This document describes in a nutshell the steps required to perform RHEL migration on an VMWare, Obviously this includes documentation to specific env that I was working on like the monitoring tools or if its an app server or db server.

**Server Preparation -**

1. Disable the server in Zabbi
2. Disable Monyog Monitoring:
3. In the case the servers are configured under a GTM, disable the server to migrate from the Wide IP list.
4. Wait until the connections start to drop on the server removed from the configuration (5 mins).
5. Backup startup files

MariaDB

cp /etc/my.cnf.d/server.cnf /etc/my.cnf.d/server.cnf.$(date '+%Y%m%d')

Galera

cp /etc/my.cnf.d/galera.cnf /etc/my.cnf.d/galera.cnf.$(date '+%Y%m%d')

1. Stop and disable the services:

MariaDB -

systemctl stop mariadb

systemctl disable mariadb

1. Create a snapshot of the VM (Centos 7)
2. Make sure the system is up-to-date and reinstall if necessary.

yum --disableexcludes=all update

1. Restart the server (In the case if necessary):

reboot

**CentOS 7 to RHEL 7**

1. Remove all packages from extra repositories:
   1. yum list installed | egrep -v "@updates|@anaconda|@base|@centos" | grep "@"
2. Remove the packages that not are part of centos installation
   1. yum erase <package1> <package2> ...
3. Remove EPEL repository
   1. yum erase epel-release
4. Remove extra repositories
   1. cd /etc/yum.repos.d
   2. mv mariadb.repo mariadb.repo.old
   3. mv oracle-java.repo oracle-java.repo.old
   4. mv zabbix.repo zabbix.repo.old
5. Install ConvertRHEL
   1. Download the Redhat GPG key:

1.curl -o /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-releasehttps://www.redhat.com/security/data/fd431d51.txt

1. Download the SSL repository:

1.curl --create-dirs -o /etc/rhsm/ca/redhat-uep.pemhttps://ftp.redhat.com/redhat/convert2rhel/redhat-uep.pem

1. Install repository:

1.curl -o /etc/yum.repos.d/convert2rhel.repohttps://ftp.redhat.com/redhat/convert2rhel/7/convert2rhel.repo

1. Install convertRHEL utility

1.yum install convert2rhel

1. Start convertrhel
   1. Edit a file with the following information:
      1. Execute the tool

2.convert2rhel --org<organization\_ID> --activationkey<activationkey>

1. Wait until Convert2RHEL installs the RHEL package s and finishes successfully.
2. Reboot the machine

1.reboot

1. Verify installation. Now the system is RedHat 7
2. Check the installed version

1.subscription-manager list --installed

1. +------------------------------------------+ Installed Product Status +------------------------------------------+ Product Name: Red Hat Enterprise Linux Server Product ID: 69 Version: 7.9 Arch: x86\_64 Status: Subscribed Status Details: Starts: 02/11/2021 Ends: 02/11/2024
2. Disable conver2rhel 1.

1.yum erase erase convert2rhel

1. mv /etc/yum.repos.d/convert2rhel.repo /etc/yum.repos.d/convert2rhel.repo.old
2. Take a second snapshot of the VM (RHEL 7)

**RHEL 7 to RHEL 8**

1. Enable base repository

#subscription-manager repos --enable rhel-7-server-rpms

1. Enable extra repository

subscription-manager repos --enable rhel-7-server-extras-rpms

1. Set the redhat subscription manager to consume the latest requests

subscription-manager release --unset

1. Update the system

#yum update

1. Reboot the system

reboot

1. Install the dnf:

1.# yum install dnf

1. Install the leap utility:

1.#dnf install leapp-upgrade

1. Uninstall pata\_acpi and floppy

1.# rmmod pata\_acpi floppy

1. Review the preupgrade report

1.# leapp preupgrade --target 8.6

1. Review the report provided byt the tool. Anwer the questions. Example:

1.#leapp answer --section check\_vdo.confirm=True

1. Upgrade the system

1.# leapp upgrade --target 8.6

1. Wait for the command to finish an reboot the system

1.#reboot

1. Take a snapshot of the VM (RHEL 8)

**RHEL 8 to RHEL 9**

1. Update the NetworkManager

2.#systemctl enable --now NetworkManager

1. Install vdo

dnf install vdo

1. Make sure to have the proper repositories installed

subscription-manager repos --enable rhel-8-for-x86\_64-baseos-rpms --enable rhel-8-for-x86\_64-appstream-rpms

