

5 этап - Automating Oracle database Shutdown and Startup by systemd

Huge Pages on Oracle Linux

```
# Проверить доступные HugePages
grep HugePages /proc/meminfo

# Проверить размер страниц
grep Hugepagesize /proc/meminfo
```

Выключить transparent hugepages

проверить

```
cat /sys/kernel/mm/transparent_hugepage/enabled

# Если always madvise [never] то все в порядке
```

Сервис для отключения transparent hugepages

```
cat > /etc/systemd/system/disable-thp.service << EOF
[Unit]
Description=Disable Transparent Huge Pages
Before=sysinit.target
DefaultDependencies=no
[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/bin/sh -c "echo 'never' > /sys/kernel/mm/transparent_hugepage/enabled && echo 'never' > /sys/kernel/mm/transparent_hugepage/defrag"
[Install]
WantedBy=sysinit.target
EOF

#Автозапуск
systemctl daemon-reload
systemctl enable disable-thp
systemctl start disable-thp
```

Затем - еще раз проверьте

Автоматический расчет

```
# Создать скрипт расчета
cat > /tmp/hugepages_settings.sh << 'EOF'
#!/bin/bash
#
# hugepages_settings.sh
#
# Linux bash script to compute values for the
# recommended HugePages/HugeTLB configuration
#
# Note: This script does calculation for all shared memory
# segments available when the script is run, no matter it
# is an Oracle RDBMS shared memory segment or not.
#
# This script is provided by Doc ID 401749.1 from My Oracle Support
# http://support.oracle.com

# Welcome text
echo "
This script is provided by Doc ID 401749.1 from My Oracle Support
(http://support.oracle.com) where it is intended to compute values for
the recommended HugePages/HugeTLB configuration for the current shared
memory segments. Before proceeding with the execution, note following:
* For ASM instance, it needs to configure ASMM instead of AMM.
* The 'pga_aggregate_target' is outside the SGA and
  you should accommodate this while calculating the overall size.
* In case you changes the DB SGA size,
  as the new SGA will not fit in the previous HugePages configuration,
  it had better disable the whole HugePages,
  start the DB with new SGA size and run the script again.
And make sure that:
* Oracle Database instance(s) are up and running
* Oracle Database 11g Automatic Memory Management (AMM) is not setup
  (See Doc ID 749851.1)
* The shared memory segments can be listed by command:
  # ipcs -m

Press Enter to proceed..."
```

read

Check for the kernel version

```
KERN=`uname -r | awk -F. '{ printf("%d.%d\n",$1,$2); }`
```

Find out the HugePages size

```
HPG_SZ=`grep Hugepagesize /proc/meminfo | awk '{print $2}`
```

```
if [ -z "$HPG_SZ" ];then
```

```
    echo "The hugepages may not be supported in the system where the script is being executed."
```

```
    exit 1
```

```
fi
```

Initialize the counter

```
NUM_PG=0
```

Cumulative number of pages required to handle the running shared memory segments

```
for SEG_BYTES in `ipcs -m | cut -c44-300 | awk '{print $1}' | grep "[0-9][0-9]*`
```

```
do
```

```
    MIN_PG=`echo "$SEG_BYTES/($HPG_SZ*1024)" | bc -q`
```

```
    if [ $MIN_PG -gt 0 ]; then
```

```
        NUM_PG=`echo "$NUM_PG+$MIN_PG+1" | bc -q`
```

```
    fi
```

```
done
```

```
RES_BYTES=`echo "$NUM_PG * $HPG_SZ * 1024" | bc -q`
```

An SGA less than 100MB does not make sense

```
if [ $RES_BYTES -lt 100000000 ]; then
```

```
    echo "*****"
```

```
    echo "*** ERROR ***"
```

```
    echo "*****"
```

```
    echo "Sorry! There are not enough total of shared memory segments allocated for  
HugePages configuration. HugePages can only be used for shared memory segments  
that you can list by command:
```

```
    # ipcs -m
```

of a size that can match an Oracle Database SGA. Please make sure that:

- * Oracle Database instance is up and running

- * Oracle Database 11g Automatic Memory Management (AMM) is not configured"

```
exit 1

fi

# Finish with results
case $KERN in
    '2.4') HUGETLB_POOL=`echo "$NUM_PG*$HPG_SZ/1024" | bc -q`;
        echo "Recommended setting: vm.hugetlb_pool = $HUGETLB_POOL" ;;
    '2.6') echo "Recommended setting: vm.nr_hugepages = $NUM_PG" ;;
    '3.8') echo "Recommended setting: vm.nr_hugepages = $NUM_PG" ;;
    '3.10') echo "Recommended setting: vm.nr_hugepages = $NUM_PG" ;;
    '4.1') echo "Recommended setting: vm.nr_hugepages = $NUM_PG" ;;
    '4.14') echo "Recommended setting: vm.nr_hugepages = $NUM_PG" ;;
    *) echo "Kernel version $KERN is not supported by this script (yet). Exiting." ;;
esac

# End
EOF

chmod +x /tmp/hugepages_settings.sh
```

Quick checking

```
# cat /proc/sys/vm/nr_hugepages
2054
```

or

```
# cat /proc/meminfo | grep Huge
AnonHugePages:      0 kB
HugePages_Total:    2054
HugePages_Free:     546
HugePages_Rsvd:     542
HugePages_Surp:      0
Hugepagesize:       2048 kB
```

