

# Tru64 UNIX Best Practice

## Moving a Cluster to Another IP Subnet

**July 2004**

**Product Version:**                   **TruCluster Server Version 5.0A and higher**

This Best Practice describes how to move a TruCluster™ Server Version 5.0A or higher cluster from one IP subnet to another IP subnet.



---

# Contents

## Moving a Cluster to Another IP Subnet

Is This Best Practice Right for You? .....	2
Before You Begin .....	3
Applying the Best Practice .....	4
Procedure .....	4
File Edit Tables .....	8
Changing External IP Addresses Only .....	8
Changing External IP Addresses and Host Names ....	9
Changing External and Internal IP Addresses and Host Names .....	10
Attribute and Checklist Tables .....	11
External Host Names and IP Addresses .....	12
Cluster Name and Cluster Aliases .....	12
Interface IP Aliases .....	13
External Servers .....	13
Checklist .....	14
Shell Scripts .....	16
getinfo Script .....	16
savefiles Script .....	18
filediffs Script .....	20
copyfiles Script .....	21
Verifying Success .....	22
Troubleshooting .....	23
Example .....	23
Example: External Host Names and IP Addresses .....	25
Example: Cluster Name and Cluster Aliases .....	25
Example: Interface IP Aliases .....	26
Example: External Servers .....	26
Example: getinfo Output .....	26
Example: Checklist .....	29
Example: filediffs Output .....	30
Comments and Questions .....	34
Legal Notice .....	34



---

## Moving a Cluster to Another IP Subnet

This Best Practice describes how to move a cluster from one IP subnet to another IP subnet. This is usually required only when a site reconfigures its network topology such that the cluster's external IP addresses will be on a different IP subnet.

Other Best Practices and the *Cluster Administration* manual describe how to change a cluster name or IP address, how to change a member host name or IP address, and how to change a cluster interconnect interface name or IP address. This Best Practice addresses the changes needed when modifying IP addresses and host names for the entire cluster.

---

### Note

---

Although a clusterwide reboot is mandatory only when changing the name of the cluster, this Best Practice halts and reboots the entire cluster during the move procedure because of the amount of changes made to system files and the move of the cluster to another IP subnet. If your site cannot tolerate any downtime and if you are not changing the name of the cluster, you may want to look at the other Best Practices and modify whichever approach best fit your needs.

---

In order of increasing complexity, moving a cluster to another IP subnet might involve changing the following items:

- IP addresses associated with the cluster's external network interfaces, interface IP aliases, and cluster aliases.
- IP addresses and host names associated with the cluster's external network interfaces, interface IP aliases, and cluster aliases.
- IP addresses and host names associated with the cluster's external network interfaces, interface IP aliases, and cluster aliases, and the IP addresses and interface names associated with the internal cluster interconnect.

---

**Note**

---

In order to simplify cluster administration, if you change a member's external host name, you might also want to change the host portion of that member's cluster interconnect name. However, because the network used by the cluster interconnect is private to the cluster, you can relocate the cluster to another network without modifying the IP names and addresses used by the cluster interconnect. The information on how to do this is provided for completeness. Whether or not you modify the host-name portion of cluster interconnect names to match member host names is entirely up to you.

The example shows an actual move that involves changing external IP names and addresses and internal IP names.

---

This Best Practice provides tables for you to use when gathering information and performing the move:

- *Applying the Best Practice* contains tables that describe the file edits required for each of the three move scenarios. Use the table that corresponds to the type of move you plan to make.
- *Attribute and Checklist Tables* help you gather information before starting the move, and a checklist you can use to keep track of your edits.

See the Tru64 UNIX Best Practices Web page for more information about Best Practices documentation.

## Is This Best Practice Right for You?

Not all Best Practices apply to all configurations. To use this Best Practice, your system or cluster must meet the requirements described in the following table:

Requirement	Description
Operating System	Compaq Tru64™ UNIX Version 5.0A or higher
Product Version	TruCluster™ Server Version 5.0A or higher

Requirement	Description
Impact on Availability	Requires a clusterwide halt and reboot. Although a clusterwide reboot is mandatory only when changing the name of the cluster, this Best Practice halts and reboots the entire cluster during the move procedure because of the amount of changes made to system files and the move of the cluster to another IP subnet. If your site cannot tolerate any downtime and if you are not changing the name of the cluster, you may want to look at the other Best Practices and modify whichever approach best fit your needs.
Skill Level	Because this Best Practice requires the manual editing of system configuration files, we recommend that only an experienced cluster administrator use this procedure.

## Before You Begin

Before you apply this Best Practice for moving a cluster to another IP subnet, you must understand why only an experienced cluster administrator should perform the procedure in *Applying the Best Practice*:

- The procedure requires a clusterwide reboot.
- Several system configuration files require careful editing. Some of these files are context-dependent symbolic links (CDSLs); you must edit the target file for each cluster member, and use `cp` rather than `mv` when copying files. (If you use `mv`, you will overwrite the CDSL instead of copying the file to the target of the CDSL.)
- When editing and copying files, you can make mistakes that might stop cluster members from booting and forming a cluster. Verify edits twice before putting the modified files into place and halting the cluster.
- When you modify an IP address, the subnet mask associated with that address might also require modification.
- The tables and scripts in this Best Practice deal with the most common system configuration files that contain host names and IP addresses. You must review, and possibly edit, any other files such as cluster

application availability (CAA) scripts that might contain host names, cluster aliases, interface IP aliases, or subnet masks.

The word *alias* has a number of closely related but different meanings:

- Default cluster alias: The host name and IP address associated with the entire cluster. By default, all cluster members join the default cluster alias at boot time.
- Other (nondefault) cluster aliases: Cluster aliases created by an administrator, who determines which members join which aliases.
- Interface IP alias: An additional IP address associated with a network interface; created using the `ifconfig` command, and sometimes used with CAA scripts as an alternative to cluster aliases.
- Name alias: A name used, for example, in `/etc/hosts` as an alias for the fully qualified host name. In the following sample line from a `hosts` file, `joey` is a name alias for the fully qualified `joey.zk3.dec.com`:

```
16.140.112.176 joey.zk3.dec.com joey
```

When preparing for the move, make sure that you know where any of these aliases are used in the cluster. In addition, be aware of any host names, IP addresses, or subnet masks associated with these aliases. Some or all of them might require changes when moving the cluster to another subnet.

