

Lab Preparation

The following labs use virtual machines created with Virtual Box. If you don't have Virtual Box installed on your computer, then it can be downloaded at <https://www.virtualbox.org/wiki/Downloads>. To install Virtual Box, simply follow the steps and accept the default options.

The two virtual machines provided to you to perform these labs contain installations of Solaris 10. The machine named "sol10" has a full installation with a Graphical User Interface in a file named Sol10.zip. The machine named "solaris10" has a custom minimal installation with a Command Line Interface in a file named Solaris10.zip.

To add the virtual machines to the Virtual Box program:

1. Unzip the Sol10.zip and Solaris10.zip files. You will browse to the directory where this is done in steps 3 and 4 below.
2. Start the Virtual Box program.
3. Click the Machine Menu, Add, Browse to Sol10.vbox file, which looks like a blue cube, and click the Open button.
4. Click the Machine Menu, Add, Browse to SOL10CLI.vbox file, which looks blue cube, and click the Open button.

To use a virtual machine:

1. Start the Virtual Box program.
2. Select the virtual machine to use in the Virtual Box window.
3. Click the Start button on the Virtual Box toolbar.
4. After the system boots up to a login screen, you can either login with the "student" or "root" account with password "student10".
5. If you place your cursor inside of the "solaris10" command line interface system only, you must press the right CTRL key to release cursor from Virtual Box window.

Troubleshooting

A large benefit of using these virtual machines is that they can easily be reset to the original state. If you experience problems with the functioning of a virtual machine, then use the following steps to recover:

1. Click the close button for the virtual machine window.
2. Select the option to "Power off the machine".
3. Click the Ok button.
4. In the Virtual Box program window, right click on the virtual machine.
5. Select "Remove" from the context menu.
6. Click the "Delete all files" button.
7. Follow the steps above to "Add a virtual machine to the Virtual Box program" for the machine that you removed.

Lab #1 - Networking Essentials

Lab A – Complete the following steps:

1. Start both the sol10 and solaris10 virtual machines and login as root.

Start the Virtual Box program, and select each virtual machine and click start. After the login screen appears, login as the root user with the password "student10".

2. Use the ifconfig command to view all network interfaces in sol10.

Open a terminal window and type: `ifconfig -a`

3. Disable the Ethernet network interface in sol10.

`ifconfig e1000g0 down`

4. Enable the Ethernet network interface in sol10.

`ifconfig e1000g0 up`

5. Use the correct command to verify if solaris10 is "alive" from sol10.

`ping solaris10`

6. In solaris10, enable the spray service.

`inetadm -e svc:/network/rpc/spray`

7. From sol10, use the spray command to test connectivity to the solaris10 machine.

`spray solaris10`

8. In solaris10 and sol10, enable the rstat service.

`inetadm -e svc:/network/rpc/rstat`

9. From sol10, use the rup command to view hosts on the network.

`rup`

10. View the network statistics for the interfaces in sol10.

`netstat -i`

11. Start capturing packets in sol10 to a file named /tmp/capture.

`snoop -o /tmp/capture`

12. Use telnet to login to the student account from solaris10 to sol10.

`telnet sol10`
login: `student`
password: `student10`

13. Stop capturing packets in sol10.

Click into the terminal where the snoop command is running, and press CTRL+c.

14. View all the captured packets in sol10.

`snoop -i /tmp/capture`

15. View the first ten packets captured in sol10.

`snoop -i /tmp/capture -p 1,10`

Lab B – There is no solution provided for this lab, as the steps provided guide you through it. If you experience problems with the machine after the lab, then follow the Troubleshooting steps in the Lab Preparation section of the labs.

