Chapter 1

Upgrading your system to UnixWare 7

Your UnixWare® 7 system contains many new and updated features and subsystems. However, much of the data from your SCO® OpenServer™ 5 or SCO UnixWare 2.1 operating system may, with some modification, be used on the new system.

The upgrade process entails saving configuration files and other important data (such as user names and associated passwords and groups) to media, from which they can be restored and modified to work on the UnixWare 7 system.

For some data, such as password files and Domain Name Service (DNS) files, the files will work as they did on your old operating system. Other data, such as MMDF mail configurations and SCO OpenServer printer configuration scripts, will not work on the UnixWare 7 system and must be re-written. In these cases, this guide describes how to determine and save your configuration, and provides pointers to additional information.

NOTE You are now reading the first version of this *Upgrade Guide*. By the time you view this guide from the UnixWare 7 CD-ROM, updated migration tools and documentation might be available for browsing and downloading from *http://www.sco.com/unixware7/documentation/*. Always check here first for the latest information.

Upgrading filesystems

This topic documents the filesystem differences between UnixWare 7 and its SCO UnixWare 2.X and SCO OpenServer Release 5 predecessors.

The recommended filesystem for UnixWare 7 is the Veritas Filesystem (VxFS). VxFS now supports filesystems up to 1TB in size. Files can be up to 1TB in length (2ˆ40) and sparse files may be of apparent length up to 2ˆ63 bytes in size. VxFS also supports UNIX95 filesystem semantics.

For a complete list of filesystem types supported by UnixWare 7, see the printed *UnixWare 7 System Handbook* or SCOhelp.

UnixWare 7 filesystems are configured for the first two disks on your system during initial system load (ISL), but can also be configured for additional disks later using **diskadd**(1M) or **disksetup**(1M).

NOTE **diskadd**(1M) limits you to 16 slices per disk; **disksetup**(1M) allows you to configure up to 184 slices per disk. Refer to the manual pages for further details.

For more information on adding and modifying filesystems after your system is installed, see the description of the **Filesystem Manager** in SCOhelp.

Upgrading SCO UnixWare 2.X filesystems

To migrate data on the primary hard drive from a SCO UnixWare 2.X system to UnixWare 7, you can either copy the data to a secondary hard disk or save the data to removable media, then restore the data after you install your new system.

To migrate data on the second hard drive, choose **Do not modify** when configuring the second hard disk during ISL. When you boot your system after installation, the system will recognize the filesystem slices on the second disk and create the relevant nodes for these filesystems. You can then use the **Filesystem Manager** to add these filesystems into */etc/vfstab*.

Upgrading SCO OpenServer Release 5 filesystems

To migrate data from SCO OpenServer Release 5 to UnixWare 7:

1. Back up the data to removable media, or to another system on the network.

   a. Change directories to the top level of the data you want to migrate. For example, to migrate user directories, you might enter:

   ```
   cd /usr
   ```

   b. Enter one of the following **cpio** commands:

   To archive to cartridge tape:

   ```
   find . -depth -print -follow | cpio -ocvdB -O /dev/rct0
   ```

   If your archive spans multiple tapes, you may also need to specify the block and volume sizes. See the manual page for **cpio**(M) for more information.

   When done creating the archive, skip to step 2.

   To archive to a file which can be transferred over the network:

   ```
   find . -depth -print -follow | cpio -ocvd > /tmp/name.cpio
   ```

   name identifies this **cpio** archive; in this example it might be **users**.

   c. Use **ftp** or another file transfer program to copy the file named in step 1b to another system on your network. You can then copy this file to your UnixWare 7 system once it is installed.

2. Install your UnixWare 7 system.

3. Restore the data.

   a. If it does not already exist, create the directory in which to extract the archive. For example, if */usr* does not exist, create it now with **mkdir**(1).

   b. If you copied the archive to another system in step 1c, copy it to the new system using **ftp** or another file transfer program.

   c. Use the **cpio** command to extract the archive.

   From cartridge tape:

   ```
   cpio -icdv -I /dev/rct0
   ```

Upgrading Mail and Messaging

SCO OpenServer 5, SCO UnixWare 2.X and UnixWare 7 all contain different mail transport systems with different methods of configuration. However, the feature sets are similar and most features do migrate easily. This topic discusses those features that feature prominently in the GUI or character based configuration tools. (The mail transport in UnixWare 7 is modelled after the SCO OpenServer one, which makes duplicating complex configuration a little easier from that direction.)

Mail folder formats are completely backwards-compatible from UnixWare 7 to both of the other systems, and so folders can be imported with ease.

The following major areas are addressed by this document:

- users' inboxes
- Mail and Messaging transport configuration
- aliases
- vacation notifications
- customized forwarding options
- virtual domains

Each of these is described according to whether the migration path is to UnixWare 7 from SCO OpenServer or from SCO UnixWare Release 2.X. In both cases, the text assumes that you are starting with a default configuration on UnixWare 7 and attempting to modify it to match your previous configuration.

Note that this topic describes how to preserve the simple configuration options that were provided in the configuration programs (SCO UnixWare Release 2.X provided a GUI-based configuration tool for *mailsurr*, while SCO OpenServer provided a GUI for MMDF and a command line script for **sendmail**).

Upgrading SCO UnixWare Release 2.X Mail and Messaging

This topic describes the migration path from SCO UnixWare Release 2.X to UnixWare 7 for various aspects of the Mail and Messaging system.

Preserving users' inboxes

User's inboxes in SCO UnixWare Release 2.X are in */var/mail*, and remain there for UnixWare 7. It is therefore sufficient to copy these files to the new machine. No data conversion of these files is required.

NOTE In the case where mailboxes are in users' home directories, these must be restored and the INBOX location configured to point there.

Using the **Mail Manager** to change the inbox location to users' home directories, change the "Users' INBOX Location" setting (within the "Folder Configuration" container) to the appropriate setting.

Preserving transport agent configuration

The SCO UnixWare Release 2.X **UNIX Mail Setup** configuration manager specifies several items. This topic explains how to duplicate similar behavior on UnixWare 7.

In the "Basic" configuration category, the following fields are displayed:

- Smarter Host

  If messages are to be routed to a "smarter host", this field contains the identifier of the desired host, and the "Route All Msgs to Smarter Host?" prompt is set to **No**.

  This feature can be duplicated in UnixWare 7 using the **Settings ➪ Bad Host Forwarding/Delay** menu option to enable forwarding to the smart host. This has the effect of sending SMTP mail addressed to an unknown host to that specified host.

  If the "Route All Msgs to Smarter Host?" prompt is set to **Yes**, then see the description of the "Route All Msgs to Smarter Host?" field for how to duplicate the functionality in UnixWare 7.

- Cluster Name

  This field is analogous to the "Mail Comes From" field in the UnixWare 7 "Basic Configuration" category except that a blank value is equivalent to setting the "Mail Comes From" value to match the host name.

- Route All Msgs to Smarter Host?

