full --skip-locked --verbose --all
```

By default, ZENworks uses 54327 as the PostgreSQL port.

- ♦ It is recommends to run these steps every week during the low activity on the database (non-business hours).

For more information, see <https://www.postgresql.org/docs/current/app-vacuumdb.html>

---







# External Database Maintenance

- ♦ Chapter 11, “Configuring the ZENworks Server to Point to the New MS SQL Database Containing Data Moved from Another MS SQL Database,” on page 51
- ♦ Chapter 12, “Configuring the ZENworks Server to Point to the New Oracle Database Containing Data Moved from Another Oracle Database,” on page 53
- ♦ Chapter 13, “Configuring PostgreSQL,” on page 55
- ♦ Chapter 14, “Creating External PostgreSQL Database Schema,” on page 57
- ♦ Chapter 15, “Configuring a ZENworks Server using an Existing Schema,” on page 59
- ♦ Chapter 16, “Backing Up and Restoring the PostgreSQL Database,” on page 61
- ♦ Chapter 17, “Migrating from md5 to scram authentication in PostgreSQL,” on page 63
- ♦ Chapter 18, “Database Activities,” on page 65



# 11 Configuring the ZENworks Server to Point to the New MS SQL Database Containing Data Moved from Another MS SQL Database

If you move the data from one MS SQL database to another MS SQL database, the Linux ZENworks Server must be configured to point to the new MS SQL database.

The following sections provide detailed information:

- ♦ [“Preparing to Move the Data” on page 51](#)
- ♦ [“Configuring the ZENworks Server to Point to the New MS SQL Database” on page 52](#)

## Preparing to Move the Data

Before configuring the server to point the new MS SQL database, do the following:

- ♦ Run the ZENworks Diagnostic Center (ZDC) and check for schema differences.

If the ZENworks database tables are not present under the default database schema DBO, then to avoid any issues, contact Micro Focus Customer Center before upgrading your zone.

To find out the database scheme, run the following query in the Microsoft SQL database:

```
SELECT distinct(SCHEMA_NAME(schema_id)) as OWNER FROM sys.objects WHERE  
type='U'
```

- ♦ Ensure that the ZENworks Server is configured to an MS SQL database. Assume that the device that currently host the MS SQL database is called MSDB1.
- ♦ Ensure that you have another device with an MS SQL database installed. Assume that this device is called MSDB2. For more information on how to install an MS SQL database, see [“Prerequisites for External Database”](#).
- ♦ Stop all the ZENworks services on all the ZENworks Servers in the Management Zone.
  - ♦ **On the Primary Server:**
    1. Execute the following command at the server prompt:  

```
/opt/microfocus/zenworks/bin/microfocus-zenworks-configure -c  
Start
```
    2. Enter the number next to the Stop action.
- ♦ Move the data from MSDB1 to MSDB2. For more information about moving the data, see the MS SQL database documentation.

# Configuring the ZENworks Server to Point to the New MS SQL Database

To configure the ZENworks Server to point to the new database (MSDB2), perform the following tasks on the ZENworks Server:

- 1 Take back up, and edit the `zdm.xml` and `zenaudit.xml` files located in `/etc/opt/microfocus/zenworks/datamodel` and `amedatasource.properties` located in `/etc/opt/microfocus/zenworks/antimalware` to do the following:
  - ♦ Ensure that the value of the `Port` entry key is the port number on which the MS SQL database is running.
  - ♦ Set the value of the `Server` entry key to the IP address of the MSDB2 device.
  - ♦ Set the value of the `Database` entry key to path of the database directory of the MSDB2 device.
  - ♦ If user name or password of the database is changed, then you must change the user name and password in the following locations:
    1. `dmaccounts.properties` and `zenaudit_dmaccounts.properties` files. (located in `/etc/opt/microfocus/zenworks/datamodel` on Linux). It is recommended that do not use the SA user for ZENworks database access.  
`username=password`  
The password will be encrypted automatically when you restart the ZENworks services.
    2. `dmmappings.properties` located in `/etc/opt/microfocus/zenworks/datamodel` on Linux.  
If you have changed the ZEN user name, then update `readonly` and `superuser` fields. If you have changed audit user, then change the `audit` field.
- 2 Run the `microfocus-zenworks-configure -c GenerateOSPProperties` command on all the ZENworks Servers.
- 3 (Conditional) If you have enabled Kafka, then Run `zman server-role-kafka-recreate-connectors -f true` command to recreate the Kafka connectors.  
For more information, See Server Commands in the [ZENworks Command Line Utilities](#).
- 4 Run `microfocus-zenworks-configure -c GenerateContentDatasourceConfigureAction` on all the ZENworks servers.
- 5 Restart the ZENworks services on all other ZENworks Servers in the Management Zone.

## On Primary Server:

1. Execute the following command at the server prompt:  
`/opt/microfocus/zenworks/bin/microfocus-zenworks-configure -c Start`
2. Enter the number next to the `Start` action.

# 12 Configuring the ZENworks Server to Point to the New Oracle Database Containing Data Moved from Another Oracle Database

If you move the data from one Oracle database to another Oracle database, the Linux ZENworks Server must be configured to point to the new Oracle database.

The following sections provide detailed information:

- ♦ [“Preparing to Move the Data” on page 53](#)
- ♦ [“Configuring the ZENworks Server to Point to the New Oracle Database” on page 54](#)

## Preparing to Move the Data

Before configuring the server to point the new Oracle database, do the following:

- ♦ Ensure that the ZENworks Server is configured to an Oracle database. The database can be installed on the ZENworks Server or on a different device. Assume that the device that currently host the Oracle database is called ORDB1.
- ♦ Ensure that you have another device with an Oracle database installed with the same database credentials as the ORDB1. Assume that this device is called ORDB2. For more information on how to install an Oracle database, see [“Prerequisites for External Database”](#).
- ♦ Move the data from ORDB1 to ORDB2. For more information about moving the data, see the Oracle database documentation.
- ♦ Stop the ZENworks services on the ZENworks Servers in the Management Zone.
  - ♦ **On Primary Server:**
    1. Execute the following command at the server prompt:

```
/opt/microfocus/zenworks/bin/microfocus-zenworks-configure -c  
Start
```
    2. Enter the number next to the `Stop` action.

# Configuring the ZENworks Server to Point to the New Oracle Database

To configure the ZENworks Primary Server to point to the new Oracle database (ORDB2), perform the following tasks on the ZENworks Primary Server:

- 1 Take back up, and edit the `zdm.xml` and `edit zenaudit.xml` files (located in `/etc/opt/microfocus/zenworks/datamodel`), and `amedatasource.properties` (located in `/etc/opt/microfocus/zenworks/antimalware` to do the following:
  - ♦ Ensure that the value of the `Port` entry key is the port number on which the Oracle database is running.
  - ♦ Set the value of the `Server` entry key to the IP address of the ORDB2 device.
  - ♦ Set the value of the `Database` entry key to net service name of the Oracle database installed on the ORDB2 device.
- 2 Run the `microfocus-zenworks-configure -c GenerateOSPProperties` command.
- 3 Run `microfocus-zenworks-configure -c GenerateContentDatasourceConfigureAction` on all the ZENworks servers.
- 4 Start the ZENworks services on all the ZENworks Servers in the Management Zone.
  - ♦ **On the Primary Server:**
    1. Execute the following command at the server prompt:

```
/opt/microfocus/zenworks/bin/microfocus-zenworks-configure -c Start
```
    2. Enter the number next to the `Start` action.

ZENworks Server should now point to the new database.

# 13

## Configuring PostgreSQL

Each Primary Server on PostgreSQL can consume up to 500 database connections. If you want to add more than one Primary Server, you should change the property values in the `postgresql.conf` file.

For the external database, the `postgresql.conf` file is available at the following location:

- ♦ **On the Primary Server:** `/var/lib/pgsql/<version>/data`

Use the following formula to calculate the property values.

`max_connections` = Number of primary servers \* 500

`max_prepared_transactions` = Number of primary servers \* 500

After changing the property values in `postgresql.conf`, restart the ZENworks and PostgreSQL service.

To start the PostgreSQL service:

To start the service, run the `systemctl start postgresql.service` command.

For more information, see [PostgreSQL Tuning and Maintenance](#) in the [ZENworks Best Practices Guide](#).





# 14 Creating External PostgreSQL Database Schema

To create external or Remote OEM PostgreSQL schema, perform the following:

- 1 Insert or mount the ZENworks installation ISO on the server where you want to install the Remote OEM or External PostgreSQL database.
- 2 Open the command prompt and run the following command:  
`setup.sh -c`
- 3 In the Select the database to be configured page, select ZENworks Database, Audit Database or both, and then click Next.
- 4 In the Select Database type page, select Remote OEM or External PostgreSQL database.
- 5 Specify the following for both the ZENworks and Audit databases:
  - ♦ Server Name
  - ♦ Port Number
  - ♦ Database Name
  - ♦ User Name
  - ♦ Password



# 15 **Configuring a ZENworks Server using an Existing Schema**

You can configure a ZENworks Server to point to the Microsoft SQL or Oracle database using an existing schema. For information on creating the schema for Microsoft SQL or Oracle, see [Configuring the First Primary Server with an Existing Schema](#).



# 16 Backing Up and Restoring the PostgreSQL Database

For more information on Back-up and restore of the PostgreSQL database, see [“Backing Up and Restoring the PostgreSQL Database” on page 39](#).



# 17 Migrating from md5 to scram authentication in PostgreSQL

From PostgreSQL 13 onwards, the default password encryption is changed from "md5" to "scram-sha-256". This document provides the complete information on how to migrate from md5 to scram authentication.

To avoid any connection issues with password encryption, ensure that you perform one of the following processes:

## MD5

The method md5 uses a custom and less secure challenge-response mechanism. It prevents password sniffing, and storing passwords on the server in plain text, but it provides no protection if an attacker manages to steal the password hash from the server, the MD5 hash algorithm is nowadays no longer considered secure against determined attacks.

To continue using the md5 encryption, change the method in `pg_hba.conf` file for all entries to md5 and restart the PostgreSQL database server services.

## SCRAM-SHA-256

The method scram-sha-256 performs SCRAM-SHA-256 authentication, as described in RFC 7677. It is a challenge-response scheme that prevents password sniffing on untrusted connections and supports storing passwords on the server in a cryptographically hashed form which is thought to acquire.

- ♦ By default, the freshly installed PostgreSQL user scram-sha=256 authentication. Hence, no modifications are required.
- ♦ If you are migrating the database to PostgreSQL 13.x from an older version, then perform the following:

---

**NOTE:** Ensure that you upgrade the ZENworks Management Zone after performing the following steps.

---

1. Modify the method in `pg_hba.conf` file for all entries to md5.
2. Modify the "password\_encryption" parameter in `postgresql.conf` file to "md5".
3. Restart the PostgreSQL database services.
4. Migrate old data (OR) restore old data from backup, along with the users which are encrypted in "md5".

5. Connect the PostgreSQL using from psql/sql-client with super user (postgres/zenpostgres user) and run the following commands:
  - a. `alter system set password_encryption = 'scram-sha-256';`
  - b. `select pg_reload_conf();`
  - c. run below command for all database users including superuser/zenworks/audit, etc..  
`ALTER USER <<user_name>> WITH PASSWORD '<<existing (or) new password>>';`
  - d. To verify if the password is encrypted to scram-sha-256 run the below query.  
`select username,passwd from pg_shadow;`
  - e. Exit
6. Modify the method in pg\_hba.conf file for all entries to scram-sha-256.
7. Restart the PostgreSQL database services.



# 18 Database Activities

This section provides the steps that database administrators should perform after successfully completing any of the below mentioned database activities:

- ♦ Migrate the database from one server to another.
- ♦ Migrate database from one provider to another (Example: MS SQL to Oracle)
- ♦ Update the External PostgreSQL database version.

If your Management Zone has multiple servers, only then perform the following steps:

---

**NOTE:** If there is only one server in the Management Zone, all ZENworks services start automatically after the database activities are completed.

---

1. From the device where you performed the database activity as mentioned, copy the following files, and paste them into the appropriate directory, on all the other Primary Servers in the zone:

- ♦ `zdm.xml`
- ♦ `dmaccounts.properties`
- ♦ `dmmappings.properties`
- ♦ `zenaudit.xml`
- ♦ `zenaudit_dmaccounts.properties`

Ensure that these files have appropriate rights. The files are in the following location:

- ♦ `/etc/opt/microfocus/zenworks/datamodel`
- ♦ Ensure that you run `permissions.sh` script located at `/opt/microfocus/zenworks/bin` on the Linux server after copying the above listed files.

2. (Optional) If Antimalware is enabled in your zone, then update the `amedatasource.properties` file:

`/etc/opt/microfocus/zenworks/antimalware`

3. Run the `microfocus-zenworks-configure -c GenerateOSPPProperties` command.

4. Run the following command on all the content servers:

```
microfocus-zenworks-configure -c  
GenerateContentDatasourceConfigureAction
```

5. Restart all the ZENworks services on all the ZENworks Servers in the Management Zone.

6. (Optional) If you have configured ZENworks Reporting in your zone, then you must reconfigure the ZENworks Reporting Server.





# Database Management - Best Practices, Tips, Troubleshooting

This section includes some tips and best practices for database management:

- ♦ [Chapter 19, “Database Ports,” on page 69](#)
- ♦ [Chapter 20, “Database Tips,” on page 71](#)
- ♦ [Chapter 21, “Troubleshooting Database Migration,” on page 77](#)
- ♦ [Chapter 22, “Customizing JDBC URL for ZENworks,” on page 85](#)
- ♦ [Chapter 23, “Using DBeaver to Export SQL Query Results,” on page 87](#)



# 19 Database Ports

Following are the default ZENworks database ports:

Port Number	Description
1433	Default port to connect with the MS SQL database. Based on your configuration, the port number might vary.
1521	Default port to connect with the Oracle database. Based on your configuration, the port number might vary.
54327	Default port to connect with the Embedded and Remote OEM PostgreSQL database.



# 20 Database Tips

- ♦ [“Changing the ZENworks Database User Password” on page 71](#)
- ♦ [“Maximum Pool Size” on page 72](#)
- ♦ [“Verifying Invalid Objects and Indexes in the Oracle Database” on page 72](#)
- ♦ [“Microsoft SQL NTLM V2 Support” on page 75](#)

## Changing the ZENworks Database User Password

Use system administrator or user with DBA privileges to execute the below commands.

Before changing the password ensure that you shutdown all Primary Servers and Reporting Server.

- ♦ **For Oracle:** After you log into Oracle database, run the below SQL statement to change the password for existing user.

```
ALTER USER <<username>> IDENTIFIED BY <<new password>>
```

- ♦ **For Microsoft SQL Server:** Users with SQL Server authentication, might run the below SQL statement to change the password for existing user.