Lab #2 - Client-Server Model

Lab A – Complete the following steps:

1. View and notice that there are no uncommented entries in the /etc/inet/inetd.conf file.

```
cat /etc/inet/inetd.conf
```

2. Since network services are managed by the inetadm command in this version of Solaris or later, use the command to view the services being managed and their state.

```
inetadm
```

3. Enable the wall network service.

```
inetadm -e svc:/network/rpc/wall
```

4. Disable the wall network service.

```
inetadm -d svc:/network/rpc/wall
```

5. List the details the of the wall network service.

```
inetadm -l svc:/network/rpc/wall
```

6. Change the default value for all services for the max_copies parameter to 5.

```
inetadm -M max_copies=5
```

7. Change the value for the wall service for the max_copies parameter to 1.

```
inetadm -m svc:/network/rpc/wall max_copies=5
```

8. List the details the of the wall network service to verify the last two changes.

```
inetadm -l svc:/network/rpc/wall
```

9. View the ports that RPC services are using.

```
rpcinfo -p
```

Lab #3 - System Management Console

Part A - Configure the CLI system for remote management with SMC

1. Configure the Solaris10 system from the command line to have the SMC Server listen for remote connections.

```
svccfg -s wbem setprop options/tcp_listen=true
```

2. Start the SMC Server and enable it to start automatically on the Solaris 10 system.

```
svcadm enable wbem
```

3. Verify that the SMC Server is listening on all interfaces on the Solaris 10 system.

```
netstat -a | grep 898
```

Part B - Configure the GUI system to remotely manage the CLI system with SMC

1. Create a backup of the toolboxes configured on the Sol10 system into a file /toolboxes.tar.

```
find /var/sadm/smc -name "*.tbx" | xargs tar -cvf toolboxes.tar
```

2. Add a toolbox to the Solaris Management Console on the Sol10 (GUI) system to manage the Solaris10 (CLI) system. Open the new toolbox when finished.

```
/usr/openwin/bin/xhost +si:localuser:root
```

```
su
```

```
smc edit
```

Select the Management Tools on the left side of the window.

From the Action menu, choose Add Toolbox URL.

Select Server Toolbox and click the Next button.

Type in the name of the server: solaris10.domain.test.

Click the Next button.

Select the first Toolbox, which should have an URL of `http://solaris10:898/toolboxes/this_computer.tbx`.

Click the Next button.

For the Toolbox Name and Description, select to Use the Toolbox defaults.

Click the Next button.

For the Toolbox Icons, select to Use the Toolbox defaults.

Click the Next button.

For the Management Scope, select Override.

Choose the file scope and set the server name to `solaris10.domain.test`.

Click the Finish button.

Save the changes to the toolbox by clicking Console, Save As..., and save the toolbox as `/var/sadm/smc/toolboxes/smc/smc.tbx`.

Exit the SMC by clicking the Console menu, and clicking exit.

Open the SMC and view the new toolbox by executing: `smc`

Lab #4 - Swap Management

Perform on the following on the system of your choice.

1. Display virtual memory statistics at 3 second intervals ten times.

```
vmstat 3 10
```

2. Show the swap space usage summary.

```
swap -s
```

3. Show the details of swap space usage.

```
swap -l
```

4. Create a swap file /var/extraswap that is about 50 Mb in size.

```
mkfile 50m /var/extraswap
```

5. Add the swap file for temporary use.

```
swap -a /var/extraswap
```

6. Verify the file has been added by viewing the details of swap space usage.

```
swap -l
```

7. Stop using the /var/extraswap file for swap.

```
swap -d /var/extraswap
```

8. Verify the file is no longer in use by viewing the details of swap space usage.

swap -l

9. Add an entry to the /etc/vfstab to make use of the /var/extra/swap file for swap persistent.

Add to the /etc/vfstab:



/opt/swapfile - - swap - no -

10. Verify that the entry is correct by rebooting the system, logging in, and listing the details of swap usage.

reboot

Login as "root" with the password "student10"

swap -l

Lab #5

Lab A – Complete the following steps on the Sol10 system:

1. Use the dumpadm command without arguments to view the current dump configuration.

`dumpadm`

2. Fill in the configuration parameters from the output:
Dump content: `kernel pages`

Dump device: `/dev/dsk/c0t0d0s1`

The savecore directory: `/var/crash`

Is savecore enabled? `yes`

Is save compression enabled: `on`

3. Use the dumpadm command to disable compression.

`dumpadm -z off`

4. Use the dumpadm command to change the dump device to `/dev/dsk/c0t2d0s5`.

`dumpadm -d /dev/dsk/c0t2d0s5`

5. Run the sync command to flush all previously unwritten system buffers out to disk, ensuring that all file modifications up to that point will be saved.