  If this prompt is set to **Yes**, enable the badhost channel. Additionally, open the "Mail Delivery Channels" category in the UnixWare 7 Mail Manager tool, then open the SMTP channel, and change the forwarding host to the "smarter host" name. This has the effect of sending all SMTP mail to the smarter host, for both known and unknown hostnames.

- Route local messages through MHS?

  A direct MHS mail gateway is not supported by UnixWare 7, so there is no analog to this feature.

- Log Messages?

  UnixWare 7 **sendmail** has a variety of logging features, some of which are turned on by default. For more information on these, see "sendmail operations" in SCOhelp. In general, **sendmail** logging goes to **syslogd**, which is the standard logging place.

The "Advanced" configuration section of the SCO UnixWare Release 2.X **UNIX Mail Setup** tool contains several parameters that have no GUI-based analogs in UnixWare 7's **sendmail**. **sendmail** does support all of those features, however, a hand edit of */etc/sendmail.cf* being necessary to precisely duplicate the SCO UnixWare Release 2.X behavior. The following is a description of what you need to do to duplicate the configuration on UnixWare 7:

- The default UnixWare 7 configuration has all of the headers in the SCO UnixWare Release 2.X advanced section enabled. To change this, you must look for a line in */etc/sendmail.cf* that starts with the letter "H" and references the header line in question. Merely remove (or comment out) the line in question to duplicate an "off" button in the SCO UnixWare Release 2.X configuration. In general, precisely duplicating header line generation is not necessary and SCO recommends leaving */etc/sendmail.cf* alone in this case.
- *mailsurr* configuration variables do not carry forward to **sendmail**, so no analog is available.
- Debug levels are available in **sendmail**, but only from the command line and not via a preconfigured value. For more on **sendmail** debug levels, see "sendmail operations" in SCOhelp or the highly recommended O'Reilly book[1] for a truly exhaustive account of **sendmail**'s numerous debugging options.

Preserving vacation notifications

Users of SCO UnixWare Release 2.X will have configured their vacation notifications from **dtmail**, the graphical mail user agent from CDE. Under UnixWare 7, they should use the **Vacation Manager**, accessible from the CDE desktop under the Mail panel.

The following SCO UnixWare Release 2.X files must be migrated:

- *~/.forward*

  SCO UnixWare Release 2.X users will have their vacation notification enabled by their *~/.forward* file. While this file may be brought as-is for use on a UnixWare 7 system, SCO recommends conversion of *~/.forward* to

[1] *sendmail*, Bryan Costales with Eric Allman, O'Reilly & Associates, Inc.

~/.maildelivery, in order to retain compatibility with the UnixWare 7 **Vacation Manager**. To enable vacation notification, an entry in *~/.forward* of the form

```
"|/usr/ucb/vacation"
```

has an equivalent *~/.maildelivery* entry of

```
*       -       pipe    R       vacation
```

For a full discussion of *~/.forward* and *~/.maildelivery* syntax, see the *forward*(4) and *maildelivery*(4) manual pages.

- *~/.vacation.msg*

  SCO UnixWare Release 2.X users will have to convert their vacation notification message from the file *~/.vacation.msg* into two corresponding UnixWare 7 files:

  – *~/tripsubject*

  – *~/tripnote*

  The body of the message in *~/.vacation.msg* should be transferred to *~/tripnote*, while the Subject header text should be put into *~/tripsubject*.

  > **NOTE** Only the Subject header text, and not the actual `Subject:` header itself, should be in *~/tripsubject*. Also, while other headers may be specified in *~/.vacation.msg*, this is not possible in the UnixWare 7 *~/tripnote* and *~/tripsubject* files.

Preserving aliases

If your SCO UnixWare Release 2.X system uses a *mailsurr* file for mail configuration (you do not use **sendmail** as mail transport agent), then the following files must be migrated to your UnixWare 7 system:

- */etc/mail/namefiles*

  On SCO UnixWare Release 2.X systems, all alias files are specified in the file */etc/mail/namefiles*. Entries in this file take the form of pathnames to files or directories. If to a directory, then files within this directory are named after the alias, and the file contents specify the actual mail address or addresses. By default, */etc/mail/namefiles* specifies the file */etc/mail/names* as the aliases file, and the directory path */etc/mail/lists*.

  Multiple alias files may also be specified for the UnixWare 7 mail system, via the **Mail Manager**. Any files specified in the SCO UnixWare Release 2.X */etc/mail/namefiles* file must be added to the list of alias files displayed in the **Mail Manager**, which will update the *sendmail.cf* configuration file appropriately. By default, these files are located in */etc/mail* (and the default aliases file on UnixWare 7 is */etc/mail/aliases*), but any path may be specified. These files must be converted to a **sendmail** format aliases file (see the description of */etc/mail/names*).

For each entry which is a directory name in */etc/mail/namefiles*, the files under that directory must be converted into **sendmail** style aliases, and then put into an existing or new alias file. For example, suppose */etc/mail/namefiles* contains the directory */etc/mail/lists*, and the file */etc/mail/lists/engr* exists with the following contents:

```
fred
barney
```

Then the following alias can be created in an alias file:

```
engr: fred, barney
```

- */etc/mail/names*

  The default aliases file on SCO UnixWare Release 2.X systems is */etc/mail/names*. This file, and any alias file defined in */etc/mail/namefiles*, must be converted to **sendmail** format for use on UnixWare 7 systems.

  Conversion to **sendmail** alias file format:

  – Comments are lines beginning with a hash sign (#) in both formats, and do not need to be altered.

  – SCO UnixWare Release 2.X alias entries have white-space separated tokens of the following form:

  ```
  name addr1 addr2 addr3
  ```

  These must be changed to the form:

  ```
  name: addr1, addr2, addr3
  ```

  The main difference is the colon separator between alias name and recipient list, and the use of commas to separate addresses. (The white-space is optional in the UnixWare 7 format.)

  – In SCO UnixWare Release 2.X alias files, lines may be continued by placing a backslash (\) at the end of the line. In a **sendmail** format alias file, a continuation line is created by indenting the line with white-space. So for example, an alias such as

  ```
  name addr1 addr2 addr3\
  addr4
  ```

  would be converted to

  ```
  name: addr1, addr2, addr3,
    addr4
  ```

  Once alias files have been converted and put onto your UnixWare 7, their corresponding databases must be created using the **newaliases**(1M) utility.

If the BSD Compatibility Package was installed on your SCO UnixWare Release 2.X system, and you are now using **sendmail** as the mail transport agent, then the following files should be migrated to the UnixWare 7 system:

- */etc/ucbmail/aliases*

  The location of the aliases file is defined by the **OA** option line in the **sendmail** configuration file */etc/ucbmail/sendmail.cf*. The default location is */etc/ucbmail/aliases*. However, you should check the *sendmail.cf* file. This aliases file may be brought forward to the UnixWare 7 system without modification, and the location specified by the **Mail Manager**. However, you must ensure that aliases which use redirection to files or pipes have corresponding directory paths or programs which exist. The aliases database file(s) should then be created using the **newaliases**(1M) utility.

