

ORACLE  
METALINK

Bookmarks Admin Profile Feedback Sign Out Help

Headlines Knowledge Service Request Configuration Support Manager Patches & Updates Forums Certify

Knowledge Browser Advanced Search Bug Search Secure Enterprise Search

Quick Find Knowledge Base   Secure Enterprise Search Advanced Saved Searches

Did this article help solve your problem?  Would you recommend this document to others?

**TIP:** Click [help](#) for a detailed explanation of this page.

Subject: **Kernel setup for Solaris 10 using project files.**

[Doc ID:](#) **Note:429191.1**

Type: **WHITE PAPER**

Last Revision Date: **31-MAY-2007**

Status: **PUBLISHED**

#### In this Document

[Abstract](#)  
[Document History](#)  
[Kernel setup for Solaris 10 using project files.](#)

#### Applies to:

Oracle Server - Enterprise Edition - Version: 9.2.0.1.0 to 10.2.0.99.0  
 Solaris Operating System (SPARC 64-bit)

#### Abstract

This is a Metalink article that covers kernel parameter setup for Solaris 10

If a Solaris System Administrator needs help implementing these concepts, they will need to contact the OS vendor. In this case, that would be Sun M

#### Document History

**Author :** grcraig  
**Create Date** 27-APR-2007

**Update Date** 31-MAY-2007  
**Expire Date** na

@ editorial review by DBMarshall , 4-27-2007

@ remark:73061.1 processed by MWood , 5-31-2007

#### Kernel setup for Solaris 10 using project files.

SUN Sparc Solaris version 10

Generally speaking, information on setting kernel parameters in Solaris 10 is limited. There is a new project file that is used to set kernel parameters ( 10 Operating System. The remainder of this while paper will be divided into 3 sections to aid in understanding the configuration of kernel parameters

#### Basics

#### Questions and Answers

#### Examples

**Permanently Setting Kernel Parameters**  
**Temporarily Setting Kernel Parameters**

#### Basics on /etc/projects

=====

1. Specifying a limit in the /etc/project file extends that limit to all processes belonging to the project.

2. Now, before I show you an example of an /etc/project file containing IPC settings for both the oracle\_oltp and oracle\_dss projects, please allow me below is just **one (1) single line with no linebreak**. Likewise, the "oracle\_dss:101:Oracle DSS:....." line below is just one (1) single line with no linebreak these lines up:

```
oracle_oltp:100:Oracle OLTP:oracle:: \
project.max-shm-memory=(privileged,51539607552,deny);\
process.max-sem-nsems=(privileged,300,deny)
oracle_dss:101:Oracle DSS:oracle:: \
project.max-shm-memory=(privileged,17179869184,deny)
```

This sets a limit of 48GB per shared memory segment and 300 semaphores for all processes in the oracle\_oltp project, and a 16GB shared memory segment limit for all processes in the oracle\_dss project.

3. The recommended method for modifying the /etc/project file is to use the "proj\*" commands, such as projadd(1) for creating a project and projmod(1) for modifying a project.

Examples of projadd and projmod  
 a.) # projadd -c "Oracle" 'user.oracle'  
 b.) # projmod -s -K "project.max-shm-memory=(privileged,6GB,deny)" 'user.oracle'

4. Resource Control assignments made in this way (in the /etc/project file) are permanent, and will survive a system re-boot. This is covered in much

5. There is also an "on-the-fly" way to temporarily set Resource Control assignments using the prctl(1) command. However, unlike the /etc/project file,

system re-boot. Again, this is covered in much more detail in the example section below.

6. Oracle support encourages the use of the "id -p" command. It shows the active and available projects for a user. If the limits (such as kernel paran for a particular user, the "id -p" will help you to see that the user is not using that project.

## Questions and Answers

Q1: Why can't we just set them the old way?

A1: Actually, you can. While it is not the preferred method, kernel parameter values specified in /etc/system are still honored, with some caveats:

- \* Values in /etc/system must be greater than the new defaults values of Solaris 10
- \* Any kernel parameters tunables that are obsolete in Solaris 10 are ignored.
- \* Values specified in /etc/system are global and affect all processes on the system
- \* If you use /etc/system to set IPC rctl values, you must reboot the system for them to take effect.

## Real World Examples - Permanently Setting Kernel Parameters

After upgrading a system from Solaris 9 to Solaris 10, it was decided that the new IPC resource control assignment should be used. The old /etc/system file contained values that were suitable for the Oracle installation on the system, but these values were too large for average users. The assumption is that all Oracle processes are run under the 'oracle' user.

The contents of the /etc/system file, as pertaining to IPC:

```
% /bin/egrep "semsys:|shmsys:|msgsys:" /etc/system
set semsys:seminfo_semmni=100 << see A.) below
set semsys:seminfo_semmns=1024
set semsys:seminfo_semmnl=256 << see B.) below
set semsys:seminfo_semvmx=32767
set shmsys:shminfo_shmmax=4294967295 << see C.) below
set shmsys:shminfo_shmmin=1
set shmsys:shminfo_shmmni=256 << see D.) below
set shmsys:shminfo_shmseg=10
```

Immediately, these lines can be ignored as the tunables are removed in Solaris 10:

```
set semsys:seminfo_semmns=1024
set semsys:seminfo_semvmx=32767
set shmsys:shminfo_shmmin=1
set shmsys:shminfo_shmseg=10
```