`sync`

6. Force the kernel to panic and save a crash dump by using the savecore -L command

`savecore -L`

7. When the command completes, make sure the crash dump succeeded by using the file command on the files of the savecore directory.

`file /var/crash/*`

Lab B Complete the following steps:

1. Use the coreadm command to the display default initial configuration.

`coreadm`

2. Execute the commands to enable global core dumps and per-process core dumps even if they are currently enabled.

`coreadm -e global`

`coreadm -e process`

3. Turn on logging to generate a message when a global core file is attempted.

`coreadm -e log`

4. Display the configuration information to verify the changes.

`coreadm`

5. In another terminal window, create a new directory named /dir, and change to that directory.

`File, Open Terminal, Default (SHIFT+CTRL+N)`

`mkdir /dir`

`cd /dir`

6. Run the ps command to get the PID of the new shell, and send a SIGFPE signal (Signal 8) to the new shell by using the kill command. (The SIGFPE signal forces a core file.) Note – The kill -8 command terminates the shell.

ps # note the PID of the bash shell

kill -8 PID # PID is the value noted in the previous step

7. In the original terminal window, check to see if a core file exists in the /dir directory of the old shell. Use the file command to verify that the core file is from the old shell.

file /dir/*

8. Use the ls command to check for a core file in the /var/core directory.

ls /var/core

9. Observe the messages generated in the /var/adm/messages file due to coreadm logging being enabled.

tail /var/adm/messages

Lab #6

Lab A – On the NFS Server, Sol10, complete the following steps:

1. Edit the /etc/dfs/dfstab file. Add an entry to share the directory that holds man pages to the solaris10 system read-only.

```
share -F nfs -o ro=solaris10 -d 'man pages' /usr/share/man
```

2. Stop and start the NFS server daemons.

```
svcadm disable network/nfs/server
```

```
svcadm enable network/nfs/server
```

3. Verify that the /usr/share/man directory is shared and that no NFS mounts are present.

```
dfshares
```

Lab B –On the NFS Client, complete the following steps:

1. Rename the /usr/share/man directory so that you can no longer access the man pages on the client system. Verify that the man pages are not available.

```
mv /usr/share/man /usr/share/man.orig
```

```
man python
```

What message does the man command report?

```
No manual entry for python.
```

2. Create a new man directory (/usr/share/man) to use as a mount point.

```
mkdir /usr/share/man
```

3. Mount the /usr/share/man directory from the server.

`mount sol10:/usr/share/man /usr/share/man`

4. Verify that the man pages are available.

`man python`

Are the man pages available?

Yes

5. Verify and record the default ro | rw options used for this mount.

`mount | grep man`

6. Attempt to write a file into the NFS-mounted file system.

`touch /usr/share/man/test`

What is the result of trying to write to the NFS-mounted file system?

`touch: cannot create /usr/share/man/test: Read-only file system`

7. Verify the list of mounts that the server provides.

`dfshares sol10`

8. Unmount the /usr/share/man directory, and verify the list of remote mounts from the server.

`umount /usr/share/man` # Executed on solaris10

`dfmounts` # Executed on sol10

Lab E – On the NFS Server, sol10, return the system back to its original state:

Edit the /etc/dfs/dfstab:

Remove any lines that start with "share".

Save the changes to the file.