## Applying the Best Practice

Before you attempt to edit files and move the cluster to another IP subnet, be sure to follow the recommendations in *Before You Begin*.

### Procedure

To move a cluster to another IP subnet, perform the following steps:

1. Obtain the IP names and addresses you will need for the move. Use the attribute tables to record this information. Note any changes in subnet masks required for the move. If the move will result in the cluster using different name servers, note the changes you will need to make to the `/etc/resolv.conf` file.
2. If this move requires changing any physical network connections, make sure the new ones are in place and ready for use.
3. Tell users and other cluster and network administrators when the move will occur. If other systems or clusters depend on any of the IP names and addresses you plan to change, their administrators will have to coordinate with you; for example, NIS, DNS, or mail servers

might be affected. If the cluster provides any services that must not be interrupted, make preparations for another system or cluster to provide these services while the cluster is shut down.

4. Run the `getinfo` script to determine where the current cluster's IP names and addresses appear in common system configuration files.
5. Look for any other files, such as CAA scripts, that might contain host names, IP addresses, cluster aliases, interface IP aliases, name aliases, or subnet masks.
6. Make both save and work copies of the configuration files you plan to modify. Use the `savefiles` script to make copies of the standard set of files you would normally edit.

---

**Note**

---

If, for example, any CAA scripts or site-specific scripts reference IP addresses, host names, aliases, or subnet masks that will change, make copies of those files and keep track of the changes you make to them.

You can modify the `savefiles` script to fit your needs.

---

7. Use the table that corresponds to the changes you plan to make:
  - Changing external IP addresses only
  - Changing external IP addresses and host names
  - Changing external and internal IP addresses and host names
- a. Using the information in the table, edit the work copies of the files (not the originals). For CDSLs, remember to edit the work files in the member-specific directories.

---

**Notes**

---

Whether you use `sysman` or an editor to make the edits is up to you. It is important that the correct edits are made, and that you are 100 percent certain that the information is correct before you copy the work files to the original file locations and halt all cluster members.

When editing the work files, if the subnet mask on the new subnet is not the same as the one for the current subnet, remember to make those edits.

If you edit the `.rhosts` or `/etc/hosts.equiv` work files, add new host names but do not remove those currently listed there. You will remove the old host names after you have successfully booted the cluster on the other subnet.

---

- b. Use the output from the `getinfo` script and the checklist table to keep track of your edits.
  - c. After you have made all edits, run the `filediffs` script to compare the contents of the work files to the original files. Use the script output to verify your edits.
8. After all edits are made to your satisfaction:
- a. Send messages to everyone who will be affected; tell them when you plan to disable access to the cluster and start the actual move.
  - b. Disable logins and all external network interfaces. For example, you can use the `wall -c` command to tell users how soon you will disable logins, then use the `touch` command to create the `/etc/nologin` file, and then use the `rcinet stop` command on each member to stop the network on that member.
  - c. Use the `copyfiles` script to copy the work files to the original file locations. Use the checklist to keep track of each file you replace with an edited version. (Remember to keep track of any other files you modify.)

To verify that the work files have been copied to the original locations, run the `filediffs` script. There should be no differences between the work files and the original files at this point. (Remember to manually verify the content of any other files you modify.)
  - d. After all the edited configuration files are in place, make sure that all the information is correct. When you shut down the cluster, the success of the reboot depends on your edits. Depending on the extent of the changes required for the move, verify the correct modifications of some or all of the following items:
    - IP addresses, interface IP aliases, and subnet masks
    - Host names and name aliases

- Default cluster alias and other cluster aliases

9. Halt each member of the cluster. For example, on each member, run the following command:

```
# shutdown -h now
```

---

**Note**

---

Because the `shutdown -c` command gets some of its information from the files you just edited, you must halt each member. (If you run `clu_get_info -full` after the edited files are in place, some of the displayed information will reflect the edits.)

---

10. When the cluster has halted, make any required changes to the network connections.
11. Boot the cluster.
12. Use the information in *Verifying Success* to determine whether everything is working as expected.
13. Remove outdated entries from `.rhosts` and `/etc/hosts.equiv`.
14. If you changed a member's hostname, we suggest that you modify the kernel configuration information for that member. Modifying this information now means that, when you later build a kernel or patch the system, the kernel configuration tools can find configuration files that match the new hostname.

For each member whose hostname has changed:

- If the member's `/sys/conf/SYSTEM_NAME` kernel configuration file is generic (you have not customized the file), you can run the `doconfig -a` command to automatically create a `/sys/conf/SYSTEM_NAME` file, a `/sys/conf/SYSTEM_NAME.list` file, and a `/sys/SYSTEM_NAME` directory for that member.
- If the member's `/sys/conf/SYSTEM_NAME` kernel configuration file is customized, copy the current file to the new name, and edit the new file to change the value of the `ident` parameter to the new hostname. Also copy the `/sys/conf/SYSTEM_NAME.list` file to the new name. Then run the `doconfig -c SYSTEM_NAME`

command, specifying the modified configuration file as the target with the `-c` option.

You can install and boot the new kernel if you want to but it is not required. The goal is to make sure that the kernel configuration information is updated to reflect the new hostname. See `doconfig(8)` for information on building kernels.

## File Edit Tables

This section provides tables that list the required edits to common system configuration files for the different types of moves:

- Changing external IP addresses only
- Changing external IP addresses and host names
- Changing external and internal IP addresses and host names

### Changing External IP Addresses Only

For common configuration files, the following table lists items that you might have to edit when changing a cluster's external IP addresses:

File	Items to Edit
<b>Shared Files</b>	
<code>/etc/hosts</code>	IP addresses associated with each of the cluster's external interfaces and cluster aliases.
<code>/etc/networks</code>	If it is defined, the address of the network.
<code>/etc/resolv.conf</code>	If the cluster will use different name servers on the new subnet, their addresses.
<b>CDSLs</b>	
<code>/etc/clu_alias.config</code>	For each member, any alias that is defined by its IP address (and possibly a subnet mask). (You should not have to modify the <code>DEFAULTALIAS</code> entry.)
<code>/etc/inet.local</code>	For each member, if this file is used to configure interface IP aliases.
<code>/etc/ntp.conf</code>	For each member, if the IP address changes for a server or peer system entry.
<code>/etc/rc.config</code> (CDSL)	For each member, the <code>IFCONFIG</code> entries. If necessary, modify subnet masks.
<code>/etc/routes</code>	For each member, if routes are defined. If necessary, modify subnet masks.
<b>Member-Specific But Not CDSLs</b>	

File	Items to Edit
/etc/gated.conf.member<n>	For each member, any entry whose IP address must change. If CLUAMGR_ROUTE_ARGS is set to nogated in a member's rc.config file, modify that member's /etc/gated.conf file (CDSL).
/cluster/admin/.member<n>.cfg	If you use these files, update any changes to IP addresses. (Otherwise you will reset the cluster to the old values if you use the files with clu_create or clu_add_member.)