Checking if transparent huge pages are disabled

```
# cat /sys/kernel/mm/transparent_hugepage/enabled`
always madvise [never]`
```

Viewing database 12.1 alert log

```
*****
Dump of system resources acquired for SHARED GLOBAL AREA (SGA)
Per process system memlock (soft) limit = 128G
Expected per process system memlock (soft) limit to lock
SHARED GLOBAL AREA (SGA) into memory: 2050M
Available system pagesizes:
4K, 2048K` Mon Oct 16 12:28:54 2017
Supported system pagesize(s):
PAGESIZE AVAILABLE_PAGES EXPECTED_PAGES ALLOCATED_PAGES ERROR(s)
  4K      Configured          3          3      NONE
2048K      2054          1025          1025      NONE
*****
```

Viewing database 11.2 alert log

```
***** Large Pages Information *****
Per process system memlock (soft) limit = 128 GB
Total Shared Global Region in Large Pages = 2050 MB (100%)
Large Pages used by this instance: 1025 (2050 MB)
Large Pages unused system wide = 4 (8192 KB)
Large Pages configured system wide = 2054 (4108 MB)
Large Page size = 2048 KB
*****
If number of Huge pages is less than required, than SGA will use both page types. Database 12.1 aler.log:
PAGESIZE AVAILABLE_PAGES EXPECTED_PAGES ALLOCATED_PAGES ERROR(s)
  4K      Configured          3      187704      NONE
2048K      659          1025          658      NONE
```

Simplified setup

1. Have the memlock user limit set in /etc/security/limits.conf file. Set the value (in KB) slightly smaller than total RAM (90%) , at least it must be bigger than HugePages size. *
soft memlock 14680064* hard memlock 14680064Atentu! If you use systemd to auto startup databases, systemd ignores limits.conf. You need to add the following to a service unit file

```
[Service]
LimitMEMLOCK=infinity
LimitNOFILE=65535
```

2. Get the required number of pages from alert.log (EXPECTED_PAGES) or by running Oracle script hugepages_settings.sh (Doc ID 401749.1)
3. edit vm.nr_hugepages in /etc/sysctl.conf as root
4. to reload the parameters, reboot Linux or use sysctl -p Proper setup and more info

HugePages on Oracle Linux 64-bit (Doc ID 361468.1)

HugePages on Linux: What It Is... and What It Is Not... (Doc ID 361323.1)

Oracle Linux: Shell Script to Calculate Values Recommended Linux HugePages / HugeTLB Configuration (Doc ID 401749.1)

Automating by systemd

alternatives are Oracle Restart or SysV init (/etc/init.d in older Linux).

Service Unit file typically has extension .service and stored in

/usr/lib/systemd/system

/etc/systemd/system

/usr/lib/systemd/user

/etc/systemd/user

Create or edit a service unit file:

```
cd /etc/systemd/system
vi oracle\_database.service
[Unit]
Description=The Oracle Database Service
After=network.target

[Service]
Type=forking
RemainAfterExit=yes
KillMode=none
TimeoutStopSec=10min
# memlock limit is needed for SGA to use HugePages
LimitMEMLOCK=infinity
LimitNOFILE=65535

User=oracle
Group=oinstall
# Please use absolute path here
# ExecStart=$ORACLE\_HOME/bin/dbstart $ORACLE\_HOME &
# First argument of dbstart is used to bring up Listener
ExecStart=/opt/oracle/product/12.1.0/se2\_1/bin/dbstart /opt/oracle/product/12.1.0/se2\_1 &
```

```
ExecStop=/opt/oracle/product/12.1.0/se2\_1/bin/dbshut /opt/oracle/product/12.1.0/se2\_1
Restart=no

[Install]
# Puts wants directive for the other units in the relationship

WantedBy=default.target
```

First argument of dbstart/dbshut is used to bring up/shutdown Oracle Listener. This script will start all databases listed in the /etc/oratab file whose third field is a "Y". If you use ASM or cluster services, read more in dbstart description.

Источник: <https://alexzy.blogspot.com/2017/10/>

Huge Pages

systemd ignores /etc/security/limits.conf. If HugePages are configured, you need to use LimitMEMLOCK and LimitNOFILE, otherwise SGA will use small pages and database alert log will show:

Increase per process memlock (soft) limit to at least 2050MB to lock 100% of SHARED GLOBAL AREA (SGA) pages into physical memory

****TimeoutStopSec****= Configures the time to wait for stop. If a service is asked to stop, but does not terminate in the specified time, it will be terminated forcibly via SIGTERM, and after another timeout of equal duration with SIGKILL (see KillMode= in systemd.kill(5)). Takes a unit-less value in seconds, or a time span value such as "5min 20s". Pass "infinity" to disable the timeout logic.

Defaults to DefaultTimeoutStopSec= from the manager configuration file (see systemd-system.conf(5)). For more info use man systemd.service

reload systemd and enable the service:

```
systemctl daemon-reload
```

```
systemctl enable oracle_database.service
```

list services:

```
systemctl list-unit-files --type service|grep oracle
```

Start the service and check its status

```
systemctl start oracle_database.service
```

```
systemctl status oracle_database.service
```

Источник: [Huge Pages on Oracle Linux](#)

Revision #10

Created 12 November 2025 15:18:00 by Admin

Updated 22 December 2025 11:19:28 by Admin