

8i | 9i | 10g | 11g | 12c | 13c | 18c | 19c | 21c | Misc | PL/SQL | SQL | RAC | WebLogic | Linux

Home » Articles » 19c » Here

Oracle Database 19c Installation On Oracle Linux 7 (OL7)

This article describes the installation of Oracle Database 19c 64-bit on Oracle Linux 7 (OL7) 64-bit. The article is based on a server installation with a minimum of 2G swap and secure Linux set to permissive. An example of this type of Linux installation can be seen here [here](#).

- [Download Software](#)
- [Hosts File](#)
- [Oracle Installation Prerequisites](#)
 - [Automatic Setup](#)
 - [Manual Setup](#)
 - [Additional Setup](#)
- [Installation](#)
- [Database Creation](#)
- [Post Installation](#)

Related articles.

- [Oracle Universal Installations \(OUI\) Silent Installations](#)
- [Database Configuration Assistant \(DBCA\) : Creating Databases in Silent Mode](#)

Download Software

Download the Oracle software from OTN or MOS depending on your support status.

- [OTN: Oracle Database 19c \(19.3\) Software \(64-bit\)](#)
- [edelivery: Oracle Database 19c \(19.3\) Software \(64-bit\)](#)

Hosts File

The "/etc/hosts" file must contain a fully qualified name for the server.

```
<IP-address> <fully-qualified-machine-name> <machine-name>
```

For example.

```
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
192.168.56.107 ol7-19.localdomain ol7-19
```

Set the correct hostname in the "/etc/hostname" file.

```
ol7-19.localdomain
```

Oracle Installation Prerequisites

Perform either the Automatic Setup or the Manual Setup to complete the basic prerequisites. The Additional Setup is required for all installations.

Automatic Setup

If you plan to use the "oracle-database-preinstall-19c" package to perform all your prerequisite setup, issue the following command.

```
# yum install -y oracle-database-preinstall-19c
```

It is probably worth doing a full update as well, but this is not strictly speaking necessary.

```
# yum update -y
```

It's worth running the all the YUM commands listed in the manual setup section. Depending on the OS package groups you have selected, some additional packages might also be needed.

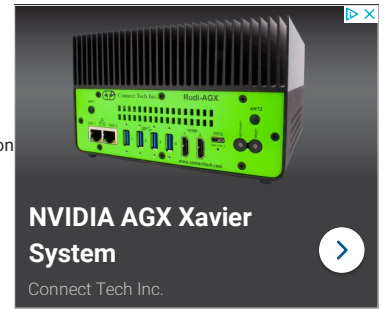
If you are using RHEL7 or CentOS7, you can pick up the PRM from the OL7 repository and install it. It will pull the dependencies from your normal repositories.

```
curl -o oracle-database-preinstall-19c-1.0-1.e17.x86_64.rpm https://yum.oracle.com/repo/OracleLinux/OL7/latest/x86_64/getPackage/oracle-database-preinstall-19c-1.0-1.e17.x86_64.rpm
yum -y localinstall oracle-database-preinstall-19c-1.0-1.e17.x86_64.rpm
```

Manual Setup

If you have not used the "oracle-database-preinstall-19c" package to perform all prerequisites, you will need to manually perform the following setup tasks.

Add the following lines to the "/etc/sysctl.conf" file, or in a file called "/etc/sysctl.d/98-oracle.conf".



```
fs.file-max = 6815744
kernel.sem = 250 32000 100 128
kernel.shmni = 4096
kernel.shmall = 1073741824
kernel.shmmax = 4398046511104
kernel.panic_on_oops = 1
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
net.ipv4.conf.all.rp_filter = 2
net.ipv4.conf.default.rp_filter = 2
fs.aio-max-nr = 1048576
net.ipv4.ip_local_port_range = 9000 65500
```

Run one of the following commands to change the current kernel parameters, depending on which file you edited.

```
/sbin/sysctl -p
# Or
/sbin/sysctl -p /etc/sysctl.d/98-oracle.conf
```

Add the following lines to a file called `/etc/security/limits.d/oracle-database-preinstall-19c.conf` file.

```
oracle soft nofile 1024
oracle hard nofile 65536
oracle soft nproc 16384
oracle hard nproc 16384
oracle soft stack 10240
oracle hard stack 32768
oracle hard memlock 134217728
oracle soft memlock 134217728
```

Someone in the comments suggested you might need to add the previous lines into the `/etc/security/limits.conf` file also for CentOS7. This is definitely not needed for OL7, but worth considering if the installer gives prerequisite failures for these settings.