### Changing External IP Addresses and Host Names

For common configuration files, the following table lists items that you might have to edit when changing a cluster's external IP addresses and host names:

File	Items to Edit
<b>Shared Files</b>	
/.rhosts	Add the new cluster name. Do not remove any entries.
/etc/cfgmgr.auth	Member host names.
/etc/hosts	IP addresses, host names, and aliases associated with the cluster's external interfaces and cluster aliases.
/etc/hosts.equiv	Add the new cluster name. Do not remove any entries.
/etc/networks	If it is defined, the address of the network.
/etc/resolv.conf	If you are changing the domain name or name servers on the new subnet.
<b>CDSLs</b>	
/etc/clu_alias.config	For each member, any alias that is defined by its host name or IP address (and possibly a subnet mask). (You should not have to modify the DEFAULTALIAS entry.)
/etc/inet.local	For each member, if this file is used to configure interface IP aliases.
/etc/ntp.conf	For each member, if the IP address or host name changes for a server or peer system entry.
/etc/rc.config (CDSL)	For each member, the HOSTNAME,IFCONFIG, and CLUSTER_NET entries. If necessary, modify subnet masks.

File	Items to Edit
/etc/routes	For each member, if routes are defined. If necessary, modify subnet masks.
/etc/sysconfigtab (CDSL)	For each member, the <code>cluster_name</code> and the <code>cluster_node_name</code> .
<b>Member-Specific But Not CDSLs</b>	
/etc/gated.conf.member<n>	For each member, any entry whose IP address must change. If <code>CLUAMGR_ROUTE_ARGS</code> is set to <code>nogated</code> in a member's <code>rc.config</code> file, modify that member's <code>/etc/gated.conf</code> file (CDSL).
/cluster/admin/.member<n>.cfg	If you use these files, update any changes to cluster names, host names, and IP addresses. (Otherwise you will reset the cluster to the old values if you use the files with <code>clu_create</code> or <code>clu_add_member</code> .)

## Changing External and Internal IP Addresses and Host Names

For common configuration files, the following table lists items that you might have to edit when changing a cluster's external and internal IP addresses and host names:

File	Items to Edit
<b>Shared Files</b>	
/.rhosts	Add the new cluster name and the new host names associated with the cluster interconnect: *-mc0 for Version 5.0A through 5.1; *-ics0 for Version 5.1A or higher. Do not remove any entries.
/etc/cfgmgr.auth	Member host names.
/etc/hosts	IP addresses, host names, and aliases associated with cluster's external interfaces, cluster aliases, and cluster interconnect interfaces.
/etc/hosts.equiv	Add the new cluster name and the new host names associated with the cluster interconnect: *-mc0 for Version 5.0A through 5.1; *-ics0 for Version 5.1A or higher. Do not remove any entries.
/etc/networks	If it is defined, the address of the network.
/etc/resolv.conf	If you are changing the domain name or name servers on the new subnet.
<b>CDSLs</b>	

File	Items to Edit
/etc/clu_alias.config	For each member, any alias that is defined by its host name or IP address (and possibly a subnet mask). (You should not have to modify the DEFAULTALIAS entry.)
/etc/ifaccess.conf	IP addresses associated with the cluster interconnect. If necessary, modify subnet masks.
/etc/inet.local	For each member, if this file is used to configure interface IP aliases.
/etc/ntp.conf	For each member, if the IP address or host name changes for a server or peer system entry. (If you change the names associated with the cluster interconnect, make sure to change those peer names.)
/etc/rc.config	For each member, the HOSTNAME,IFCONFIG, and CLUSTER_NET entries. If necessary, modify subnet masks.
/etc/routes	For each member, if routes are defined. If necessary, modify subnet masks.
/etc/sysconfigtab	For each member, the value of cluster_name, cluster_node_name, and cluster_node_inter_name.
<b>Member-Specific But Not CDSLs</b>	
/etc/gated.conf.member<n>	For each member, any entry whose IP address must change. If CLUAMGR_ROUTE_ARGS is set to nogated in a member's rc.config file, modify that member's /etc/gated.conf file (CDSL).
/cluster/admin/.member<n>.cfg	If you use these files, update any changes to cluster names, host names, cluster interconnect names, and IP addresses. (Otherwise you will reset the cluster to the old values if you use the files with clu_create or clu_add_member.)

## Attribute and Checklist Tables

Use the tables in this section to record the IP name and address information you will need to move the cluster to its new subnet. If you have more than four cluster members, or more than three cluster aliases, make copies of the pertinent tables and relabel rows as needed.

### External Host Names and IP Addresses

Member	Attribute	Value	
Member 1	Host Name	Old	
		New	
	IP Address (and subnet mask)	Old	
		New	
Member 2	Host Name	Old	
		New	
	IP Address (and subnet mask)	Old	
		New	
Member 3	Host Name	Old	
		New	
	IP Address (and subnet mask)	Old	
		New	
Member 4	Host Name	Old	
		New	
	IP Address (and subnet mask)	Old	
		New	

### Cluster Name and Cluster Aliases

Cluster Alias	Value	
Fully qualified cluster name (the cluster name is the default cluster alias)	Old	
	New	
Default cluster alias IP address (and subnet mask)	Old	
	New	
Name of additional cluster alias #1	Old	
	New	
IP Address of additional cluster alias #1 (and subnet mask)	Old	
	New	
Name of additional cluster alias #2	Old	
	New	

Cluster Alias	Value	
IP Address of additional cluster alias #2 (and subnet mask)	Old	
	New	

### Interface IP Aliases

Member	Attribute	Value	
Member 1	IP Alias #1 (and subnet mask)	Old	
		New	
	IP Alias #2 (and subnet mask)	Old	
		New	
Member 2	IP Alias #1 (and subnet mask)	Old	
		New	
	IP Alias #2 (and subnet mask)	Old	
		New	
Member 3	IP Alias #1 (and subnet mask)	Old	
		New	
	IP Alias #2 (and subnet mask)	Old	
		New	
Member 4	IP Alias #1 (and subnet mask)	Old	
		New	
	IP Alias #2 (and subnet mask)	Old	
		New	

### External Servers

If the cluster will use different servers for network services such as BIND, NIS, or NTP on the new subnet, record the old and new IP addresses used by these services in the following table:

Server	IP Address	
	Old	
	New	

Server	IP Address	
		Old
	New	
	Old	
	New	
	Old	
	New	

### Checklist

Use the checklist's `Status` column to keep track of edits to the work copies of configuration files. *Example: Checklist* shows how we used the `Status` column to record when edits to a work file were complete, and when the work file was copied to the original file's location.