- *~/.forward*

  User-defined *~/.forward* files may be migrated to UnixWare 7 systems (with the modification discussed in "Preserving vacation notifications" (page 6)) to use the vacation notification feature.

Preserving custom forwarding

Users on SCO UnixWare Release 2.X systems had the ability to configure mail forwarding via the **/bin/mail** program's **-F** option. On UnixWare 7, **/bin/mail** is now equivalent to **mailx**(1), so the **-F** option is not available. However, similar functionality is supplied by the use of *~/.forward* and *~/.maildelivery*. The following lists the SCO UnixWare Release 2.X **/bin/mail** forwarding options, and their equivalent in either *~/.forward* or *~/.maildelivery*.

- **mail -F *recipient*** . . .

  – *~/.forward*

    In a *~/.forward* file, add each recipient you wish to forward your mail to. Separate recipients with commas if they all appear on the same line, or put a single recipient per line. For example:

    ```
    recipient1, recipient2@thishost.com, recipient3
    recipient4
    recipient5
    ```

    This causes your mail to be forwarded to five other addresses.

  – *~/.maildelivery*

    In a *~/.maildelivery* file, use the *maildelivery*(4) pipe (|) action to pipe the message to **sendmail** for redelivery. For example, to forward your mail to *recipient1* and *recipient2*, create the following line in *~/.maildelivery*:

    ```
    *   -   |   A   /usr/lib/sendmail recipient1 recipient2
    ```

Note that this does not create any Recent-headers, so the message will appear in the forwarded recipient's mailbox with only the original From and To addresses.

- **mail -F '|/usr/bin/sh -c "shell_command_line"'**

  – *~/.forward*

  The SCO UnixWare Release 2.X **/bin/mail -F** option allowed recipients to be commands (called "Personal Surrogates") to which the mail message would be piped.

  Similar functionality is available in *~/.forward* with the following syntax:

  ```
  "|shell_command_line"
  ```

  The quotes are necessary if ***shell_command_line*** contains white space; that is, if the command has arguments. For example, to set automatic replies for extended absences, you could add the following entry to your *~/.forward* file:

  ```
  \user, "|/usr/bin/vacation user"
  ```

  The recipient "\user" tells **sendmail** to also deliver the message to the user's mailbox, and skip additional alias transformations. This may be used to replace the ">|" prefix, which allows SCO UnixWare Release 2.X mail to append the message to the user's mailbox, in addition to executing the pipe command ("Post-processed Personal Surrogates").

  > **NOTE** **sendmail** sorts all addresses in *~/.forward* and deletes duplicates before delivering to any of them, therefore ensure that programs in *~/.forward* are unique. This can usually be accomplished through the use of the program's command-line arguments.

  – *~/.maildelivery*

  Piping a message to a program can be accomplished in a *~/.maildelivery* entry which uses the pipe (|) action. Using the same example, the equivalent *~/.maildelivery* entry would be as follows:

  ```
  *   -   |   R   /usr/bin/vacation
  ```

  The "R" result specified in the fourth column specifies that the message is not to be considered delivered by the action, causing the message to be delivered into the user's mailbox as well. This is the equivalent of the ">|" prefix for the SCO UnixWare Release 2.X mail Post-processed Personal Surrogate functionality.

> **NOTE** There is no *~/.forward* or *~/.maildelivery* equivalent of the **%c** or **%S** keys used to textually substitute the value of the `Content-Type:` header, or `Subject:` header lines in mail **-F** pipe program recipients. The **%R** key, which represents the return path to the message sender, may be replaced with the $(***sender***) variable in *~/.maildelivery*. There is no equivalent syntax for *~/.forward*.

Those SCO UnixWare Release 2.X systems with the BSD Compatibility Package installed, and which use **sendmail** as the mail transport agent, will be able to transfer users *~/.forward* files to UnixWare 7 systems without modification, as long as those programs, files, and recipients referenced in the *~/.forward* files are accessible.

Upgrading SCO OpenServer Release 5 Mail and Messaging

This topic describes the migration path from SCO OpenServer Release 5 to UnixWare 7 for various aspects of the Mail and Messaging system.

Preserving users' inboxes

In SCO OpenServer Release 5, users' inboxes are in */usr/spool/mail*, and in */var/mail* in UnixWare 7. There is a symbolic link to */usr/spool/mail* from */var/mail*, and it is therefore sufficient to copy these files to the new machine. No data conversion of these files is required.

> **NOTE** In the case where mailboxes are in users' home directories, these must be restored and the INBOX location configured to point there.

Using the **Mail Manager** to change the inbox location to the users' home directories, change the "Users' INBOX Location" setting (within the "Folder Configuration" category) to the appropriate setting.

Preserving transport agent configuration

SCO OpenServer Release 5 supports two mail transport agents: MMDF and **sendmail**. Preserving these configurations is described separately:

- Preserving SCO OpenServer MMDF configurations.

  UnixWare 7 **sendmail** has been modelled after MMDF to make the transition easier: even advanced features such as the domain table and channel support are supplied. Not all of these are covered in this topic, but after you become familiar with UnixWare 7's **sendmail** configuration, it should become clear how to port even advanced MMDF configurations to UnixWare 7's **sendmail**.

  The primary screen of SCO OpenServer Release 5's **MMDF Configuration Manager** allows you to configure the host name, enable SMTP and UUCP, enable domain hiding, and enable or disable the name service methods.

- The entry in the "Host Name" field under MMDF should be exactly the same as that given in the "Host Name" field in the UnixWare 7 manager's "Basic Configuration" category.
- Selecting entries in "Networks" amounts to adding channels for each network desired. SMTP is on by default in UnixWare 7, and does not need to be disabled even if no network card is in the machine.

  The UnixWare 7 **Mail Manager** enables UUCP by selecting the **Settings ➪ UUCP Enable/Disable** menu option.
- "Format for Mail User Addresses" in MMDF is analogous to "Mail comes from" in the "Basic configuration" category of the UnixWare 7 manager. Duplicate the domain setting (the right hand part of the mail address) in the UnixWare 7 configuration.
- "Name Service Lookup Methods" are automatic in UnixWare 7, and do not require configuring.

The additional options under "Forwarding" in the **MMDF Configuration Manager** are as follows:

- "Unknown Hosts Forwarding" creates a badhost channel in MMDF. The badhost channel in UnixWare 7 functions equivalently and may be enabled by selecting the **Settings ➪ Bad Host Forwarding/Delay** menu option and entering the host name from the MMDF configuration to preserve this functionality. If the MMDF setting is "Return mail for unknown hosts to sender", then do nothing (that is, do not create a badhost channel), as this is the default behavior in UnixWare 7.
- "Unknown User Forwarding" creates a baduser channel in MMDF. The baduser channel in UnixWare 7 functions equivalently and may be enabled by selecting the **Settings ➪ Baduser Enable/Disable** menu option and entering the host name from the **MMDF Configuration Manager** to preserve this functionality. If the MMDF setting is "Return mail for unknown users to sender", then do nothing (that is, do not create a baduser channel), as this is the default behavior in UnixWare 7.