The following packages are listed as required. Many of the packages should be installed already.

```
yum install -y bc
yum install -y binutils
yum install -y compat-libcap1
yum install -y compat-libstdc++-33
#yum install -y dtrace-modules
#yum install -y dtrace-modules-headers
#yum install -y dtrace-modules-provider-headers
yum install -y dtrace-utils
yum install -y elfutils-libelf
yum install -y elfutils-libelf-devel
yum install -y fontconfig-devel
yum install -y glibc
yum install -y glibc-devel
yum install -y ksh
yum install -y libaio
yum install -y libaio-devel
yum install -y libdtrace-ctf-devel
yum install -y libXrender
yum install -y libXrender-devel
yum install -y libX11
yum install -y libXau
yum install -y libXi
yum install -y libXtst
yum install -y libgcc
yum install -y librdmacm-devel
yum install -y libstdc++
yum install -y libstdc++-devel
yum install -y libxcb
yum install -y make
yum install -y net-tools # Clusterware
yum install -y nfs-utils # ACFS
yum install -y python # ACFS
yum install -y python-configshell # ACFS
yum install -y python-rtslib # ACFS
yum install -y python-six # ACFS
yum install -y targetcli # ACFS
yum install -y smartmontools
yum install -y sysstat

# Added by me.
yum install -y unixODBC
```

Create the new groups and users.

```
groupadd -g 54321 oinstall
groupadd -g 54322 dba
groupadd -g 54323 oper
#groupadd -g 54324 backupdba
#groupadd -g 54325 dgdba
#groupadd -g 54326 kmdba
#groupadd -g 54327 asmdba
#groupadd -g 54328 asmoper
#groupadd -g 54329 asmadmin
#groupadd -g 54330 racdba

useradd -u 54321 -g oinstall -G dba,oper oracle
```

Uncomment the extra groups you require.

Additional Setup

The following steps must be performed, whether you did the manual or automatic setup.

Set the password for the "oracle" user.

```
passwd oracle
```

Set secure Linux to permissive by editing the "/etc/selinux/config" file, making sure the SELINUX flag is set as follows.

```
SELINUX=permissive
```

Once the change is complete, restart the server or run the following command.

```
# setenforce Permissive
```

If you have the Linux firewall enabled, you will need to disable or configure it, as shown [here](#). To disable it, do the following.

```
# systemctl stop firewalld
# systemctl disable firewalld
```

If you are not using Oracle Linux and UEK, you will need to manually [disable transparent huge pages](#).

Create the directories in which the Oracle software will be installed.

```
mkdir -p /u01/app/oracle/product/19.0.0/dbhome_1
mkdir -p /u02/oradata
chown -R oracle:oinstall /u01 /u02
chmod -R 775 /u01 /u02
```

Putting mount points directly under root without mounting separate disks to them is typically a bad idea. It's done here for simplicity, but for a real installation "/" storage should be reserved for the OS.

Unless you are working from the console, or using SSH tunnelling, login as root and issue the following command.

```
xhost +<machine-name>
```

The scripts are created using the cat command, with all the "\$" characters escaped. If you want to manually create these files, rather than using the cat command, remember to remove the "\" characters before the "\$" characters.

Create a "scripts" directory.

```
mkdir /home/oracle/scripts
```

Create an environment file called "setEnv.sh". The "\$" characters are escaped using "\". If you are not creating the file with the cat command, you will need to remove the escape characters.

```
cat > /home/oracle/scripts/setEnv.sh <<EOF
# Oracle Settings
export TMP=/tmp
export TMPDIR=$TMP

export ORACLE_HOSTNAME=ol7-19.localdomain
export ORACLE_UNQNAME=cdb1
export ORACLE_BASE=/u01/app/oracle
export ORACLE_HOME=$ORACLE_BASE/product/19.0.0/dbhome_1
export ORA_INVENTORY=/u01/app/oraInventory
export ORACLE_SID=cdb1
export PDB_NAME=pdb1
export DATA_DIR=/u02/oradata
```

```
export PATH=/usr/sbin:/usr/local/bin:\$PATH
export PATH=\$ORACLE_HOME/bin:\$PATH

export LD_LIBRARY_PATH=\$ORACLE_HOME/lib:/lib:/usr/lib
export CLASSPATH=\$ORACLE_HOME/jlib:\$ORACLE_HOME/rdbms/jlib
EOF
```

Add a reference to the "setEnv.sh" file at the end of the "/home/oracle/.bash_profile" file.

```
echo ". /home/oracle/scripts/setEnv.sh" >> /home/oracle/.bash_profile
```

Create a "start_all.sh" and "stop_all.sh" script that can be called from a startup/shutdown service. Make sure the ownership and permissions are correct.

```
cat > /home/oracle/scripts/start_all.sh <<EOF
#!/bin/bash
. /home/oracle/scripts/setEnv.sh

export ORAENV_ASK=NO
. oraenv
export ORAENV_ASK=YES

dbstart \$ORACLE_HOME
EOF

cat > /home/oracle/scripts/stop_all.sh <<EOF
#!/bin/bash
. /home/oracle/scripts/setEnv.sh

export ORAENV_ASK=NO
. oraenv
export ORAENV_ASK=YES

dbshut \$ORACLE_HOME
EOF

chown -R oracle:oinstall /home/oracle/scripts
chmod u+x /home/oracle/scripts/*.sh
```

Once the installation is complete and you've edited the "/etc/oratab", you should be able to start/stop the database with the following scripts run from the "oracle" user.

```
~/scripts/start_all.sh
~/scripts/stop_all.sh
```

You can see how to create a Linux service to automatically start/stop the database [here](#).