File	Status	
<b>Shared Files</b>		
<code>/.rhosts</code>		
<code>/etc/cfgmgr.auth</code>		
<code>/etc/hosts</code>		
<code>/etc/hosts.equiv</code>		
<code>/etc/networks</code>		
<code>/etc/resolv.conf</code>		
<b>CDSLs</b>		
<code>/etc/clu_alias.config</code>	Member 1	
	Member 2	
	Member 3	
	Member 4	
<code>/etc/ifaccess.conf</code>	Member 1	
	Member 2	
	Member 3	
	Member 4	

File	Status	
/etc/inet.local	Member 1	
	Member 2	
	Member 3	
	Member 4	
/etc/ntp.conf	Member 1	
	Member 2	
	Member 3	
	Member 4	
/etc/rc.config	Member 1	
	Member 2	
	Member 3	
	Member 4	
/etc/routes	Member 1	
	Member 2	
	Member 3	
	Member 4	
/etc/sysconfigtab	Member 1	
	Member 2	
	Member 3	
	Member 4	
<b>Member-Specific But Not CDSLs</b>		
/etc/gated.conf.member<n>	Member 1	
	Member 2	
	Member 3	
	Member 4	
/cluster/admin/.member<n>.cfg	Member 1	
	Member 2	
	Member 3	
	Member 4	

## Shell Scripts

This section contains the following shell scripts that you can use when moving a cluster to another IP subnet:

- `getinfo`: Gets information about the current cluster's use of IP addresses and host names. Use the output to help determine which files require edits.
- `savefiles`: Makes date-stamped save copies and work copies of the most often edited system files. Edit the work copies; keep the date-stamped copies as backups.
- `diff_files`: Compares contents of work copies to original files. Use the output to verify edits before copying work files to original locations.
- `copyfiles`: Copies work files to original file locations.

### getinfo Script

The `getinfo` script uses `clu_get_info` to get information about the current cluster, and then uses `grep` to search for that information in common system configuration files. The script writes the output to `/tmp/GetInfoOutput`. We recommend that you run this script before making any changes, and that you use the script along with the checklist when making and verifying edits.

```
#!/bin/ksh
## getinfo
# Look in stock set of system configuration files to determine
# where current cluster's IP addresses and host names are used

typeset -r awk=/usr/bin/awk
typeset -r cat=/sbin/cat
typeset -r cluamgr="/usr/sbin/cluamgr -s all"
typeset -r clu_get_info="/usr/sbin/clu_get_info -raw"
typeset -r grep=/sbin/grep
typeset -r mkdir=/sbin/mkdir
typeset -r sort=/sbin/sort

typeset GrepFile=tmpgrep1
typeset GrepFile2=tmpgrep2
typeset OutFile=GetInfoOutput
typeset -r String1="-----"
typeset -r String2="\n=====

# mkdir a temporary directory in this member's /tmp
MKDIR_LIM=10
i=$MKDIR_LIM
while [ $i -gt 0 ]
do
    TMPDIR=/tmp/tmp.$$.`date +%j.%S`
    ${mkdir} -m 700 $TMPDIR && break
    i=`expr $i - 1`
    sleep 1
```

```

done

if [ $i -eq 0 ]
then
echo 1>&2 "Could not create temporary directory, aborting"
exit 2
fi

echo "    ${OutFile}" >> ${TMPDIR}/${OutFile}
echo ${String1}\n >> ${TMPDIR}/${OutFile}
# gather information for later greps
# full name of cluster
echo '${clu_get_info} -raw| ${grep} ^C | ${awk} -F: '{print $2}' \
    >> ${TMPDIR}/${GrepFile}

# member names (long and short forms) and interconnect names
for a in '${clu_get_info} -raw | ${grep} ^M | ${awk} -F: '{print $3 "\n" $4}'
do
    echo ${a} >> ${TMPDIR}/${GrepFile}
    [[ ${a} = *. .* ]] && {
    echo ${a} | ${awk} -F. '{print $1}' >> ${TMPDIR}/${GrepFile}
    }
done

# cluster alias names (long and short forms)
for b in '${cluamgr} | ${grep} Status | ${awk} -F: " '{print $2}'
do
    echo ${b} >> ${TMPDIR}/${GrepFile}
    echo ${b} | ${awk} -F. '{print $1}' >> ${TMPDIR}/${GrepFile}
done

# using what we have, what else can we find in hosts
for c in '${cat} ${TMPDIR}/${GrepFile}'
do
    ${grep} ${c} /etc/hosts | awk '{print $1}' \
    | ${sort} -u >> ${TMPDIR}/${GrepFile2}
done

# consolidate information in one file for later "grep -f <file>"
cat ${TMPDIR}/${GrepFile2} >> ${TMPDIR}/${GrepFile}

echo "grep'd for these strings in the following files:" \
    >> ${TMPDIR}/${OutFile}
echo ${String1} >> ${TMPDIR}/${OutFile}
cat ${TMPDIR}/${GrepFile} | ${sort} -u >> ${TMPDIR}/${OutFile}
echo ${String2} >> ${TMPDIR}/${OutFile}

# shared files
for d in /.rhosts \
    /etc/cfgmgr.auth \
    /etc/hosts \
    /etc/hosts.equiv \
    /etc/networks \
    /etc/resolv.conf
do
    echo ${d} >> ${TMPDIR}/${OutFile}
    echo ${String1} >> ${TMPDIR}/${OutFile}
    ${grep} -f ${TMPDIR}/${GrepFile} ${d} >> ${TMPDIR}/${OutFile}
    echo ${String2} >> ${TMPDIR}/${OutFile}

