

**Disk operations**

**/dev/dsk/cXtXdXsX** – block device  
**/dev/rdisk/cXtXdXsX** – raw/character device

**format [-e]** – handles label/vtoc partitions in Solaris.

**-e** – enables expert menu (like creating EFI/SMI labels)

**format -e → choose disk → label → choose SMI[0] or EFI[1] → yes** - this sequence will label a disk to SMI or EFI label.

**format -e → choose disk → p → p →** - this sequence will list partition table of the chosen disk.

**format -e → choose disk → p → 6 → tag → permissions → enter starting cylinder → enter size, label → yes** – this sequence will create partition 6 with the size you have chosen.

**rmformat** – command to show remote magnetic devices like USB disks and CD/DVD devices.

**fdisk** – handles MBR partitions.

**prtvtoc /dev/rdisk/cXtXdXsX** – prints partition virtual table of contents (vtoc).  
**fmthard -s <vtoc-file>**

**/dev/rdisk/cXtXdXsX** – enables writing a partition table/vtoc represented by the file - <vtoc-file> to a new hard disk.

**stmsboot -e** – enables Solaris I/O multi pathing.

**stmsboot -d** – disables Solaris I/O multi pathing.

**Device operations**

**devfsadm -v** – refreshes the /dev directory to accept changes after device changes.

**volcheck** – refreshes the /dev/vol directory to accept changes after USB/CD/DVD device changes.

**Network device operations**

**dladm show-phys** – shows the available physical interfaces in the system.

**dladm show-link** – shows all available network elements (physical and logical).

**NetWork AutoMagic (NWAM) operations**

**netadm** – a command to administer NWAM daemon. (only when NWAM is activated).

**netcfg** – a command to configure profiles and locations using NWAM daemon. (only when NWAM is activated).

**netadm list** – shows current network profile configuration.

**netadm enable -p ncp automatic** – enables NWAM daemon.

**netadm enable -p ncp defaultfixed** – disables NWAM daemon and enables fixed IP configuration.

**dlstat net0 -i 1** – shows network statistics on interface named net0.

**NIC operations**

**ipadm create-ip <if-dev>** – creates an ip interface that can handle ipv4 or ipv6.

**ipadm show-if** – shows configured interfaces and the status of the link.

**ipadm show-ifprop <if-name>** – shows configured interface properties. (similar to ndd)

**DHCP operations**

**ipadm create-addr -T dhcp <if-dev>/v4dhcp** – creates an ipv4 dhcp client configuration over the ip device <if-dev>.

**ipadm delete-addr <if-dev>/v4dhcp** – deletes an ipv4 dynamic configuration.

**STATIC operations**

**ipadm create-addr -T static -a 10.0.0.1/24 <if-dev>/v4static1** – creates an ipv4 static configuration over the ip device <if-dev>.

v4static1 can be replaced with v4staticN where N is a positive number representing a different static configuration.

**ipadm delete-addr <if-dev>/v4static1** – deletes an ipv4 static configuration.

**NIC operations**

**ipadm up-addr <if-dev>/v4dhcp** – turns on the ip address named <if-dev>/v4dhcp.

**ipadm down-addr <if-dev>/v4static1** – turns down the ip address named <if-dev>/v4static1.

**ipadm enable-addr -t <if-dev>/v4dhcp** – enables ip address named <if-dev>/v4dhcp.

**ipadm disable-addr -t <if-dev>/v4static1** – disables ip address named <if-dev>/v4static1.

**Filesystem operations**

**mount /dev/dsk/cXtXdXsX /dir1** – mount the block device /dev/dsk/cXtXdXsX to the directory /dir1 .

**umount /dir1** – unmounts /dir1 from the system.

**Package operations**

Package IFO flags–  
 Installed/Frozen/Obsolete(or renamed)

**/usr/lib/pm-launch packagemanager** – launches the package manager GUI.

**pkg publisher** – lists the current package publishers.

**pkg list** – lists all installed packages in the system.

**pkg list -a** – lists all the packages that are found locally and remotely (using all your publishers).

**pkg info entire** – Shows the current SRU(Software Release Update) version.

**pkg info <pkg-fmri>** – lists the information about a package named - <pkg-fmri>.

**pkg contents <pkg-fmri>** – lists the file content of a package named <pkg-fmri>.

**pkg install [-nv] <pkg-fmri>** – installs the package named <pkg-fmri> along with all it's dependencies.

**-n** – perform a dry run w/o actually installing.

**-v** – perform verbose operation.

**pkg verify <pkg-fmri>** – verifies a package named <pkg-fmri> installation.

**pkg search [-l] <pkg-fmri>** – search for a package named <pkg-fmri>, the search matches current installed or newer versions.