Installation

Log into the oracle user. If you are using X emulation then set the DISPLAY environmental variable.

```
DISPLAY=<machine-name>:0.0; export DISPLAY
```

Switch to the ORACLE_HOME directory, unzip the software directly into this path and start the Oracle Universal Installer (OUI) by issuing one of the following commands in the ORACLE_HOME directory. The interactive mode will display GUI installer screens to allow user input, while the silent mode will install the software without displaying any screens, as all required options are already specified on the command line.

```
# Unzip software.
cd $ORACLE_HOME
unzip -oq /path/to/software/LINUX.X64_193000_db_home.zip

# Interactive mode.
./runInstaller

# Silent mode.
./runInstaller -ignorePrereq -waitforcompletion -silent \
  -responseFile ${ORACLE_HOME}/install/response/db_install.rsp \
  oracle.install.option=INSTALL_DB_SWONLY \
  ORACLE_HOSTNAME=${ORACLE_HOSTNAME} \
  UNIX_GROUP_NAME=oinstall \
  INVENTORY_LOCATION=${ORA_INVENTORY} \
  SELECTED_LANGUAGES=en, en_GB \
  ORACLE_HOME=${ORACLE_HOME} \
  ORACLE_BASE=${ORACLE_BASE} \
  oracle.install.db.InstallEdition=EE \
  oracle.install.db.OSDBA_GROUP=dba \
  oracle.install.db.OSBACKUPDBA_GROUP=dba \
  oracle.install.db.OSGDDBA_GROUP=dba \
  oracle.install.db.OSKMDBA_GROUP=dba \
  oracle.install.db.OSRACDBA_GROUP=dba \
  SECURITY_UPDATES_VIA_MYORACLESUPPORT=false \
  DECLINE_SECURITY_UPDATES=true
```

Run the root scripts when prompted.

```
As a root user, execute the following script(s):
1. /u01/app/orainventory/orainstRoot.sh
2. /u01/app/oracle/product/19.0.0/dbhome_1/root.sh
```

You can read more about silent installations [here](#).

You are now ready to create a database.

Database Creation

You create a database using the Database Configuration Assistant (DBCA). The interactive mode will display GUI screens to allow user input, while the silent mode will create the database without displaying any screens, as all required options are already specified on the command line.

```
# Start the listener.
lsnrctl start

# Interactive mode.
dbca

# Silent mode.
dbca -silent -createDatabase \
      -templateName General_Purpose.dbc \
      -gdbname ${ORACLE_SID} -sid ${ORACLE_SID} -responseFile NO_VALUE \
      -characterSet AL32UTF8 \
      -sysPassword SysPassword1 \
      -systemPassword SysPassword1 \
      -createAsContainerDatabase true \
      -numberOfPDBs 1 \
      -pdbName ${PDB_NAME} \
      -pdbAdminPassword PdbPassword1 \
      -databaseType MULTIPURPOSE \
      -memoryMgmtType auto_sga \
      -totalMemory 2000 \
      -storageType FS \
      -datafileDestination "${DATA_DIR}" \
      -redoLogFileSize 50 \
      -emConfiguration NONE \
      -ignorePreReqs
```

You can read more about silent database creation [here](#).

Post Installation

Edit the "/etc/oratab" file setting the restart flag for each instance to 'Y'.

```
cdb1:/u01/app/oracle/product/19.0.0/dbhome_1:Y
```

Enable Oracle Managed Files (OMF) and make sure the PDB starts when the instance starts.

```
sqlplus / as sysdba <<EOF
alter system set db_create_file_dest='${DATA_DIR}';
alter pluggable database ${PDB_NAME} save state;
exit;
EOF
```

For more information see:

- [Oracle Database 19c : Installation Guide for Linux](#)
- [Automating Database Startup and Shutdown on Linux](#)
- [Oracle Universal Installations \(OUI\) Silent Installations](#)
- [Database Configuration Assistant \(DBCA\) : Creating Databases in Silent Mode](#)

Hope this helps. Regards Tim...

[Back to the Top.](#)

[Contact Us](#)



[Home](#) | [Articles](#) | [Scripts](#) | [Blog](#) | [Certification](#) | [Videos](#) | [Misc](#) | [About](#)

[About Tim Hall](#)

[Copyright & Disclaimer](#)