1. Lock the system to RHEL 8.7

6.subscription-manager release --unset

1. Update the system

8.#dnf --disableexcludes=all update --allowerasing

1. Reboot

10.#reboot

1. Install the leap utility

12.#dnf install leapp-upgrade

1. Delete previous runs of leapp-upgrade

14.# rm -rf /root/tmp\_leapp\_py3

1. Perform the pre-upgrade analysis

16.#leapp preupgrade --target 9.3

1. Review the report provided byt the tool. Answer the questions. Example:

18.#leapp answer --section check\_vdo.confirm=True

1. Start the update process

20.#leapp upgrade --target 9.3

1. Manually reboot the system

22.#reboot

1. The system is now on RHEL 9.
2. Set the redhat subscription manager to consume the latest requests

subscription-manager release --unset

1. Update the system

dnf update

1. Reboot

27.reboot

1. Run this command.

29.find /etc -name logrotate.timer

1. If it is not any file, enable logrotate to fix it (Due to a bug in leapp)

31.systemctl enable --now logrotate.timer

1. Activate iptables. (Iptables use is suspended on RHEL 9

33.systemctl start iptables

34.systemctl enable iptables

1. If the server has IPv6 configured, activate ip6tables

36.systemctl start ip6tables

37.systemctl enable ip6tables

1. Install EPEL

39.subscription-manager repos --enable codeready-builder-for-rhel-9-$(arch)-rpms

40.dnf installhttps://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm

1. Install MariaDB
   1. cd /opt
   2. wget <https://dlm.mariadb.com/enterprise-release-helpers/mariadb_es_repo_setup>
   3. chmod +x mariadb\_es\_repo\_setup
   4. ./mariadb\_es\_repo\_setup --token="token" --apply --mariadb-server-version="10.6"
   5. dnf install MariaDB-server MariaDB-client
2. Restore the init file:
   1. cp /etc/my.cnf.d/server.cnf.$(date '+%Y%m%d') /etc/my.cnf.d/server.cnf
   2. cp /etc/my.cnf.d/galera.cnf.$(date '+%Y%m%d') /etc/my.cnf.d/galera.cnf
   3. ls -l /etc/my.cnf.d/
3. Start and enable the services:
   1. MariaDB

1.systemctl start mariadb

2.systemctl enable mariadb

1. Install vm-tools

# dnf install open-vm-tools

# systemctl enable vmtoolsd

# systemctl start vmtoolsd

1. Delete snapshots of the VM

[**Migrate to RHEL 9 physical db servers**](https://wecann.icann.org/people/mauro.lozano@icann.org/blog/2023/01/25/migrate-to-rhel-9-physical-db-servers)

**Previous steps**

1. Install RHEL 9 on the new server
2. Configure the new server to be on AD groups, zabbix, splunk, etc.

**Migration Steps**

1. Connect to the source server
2. Convert to root
3. Give permissions to the mysql user to connect:

1.vi /etc/passwd

2.# Assign /bin/bash to the mysql user

3.exit

4.passwd mysql

1. Stop the database service

1.systemctl stop mariadb

2.systemctl disable mariadb

1. Connect to the destination server
2. Convert to root
3. Stop mariadb service

1.systemctl stop mariadb

2.systemctl disable mariadb

1. export variables

1.export source\_server=

1. Test the connection

1.ssh mysql@${source\_server} uptime

1. Copy the configuration files:

1.scp -rpmysql@${source\_server}:/etc/my.cnf.d/server\* /etc/my.cnf.d/

1. Create the /var/log/mysql directory:

1.mkdir /var/log/mysql

2.chown mysql:mysql /var/log/mysql/

3.chmod 755 /var/log/mysql/

1. Convert to mysql
2. Export variable

1.export source\_server=

1. Copy the log files:

1.scp -rpmysql@${source\_server}:/var/log/mysql/\* /var/log/mysql/

1. Copy the data files:

1.scp -rpmysql@${source\_server}:/var/lib/mysql/\* /var/lib/mysql/

1. Change the destination and source ip addresses
2. Revoke permissions from the mysql user to connect:

1.vi /etc/passwd

2.# Assign /sbin/nologin to the mysql user

3.exit

1. Shutdown the source server
2. Start the destination server
3. Start mysql

1.systemctl start mysql

2.systemctl enable mysql

1. Verify the service

[Postgresql Cluster Active Active](https://wecann.icann.org/people/mauro.lozano@icann.org/blog/2023/06/28/postgresql-cluster-active-active)

1.            Create the virtual machines. In this example we are using pdb{3-4}.[dev.org](http://dev.icann.org/)

2.            Connect to the server

3.            Convert to root

4.            Aggregate codeready

subscription-manager repos --enable codeready-builder-for-rhel-9-$(arch)-rpms

5.            Install postgresql

6.                   dnf --disableexcludes=all install postgresql postgresql-server

7.            Initialize database

postgresql-setup --initdb

8.            Enable and activate postgresql in the server

systemctl enable postgresql

systemctl start postgresql

9.            Add the firewall rules

firewall-cmd --zone=public --add-service=postgresql

firewall-cmd --permanent --zone=public --add-service=postgresql

10.         Check configuration file (add the following line)

vi /var/lib/pgsql/data/postgresql.conf

listen\_addresses = [pdb3.dev.org](http://pdb3.dev.icann.org/)

11.         Configure the replication users (add the line)

vi /var/lib/pgsql/data/pg\_hba.conf

host replication all [10.47.230.201/30](http://10.47.230.201/30) ident

12.         Restart postgresql

systemctl restart postgresql

13.         Install Java 8

dnf install jdk1.8

14.         Get Symmetricds

cd /opt

wget <https://sourceforge.net/projects/symmetricds/files/symmetricds/symmetricds-3.14/symmetric-server-3.14.7.zip/download> donwnload.zip

unzip download.zip