**pkg update [-n] <pkg-fmri>** – updates the package named <pkg-fmri>.

**-n** – perform a dry run w/o actually installing.

**-v** – perform verbose operation.

**pkg uninstall <pkg-name>** – uninstall the package named <pkg-name> from the system.

**Boot Environment operations**

A boot environment is a way to snapshot your local file-system while enabling boot to the specific boot-environment

N- Active now.

R- Active on next reboot.

**beadm list** – list all available boot environments.

**beadm create <BE-name>** – create a new boot environment named <BE-name>.

**beadm rename <old-BE-name> <new-BE-name>** – renames and existing boot environment named <old-BE-name> to be named <new-BE-name>.

**beadm destroy <BE-name>** – destroys a boot environment named <BE-name>.

**beadm activate <BE-name>** – activates an existing boot environment named <BE-name> to be active in next reboot (requires - init 6).

On X86 grub shows all available BE's. On SPARC issue: **boot -L** to show all BE's.

**Services operations**

**svcs [-a]** – list all enabled service  
**-a** – list all services (also the disabled ones).

**svcadm enable <FMRI>** – enable the service named <FMRI>.

**svcadm disable <FMRI>** – disable the service named <FMRI>.

**netservices [limited | open] – limited** – will secures the Solaris system to enable only services that are secured by default.

**Open** – will open all services like telnet/ftp which are not secured by default.

**inetconv** – checks /etc/inetd.conf file and converts inetd services to SMF services.

**User/group operations**

**useradd -d /export/home/<login> -u <UID> -g <GID> <login>** – creates the user named <login> with user id <UID> and primary group id <GID>.

**userdel [-r]<login>** – deletes the user named <login> .

**-r** – also deletes user's home directory.

**groupadd -g <GID> <group-name>** – creates a group-name named <group-name> with specific gid value of <GID>.

**groupdel <group-name>** – deletes the groupname named <group-name>.

**Swap operations**

Virtual memory area and is combined from the RAM and a dedicated disk storage known as swap space.

**swap -s** – lists summary of system's virtual space.

**swap -l** – lists details of system's physical swap areas.

**swap -a <dev | file>** – Add device or swap file to be a swap device.

**swap -d <dev | file>** – deletes device or swap file from being a swap device.

**zfs create -V <size> rpool/swap2 –**

Please mail [info@grigale.com](mailto:info@grigale.com) if you find any errors or typos.

Copyright © 2012 Grigale Ltd. This document is licensed under the terms of [Creative Commons Attribution 3.0 License](https://creativecommons.org/licenses/by/3.0/)

Creates a swap volume named swap2 using zfs.  
**swap -a /dev/zvol/dsk/rpool/swap2** – Adds the device /dev/zvol/dsk/rpool/swap2 as a swap device.

### NFS operations

NFS – network file-system.

#### NFS Server side

/etc/dfs/dfstab – configuration file for configuring shares (only needed for ufs shares).

**dfshares** – lists summary of local nfs shares on the system.

**dfmounts** – lists summary of nfs clients connected to local nfs server.

**share** – lists summary of local nfs shares on the system.

**share /export/home/fs1** – creates a temporary share on the directory /export/home/fs1.

#### NFS client side

/etc/vfstab – configuration file for using nfs shares as client.

**mount server:/export/home/fs1 /local/fs1** – mounts the remote nfs share named server:/export/home/fs1 to a local directory named /local/fs1.

### Auto File-system operations

/etc/auto\_master – contains all direct and indirect maps.

Direct map – map of an absolute directory path to NFS share.

Indirect map - map of an relative directory path to NFS share.

auto\_master direct map configuration -  
 /- auto\_direct

auto\_master indirect map configuration -  
 /home auto\_home

Direct mapping -

/etc/auto\_direct:  
 /opt/SUNWspro server:/opt/SUNWspro

Indirect mapping -

/etc/auto\_home:  
 haim server:/export/home/haim

\* server:/export/home/&

Auto file-system mounting and unmounting service name: **autofs**

Enabling the service:  
**svcadm enable autofs**

Disabling the service:  
**svcadm disable autofs**

Refreshing the service after creating a new mapping:  
**svcadm refresh autofs**

Temporary automatic umount duration configuration:

**automount -t <new\_duration>**

Permanent automatic umount duration (in seconds) configuration:

/etc/default/autofs

change the line  
 #AUTOMOUNT\_TIMEOUT=600

to be:  
 AUTOMOUNT\_TIMEOUT=200 - this will change the duration time to be 200 seconds.

### RBAC

**right** – ability to perform a command with special permissions.

**Authorization** – ability to perform a task that is not backed up by a special command.

**Profile** – set of rights and authorizations.

**Role** – special user that can not logged in.

Handling roles -

**roleadd <rolename>**

**roledel [-d] <rolename>**

Handling profiles -

/etc/security/prof\_attr – profiles DB file.