SCO UnixWare Release 2.X MMDF mailbox locations may be either */usr/spool/mail* or the users' home directories. UnixWare 7 implements exactly the same feature, which is accessed via the "Folder Configuration" category in the UnixWare 7 mail configuration tool.

The "Redirection" option in the MMDF configuration calls up the aliases editor. The same functionality exists in UnixWare 7 and is accessed by opening the "Alias Files and Maps" category and selecting the alias file entry. Then select the "Edit" facility in the dialog box and the alias editor is started for the system-wide mail alias file.

SCO recommends importing alias files rather than typing in the aliases by hand into the alias editor.

- Preserving SCO OpenServer **sendmail** configurations.

  The **sendmail** configuration is obtained and/or modified by running **mkdev cf** on SCO OpenServer. It contains several options, each of which is described here. In addition, SCO OpenServer **sendmail** supports some of MMDF's features, notably configurable INBOX location which is also described here.

  – UUCP connections are enabled and disabled in the UnixWare 7 mail manager via the **Settings ➪ UUCP Enable/Disable** option. Manual editing is not allowed in the GUI, as UUCP's *Systems* file is automatically parsed into the mail configuration. If you wish to have a private map (in other words, you do not have all of the systems in UUCP *Systems* file used as mail destinations), merely change the name of the UUCP channel's table file and edit it yourself.

  – The manual domain entry for **mkdev cf** is analogous to the "Mail Comes From" entry in the "Basic Configuration" category of the UnixWare 7 mail configuration utility. You may do domain-hiding (make one machine pretend to be part of a larger entity) if you like.

  – NIS support is automatic in UnixWare 7 if NIS is installed and configured.

  – Alternate host names may be entered and they function the same on UnixWare 7 with the exception that you do not have to enter the short name of a machine into the alternate name list: instead, the short name is added automatically.

  – UnixWare 7 does not have an option to automatically forward UUCP style addresses to the machine *uunet*.

  – UnixWare 7 does have the ability to forward UUCP traffic destined for a host (which must be known to use UUCP) to a UUCP gateway. UUCP has its own configuration file in */etc/uucp/Systems*, which keeps a list of all hosts known to UUCP. When a UUCP channel is present, the **Mail Manager** will automatically update the mail system's internal tables to match the UUCP configuration. Thereafter, any mail addressed to a host in the UUCP table will be forwarded to the UUCP system for delivery.

    This feature is enabled by using the **Settings ➪ UUCP Enable/Disable** menu option to enable the UUCP channel. Then edit the "Forward all mail to" entry in the UUCP channel to add the UUCP gateway name.

    Note that, if the UUCP gateway is on the Internet, you would use the same strategy with a custom SMTP channel (that is, leave the DNS one alone, and create a new one for this option). To do this, create a new channel with channel program as SMTP, table type as UUCP *Systems* file, and the channel name something like *uucpgate*. Then edit the "Forward all mail to" option to point to the UUCP gateway.

- A “smart host” is a host to which all mail addresses containing an at-sign (@) are sent. The intent is for machines that do not have access to a name service to forward their mail traffic to a machine that does have such access.

  This feature can be be duplicated on UnixWare 7 by configuring the default SMTP channel to have the “Forward all mail to” option pointing at the smart host. Also, add a badhost channel (using the **Settings ➪ Bad Host Forwarding/Delay** option) that also points to the smart host.

  This will cause all known (via the SMTP channel) and unknown (via the badhost channel) hosts to be resolved to the smart host.

- The UnixWare 7 configuration tool does not directly support adding an X.400 gateway product to the system. However, adding a new SMTP channel with a list of hosts to be forwarded to the X.400 gateway is possible.

Preserving virtual domain support

The SCO OpenServer 5.0.4 **sendmail** supports virtual domains under POP. UnixWare 7 supports virtual domains and adds IMAP support as well.

SCO OpenServer used virtual users (fake users known only to the POP server), whereas UnixWare 7 uses real users and exports them via a user map into the virtual domains.

Refer to “The Virtual Domain User Manager” in SCOhelp for details of how to set up virtual domains and export users to those domains. The virtual domain POP users’ inboxes on SCO OpenServer are contained in the directory */usr/internet/ip/**ip_addr**/sco_mail/spool*, where ***ip_addr*** is the IP address of a virtual domain. The inboxes are named after the virtual user names.

These mailboxes do not need to be reformatted, but do need to be moved into the standard place for inboxes (by default, */var/mail*) on UnixWare 7.

In UnixWare 7, virtual users’ password information is contained in */var/internet/ip/**ip_addr**/passwd*. This file is in the old style UNIX */etc/passwd* format where the encrypted password is contained in the same file as the user ID information. SCO OpenServer normally splits out the encrypted password into */etc/shadow*, and places an “x” in that field in */etc/passwd*.

Preserving vacation notifications

For additional information on the vacation notification features, see “The Vacation Notification Manager” in SCOhelp and the **vacation**(1) and *mail-delivery*(4) manual pages.

- Migration from OpenServer MMDF MTA:

  The following SCO OpenServer Release 5 files must be migrated:

  – *~/.maildelivery*

    OpenServer users who have MMDF as their mail transport agent must make the following change in their *~/.maildelivery* file:

    ```
    *       -       pipe    R       rcvtrip
    ```

    becomes

    ```
    *       -       pipe    R       vacation
    ```

  – *~/.alter_egos*
    ~/tripnote
    ~/triplog

    If users have any of the above files on SCO OpenServer Release 5, they may be brought as-is onto a UnixWare 7 system. Their functionality is identical to that on OpenServer.

- Migration from SCO OpenServer Release 5 **sendmail** MTA:

  OpenServer users who have **sendmail** as their mail transport agent should convert their *~/.forward* and *~/.vacation.msg* files as described for migrations from SCO UnixWare Release 2.X.

Preserving aliases

- If you are using MMDF as your mail transport agent:

  All alias files in SCO OpenServer Release 5 are specified by the **ALIAS** keyword and ***table*** parameter in */usr/mmdf/mmdftailor*. For example, the entry

  ```
  ALIAS table=aliases
  ```

  specifies an alias file named *aliases*. The directory under which the alias files are located is specified in *mmdftailor* by the **MTBLDIR** keyword. For example:

  ```
  MTBLDIR=/usr/lib/mail/aliasfiles
  ```

  In this case, the full pathname to the alias file would be */usr/lib/mail/aliasfiles/aliases*. By default, if you do not have **MTBLDIR** defined in the *mmdftailor* file, the directory is defined to be */usr/mmdf/table*.

  All alias files may be specified for your **sendmail** configuration on UnixWare 7 systems by using the **Mail Manager**. In addition, the files must be converted to **sendmail** alias file format:

  – Comments are lines beginning with a hash sign (#) in both formats, and do not need to be altered.

- Simple aliases of the form

  ```
  name: addr1, addr2, addr3
  ```

  are used in both formats, and will not need modification.