```

```

done

for f in `clu_get_info | grep ^M | ${awk} -F: '{print $2}'`
do
    # one-off for sysconfigtab
    y=/cluster/members/member${f}/boot_partition/etc/sysconfigtab
    echo ${y} >> ${TMPDIR}/${OutFile}
    echo ${String1} >> ${TMPDIR}/${OutFile}
    ${grep} -f ${TMPDIR}/${GrepFile} ${y} >> ${TMPDIR}/${OutFile}
    echo ${String2} >> ${TMPDIR}/${OutFile}

    # rest are in /cluster/members/{memb}/etc/
    for g in clu_alias.config \
            ifaccess.conf \
            inet.local \
            ntp.conf \
            rc.config \
            routes
    do
        y=/cluster/members/member${f}/etc/${g}
        echo ${y} >> ${TMPDIR}/${OutFile}
        echo ${String1} >> ${TMPDIR}/${OutFile}
        ${grep} -f ${TMPDIR}/${GrepFile} ${y} >> ${TMPDIR}/${OutFile}
        echo ${String2} >> ${TMPDIR}/${OutFile}
    done

    # member-specific files but not CDSLs
    for h in /cluster/admin/.member${f}.cfg \
            /etc/gated.conf.member${f}
    do
        echo $h >> ${TMPDIR}/${OutFile}
        echo ${String1} >> ${TMPDIR}/${OutFile}
        ${grep} -f ${TMPDIR}/${GrepFile} $h >> ${TMPDIR}/${OutFile}
        echo ${String2} >> ${TMPDIR}/${OutFile}
    done
done

# Make a copy before removing TMPDIR
cp ${TMPDIR}/${OutFile} /tmp
print "Output is in /tmp/${OutFile}"

/bin/rm -rf ${TMPDIR} 2> /dev/null

```

## savefiles Script

The savefiles script makes copies of a stock set of system configuration files. The script makes two copies of each file: a date-stamped save copy, and a work copy for editing.

```

#!/bin/ksh
## savefiles
# make save (date-stamped) and work copies of files before editing

typeset -r awk=/usr/bin/awk
typeset -r cp=/sbin/cp
typeset -r date=/sbin/date

datestamp=`date | ${awk} '{print $3 $2 $6}'`
work=work

```

```

# shared files
for i in /.rhosts \
    /etc/cfgmgr.auth \
    /etc/hosts \
    /etc/hosts.equiv \
    /etc/networks \
    /etc/resolv.conf
do
    [ -s ${i} ] && {
        ${cp} ${i} ${i}.${datestamp}
        ${cp} ${i} ${i}.${work}
    }
done

# CDSLs, need member IDs for all members
for i in `clu_get_info -raw | grep ^M | awk -F: '{print $2}'`
do
    # one-off for sysconfigtab
    y=/cluster/members/member${i}/boot_partition/etc/sysconfigtab
    [ -s ${y} ] && {
        ${cp} ${y} ${y}.${datestamp}
        ${cp} ${y} ${y}.${work}
    }

    # the rest are in /cluster/members/{memb}/etc/
    for j in clu_alias.config \
        ifaccess.conf \
        inet.local \
        ntp.conf \
        rc.config \
        routes
    do
        y=/cluster/members/member${i}/etc/${j}
        [ -s ${y} ] && {
            ${cp} ${y} ${y}.${datestamp}
            ${cp} ${y} ${y}.${work}
        }
    done

# member-specific files but not CDSLs
for k in /cluster/admin/.member${i}.cfg \
    /etc/gated.conf.member${i}
do
    ${cp} ${k} ${k}.${datestamp}
    ${cp} ${k} ${k}.${work}
done
done

cat <<EOF

${0}: Made save (date-stamped) and work copies for files in the
following list that are nonzero in length.

# shared files
.rhosts
/etc/cfgmgr.auth
/etc/hosts
/etc/hosts.equiv
/etc/networks

```

```

/etc/resolv.conf

# CDSLs
/etc/clu_alias.config
/etc/ifaccess.conf
/etc/inet.local
/etc/ntp.conf
/etc/rc.config
/etc/routes
/etc/sysconfigtab (only CDSL whose target is not {memb}/etc)

# member-specific files but not CDSLs
/cluster/admin/.member<n>.cfg
/etc/gated.conf.member<n>

EOF
exit 0

```

## filediffs Script

The `filediffs` script uses the `diff` command to compare the edited contents of the work files with the original files. It writes output to `/tmp/FileDiffsOutput`. Use this script to help verify the correctness of your edits before copying the work files to the original file locations.

```

#!/bin/ksh
## filediffs

typeset -r diff=/sbin/diff
typeset -r work=work
typeset -r OutFile=/tmp/FileDiffsOutput
typeset -r String1="-----"
typeset -r String2="\n=====

[ -s ${OutFile} ] && rm ${OutFile}

echo " ${OutFile}" >> ${OutFile}
echo ${String1}\n >> ${OutFile}

# diff work copies of shared files
for d in /.rhosts \
        /etc/cfgmgr.auth \
        /etc/hosts \
        /etc/hosts.equiv \
        /etc/networks \
        /etc/resolv.conf
do
    [ -s ${d}.${work} ] && {
        echo "diff ${d} \n      ${d}.${work}" >> ${OutFile}
        echo ${String1} >> ${OutFile}
        ${diff} ${d} ${d}.${work} >> ${OutFile}
        echo $String2 >> ${OutFile}
    }
done

# one-off diff for sysconfigtab
for f in `clu_get_info -raw | grep ^M | awk -F: '{print $2}'`
do
    y=/cluster/members/member${f}/boot_partition/etc/sysconfigtab

```

```

[ -s ${y}.${work} ] && {
echo "diff ${y}\n      ${y}.${work}" >> ${OutFile}
echo ${String1} >> ${OutFile}
${diff} ${y} ${y}.${work} >> ${OutFile}
echo ${String2} >> ${OutFile}
}

# diff files in /cluster/members/{memb}/etc/
for g in clu_alias.config \
  ifaccess.conf \
  inet.local \
  ntp.conf \
  rc.config \
  routes
do
y=/cluster/members/member${f}/etc/${g}
[ -s ${y}.${work} ] && {
echo "diff ${y}\n${y}.${work}" >> ${OutFile}
echo ${String1} >> ${OutFile}
${diff} ${y} ${y}.${work} >> ${OutFile}
echo ${String2} >> ${OutFile}
}
done

# diff member-specific files but not CDSLs
for h in /cluster/admin/.member${f}.cfg \
  /etc/gated.conf.member${f}
do
[ -s ${h}.${work} ] && {
echo "diff ${h}\n      ${h}.${work}" >> ${OutFile}
echo ${String1} >> ${OutFile}
${diff} ${h} ${h}.${work} >> ${OutFile}
echo ${String2} >> ${OutFile}
}
done
done

print "diffs are in ${OutFile}"