Execute on sol10:

`svcadm disable network/nfs/server`

Lab #7

Lab A – On the Server Host, sol10, complete the following steps:

1. Edit the /etc/dfs/dfstab file, and add a line to share the man pages.

```
share -F nfs -o ro=solaris10 -d 'man pages' /usr/share/man
```

```
svcadm enable network/nfs/server
```

Lab B – On the Client Host, solaris10, complete the following steps:

1. Rename the /usr/share/man directory so that you cannot view the man pages installed on the client system.

```
mv /usr/share/man /usr/share/man.orig
```

2. Edit the /etc/auto_master file to add an entry for a direct map.

Add to /etc/auto_master:

```
/- auto_direct -ro
```

3. Use the vi editor to create a new file called /etc/auto_direct, and add an entry to the file to share the man pages.

Add to /etc/auto_direct:

```
/usr/share/man -ro solaris10:/usr/share/man
```

4. Run the automount command to update the list of directories managed by the automountd daemon.

```
automount
```

5. Test the configuration, and verify that a mount for the /usr/share/man directory exists after accessing the man pages.

```
ls /usr/share/man
```

`mount | grep man`

Lab C – On the Server Host, sol10, complete the following steps:

1. Verify that the /export/home directory exists. If it does not exist, create it.

`ls -ld /export/home`

2. Add a user account with the following characteristics:

User ID: 3001
Primary group: 10
Home directory: /export/home/usera
Login shell: /usr/bin/bash
User name: usera

`useradd -u 3001 -g 10 -d /export/home/usera \`
`-s /usr/bin/bash -m usera`

3. Remove the lock string from the new user's /etc/shadow file entry.

`passwd -d usera`

Lab D – On the Client Host, complete the following steps:

1. Verify that the /export/home directory exists. If it does not exist, create it.

`ls -ld /export/home`

2. Add a user account with the following characteristics:

User ID: 3001
Primary group: 10
Home directory: /export/home/usera
Login shell: /usr/bin/bash

User name: usera

```
useradd -u 3001 -g 10 -d /export/home/usera \  
-s /usr/bin/bash -m usera
```

3. Remove the lock string from the new user's /etc/shadow file entry.

```
passwd -d usera
```

Lab E – On Both Systems, complete the following steps:

1. Edit the /etc/passwd file, and change the home directory for the new user from the /export/home/usera to /home/usera.

Modify /etc/passwd so the line starting with "usera" reads:

```
usera:x:3001:10::/home/usera:/usr/bin/bash
```

2. Edit the /etc/auto_home file. Add the following line, and replace username with the name of your new user:

```
usera sol10:/export/home/usera
```

Lab F – On the Server Host, complete the following steps:

1. Edit the /etc/dfs/dfstab file, and add a line to share the /export/home directory.

```
share -F nfs -o rw=solaris10 -d 'usera home' \  
/export/home/usera
```

2. Use the svcadm command to restart the nfs server.

```
svcadm restart nfs/server
```

Lab G – On Both Systems, complete the following step:

Log in as the new user.

Do both systems automatically mount the new user's home directory?

Yes.

Which directory is mounted, and what is the mount point:
On the server?

/home/user is mounted on /export/home/usera.

On the client?

/home/user is mounted on sol10:/export/home/usera.

Lab H – On the Client Host, complete the following steps:

1. On the client, log off as usera.

exit

2. Remove the entry for usera from the /etc/auto_home map.

Login as root

vi /etc/auto_home

Gdd:wq

3. Remove the entry for the auto_home map from the
/etc/auto_master map.

vi /etc/auto_home

Gdd:wq

4. Reboot the client.

reboot

5. Remove the /usr/share/man directory.

```
rm -rf /usr/share/man
```

6. Rename the /usr/share/man.orig directory to /usr/share/man.

```
mv /usr/share/man.orig /usr/share/man
```

Lab H – On the Server Host, complete the following steps:

1. Log off as usera.

Click the Launch menu and select Log off usera.