- Redirection of a message to a file in an SCO OpenServer Release 5 MMDF-style alias uses the following syntax:

  ```
  alias:login/file
  ```

  For example, "foobar:dpk/foobar" would cause user and group IDs to be set to those of the user *dpk*, and the text of the message to be appended to the file *foobar* in *dpk*'s default login directory. Similarly, "foobar:dpk//tmp/foobar" would do the same for file */tmp/foobar*.

  On UnixWare 7, redirection of a message to a file is specified differently. If any addresses to the right of the colon in the alias list begin with a slash (/) character, **sendmail** interprets the address as the name of a file, and appends the mail message to that file. The filename must be a full pathname, and there is no notion of being able to specify a file relative to a user's default login directory. So, using the previous example, if the home directory of user *dpk* is known to be */home/dpk*, then the alias "foobar:dpk/foobar" would be converted as follows:

  ```
  foobar: /home/dpk/foobar
  ```

  The user and group ID of the file is specified by other **sendmail** mail delivery agent options. See "sendmail operations" in SCOhelp for more information.

- Redirection of a message to a pipe is available for use with MMDF in SCO OpenServer Release 5. The following would cause a message to be passed into a UNIX pipe (see **pipe**(2)) with user ID and group ID set to those of the user news:

  ```
  news-inject:news|/usr/lib/news/uurec
  ```

  In UnixWare 7, **sendmail** format alias files, a program address may take the following forms:

  - |prg
  - "|prg args"
  - |"prg args"

  The ***prg*** is the full path of the program to be run. Note that if command-line arguments are needed for the program, they must follow ***prg***, and the entire expression must be quoted (the leading quotation mark may either precede or follow the |).

  So, converting the previous example to **sendmail** alias file format would yield:

  ```
  news-inject:|/usr/lib/news/uurec
  ```

Again, the OpenServer specification of the user ID and group ID by the "news" prefix is not allowed for UnixWare 7. **sendmail** uses other methods for this functionality.

- OpenServer, MMDF-style aliases also allow you to specify the value-part of an entry as a filename, so that the actual value is taken from the file. There are two possible notations for this:

  1. By having left-angle bracket (<) precede the value specification. For example:

     ```
     mother: < /etc/mmdf/mother_list@udel-relay.arpa
     ```

  2. By using a data type with value ":include:". For example:

     ```
     mother: :include: /etc/mmdf/mother@udel-relay.arpa
     ```

  In both cases, the *@HOST* is optional.

  On UnixWare 7 systems, **sendmail** does allow a special notation in the right-hand side of an alias to read its list of recipients from an external file. The format is as follows:

  ```
  aliasname: :include:/path
  ```

  The expression ":include:" must appear exactly as shown, but optional white space is allowed between the ":include:" and "/path". The "/path" is the full pathname of a file containing a list of recipients. However, you may not specify an *@HOST* as is possible for SCO OpenServer Release 5. See "sendmail operations" in SCOhelp for more information concerning ":include:" lists.

- If you are using **sendmail** as your MTA:

  - */usr/lib/mail/aliases*

    The location of the aliases file is defined by the **OA** option line in the **sendmail** configuration file */usr/lib/sendmail.cf*. The default location is */usr/lib/mail/aliases*. However, you should check the *sendmail.cf* file. This aliases file is compatible with the UnixWare 7 system without modification, and the name and location may be specified by the **Mail Manager**. However, you must ensure that aliases which use redirection to files or pipes have corresponding directory paths or programs which exist. The aliases database file(s) should then be created using the **newaliases**(1M) utility.

  - *~/.forward*

    User-defined *~/.forward* files may be brought forward to UnixWare 7 systems (with the modification discussed in "Preserving vacation notifications" (page 14)) to use the vacation notification feature.

Preserving custom forwarding

On SCO OpenServer Release 5 systems using MMDF as mail transport agent, users' *~/.maildelivery* files are compatible with UnixWare 7, and may be transferred with little modification. However, the following programs popular with use in *~/.maildelivery* on OpenServer are not available in UnixWare 7:

- */usr/bin/rcvalert*
- */usr/bin/rcvfile*
- */usr/bin/rcvprint*
- */usr/bin/rcvtrip*
- */usr/bin/resend*

Use of */usr/bin/vacation* in *~/.maildelivery* to replace *rcvtrip* is discussed in "Preserving vacation notifications" (page 14).

On OpenServer systems that use **sendmail** as the mail transport agent, users' *~/.forward* files are also compatible with UnixWare 7 and may be transferred with little modification. Users must simply ensure that programs, files, and recipients referenced in their *~/.forward* files are accessible. See also "Preserving vacation notifications" (page 14).

Upgrading users and groups

Before installing UnixWare 7, take a copy of your */etc/passwd*, */etc/shadow*, and */etc/group* files from the system to be upgraded and store them in a safe place.

After you have installed UnixWare 7, perform the following steps as *root*,

1. If the system has been installed with the audit package, the audit package must be removed before proceeding. This procedure will *not* work with the audit package installed.
2. Make copies of your */etc/passwd*, */etc/shadow*, and */etc/group* files in case something goes wrong.

3. After the installation, the */etc/passwd* file will look something like this.

```
root:x:0:3:0000-Admin(0000):/:/sbin/sh
daemon:x:1:12:0000-Admin(0000):/:
bin:x:2:2:0000-Admin(0000):/usr/bin:
sys:x:3:3:0000-Admin(0000):/:
adm:x:4:4:0000-Admin(0000):/var/adm:
uucp:x:5:5:0000-uucp(0000):/usr/lib/uucp:
mail:x:6:6:Mail Processes:/etc/mail:
nuucp:x:10:10:0000-uucp(0000):/var/spool/uucppublic:/usr/lib/uucp/uucico
nobody:x:60001:60001:uid no body:/:
noaccess:x:60002:60002:uid no access:/:
lp:x:7:9:0000-LP(0000):/var/spool/lp:/usr/bin/sh
listen:x:37:4:Network Admin:/usr/net/nls:/usr/bin/ksh
mhsmail:x:61:6:MHS Admin Processes:/var/spool/smf:/usr/bin/ksh
owner:x:101:1:owner:/home/owner:/usr/bin/ksh
```

   These entries should remain intact. Below the line beginning owner, lines from the */etc/passwd* file of the upgraded system should be added. This is a cut and paste exercise with your favourite editor.

4. After the installation, the */etc/shadow* file will look something like this:

```
root:nhQ.Mx5msjFzg:10141::::::
daemon:NP:6445::::::
bin:NP:6445::::::
sys:NP:6445::::::
adm:NP:6445::::::
uucp:NP:6445::::::
mail:NP:6445::::::
nuucp:NP:6445::::::
nobody:NP:6445::::::
noaccess:NP:6445::::::
lp:*LK*:::::::
listen:*LK*:::::::
mhsmail:*LK*:::::::
owner:wjMKLa1PCZsCs:10140::::::
```

   These entries should remain intact. Below the line beginning owner, lines from the */etc/shadow* file of the upgraded system should be added. This is a cut and paste exercise with your favourite editor.