```
ALTER LOGIN <<username>> WITH PASSWORD = 'new password' OLD_PASSWORD = 'old password'
```

Users with authentication, might change the Windows user account password and no changes required in SQL Server.

After changing the database user password, perform the following steps to change the password in all the Primary Servers:

- 1 Go to the following path;  
`/etc/opt/microfocus/zenworks/datamodel`
- 2 Back up the `dmaccounts.properties` and `zenaudit_dmaccounts.properties` files.
- 3 Open the `dmaccounts.properties` and `zenaudit_dmaccounts.properties` files.
- 4 Change the existing password that is in encrypted form.  
For example, the existing value is `ZENWORKS=@OB@6a7278626bba`  
and the changed value is `ZENWORKS=<<new password>>`
- 5 Run the `microfocus-zenworks-configure -c GenerateOSPPProperties` command.
- 6 Run the `microfocus-zenworks-configure -c GenerateContentDatasourceConfigureAction` command on all the ZENworks servers.
- 7 On all the Primary Servers in the zone, restart the ZENworks services by running the following command: `microfocus-zenworks-configure -c Start`.

---

**IMPORTANT:** Do not encrypt the new password. The new password is encrypted automatically when you restart the ZENworks services.

---

## Maximum Pool Size

The MaxPoolSize value configured in the `zdm.xml` file governs the maximum number of connections allowed in a database connection pool from a Primary Server.

The `zdm.xml` file is located on the Primary Server:

```
/etc/opt/microfocus/zenworks/datamodel
```

With the default MaxPoolSize value of 100, the ZENServer and ZENLoader services currently create a single thread pool. As a result, under the peak load, there are 100 possible connections each from ZENloader and ZENserver.

However, the database server should be able to accept and serve  $200 * <N>$  concurrent connections from the ZENworks context, where  $N$  is the number of Primary Servers in the ZENworks Zone.

The current default value is sufficient for most configurations and loads. It is recommended not to customize the MaxPoolSize value; to closely monitor the database, functionality and performance of ZENworks.

If you want to configure a higher value for the MaxPoolSize parameter, ensure that the number of concurrent connections that can be accepted or served by the database server is greater than or equal to  $2 * <Configured MaxPoolSize> * <Number of Primary Servers>$ .

---

**NOTE:** The default MinPoolSize value is 5.

---

## Verifying Invalid Objects and Indexes in the Oracle Database

If you encounter Invalid Objects or Invalid Index in ZENworks Control Center or ZENworks logs, you can use the following queries to verify if there are any Invalid objects and Indexes in the Oracle database

---

**NOTE:** ♦Do NOT use Squirrel

---

```
SELECT * FROM USER_OBJECTS WHERE STATUS != 'VALID' ;
SELECT * FROM USER_INDEXES WHERE FUNCIDX_STATUS = 'DISABLED' ;
SELECT * FROM USER_INDEXES WHERE STATUS NOT IN ('VALID', 'USABLE', 'N/A') ;
SELECT * FROM USER_IND_PARTITIONS WHERE STATUS NOT IN ('N/A', 'USABLE') ;
SELECT * FROM USER_IND_SUBPARTITIONS WHERE STATUS NOT IN ('USABLE') ;
SELECT * FROM USER_CONSTRAINTS WHERE STATUS != 'ENABLED' ;
```

The above queries should NOT return any records.

If these queries return any records, then you need to run the below scripts in the respective schema:



```

DECLARE
    V_CNT NUMBER;
BEGIN
    SELECT COUNT(1)
    INTO V_CNT
    FROM USER_OBJECTS
    WHERE STATUS      = 'INVALID'
    AND OBJECT_TYPE IN ( 'PROCEDURE', 'TRIGGER', 'FUNCTION', 'VIEW' );
    WHILE V_CNT      > 0
    LOOP
        FOR REC IN
            (SELECT *
            FROM USER_OBJECTS
            WHERE STATUS      = 'INVALID'
            AND OBJECT_TYPE IN ( 'PROCEDURE', 'TRIGGER', 'FUNCTION', 'VIEW' )
            )
        LOOP
            BEGIN
                EXECUTE IMMEDIATE 'ALTER ' || REC.OBJECT_TYPE || '
' || REC.OBJECT_NAME || ' " COMPILE';
            EXCEPTION
                WHEN OTHERS THEN
                    NULL;
            END;
        END LOOP;
        SELECT COUNT(1)
        INTO V_CNT
        FROM USER_OBJECTS
        WHERE STATUS      = 'INVALID'
        AND OBJECT_TYPE IN ( 'PROCEDURE', 'TRIGGER', 'FUNCTION', 'VIEW' );
    END LOOP;
END;
/

```

```

DECLARE
    V_CNT NUMBER;
BEGIN
    SELECT COUNT(1)
    INTO V_CNT
    FROM USER_INDEXES
    WHERE FUNCIDX_STATUS= 'DISABLED'
    AND TEMPORARY      = 'N'
    AND INDEX_TYPE!    = 'LOB'
    AND PARTITIONED    = 'NO'
    AND TABLE_NAME    IN
        (SELECT TABLE_NAME FROM USER_TABLES WHERE TEMPORARY= 'N'
        )
    AND TABLE_NAME NOT IN ( 'PLAN_TABLE', 'ZENUPGRADELOG', 'ZENUPG_INDEX' );
    WHILE V_CNT      > 0
    LOOP
        FOR REC IN
            (SELECT *
            FROM USER_INDEXES
            WHERE FUNCIDX_STATUS= 'DISABLED'

```