**profiles** – shows which profiles are assigned to the current logged in user.

Adding profile to a user or role -  
**roleadd/rolemod -P**

**“profile1,profile2” <rolename>**

**useradd/usermod -P**

**“profile1,profile2” <rolename>**

Deleting profiles -

**roleadd/rolemod -P “” <rolename>**

**useradd/usermod -P “” <rolename>**

Running commands using RBAC -

**pfexec <cmd>**

use - **/usr/bin/pfsh, /usr/bin/pfsh**

Allowing a user to swite to role -

**useradd/usermod -R <rolename>**

**<loginname>**

Switching from user to a role -

**su - <rolename>**

### Zones operations

Configuring zone -

**zonecfg -z <zone-name> “create;set autoboot=<true|false>;set**

**zonepath= /full/zone/path;add net;set**

**physical=e1000g0;set**

**address=10.0.0.10;end;commit”**

Installing zone -

**zoneadm -z <zone-name> install**

Zone boot/halt -

**zoneadm -z <zone-name> <boot|halt>**

First time login:

**zlogin -C <zone-name>**

Next time logins:

**zlogin <zone-name> (requires root privileges)**

Cloning zone -

**zoneadm -z <zone-name> clone**

**<original-zone>**

Detaching zones -

**zoneadm -z <zone-name> halt**

**zoneadm -z <zone-name> detach**

Attaching zones -

We need to configure the zone.

### zoneadm -z <zone-name> attach

### Virtual Switch device operations

**dladm show-etherstub** – shows the available virtual switches in the system.

**dladm create-etherstub sw1** – creates a virtual switch named sw1 in the system.

**dladm delete-etherstub sw1** – deletes a virtual switch named sw1 in the system.

### VMNIC device operations

**dladm show-vmnic** – shows the available physical interfaces in the system.

**dladm create-vmnic -I <net-name>**

**<vmnic-name>** – creates a vmnic named

**<vmnic-name>** binded to a physical

interface or switch named **<net-name>**.

(e.g net-name= sw1|net0).

**dladm delete-vmnic <vmnic-name>** –

deletes a vmnic named **<vmnic-name>**.

### flow operations

**flowadm show-flow** – shows the configured flows in the system.

**flowadm show-flowprop** – shows the configured flows properties in the system.

**flowadm add-flow -I <net-name> -a**

**transport=[UDP|**

**TCP],local\_port=2049 -p**

**maxbw=100M nfs1** – adds a flow

named nfs1 which sets the bandwidth of nfs tcp to be 100Mbps.

**flowadm set-flowprop maxbw=10M**

**nfs1** – sets flow bandwidth of an

existing flow named **nfs1** to be 10Mbps.

### Aggregation

**dladm show-aggr** – show existing

aggregations

**dladm create-aggr -d bge0 -d bge1 1** –

creates a new aggregation with key=1.

**dladm add-aggr -d bge2 1** – adds a

new interface to existing aggregation

with key=1.

**dladm remove-aggr -d bge0 1** –

removes bge0 from participating in the

aggregation with key=1.

**dladm delete-aggr 1** – deletes a whole aggregation with key=1.

### Vlan tagging

**dladm createvlan -I bge2 -v 667 bge2v30** – creates vlan tagged interface named bge2v30 that uses bge2 with properties of vlan ID=30.

**dladm createvlan -I aggr1 -v 667 aggr1v30** – creates vlan tagged interface named aggr1v30 that uses aggr1 with properties of vlan ID=30.

### Aggregation + Vlan tagging

**dladm createvlan -I aggr1 -v 667 bge2v30** – creates vlan tagged interface named aggr1v30 that uses aggr1 with properties of vlan ID=30.

### Routing

**netstat -rn** – shows current routing table

**route [-p] add <network> <gateway>**

– creates a temporary route. -p – for creating a permanent route.

**route [-p] delete <network>**

**<gateway>** – deletes temporarily a route. -p – for deleting permanently.

### Enabling dynamic routing

**svcadm enable route:default** – enables dynamic routing using RIPv1/RIPv2 and RDISC protocols.

**svcadm disable route:default** – disables dynamic routing using RIPv1/RIPv2 and RDISC protocols.

### Default router

**route [-p] add default <gateway>** – adds a new default gateway. -p – for permanent use.

**Route [-p] change default <gateway>** – changes the default gateway. -p – for permanent use.

**Route [-p] change default <gateway>** – changes the default gateway. -p – for permanent use.

**Route [-p] change default <gateway>** – changes the default gateway. -p – for permanent use.

**Route [-p] change default <gateway>** – changes the default gateway. -p – for permanent use.

**Route [-p] change default <gateway>** – changes the default gateway. -p – for permanent use.

**Route [-p] change default <gateway>** – changes the default gateway. -p – for permanent use.

**Route [-p] change default <gateway>** – changes the default gateway. -p – for permanent use.

**Route [-p] change default <gateway>** – changes the default gateway. -p – for permanent use.