5. After the installation, the */etc/group* file will look something like this:

   ```
   root::0:root
   other::1:root
   bin::2:root,bin,daemon
   sys::3:root,bin,sys,adm
   adm::4:root,adm,daemon
   uucp::5:root,uucp
   mail::6:root
   tty::7:root,adm
   audit::8:root
   nuucp::10:root,nuucp
   daemon::12:root,daemon
   cron::23:root
   dtadmin::25:root
   priv::47:root
   nobody::60001:
   noaccess::60002:
   lp::9:root,lp
   dos::100:
   ```

   These entries should remain intact. Below the line beginning owner, lines from the */etc/group* file of the upgraded system should be added. This is a cut and paste exercise with your favourite editor.

6. Execute **creatiadb**(1M).
7. Re-install the audit package if required.

Upgrading locales

Locale configuration is forced during UnixWare 7 installation — you must set the locale as you install. More choices have been added to both locale and keyboard definitions in UnixWare 7, which is also backwards-compatible with SCO UnixWare Release 2.X and SCO OpenServer Release 5.

You can also set the locale after installation on a system-wide basis using the SCOadmin **International Settings Manager** or on a per-user basis using the SCOadmin **Account Manager**.

Upgrading SCO UnixWare Release 2.X locales

On your SCO UnixWare Release 2.X system, view the file */etc/default/locale*. This contains the system locale, and consists of a vairable (**LANG**) and its setting, for example `LANG=C`. Record this locale so you have access to it during UnixWare 7 installation.

When you install UnixWare 7 and are prompted for the locale, you will select a locale that appears in spelled-out form; for example, the locale "C" corresponds to "C (English)". You can also select a keyboard that maps to that particular locale.

Upgrading SCO OpenServer Release 5 locales

On your SCO OpenServer Release 5 system, start the SCOadmin **International Settings Manager** to view the current locale. The locale is highlighted when you enter the manager; it is listed in the "Language" selection box. For example, your locale might be "en_US". Record this locale so you have access to it during UnixWare 7 installation.

When you install UnixWare 7 and are prompted for the locale, you will select a locale that appears in spelled-out form; for example, the locale "en_US" corresponds to "English for United States". You can also select a keyboard that maps to that particular locale.

Upgrading video adapters

UnixWare 7 supports a large number of video adapters including those supported under SCO UnixWare 2.X and SCO OpenServer Release 5. In addition, UnixWare 7 provides the **vesa** X server driver. This generic driver can operate any new video card that honors the VESA BIOS interface, and is useful in supplying high resolution support to video cards that do not have a specific accelerated driver. For more information on this feature, including performance implications, see SCOhelp on your installed UnixWare 7 system.

Most video adapters are automatically configured when you install your UnixWare 7 system. However, you should record your video configuration from your previous operating system in case:

- UnixWare 7 cannot automatically configure the adapter
- UnixWare 7 incorrectly configures the adapter
- you incorrectly configure the adapter manually and need to restore the default configuration

To manually configure a video adapter in UnixWare 7, use the SCOadmin **Video Configuration Manager**.

Upgrading SCO UnixWare 2.X video adapters

On your SCO UnixWare 2.X system, view or print the file */usr/X/defaults/Xwinconfig*. This file contains keyboard, video adapter, and monitor definitions. The important lines are shown here:

```
       chipset = GD54xx              # video chipset
         model = "GD54xx"            # the core drawing lib for this class
    vendor_lib = gd54xx_256.so.2     # chip specific drawing lib
  virtual_size = 1024x768            # actual Frame Buffer size
        vendor = "Cirrus Logic - Generic"             # vendor name
```

From this information, you can determine that the configured video adapter is a Cirrus Logic GD54xx series model configured for 1024x768 mode.

Record this information, then (if auto-detection or auto-configuration fails) use it to configure your adapter on UnixWare 7 using the SCOadmin **Video Configuration Manager**.

Upgrading SCO OpenServer Release 5 video adapters

To obtain information about the currently configured adapter, run the **Video Configuration Manager**.

The display at the top of the screen lists the name of the adapter, any configured monitor, and the resolution.

Record this information, then (if auto-detection or auto-configuration fails) use it to configure your adapter on UnixWare 7 using the SCOadmin **Video Configuration Manager**.

Troubleshooting video configuration

If you install your UnixWare 7 system and find that your video adapter is incorrectly configured, or you want to modify configuration, try the following.

To run your system in a safe video mode

Enter **/usr/bin/X11/setvideomode -stdvga**. This This sets IBM VGA 640x480-16 mode, which is almost always safe for any adapter.

To restore the adapter's default configuration

Enter **/usr/bin/X11/setvideomode -default**.

Do this if initial auto-configuration worked well enough to get the video working, but you manually configured the adapter to a different setting and lost the use of the video adapter.

This **-default** option restores the settings to initial auto-configuration defaults.

To determine the video adapter in the system

Enter **/usr/bin/X11/VideoHelp**.

This command lets you know what video adapter is present on your system.

Upgrading documentation

SCOhelp, based on Netscape Navigator Gold™ 3, is used to view online topics and other information in UnixWare 7. This version of SCOhelp is based on Hyper Text Markup Language (HTML) as was the SCO OpenServer SCOhelp. However, significant enhancements and changes have been made, including the addition of the Verity search engine and frames. While SCO OpenServer Release 5 help files can be viewed with the new SCOhelp, they cannot be migrated directly to the UnixWare 7 system.

There is no compatibility between either version of SCOhelp and the **dtext** browser found in SCO UnixWare 2.X.

Because all documentation carried forward from both predecessor operating systems has been converted to the new format, most users will not need to migrate any online documentation.

For those who want to do so, or who want to create their own help files, extensive online help describes the SCOhelp architecture and provides step-by-step instructions for integrating your documentation into the view. See SCOhelp for more details.

The **man**(1) command on UnixWare 7 is similar to predecessor commands, but also allows for browsing of manual pages in HTML format. **nroff** or ascii manual pages can be migrated from other operating systems to UnixWare 7. Again, see SCOhelp for information on integrating your manual pages into SCOhelp.

Upgrading networking

This guide covers the migration of these interfaces and protocols:

- Network adapters (this page)
- TCP/IP (this page)
- NetBIOS (page 36)
- NetWare (page 35)
- Point-to-Point Protocol (PPP) (page 37)

Upgrading network interface configuration

Configuration of network interface hardware in UnixWare 7 can be done at install time (for one network adapter only), or it can be performed at a later time by using the **Network Configuration Manager** as in SCO OpenServer Release 5.

Note the configuration details of the network adapter hardware (IRQ, I/O address range, memory address range, DMA channel) in your system so that you can configure your UnixWare 7 system with these values. For SCO UnixWare 2.1, note the details displayed by **niccfg**. For SCO OpenServer Release 5, note the details displayed by the **Network Configuration Manager**.