```

## copyfiles Script

The copyfiles script copies the \*.work files to the original file locations.

```

#!/bin/ksh
## copyfiles
# copy work files to original locations
# to see output, change /bin/ksh to /bin/ksh -x

typeset -r cp=/sbin/cp
work=work

# shared files
for i in /.rhosts \
  /etc/cfgmgr.auth \
  /etc/hosts \
  /etc/hosts.equiv \
  /etc/networks \
  /etc/resolv.conf

```

```

do
    [ -s ${i}.${work} ] && cp ${i}.${work} ${i}
done

# CDSLs, need member IDs for all members
for i in `clu_get_info -raw | grep ^M | awk -F: '{print $2}'`
do
    # one-off for sysconfigtab
    y=/cluster/members/member${i}/boot_partition/etc/sysconfigtab
    [ -s ${y}.${work} ] && cp ${y}.${work} ${y}

    # the rest are in /cluster/members/{memb}/etc/
    for j in clu_alias.config \
        ifaccess.conf \
        inet.local \
        ntp.conf \
        rc.config \
        routes
    do
        y=/cluster/members/member${i}/etc/${j}
        [ -s ${y}.${work} ] && \
        cp ${y}.${work} ${y}
    done

    # member-specific files but not CDSLs
    for k in /cluster/admin/.member${i}.cfg \
        /etc/gated.conf.member${i}
    do
        [ -s ${k}.${work} ] && cp ${k}.${work} ${k}
    done
done
done

```

## Verifying Success

After you apply the Best Practice for moving a cluster to another IP subnet, you can verify whether it was successful.

If all edits are made correctly, and the edited files are put in their proper places, the systems will boot, form a cluster, and assume their new identities. Use the following commands to verify that the cluster and its subsystems are operating correctly:

```

# hostname
# ifconfig -a
# netstat -i
# clu_get_info -full | more
# cluamgr -s all
# caa_stat

```

You can also use standard networking commands like `ping`, `rlogin`, and `rpcinfo` to verify that the cluster members are available for use, will accept logins, and can communicate with other systems.

If the Best Practice was not successful, see *Troubleshooting* for information about identifying and solving problems.

## Troubleshooting

If you determine that the Best Practice was not successful, as described in *Verifying Success*, use the following table to identify and solve problems:

Problem	Possible Solutions
Member cannot boot.	<p>If the failure occurs early in the boot path, you probably made a mistake editing the <code>sysconfigtab</code> file.</p> <p>If you can get one member to boot, mount the <code>boot_partition</code> of the member that cannot boot, and fix the edits.</p> <p>If no members can boot, first determine whether any members fail because they cannot gain quorum. If so, perform an interactive boot, as described in <i>Cluster Administration</i>, for one member and set the value of expected votes to zero. Boot that member, fix the other members' files, boot those members, and readjust the expected votes for quorum.</p> <p>If you cannot boot any members interactively, boot the Tru64 UNIX operating system, mount the <code>boot_partition</code> for the first member of the cluster, fix the edits for that member, halt the Tru64 UNIX system, boot the cluster member (interactively, if necessary), and fix the remaining members.</p>
Cluster boots but some network problems were encountered in multiuser mode.	Define the problem and decide which files on which members are most likely to have bad edits. Fix the edits, then stop and restart network services on those systems.
Even after applying the preceding solutions, you are unsuccessful.	Restore the saved copies of the original files. Restore the original network connections. Boot the cluster on its old subnet so the cluster can continue to serve clients while you figure out what went wrong.

## Example

This section shows the steps taken to move a two-member cluster to another IP subnet. We performed the most complex type of move, modifying both external and cluster interconnect IP addresses and host names.

The following notes were made before starting:

- Because these are test clusters, no general notification or coordination is needed.

- We will change host names, default cluster alias, and the IP addresses associated with those names.
  - Because we are changing member host names, we will, for consistency, change the names (but not the IP addresses) associated with the cluster interconnect interfaces.
  - We will not add or remove any network interface cards.
  - Because no CAA scripts reference IP addresses or host names, no modifications are required for CAA.
  - No interface IP aliases are in use.
  - No site-specific scripts are in use.
1. Used the tables in *Attribute and Checklist Tables* to record information. The filled-in tables are:
    - *Example: External Host Names and IP Addresses*
    - *Example: Cluster Name and Cluster Aliases*
    - *Example: Interface IP Aliases*
    - *Example: External Servers*
  2. Ran the `getinfoscript` to determine which files had entries associated with the current cluster IP addresses and host names. The script output is in *Example: getinfo Output*.
  3. Ran the `savefiles` script to make save and work copies of all the files that might be edited.
  4. Used the Changing External and Internal IP Addresses and Host Names table as a guide for editing each file.
 

Used the `getinfo` script output and the checklist to keep track of the edits. The `getinfo` output is in *Example: getinfo Output*. The filled-in checklist is in *Example: Checklist*.
  5. After all edits were made, ran the `filediffs` script and used its output as a final checkpoint before continuing. The `filediffs` output is in *Example: filediffs Output*.
  6. Disabled all external network interfaces. (Did not disable logins because the cluster is used only for testing. Sent a `wall` message before disabling the interfaces.)
  7. Used the `copyfiles` script to copy (not move) the edited\* `.work` versions of files to the correct locations. For each file, manually verified that the edited version had replaced the original version, and that the contents were correct.

8. Halted each member of the cluster.
9. Disconnected the network cables from the 16.140.160 subnet and connected those from the 16.140.112 subnet.
10. Booted the cluster.
11. Verified that the cluster and both members had correctly assumed their new identities. For example, ran `clu_get_info -full` and `ifconfig -a`.

