

How to do a data migration between two DNSManager servers

This article explains how to perform data migration between two DNSManager servers avoiding the XML export - import option from the web interface. The indications included here are recommended in case you want to move your installation onto a new server, or if your server suffered a failure and you want to import your data on a new server.

Step-by-step guide

To migrate DNS Manager between two servers, follow the steps below and keep in mind that this operation can only be done manually.

There are two ways you can perform a migration:

- from the web interface, using the XML export option on the source server and then the XML import for the destination server
- using the latest database dump created by the dbdump script

Back up the Database

1. On the old server, run the `/usr/local/dnsmanager/bin/dbdump` script. This will perform a full database backup.
2. Once it's complete, the file `/var/lib/dnsmanager/dumps/dnsmanagermysql.daily.dump.0.gz` should be copied to the new server in the `/root` directory.

The `dbdump` script uses the `/etc/dnsmanager/dnsmanager.conf` file to read the location of the backups. By default `DUMPS_D` from `local.conf` is set to `/var/lib/dnsmanager/dumps`.

In case this was modified, please note that the database backup will no longer be carried out in `/var/lib/dnsmanager/dumps/dnsmanagermysql.daily.dump.0.gz`.

Get database login from old server

The database login username is taken from the `/etc/dnsmanager/dnsmanager.conf` using the following command:

```
cat /etc/dnsmanager/dnsmanager.conf | grep DB_USER | awk '{print $2}'
```

The output of that command will be called `OLD_DB_USER`.

Set up the new server

If you haven't already installed the latest version of DNSManager, do it now using the [Command Line Installer](#).

Once installed, take the following steps:

1. Obtain the value below. This value will be called `NEW_DB_USER`.

```
cat /etc/dnsmanager/dnsmanager.conf | grep DB_USER | awk '{print $2}'
```

2. Unzip database backup from `/root`.

```
gzip -d dnsmanagermysql.daily.dump.0.gz
```

3. Change user in the dump.

```
sed -i 's/OLD_DB_USER/NEW_DB_USER/g' dnsmanagermysql.daily.dump.0
```

Replace `OLD_DB_USER` with `DB_USER` value from the old server and `NEW_DB_USER` with `DB_USER` value from the new server.

4. Log in to MySQL using the following command:

```
mysql -u`cat /etc/dnsmanager/dnsmanager.conf |grep DB_USER | awk '{print $2}'` -p`cat /etc/dnsmanager/dnsmanager.conf | grep DB_PASSWD | awk '{print $2}'`
```

5. Then execute this:

```
DROP DATABASE dnsmanager;
CREATE DATABASE dnsmanager;
quit
```

6. Load the old database.

```
mysql --force -u`cat /etc/dnsmanager/dnsmanager.conf |grep DB_USER | awk '{print $2}'` -p`cat /etc/dnsmanager/dnsmanager.conf | grep DB_PASSWD | awk '{print $2}'` --default-character-set=utf8 dnsmanager < dnsmanagermysql.daily.dump.0
```

7. In case you migrate from an old version, you need to run `mysql_upgrade` utility.

```
mysql_upgrade -u`cat /etc/dnsmanager/dnsmanager.conf |grep DB_USER | awk '{print $2}'` -p`cat /etc  
/dnsmanager/dnsmanager.conf | grep DB_PASSWD | awk '{print $2}'` --force
```

8. Force install dnsmanager-core.

```
yum -c /etc/dnsmanager-yum.conf reinstall dnsmanager-core
```

9. Restore zone files from the database:

```
/usr/local/dnsmanager/admin/sbin/zonemng -r
```

10. Restart dnsmanager, named, zonemngd and updateurld and that should be it.

Related articles

- [Primary and secondary server setup for 4PSA DNS Manager](#)
- [How to block specific countries from accessing your server](#)
- [How to find out how many DNS queries are being made](#)
- [How to dump zones remotely from a Plesk Windows server](#)
- [How to debug Asterisk and Kamailio](#)