2. After the client reboots as described in Step 4 of “Task 8 – On the Client Host” section on page 7-24, remove the entry for usera from the /etc/auto_home map.

Login as root

vi /etc/auto_home

Gdd:wq

3. Remove the entries from /etc/dfs/dfstab file.

Login as root

vi /etc/auto_home

Gdd:wq

4. Unshare mounted directories.

svcadm disable svc:/network/nfs/server

Lab #10

Complete the following steps:

1. Log in as the root user, and open a terminal window.

Login as root.

Right-click desktop and click Open Terminal.

2. If user10 does not exist on your system, create it with a user ID of 1010, a group ID of 10, a Bash login shell, and a home directory of /export/home/user10.

```
useradd -u 1010 -g 10 -s /usr/bin/bash -d /export/home/user10 \
-m user10
```

3. Create the directory named /var/test, and change directories to that location.

```
mkdir /var/test
```

```
cd /var/test
```

4. Create two new files named file1 and file2. Record the permissions applied to each.

```
touch file1 file2
```

```
ls -l
```

```
total 0
```

```
-rw-r--r--  1 root   root      0 Jun 16 14:07 file1
```

```
-rw-r--r--  1 root   root      0 Jun 16 14:07 file2
```

5. Use the echo command to create file1, and have the file contain the text string "Success for life."

```
echo 'Success for life' > file1
```

```
ls -l file1
```

```
-rw-r--r--  1 root   root    17 Jun 16 14:09 file1
```

6. Display the ACL for the file named file1.

getfacl file1

```
# file: file1
# owner: root
# group: root
user::rw-
group::r--      #effective:r--
mask:r--
other:r--
```

Do the permissions in the ACL match the permissions indicated by the ls command?

Yes.

7. Change permissions on the file named file1 so that only the owner (root) and group (other) have read access.

chgrp other file1

ls -l file1

```
-rw-r--r--  1 root   other    17 Jun 16 14:09 file1
```

8. Change your user identity to user10.

su - user10

9. Display the contents of the file named file1.

cat /var/test/file1

What is the output?

Success for life

10. What happens if you try to append file1 with the text: "Added by user10"?

```
echo 'Added by user10' >> /var/test/file1  
-bash: /var/test/file1: Permission denied
```

11. Exit your su session.

```
exit
```

12. Use the setfacl command to add an ACL entry that allows read and write access for user10 to the ACL for the file named file1.

```
setfacl -m u:1010:6 file1
```

13. Verify that the new ACL entry exists.

```
getfacl file1
```

```
# file: file1  
# owner: root  
# group: other  
user::rw-  
user:user10:rw-    #effective:r--  
group::r--    #effective:r--  
mask:r--  
other:r--
```

14. Does the user10 permissions acl match the permissions associated with the mask entry?

Yes. Effectively, the permissions for user10 is still read only.

15. Set the mask permissions for file1 to read and write.

```
setfacl -m m:6 file1
```

16. Display the ACL and a long listing for file1.

```
getfacl file1
```

```
ls -l file1
```

Do the mask permissions match the user10 permissions?

Yes. Now user10's permissions match the mask permissions.

In the long listing output, do you find an indication that file1 has additional ACL entries?

The + at the end of the normal permissions.

17. Change your user identity to user10.

su - user10

18. Append file1 with the text: "Added by user10"?

echo 'Added by user10' >> /var/test/file1

19. Display the contents of the file named file1.

cat /var/test/file1

What is the output?

Success for life

Added by user10

20. Exit your su session.

exit

21. Set the normal permissions on the file to 600.

chmod 600 file1

Did changing the permissions for the file affect the mask permissions?

getfacl file1

```
# file: file1
# owner: root
# group: other
user::rw-
user:user10:rw-      #effective:---
group::---           #effective:---
mask:---
other:---
Yes.
```

How did this affect the acl for user10?

It made the permissions effectively "---" or none.