### Example: External Host Names and IP Addresses

Member	Attribute	Value	
Member 1	Host Name	Old	swiss.zk3.dec.com
		New	ernest.zk3.dec.com
	IP Address (and subnet mask)	Old	16.140.160.121, 255.255.255.0
		New	16.140.112.238, 255.255.255.0
Member 2	Host Name	Old	rye.zk3.dec.com
		New	joey.zk3.dec.com
	IP Address (and subnet mask)	Old	16.140.160.122, 255.255.255.0
		New	16.140.112.176, 255.255.255.0

### Example: Cluster Name and Cluster Aliases

Cluster Alias	Value	
Fully qualified cluster name (the cluster name is the default cluster alias)	Old	deli.zk3.dec.com
	New	gallo.zk3.dec.com
Default cluster alias IP address (and subnet mask)	Old	16.140.160.124, 255.255.255.0
	New	16.140.112.209, 255.255.255.0
Name of additional cluster alias #1	Old	No other cluster aliases defined

### Example: Interface IP Aliases

Member	Attribute	Value	
Member 1	IP Alias #1	Old	No interface IP aliases defined on this member
		New	
	IP Alias #2	Old	
		New	
Member 2	IP Alias #1	Old	No interface IP aliases defined on this member
		New	
	IP Alias #2	Old	
		New	

### Example: External Servers

Server	IP Address	
BIND	Old	Not changing servers
	New	
NTP	Old	Not changing servers
	New	

### Example: getinfo Output

```

GetInfoResults
-----

grep'd for these strings in the following files:
-----
10.0.0.1
10.0.0.2
16.140.160.121
16.140.160.122
16.140.160.124
deli
deli.zk3.dec.com
rye
rye-ics0
rye.zk3.dec.com
swiss
swiss-ics0
swiss.zk3.dec.com

=====
/.rhosts
-----

```

```

deli.zk3.dec.com
swiss-ics0
rye-ics0

=====
/etc/cfgmgr.auth
-----
rye.zk3.dec.com

=====
/etc/hosts
-----
16.140.160.121 swiss.zk3.dec.com swiss
16.140.160.124 deli.zk3.dec.com deli
10.0.0.1 swiss-ics0
10.0.0.2 rye-ics0
16.140.160.122 rye.zk3.dec.com rye

=====
/etc/hosts.equiv
-----
deli.zk3.dec.com
swiss-ics0
rye-ics0

=====
/etc/networks
-----

=====
/etc/resolv.conf
-----

=====
/cluster/members/member1/boot_partition/etc/sysconfigtab
-----
cluster_name=deli
cluster_node_name=swiss
cluster_node_inter_name=swiss-ics0

=====
/cluster/members/member1/etc/clu_alias.config
-----

=====
/cluster/members/member1/etc/ifaccess.conf
-----

=====
/cluster/members/member1/etc/inet.local
-----

=====
/cluster/members/member1/etc/ntp.conf
-----
peer rye-ics0 version 3

=====
/cluster/members/member1/etc/rc.config

```

```

-----
HOSTNAME="swiss.zk3.dec.com"
IFCONFIG_0="16.140.160.121 netmask 255.255.255.0"
IFCONFIG_1="10.0.0.1 netmask 255.255.255.0"
CLUSTER_NET="swiss-ics0"

=====
/cluster/members/member1/etc/routes
-----

=====
/cluster/admin/.member1.cfg
-----
# date: Thu Jun 28 16:01:14 EDT 2001 hostname swiss.zk3.dec.com
clu_alias_ip=16.140.160.124
clu_ics_host=swiss-ics0
clu_ics_ip=10.0.0.1
clu_name=deli.zk3.dec.com

=====
/etc/gated.conf.member1
-----
    host 16.140.160.124 interface 127.0.0.1;
export proto rip interface 10.0.0.1 {
export proto rip interface 16.140.160.121 {
    16.140.160.124 mask 255.255.255.255 metric 14;

=====
/cluster/members/member2/boot_partition/etc/sysconfigtab
-----
cluster_name=deli
cluster_node_name=rye
cluster_node_inter_name=rye-ics0

=====
/cluster/members/member2/etc/clu_alias.config
-----

=====
/cluster/members/member2/etc/ifaccess.conf
-----

=====
/cluster/members/member2/etc/inet.local
-----

=====
/cluster/members/member2/etc/ntp.conf
-----
peer swiss-ics0 version 3

=====
/cluster/members/member2/etc/rc.config
-----
CLUSTER_NET="rye-ics0"
IFCONFIG_0="10.0.0.2 netmask 255.255.255.0"
XNTP_SERV2="swiss-ics0"
HOSTNAME="rye.zk3.dec.com"
IFCONFIG_1="16.140.160.122 netmask 255.255.255.0"

=====

```

```

/cluster/members/member2/etc/routes
-----

=====
/cluster/admin/.member2.cfg
-----
# date: Thu Jun 28 16:19:13 EDT 2001 hostname swiss.zk3.dec.com
clu_ics_host=rye-ics0
clu_ics_ip=10.0.0.2
unix_host=rye.zk3.dec.com

=====
/etc/gated.conf.member2
-----
    host 16.140.160.124 interface 127.0.0.1;
export proto rip interface 10.0.0.2 {
export proto rip interface 16.140.160.122 {
    16.140.160.124 mask 255.255.255.255 metric 14;

=====

```

## Example: Checklist

File	Status	
<b>Shared Files</b>		
/.rhosts	Edits: done. Replace original file: done.	
/etc/cfgmgr.auth	Edits: done. Replace original file: done.	
/etc/hosts	Edits: done. Replace original file: done.	
/etc/hosts.equiv	Edits: done. Replace original file: done.	
/etc/networks	Checked; none defined.	
/etc/networks	Checked; not modifying for test. Will be in same domain and use same name servers.	
<b>CDSLs</b>		
/etc/clu_alias.config	Member 1	Checked; no IP addresses or host names.
	Member 2	Checked; no IP addresses or host names.
/etc/ifaccess.conf	Member 1	Checked; not changing interconnect IP addresses.
	Member 2	Checked; not changing interconnect IP addresses.
/etc/inet.local	Member 1	Checked; nothing defined.
	Member 2	Checked; nothing defined.

File	Status	
/etc/ntp.conf	Member 1	Edits: done. Replace original file: done. (Set peer to new name of other member.)
	Member 2	Edits: done. Replace original file: done. (Set peer to new name of other member.)
/etc/rc.config	Member 1	Edits: done. Replace original file: done. (HOSTNAME, IFCONFIG_0, CLUSTER_NET)
	Member 2	Edits: done. Replace original file: done. (HOSTNAME, IFCONFIG_0, CLUSTER_NET, XNTP - member 1 is a peer)
/etc/routes	Member 1	No routes defined.
	Member 2	No routes defined.
/etc/sysconfigtab	Member 1	Edits: done. Replace original file: done. (cluster_name, cluster_node_name, cluster_inter_name)
	Member 2	Edits: done. Replace original file: done. (cluster_name, cluster_node_name, cluster_inter_name)
<b>Member-Specific But Not CDSLs</b>		
/etc/gated.conf.member<n>	Member 1	Edits: done. Replace original file: done.
	Member 2	Edits: done. Replace original file: done.
/cluster/admin/.member<n>.cfg	Member 1	Edits: done. Replace original file: done.
	Member 2	Edits: done. Replace original file: done.