```

AND TEMPORARY          = 'N'
AND INDEX_TYPE!        = 'LOB'
AND PARTITIONED        = 'NO'
AND TABLE_NAME        IN
    (SELECT TABLE_NAME FROM USER_TABLES WHERE TEMPORARY='N'
    )
AND TABLE_NAME NOT IN ( 'PLAN_TABLE', 'ZENUPGRADELOG', 'ZENUPG_INDEX' )
)
LOOP
    EXECUTE IMMEDIATE 'ALTER INDEX ' || REC.INDEX_NAME || ' ENABLE';
END LOOP;
SELECT COUNT(1)
INTO V_CNT
FROM USER_INDEXES
WHERE FUNCIDX_STATUS='DISABLED'
AND TEMPORARY          = 'N'
AND INDEX_TYPE!        = 'LOB'
AND PARTITIONED        = 'NO'
AND TABLE_NAME        IN
    (SELECT TABLE_NAME FROM USER_TABLES WHERE TEMPORARY='N'
    )
AND TABLE_NAME NOT IN ( 'PLAN_TABLE', 'ZENUPGRADELOG', 'ZENUPG_INDEX' );
END LOOP;
END;
/

```

```

BEGIN
FOR REC1 IN
    (SELECT *
    FROM USER_INDEXES
    WHERE PARTITIONED='YES'
    AND TABLE_NAME IN
        (SELECT TABLE_NAME FROM USER_TABLES WHERE TEMPORARY='N'
        )
    AND TABLE_NAME NOT IN ( 'PLAN_TABLE', 'ZENUPGRADELOG', 'ZENUPG_INDEX' )
    )
LOOP
    IF REC1.STATUS='N/A' THEN
        FOR REC2 IN
            (SELECT *
            FROM USER_IND_PARTITIONS
            WHERE INDEX_NAME=REC1.INDEX_NAME
            AND STATUS!      = 'USABLE '
            )
        LOOP
            IF REC1.STATUS='N/A' THEN
                FOR REC3 IN
                    (SELECT *
                    FROM USER_IND_SUBPARTITIONS
                    WHERE INDEX_NAME =REC2.INDEX_NAME
                    AND PARTITION_NAME=REC2.PARTITION_NAME
                    AND STATUS!      = 'USABLE '
                    )
                LOOP
                    EXECUTE IMMEDIATE 'ALTER INDEX ' || REC3.INDEX_NAME || ' REBUILD

```

```

SUBPARTITION '||REC3.SUBPARTITION_NAME;
        END LOOP;
    ELSE
        EXECUTE IMMEDIATE 'ALTER INDEX '||REC2.INDEX_NAME||' REBUILD
PARTITION '||REC2.PARTITION_NAME;
        END IF;
    END LOOP;
END IF;
END LOOP;
FOR REC IN
(SELECT * FROM USER_INDEXES WHERE FUNCIDX_STATUS!='ENABLED'
)
LOOP
    EXECUTE IMMEDIATE 'ALTER INDEX '||REC.INDEX_NAME||' ENABLE';
END LOOP;
END;
/

```

Verify the invalid objects after executing above scripts. The following queries should NOT return any records:

```

SELECT * FROM USER_OBJECTS WHERE STATUS='INVALID';
SELECT * FROM USER_INDEXES WHERE FUNCIDX_STATUS='DISABLED';
SELECT * FROM USER_INDEXES WHERE STATUS='INVALID';
SELECT * FROM USER_IND_PARTITIONS WHERE STATUS NOT IN ('N/A','USABLE');
SELECT * FROM USER_IND_SUBPARTITIONS WHERE STATUS NOT IN ('USABLE');
SELECT * FROM USER_CONSTRAINTS WHERE STATUS!='ENABLED';

```

## Microsoft SQL NTLM V2 Support

The Microsoft JDBC Driver for SQL Server supports NTLM v2 only, which offers enhanced security compared to the older NTLM v1 protocol.

To configure the system to allow only NTLM v2 and block NTLM v1, perform the following steps:

1. Open Group Policy Editor.
2. Navigate to Computer Configuration > Windows Settings > Security Settings > Local Policies > Security Options.
3. Right click on Network Security: LAN Manager authentication level, select properties, and then select Send NTLMv2 response only. Refuse LM & NTLM from the drop-down menu.

---

**IMPORTANT:** Disabling NTLM v1 could affect other applications that rely on it for authentication.

---

For more information, see [Using NTLM Authentication](#).



# 21 Troubleshooting Database Migration

- ♦ “After database migration ZEUS is unable to contact the new database server” on page 77
- ♦ “During Appliance deployment, configuring External PostgreSQL using the New Schema option fails to create Audit database” on page 78
- ♦ “Troubleshooting a Java Heap Space Exception” on page 78
- ♦ “Troubleshooting an Oracle Database Crash” on page 79
- ♦ “Troubleshooting an Oracle Tablespace Issue” on page 79
- ♦ “Troubleshooting the Database Migration Failure Issue” on page 79
- ♦ “Troubleshooting the Database Migration by Using An Existing User Schema” on page 80
- ♦ “Troubleshooting the ORA-01652: unable to extend temp segment by 128 in tablespace TEMP” on page 80
- ♦ “Troubleshooting the ORA-01400: cannot insert NULL” on page 81
- ♦ “Troubleshooting the ORA-12516, TNS: listener could not find available handler with matching protocol stack” on page 81
- ♦ “Troubleshooting the database migration failure, duplication of serial numbers in the zDevice table” on page 81
- ♦ “Configure action fails with exception `ZEN configure action container is not responding`” on page 82

## After database migration ZEUS is unable to contact the new database server

If you have added the `database.conf` file to override the default JDBC URL and drivers values. After database migration, the content of the `database.conf` file might not be updated with the latest database information. Hence, ZEUS might not be able to contact the new database server.

### Workaround:

By default, the details in the `ZDM.xml` file is used to communicate between database server and ZEUS. If you want to override the default JDBC driver class and JDBC URL values, then updated the `database.conf` file with the new database information.

To use the default values for driver class and JDBC URL, delete the `database.conf`, else value in this file will be used to communicate with ZEUS.

The `database.conf` file is available in the following location:

```
/etc/opt/microfocus/zenworks/datamodel
```

# During Appliance deployment, configuring External PostgreSQL using the New Schema option fails to create Audit database

While deploying the ZENworks 2017 Appliance, if you configure external PostgreSQL as the database (ZENworks and Audit) using the New Schema option, then the Audit database might not be created.

**Workaround:** Create a PostgreSQL database schema and then deploy the appliance. During the appliance deployment, configure the external PostgreSQL database using the Existing Schema option.

For more information on creating PostgreSQL schema, see the [“Creating External PostgreSQL Database Schema” on page 57](#) section.

## Troubleshooting a Java Heap Space Exception

If you encounter a Java heap space exception during the database migration because of low memory:

- 1 Edit the `ZENworks_installation_path\bin\novell-zenworks-configure.bat` file on `/opt/novell/zenworks/bin/novell-zenworks-configure` on Linux to change the heap space value in the following line, depending upon the RAM of the device where the migration utility is running:

```
"%JAVA_HOME%\bin\java" -Djava.library.path=%ZENLIB% -cp "%MYCP%"
%DEBUG_OPTS% %JAVA_OPTS% -Xmx1024m
com.novell.zenworks.configure.ConfigureLoader %CONFIG_OPTS%
```

The heap space value is represented in megabytes (MB) within `-Xmx1024m`. By default, it is 1024.

For example, if the RAM of the device is 2048 MB, then the line in the `novell-zenworks-configure.bat` file can be updated as follows:

```
"%JAVA_HOME%\bin\java" -Djava.library.path=%ZENLIB% -cp "%MYCP%"
%DEBUG_OPTS% %JAVA_OPTS% -Xmx2048m
com.novell.zenworks.configure.ConfigureLoader %CONFIG_OPTS%
```

---

**IMPORTANT:** The heap space value must be either equivalent to or less than the RAM of the device. It is recommended to have a minimum 2048 MB to continue with database migration.

---

- 2 At the console prompt, run the `\bin\novell-zenworks-configure.bat` file on `/opt/novell/zenworks/bin/novell-zenworks-configure`.
- 3 Follow the prompts.

When you are prompted to enter the location of the file required for resuming the migration, enter the complete path of `DBMigration.xml`. The file is located in the `/opt/novell/zenworks/bin` directory.

The XML file contains a list of tables and a flag indicating whether the table was successfully migrated or not. When the database migration resumes, only the tables with flag value set to False are migrated.

# Troubleshooting an Oracle Database Crash

If the Oracle database crashes during the database migration:

- 1 At the console prompt, run the `ZENworks_installation_path\bin\novell-zenworks-configure.bat` file on `/opt/novell/zenworks/bin/novell-zenworks-configure`.
- 2 Follow the prompts.

When you are prompted to enter the location of the file required for resuming the migration, enter the complete path of `DBMigration.xml`. The file is located in the `/opt/novell/zenworks/bin` directory on Linux.

The XML file contains a list of tables and a flag indicating whether the table was successfully migrated or not. When the database migration resumes, only the tables with flag value set to False are migrated.

---

**IMPORTANT:** Do not edit the contents of `DBMigration.xml`.

---

## Troubleshooting an Oracle Tablespace Issue

If the Oracle USERS tablespace does not have sufficient space to create and store the ZENworks database schema, the database migration fails with the following error messages while trying to create the tables:

```
SEVERE: Terminating the database migration.  
SEVERE: An error has occurred while migrating the database.
```

To resolve this issue, perform the following:

- The Oracle database administrator must increase the size of the USERS tablespace.
- Ensure that the tablespace has a minimum of 100 MB to create ZENworks database schema without any data in it.
- Ensure to have some extra space depending upon the size of the database to be migrated.

## Troubleshooting the Database Migration Failure Issue

If the `NLS_CHARACTERSET` parameter is not set to `AL32UTF8` and the `NLS_NCHAR_CHARACTERSET` parameter is not set to `AL16UTF16`, the database migration fails with the following error messages:

```
Failed to run the sql script: localization-updater.sql,  
message:Failed to execute the SQL command: insert into  
zLocalizedMessage(messageid,lang,messagestr)  
values('POLICYHANDLERS.EPE.INVALID_VALUE_FORMAT','fr','La stratÃ©gie {0}  
n'a  
pas pu Ãªtre appliquÃ©e du fait que la valeur de la variable "{1}" n'est  
pas  
dans un format valide.'),  
message:ORA-00600: internal error code, arguments: [ktfbbsearch-7], [8],  
[],  
[], [], [], [], []
```

To resolve the issue of database migration failure:

- ♦ Set the NLS\_CHARACTERSET parameter to AL32UTF8 and the NLS\_NCHAR\_CHARACTERSET parameter to AL16UTF16..
- ♦ Ensure that the character set parameters are configured with the recommended values, run the following query at the database prompt:

```
select parameter, value from nls_database_parameters where parameter
like '%CHARACTERSET%';
```

## Troubleshooting the Database Migration by Using An Existing User Schema

If you choose to migrate the database by using an existing user schema, the database migration utility creates the ZENworks database but it might fail to migrate the data.

To resolve this issue:

- 1 Ensure that the ZENworks tables, views, and user sequence are deleted from the newly created ZENworks database by the database administrator. Later on, clear the `user_recyclebin` database table.
- 2 Start the database migration again by using the same user schema.

## Troubleshooting the ORA-01652: unable to extend temp segment by 128 in tablespace TEMP

The **ORA-01652: unable to extend temp segment by 128 in tablespace TEMP** error appears, when the temp tablespace have default size and the current transaction involves more number of records with `clob` data type.

To troubleshoot this scenario, perform either of the following or both:

1. Add extra `dbf` file for temp tablespace.
2. Reduce the number of records for the transaction (For Database Migration if you are using default batch size 10,000 that can be reduced to 100 or 500 or 1000 based on the nature of each record having `clob` data.).

---

**IMPORTANT:** For better performance depending on the database size and resource availability, you can increase the `batchsize` in the `db-migration-mssql-to-oracle.properties` file. It is located in `/etc/opt/novell/zenworks/conf/` on Linux.

---



# Troubleshooting the ORA-01400: cannot insert NULL

The **cannot insert NULL** error appears, when you are migrating MS SQL to Oracle database.

To troubleshoot this scenario, perform the following:

1. Go to the `database-table-column-mapping.properties` file located at the following path:  
  
`/etc/opt/novell/zenworks/conf`
2. Open the `database-table-column-mapping.properties` file.
3. Append the Table name, Column, and the Replacement value according to the the `database-table-column-mapping.properties` file format.

---

**IMPORTANT:** If you donot provide a replacement value, then it will take an empty character as a replacement value.

---

# Troubleshooting the ORA-12516, TNS: listener could not find available handler with matching protocol stack

The **TNS: listener could not find available handler with matching protocol stack** error appears, when you are migrating MS SQL to Oracle database.

To troubleshoot this scenario, increase the oracle processes by using the below SQL statement and bounce the database to reflect the changes.

```
SQL> alter system set processes=500 scope=spfile
```

# Troubleshooting the database migration failure, duplication of serial numbers in the zDevice table

In the ZENworks database on the Microsoft SQL server, if the unique constraint is not created on the serial number column of the zDevice table, duplicate serial numbers might be assigned to the devices that are connected to the Management Zone. Because of the duplicate serial numbers, the database migration from the Microsoft SQL database to the Oracle database might fail.

To identify and remove the duplicate serial number in the zDevice table, perform the following steps in the Microsoft SQL database:

1. Use the following query to identify the duplicate serial numbers in the Microsoft SQL database:

```
SELECT * FROM zDevice WHERE serialnumber IN (SELECT serialnumber FROM zDevice GROUP BY serialnumber HAVING COUNT(1) > 1)
```

2. If the duplicate serial numbers are found on the zDevice table, do one of the following:
  - ♦ Remove the device from the Management Zone and register the device again.
  - ♦ Modify the duplicate serial numbers so that each serial number is unique.

To resume the database migration, perform the following in the Oracle database:

1. Use the following query to identify the duplicate serial numbers in the Oracle database:

```
SELECT * FROM zDevice WHERE serialnumber IN (SELECT serialnumber FROM
zDevice GROUP BY serialnumber HAVING COUNT(1) > 1)
```

2. Run the following SQL query to remove the duplicate entry from the target Oracle database:

```
begin
for rec in (select zuid from zDevice where serialnumber is null)
loop
update zDevice
set serialnumber=NVL(serialnumber,'_')||'_'||dbms_random.string('U',
10)
where zuid=rec.zuid;
commit;
end loop;

for rec in (select serialnumber from zDevice group by serialnumber
having count(1) > 1)
loop
update zDevice
set serialnumber=serialnumber||'_'||dbms_random.string('U', 10)
where serialnumber=rec.serialnumber;
commit;
end loop;
```

3. If there are any duplicate serial numbers, run the following query to verify.

If there are any duplicate numbers, run `select serialnumber from zDevice group by serialnumber having count(1) > 1;` and remove the duplicate serial numbers.

If there are no duplicates serial numbers, resume with the migration process.

---

**NOTE:** It is recommends that you clean up the duplicate serial numbers data before starting the database migration. You can continue to use the existing Management Zone without any loss functionality.

---

## Configure action fails with exception ZEN configure action container is not responding

The configure action completes with the following warning:

```
com.novell.zenworks.configure.NonFatalConfigurationException: ZEN configure
action container is not responding
```

This issue occurred because the ZENConfigure container failed initialize within the wait time.

To resolve this issue, increase the wait time of ZENConfigure container by using the `max-retries-for-configure-server=<number of retries>` configuration property in the `/etc/opt/microfocus/zenworks/host-configure-action.properties` directory. The default value for `max-retries-for-configure-server` is 60.

---

**NOTE:** Create the file with the entry if it does not exist.

---



# 22 Customizing JDBC URL for ZENworks

ZENworks provides a way to configure customized JDBC URLs. After providing the URLs, the newer JDBC URL will be used to establish a connection with the database. ZENworks, Audit, and Antimalware databases can be configured to use the customized JDBC URLs.

Perform the following steps to customize the JDBC URLs:

For the ZENworks database, add the following entry in the zdm.xml file:

```
<entry key="Jdbc_Url">JDBC URL HERE</entry>
```

```
/etc/opt/microfocus/zenworks/datamodel/zdm.xml
```

For the Audit database, add the following entry in the zenaudit.xml file:

```
<entry key="Jdbc_Url">JDBC URL HERE</entry>
```

```
/etc/opt/microfocus/zenworks/datamodel/zenaudit.xml
```

---

**NOTE:** Replace JDBC URL HERE with the actual JDBC URL. The URLs can be obtained from your database administrator.

---

## **Antimalware is not enabled, and you want to configure a custom JDBC URL:**

In this scenario, you need to apply the FTF (20.2.0.996 or later) and update the 'database.jdbc.url' key in the amedatasource.properties file available in the following location:

```
/etc/opt/microfocus/zenworks/datamodel/zenaudit.xml
```

For example:

```
database.jdbc.url=jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=[SCAN_VIP])(PORT=[SCAN_PORT])))(CONNECT_DATA=(SERVICE_NAME=[SERVICE_NAME]))),  
even before enabling and starting Antimalware enablement(The other key-value pairs in the file  
should be empty at this point). If this step is performed, the jdbcUrl will be given preference during  
and after Antimalware enablement.
```

## **Antimalware is already enabled and you want to configure a custom JDBC URL:**

In this scenario, you just need to update the in the amedatasource.properties.

However, if you perform a forced migration to the Antimalware database at a later point, then you need to apply the patch and restart the Antimalware service.

After adding the URLs, restart ZENAdmin, ZENClient and ZENLoader services.

---

**NOTE:** ♦If you have multiple primary servers in the zone, and you are configuring a custom JDBC URL, then you need to manually modify the config files on all the primaries.

- ♦ If you are adding a new Primary Server to the zone (which is already configured with custom JDBC URL), then ZENworks takes care of updating the config file with the custom JDBC URL.
-

For example, this ability can be used to configure ZENworks to use Oracle DataGuard.

Following is the DataGuard JDBC driver URL format:

```
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=[SCAN_VIP])(PORT=[SCAN_PORT])))(CONNECT_DATA=(SERVICE_NAME=[SERVICE_NAME])))
```

# 23 Using DBeaver to Export SQL Query Results

The following section explains how to use DBeaver with ZENworks zone database to query and export the data into a CSV file.

1. Install DBeaver on a workstation (not on the ZENworks Primary Server).
2. Open the DBeaver, click Connections, and then select Create Connection.
3. Select PostgreSQL.

Edit the connection properties as shown in the below image.



On the Primary Server, run `zman dgc` to get the zenadmin and password.

Go to `zdm.xml` to collect the database name.

On Linux: `/etc/opt/microfocus/zenworks/datamodel`

4. After connecting, right click on the database on the left side and select SQL Editor > Open SQL Script as show in the below image, and start executing SQL queries.



5. Go to > Window > Preferences > DBeaver > Editors > Data Editor > Lob Editors > Binary data formatter and then select “Hex”.



6. Right click on all the results of the query.





---

**NOTE:** For some tables where comma delimited is not useful (example inventory tables) just copy the results and paste into Excel.

---

7. Select Export Data, and then select CSV.



Ensure that the output includes hex formatting for the binary fields.



8. Send the resulting CSV file.