The remaining lines need to have resource controls set up for them. To do so, the first step is that a project must be created. Since all processes are run under the 'oracle' user, the first step is that a project must be created. Since all processes are run under the 'oracle' user, the first step is that a project must be created. Since all processes are run under the 'oracle' user, the first step is that a project must be created.

```
# projadd -c "Oracle" 'user.oracle'
```

A.) Once the project is created, we will assign resource controls corresponding to the remaining lines of the old /etc/system file that need to be "converted" to a resource control assignment as follows:

```
set semsys:seminfo_semmni=100
```

Since this "100" value is less than the new Solaris 10 default for project.max-sem-ids of "128", we could either artificially lower the value to 100 with a resource control assignment, or we could accept the new Solaris 10 default value by simply ignore the old /etc/system line. We chose to simply ignore it.

B.) The next line of the old /etc/system file that need to be "converted" to a resource control is:

```
set semsys:seminfo_semmnl=256
```

Again, this value is less than the new Solaris 10 default value for process.max-sem-nsems of 512. In this case however, we wish to artificially limit Oracle processes to 256. In this case however, we wish to artificially limit Oracle processes to 256. In this case however, we wish to artificially limit Oracle processes to 256.

```
# projmod -s -K "process.max-sem-nsems=(privileged,256,deny)" 'user.oracle'
```

C.) The next line of the old /etc/system file that need to be "converted" to a resource control is:

```
set shmsys:shminfo_shmmax=4294967295
```

This system has 8GB of memory. Therefore, this value (4GB, in bytes) is larger than new Solaris 10 OS default value for project.max-shm-memory. Since this value is larger than the new Solaris 10 OS default value for project.max-shm-memory, another resource control assignment must be created:

```
# projmod -s -K "project.max-shm-memory=(privileged,4GB,deny)" 'user.oracle'
```

D.) The final line of the old /etc/system file that need to be "converted" to a resource control is:

```
set shmsys:shminfo_shmmni=256
```

Again, this value is larger than the new default for project.max-shm-ids so another resource control assignment needs to be created:

```
# projmod -s -K "project.max-shm-ids=(privileged,256,deny)" 'user.oracle'
```

Since there are no more lines from /etc/system that pertain to IPC, we remove the old lines:

```
# /bin/cp /etc/system /etc/system.solaris9
# /bin/egrep -v "semsys:|shmsys:|msgsys:" /etc/system > /etc/system.solaris10
# /bin/mv /etc/system.solaris10 /etc/system
```

Now, before I show you our final /etc/project file, please allow me to emphasize that the "user.oracle:100:oracle:::process....." line below is just one (1) line in the final document that I break it up:

```
# cat /etc/project
system:0:::
user.root:1:::
noproject:2:::
default:3:::
group.staff:10:::

*****
NOTICE this is ONE LINE!
*****

user.oracle:100:oracle:::process.max-sem-nsems=(privileged,256,deny);project.max-shm-ids=(privileged,256,deny);project.max-shm-memory=(privile
NOTICE this is ONE LINE!
*****
```

---

### Real World Examples - Temporarily Setting Kernel Parameters

Resource controls can also be set "on the fly" using prctl(1). Unlike the /etc/project file, resource assignments made in this way will NOT survive a sys privileges.

The syntax of prctl(1) can, at first, seem complex. Some common usages are:

```
# prctl -i process <pid>
to list all resource controls for process <pid>

# prctl -i project <project>
to list all resource controls for project <project>

# prctl -n <rctl> -i process <pid>
lists only the resource control named <rctl> for process <pid>

# prctl -n <rctl> -r -v <value> -i process <pid>
replaces (-r) the named rctl setting with the value <value> for process <pid>
```

Unlike the /etc/project file, prctl allows the use of "scale factors" to simplify resource control management. Values specified with the -v switch can be "human readable" values such as 48GB instead of the 51539607552 bytes required in the project database.

For example, assuming the preceding /etc/project file we can check the values for the Shared Memory setting for the oracle\_dss project:

```
% prctl -n project.max-shm-memory -i project oracle_dss
project: 101: oracle_dss
NAME PRIVILEGE VALUE FLAG ACTION RECIPIENT
project.max-shm-memory
privileged 16.0GB - deny -
system 16.0EB max deny -
```

Should we need to temporarily increase the setting to 24GB:

```
% prctl -n project.max-shm-memory -r -v 24GB -i project oracle_dss
% prctl -n project.max-shm-memory -i project oracle_dss
project: 101: oracle_dss
NAME PRIVILEGE VALUE FLAG ACTION RECIPIENT
project.max-shm-memory
privileged 24.0GB - deny -
system 16.0EB max deny
```

### REFERENCES

The prctl(1) man page and System Administration Guide: Solaris Containers-Resource Management and Solaris Zones provide several useful examples as well.

---

Copyright © 2007 Oracle. All rights reserved. Oracle is a registered trademark of Oracle. Various product and service names referenced herein may be trademarks of Oracle or their respective owners.

**Disclaimer:** This document is provided for information purposes only and the contents hereof are subject to change without notice. Oracle does not warrant that this document conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. Oracle specific contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without the prior written permission of Oracle.

Help us improve our service. Please [email](#) us your comments for this document. .

---

[Bookmarks](#) [Admin](#) [Profile](#) [Feedback](#) [Sign Out](#) [Help](#)