Upgrading TCP/IP

In UnixWare 7, TCP/IP is configured over a network interface using the **Network Configuration Manager** as in SCO OpenServer Release 5. You should note the hostname, domain name, IP address, netmask, broadcast address and frame type of the existing network interfaces so that you can configure these on your UnixWare 7 system. To obtain these values, run the **Network Configuration Manager** in SCO OpenServer Release 5, or run **/etc/inet/menu** in SCO UnixWare 2.1.

Files to migrate

You may need to copy over the file */etc/hosts* from the SCO UnixWare 2.1 or SCO OpenServer Release 5 system. This contains information about the hostnames and IP addresses of *localhost* and other systems. It is recommended that you merge this information with the existing */etc/hosts* file to avoid accidentally removing the *localhost* entry.

You may also need information from the */etc/tcp* file on an SCO OpenServer Release 5 system such as the IP address of a statically configured default router. Look for an entry such as:

```
/etc/route add default gateway
```

In SCO UnixWare 2.1, the **/etc/inet/menu** command shows the IP address of the default router on the local network to which the interface is attached. (It also displays information about DNS name servers that should be used. See "Upgrading DNS" (page 28) for more information.) In UnixWare 7, use the **Network Configuration Manager** to configure the default router that should be used with TCP/IP.

The */etc/tcp* file on an SCO OpenServer Release 5 system and the */etc/inet/config* file on an SCO UnixWare 2.1 system also contain information about which TCP/IP services should be configured in the */etc/inet/config* file on your UnixWare 7 system. The */etc/inetd.conf* file will also show what services were available through the **inetd** daemon. Again, you should only consult this file so that you can amend the */etc/inetd.conf* file on your UnixWare 7 system. (Note that UnixWare 7 is bundled with TCP Wrappers which allow you to control who can access the services listed in */etc/inetd.conf*.)

Upgrading DHCP or AAS from SCO OpenServer Release 5.0.4

If you configured the Dynamic Host Configuration Protocol (DHCP) or the Address Allocation Server (AAS) on your SCO OpenServer Release 5.0.4 system, you can migrate their daemon configuration files, */etc/inet/dhcpd.conf* and */etc/inet/aasd.conf*, to UnixWare 7. Both will work without additional modification.

DHCP and AAS were not available on previous versions of SCO OpenServer or SCO UnixWare.

Upgrading routing

This section discusses differences between UnixWare 7, SCO UnixWare 2.1 and SCO OpenServer Release 5, and how the upgrade of routing may be accomplished.

Differences

UnixWare 7 contains updated **gated** and **routed** daemons (named **in.gated** and **in.routed**) and an updated **route** command. Both **gated** and **routed** support RIP Version 1 and Version 2, and router discovery. The separate router discovery daemon, **irdd**, that was available in SCO OpenServer Release 5 does not exist in UnixWare 7.

The release of **gated** in UnixWare 7 (Version 3-5-7) is similar to the version in SCO OpenServer Release 5. It is significantly improved over the SCO UnixWare 2.1 version, which did not support either OSPF or RIPv2.

gated in UnixWare 7 supports RIPv1, RIPv2, OSPFv2, EGPv2, BGPv2-v4, and router discovery. The HELLO routing protocol was supported by the SCO UnixWare 2.1 **gated**, but it is not supported in either the SCO OpenServer Release 5 or UnixWare 7 versions of **gated**.

Updated support commands for **gated** in UnixWare 7 include **gdc**, **ripquery** and **ospf_monitor**. The commands **ospf_monitor** and **gdc** did not exist in SCO UnixWare 2.1.

routed in UnixWare 7 supports RIPv1, RIPv2, and router discovery. **routed** in SCO UnixWare 2.1 and SCO OpenServer Release 5 only supported RIPv1. A new support command, **rtquery**, allows you to query routing daemons in the manner of **ripquery**. Additionally, it provides additional control over **routed**, by allowing you to raise or lower the trace level for debugging.

gated conforms to the RFCs shown in the following table:

	SCO Open-Server Release 5	SCO UnixWare Release 2.1	UnixWare 7	Description
RFC 891	Yes	Yes	Yes	DCN local network protocols
RFC 904	Yes	Yes	Yes	EGP specification
RFC 911	Yes	Yes	Yes	EGP gateway
RFC 1058	Yes	Yes	Yes	RIPv1 specification
RFC 1163	RFC 1267	Yes	RFC 1267	BGP specification
RFC 1164	RFC 1268	Yes	RFC 1268	BGP application
RFC 1253	Yes		Yes	OSPFv2 MIB
RFC 1256	Yes		Yes	Router discovery
RFC 1267	Yes		Yes	BGP-3 specification
RFC 1268	Yes		Yes	BGP-3 application
RFC 1269	Yes		Yes	BGP-3 managed objects
RFC 1389	Yes		Yes	RIPv2 MIB
RFC 1403			Yes	BGP OSPF interaction
RFC 1583	Yes		Yes	OSPFv2 specification
RFC 1723	Yes		Yes	RIPv2 specification

routed conforms to the RFCs shown in the following table:

	SCO Open-Server Release 5	SCO UnixWare Release 2.1	UnixWare 7	Description
RFC 1058	Yes	Yes	Yes	RIPv1 specification
RFC 1256			Yes	Router discovery
RFC 1723			Yes	RIPv2 specification

Configuring routing

UnixWare 7 does not provide a graphical manager for configuring routing. The **Network Client Manager** does include support for the **traceroute** and **ping** commands but not for configuring routing.

You can use the **Network Configuration Manager** to configure a default router.

Files to migrate

In UnixWare 7, as in SCO UnixWare 2.1, all routing configuration files are located in */etc/inet*, rather than in */etc* as in SCO OpenServer Release 5.

Configuration files are:

/etc/inet/gated.conf
gated configuration file

/etc/inet/gateways
routed configuration file

The following sample files are provided in */etc/inet* for *gated* configuration:

gated.bgp
BGP configuration

gated.egp
EGP configuration

gated.ospf
OSPF configuration

gated.rip
RIP configuration

Migrating gated and routed files to UnixWare 7

For **gated**, changes are required to */etc/inet/gated.conf*. Some keywords recognised by **gated** in SCO OpenServer Release 5 have changed and affect the default behaviour. In particular, a new **aggregate** keyword may be required as route aggregation was always enabled in SCO OpenServer Release 5. Additionally, more extensive tracing is provided; see **gated.conf**(4tcp) for further details).

The **gdc checkconf** command is useful for checking the integrity of the *gated.conf* file. It should be run in multi-user mode (that is, with networking running). Otherwise, it will be unable to pick up valid network interfaces to use.

For **routed**, the */etc/inet/gateways* configuration file supports many more command keywords. In particular, the **no_rdisc** keyword can be used to disable router discovery (enabled by default). See **routed**(1Mtcp) for details.

The **gdc** and **rtquery** commands provide the ability to dump a snapshot of the routing daemon's routing table and interface list to a log file for debugging purposes.

The files */var/adm/syslog* and */var/adm/log/osmlog* are used to log messages by default.