Lab #11

Lab A– Complete the following steps:

1. Create a user named "admin1" with a uid of 1111. Assign the password "11shutdown" and the primary group "other" to the user.

`/usr/openwin/bin/xhost +si:localuser:root`

`su`

`smc`

Click "This Computer (127.0.0.1)"

Double-click "System Configuration"

Double-click "Users"

Click the Action menu and select Add User

Type "admin1" for the user name

Click the Next button

Set the User ID Number to 1111.

Click the Next button

Set the password that user will use at first login to "10student" and confirm it.

Click the Next button

Select the Primary Group "other" from the list.

Click the Next button

Leave the Path /export/home

Click the Next button

Acknowledge the mail server

Click the Next button

Review the user's information

Click the Finish button

2. Create a role named uadmin. Give it a user ID of 5000 and a full name of Shutdown.

`/usr/openwin/bin/xhost +si:localuser:root`

`su`

`smc`

Click "This Computer (127.0.0.1)"

Double-click "System Configuration"

Double-click "Users"

Login with the User Name "root" and the Password "student10"

Double-click "Administrative Roles"

Select the Action Menu, Add Administrative Role

Enter "uadmin" for the Role Name

Enter "User Administrator" for the Full Name

Set the Role ID to 5000

Uncheck Create a role mailing list

Click the Next button

Type and confirm the role password of "10student "

Click the Next button

Select the "User Management"

Click the Add button and "User Management" should appear in the Granted Rights list

Select the "User Security"

Click the Add button and "User Security" should appear in the Granted Rights list

Click the Next button

Use a home directory path of /export/home

Click the Next button

Add the "admin1" user, and click the Next button

Click the Finish button

Close the SMC

3. Verify that the role is included in the /etc/user_attr file and assigned to the shut1 user.

```
grep uadmin /etc/user_attr
```

```
uadmin::::profiles=User Management,User Security;type=role
```

```
admin1::::roles=uadmin;type=normal
```

4. Create another user named "admin2" with a uid of 1112. Assign the password "10student" and the primary group "other" to the user.

```
/usr/openwin/bin/xhost +si:localuser:root
```

```
su
```

```
smc
```

Click "This Computer (127.0.0.1)"

Double-click "System Configuration"

Double-click "Users"

Click the Action menu and select Add User

Type "admin2" for the user name

Click the Next button

Set the User ID Number to 1112.

Click the Next button

Set the password that user will use at first login to "10student" and confirm it.

Click the Next button

Select the Primary Group "other" from the list.

Click the Next button

Leave the Path /export/home

Click the Next button

Acknowledge the mail server

Click the Next button

Review the user's information

Click the Finish button

5. Assign the "uadmin" role to the "admin2" user.

/usr/openwin/bin/xhost +si:localuser:root

su

smc

Click "This Computer (127.0.0.1)"

Double-click "System Configuration"

Double-click "Users"

Double-click "admin2"

Click on the Roles tab

Select the "uadmin" role on the left

Click the Add button

Click the OK button

9. To test the configuration, log in as "admin1" and open a terminal.

Click the Launch button, and Log out Student

Type the username "admin1"

Click the OK button

Type the password "10student"

Click the OK button to login.

Click the OK for the default desktop.

Click the Launch button, Applications, Utilities, and Terminal

10. From this login, switch to the "uadmin" role.

su uadmin

Enter the password "10student"

11. Create a user named "admin3" with a password of "10student"

`pfexec /usr/sbin/useradd admin3`

`pfexec passwd admin3`

Enter the "10student" password twice

10. Execute the profiles command to determine which RBAC profiles are associated with "admin1".

`su`

Enter the password "student10"

`profiles admin1`

`exit`

11. Execute the roles command to determine which RBAC roles are associated with "admin1".

`su`

Enter the password "student10"

`roles admin1`

`exit`

12. Execute the auths command to determine which RBAC authorizations are associated with "admin1".

`su`

Enter the password "student10"

```
auths admin1 | tr "," "\n"
```

```
exit
```