### Example: filediffs Output

```

/tmp/FileDiffsOutput
-----
diff /.rhosts
    /.rhosts.work
-----
7a8,10
> gallo.zk3.dec.com
> ernest-ics0
> joey-ics0

```

```

=====
diff /etc/cfgmgr.auth
    /etc/cfgmgr.auth.work
-----
1c1
< rye.zk3.dec.com
---
> joey.zk3.dec.com

=====
diff /etc/hosts
    /etc/hosts.work
-----
40,41c40,41
< 10.0.0.1 swiss-ics0
< 10.0.0.2 rye-ics0
---
> # 10.0.0.1 swiss-ics0
> # 10.0.0.2 rye-ics0
42a43,48
>
> 16.140.112.209 gallo.zk3.dec.com gallo
> 16.140.112.238 ernest.zk3.dec.com ernest
> 16.140.112.176 joey.zk3.dec.com joey
> 10.0.0.1 ernest-ics0
> 10.0.0.2 joey-ics0

=====
diff /etc/hosts.equiv
    /etc/hosts.equiv.work
-----
50a51,54
>
> gallo.zk3.dec.com
> ernest-ics0
> joey-ics0

=====
diff /etc/networks
    /etc/networks.work
-----

=====
diff /etc/resolv.conf
    /etc/resolv.conf.work
-----

=====
diff /cluster/members/member1/boot_partition/etc/sysconfigtab
    /cluster/members/member1/boot_partition/etc/sysconfigtab.work
-----
429,431c429,431
< cluster_name=deli
< cluster_node_name=swiss
< cluster_node_inter_name=swiss-ics0
---
> cluster_name=gallo
> cluster_node_name=ernest
> cluster_node_inter_name=ernest-ics0

```

```

=====
diff /cluster/members/member1/etc/clu_alias.config
/cluster/members/member1/etc/clu_alias.config.work
-----

=====
diff /cluster/members/member1/etc/ifaccess.conf
/cluster/members/member1/etc/ifaccess.conf.work
-----

=====
diff /cluster/members/member1/etc/inet.local
/cluster/members/member1/etc/inet.local.work
-----

=====
diff /cluster/members/member1/etc/ntp.conf
/cluster/members/member1/etc/ntp.conf.work
-----
33c33
< peer rye-ics0 version 3
---
> peer joey-ics0 version 3

=====
diff /cluster/members/member1/etc/rc.config
/cluster/members/member1/etc/rc.config.work
-----
106c106
< HOSTNAME="swiss.zk3.dec.com"
---
> HOSTNAME="ernest.zk3.dec.com"
134c134
< IFCONFIG_0="16.140.160.121 netmask 255.255.255.0"
---
> IFCONFIG_0="16.140.112.238 netmask 255.255.255.0"
160c160
< CLUSTER_NET="swiss-ics0"
---
> CLUSTER_NET="ernest-ics0"

=====
diff /etc/gated.conf.member1
/etc/gated.conf.member1.work
-----
42c42
< host 16.140.160.124 interface 127.0.0.1;
---
> host 16.140.112.209 interface 127.0.0.1;
69c69
< export proto rip interface 16.140.160.121 {
---
> export proto rip interface 16.140.112.238 {
72c72
< 16.140.160.124 mask 255.255.255.255 metric 14;
---
> 16.140.112.209 mask 255.255.255.255 metric 14;

=====
diff /cluster/members/member2/boot_partition/etc/sysconfigtab

```

```

        /cluster/members/member2/boot_partition/etc/sysconfigtab.work
-----
424,426c424,426
< cluster_name=deli
< cluster_node_name=rye
< cluster_node_inter_name=rye-ics0
---
> cluster_name=gallo
> cluster_node_name=joey
> cluster_node_inter_name=joey-ics0

=====
diff /cluster/members/member2/etc/clu_alias.config
/cluster/members/member2/etc/clu_alias.config.work
-----

=====
diff /cluster/members/member2/etc/ifaccess.conf
/cluster/members/member2/etc/ifaccess.conf.work
-----

=====
diff /cluster/members/member2/etc/inet.local
/cluster/members/member2/etc/inet.local.work
-----

=====
diff /cluster/members/member2/etc/ntp.conf
/cluster/members/member2/etc/ntp.conf.work
-----
49c49
< peer swiss-ics0 version 3
---
> peer ernest-ics0 version 3

=====
diff /cluster/members/member2/etc/rc.config
/cluster/members/member2/etc/rc.config.work
-----
109c109
< CLUSTER_NET="rye-ics0"
---
> CLUSTER_NET="joey-ics0"
139c139
< XNTP_SERV2="swiss-ics0"
---
> XNTP_SERV2="ernest-ics0"
149c149
< HOSTNAME="rye.zk3.dec.com"
---
> HOSTNAME="joey.zk3.dec.com"
155c155
< IFCONFIG_1="16.140.160.122 netmask 255.255.255.0"
---
> IFCONFIG_1="16.140.112.176 netmask 255.255.255.0"

=====
diff /etc/gated.conf.member2
/etc/gated.conf.member2.work

```

```
-----
42c42
<      host 16.140.160.124 interface 127.0.0.1;
---
>      host 16.140.112.209 interface 127.0.0.1;
69c69
< export proto rip interface 16.140.160.122 {
---
> export proto rip interface 16.140.112.176 {
72c72
<      16.140.160.124 mask 255.255.255.255 metric 14;
---
>      16.140.112.209 mask 255.255.255.255 metric 14;
=====
```

## Comments and Questions

We value your comments and questions on the information in this document. Please mail your comments to us at this address:

[best\\_practices@zk3.dec.com](mailto:best_practices@zk3.dec.com)

## Legal Notice

Compaq, the Compaq logo, Tru64, and TruCluster are trademarks of Compaq Information Technologies Group, L.P. in the U.S. and/or other countries. UNIX is a trademark of The Open Group in the U.S. and/or other countries. All other product names mentioned herein may be trademarks of their respective companies.

Confidential computer software. Valid license from HP required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.