Upgrading DNS

UnixWare 7 is shipped with BIND Version 4.9.6, and includes a number of bug fixes, security fixes security fixes and new features over versions of BIND that shipped with SCO UnixWare 2.1 and SCO OpenServer Release 5.

Configuring DNS

DNS may be configured using the **DNS Manager**. However, if you migrate configuration files from SCO OpenServer Release 5 or SCO UnixWare 2.1, the **DNS Manager** may not be able to understand their structure or naming conventions. In this case, you must edit the files yourself.

DNS files to be migrated from SCO OpenServer Release 5

The file */etc/named.boot* must be relocated as */etc/inet/named.boot*. Similarly, any configuration files in the */etc/named.d* hierarchy should be relocated below */etc/inet/named.d*. You may also need to edit the files to correct any pathnames such as those specified by the *directory* directive. You do not need to copy over the cache hints file (see *root.cache*(4tcp)) as one is provided with the system (*/etc/inet/named.d/db.cache*). If necessary, you can use the **DNS Manager** to update this file.

Remove any **hostresorder** line in the resolver configuration file, */etc/resolv.conf*. In UnixWare 7, name resolution order and methods are controlled using entries in */etc/netconfig*. It is recommended that you do not edit this file directly. Use the **Network Client Manager** to configure entries in this file.

Migrating DNS files

The recommended upgrade path is to use the **DNS Manager** to configure a caching-only nameserver.

Next, configure any zones that the system serves as a primary name server. Use the **ndc restart** command to restart **named**. Check the contents of */var/adm/syslog* and */var/adm/log/osmlog* for any named errors. You may notice that hostnames containing an underbar ("_") character are logged as this is an illegal character for an Internet hostname. You should rename these hosts if possible.

Finally, configure any zones that the system serves as a secondary or stub name server and restart **named**. Check the logs again and check that the zone data has been written to the correct files.

The interpretation of a decimal point in the SOA serial number has changed. Previous versions of BIND would interpret 1.234 as 1000234 instead of 1234. The recommended serial number format is ***YYYYMMDDNN*** where ***YYYY*** is the year, ***MM*** is the month (01-12), ***DD*** is the day (01-31), and ***NN*** is the serial number of the change during that day (00-99) This allows you to make 100 changes a day until the year 4294.

Upgrading NIS

The version of NIS in UnixWare 7 is based on that in SCO UnixWare 2.1. No significant changes have been made to NIS since SCO UnixWare 2.1 shipped. NIS in UnixWare 7 does not support the copy-only servers that could be configured in SCO OpenServer Release 5's version of NIS.

Configuring NIS in UnixWare 7

NIS may be configured using **ypinit** as in SCO UnixWare 2.1. Alternatively, you can use the **Network Client Manager** to configure a NIS client.

Migrating NIS files to UnixWare 7

In UnixWare 7 and SCO UnixWare 2.1, NIS files are located in the */var/yp* hierarchy rather than in the */etc/yp* hierarchy which SCO OpenServer Release 5 uses.

NIS master and slave servers should set up */etc/passwd* and */etc/group* files using the **Account Manager** as normal but the copies of these files that are used to generate the corresponding NIS maps can be located elsewhere if the **DIR** variable is redefined in */var/yp/Makefile*.

Run **ypinit** with the appropriate option on all systems that need to use NIS:

-m Configure a master server.

-s *master*
Configure a slave server specifying the ***master***.

-c Configure a client. Alternatively, use the **Network Client Manager**.

Finally, on NIS clients, add escapes (+:) to files such as */etc/passwd* and */etc/group* so that they can access the corresponding NIS maps.

Upgrading UUCP

The SCO UnixWare 2.1 UUCP subsystem is carried forward to UnixWare 7. For SCO OpenServer Release 5 users, there are new API's, **dials**(3N) and **cs_connect**(3N), which are used to dial out to remote systems. The SCO OpenServer Release 5 modem dialers (based on **atdialer**) have been carried forward to UnixWare 7. This allows for the configuration of over 900 different modems.

UnixWare 7 includes support for ISDN BRI adapters and call service handling. Both of these features are proprietry to SCO.

Configuring UUCP

To configure entries for modems and ISDN adapters in the */etc/uucp/Devices* file, use the **Hardware** menu under the WAN view of the **Network Configuration Manager**.

To configure call services and filters defined in the files */etc/ics/Callfilter* and */etc/ics/Callservices*, select **Call Services** ➪ **Incoming** in the WAN view of the **Network Configuration Manager**.

To configure entries for remote systems in the */etc/uucp/Systems* file, select **Call Services** ➪ **Outgoing** in the WAN view of the **Network Configuration Manager**.

Files to migrate

For SCO OpenServer Release 5, the following files should be moved to */etc/uucp*:

/usr/lib/uucp/Devices

/usr/lib/uucp/Permissions

/usr/lib/uucp/Poll

/usr/lib/uucp/Systems

These files should not need modification.

For SCO UnixWare 2.1, you will need to migrate:

/etc/uucp/Config

/etc/uucp/Devices

/etc/uucp/Dialcodes

/etc/uucp/Grades

/etc/uucp/Limits

/etc/uucp/Permissions

/etc/uucp/Poll

/etc/uucp/Sysfiles

/etc/uucp/Systems

The *Devices* file may need modifying to reflect the device naming scheme used by UnixWare 7. See "Serial device node naming conventions" in SCOhelp for details.

Upgrading the FTP server

The FTP servers in SCO OpenServer Release 5 and UnixWare 7 are based on the Washington University FTP server, **wu-ftpd**. The UnixWare 7 version is based on the latest version (2.4). It includes additional features and many bug fixes compared to the SCO OpenServer Release 5 version.

The FTP server in SCO UnixWare 2.1 is not based on **wu-ftpd** and lacks many of the features of the SCO OpenServer Release 5 and UnixWare 7 servers.

The FTP servers in SCO OpenServer Release 5, SCO UnixWare 2.1 and UnixWare 7 conform to RFC 959. Only the SCO OpenServer Release 5 and UnixWare 7 versions conform to RFC 1123.

Configuring the UnixWare 7 FTP server

The FTP server in UnixWare 7 may be configured using the **FTP Server Manager**.

Files to migrate

The following files need to be migrated from SCO UnixWare 2.1:

/etc/ftpusers

/etc/shells

The following files need to be migrated from SCO OpenServer Release 5:

/etc/ftpusers

/etc/shells

/etc/ftpaccess

/etc/ftpconv becomes */etc/ftpconversions*

Procedure for migrating FTP files

Any user names added to the SCO OpenServer Release 5 or SCO UnixWare 2.1 */etc/ftpusers* file should be added to the UnixWare 7 */etc/ftpusers* file in order to continue to deny access to those users.

Any shells added to the SCO OpenServer Release 5 or SCO UnixWare 2.1 */etc/shells* file should be added to the UnixWare 7 */etc/shells* file in order to continue to allow access to a user who has one of those shells as their login shell. The pathnames of some entries may need changing to match the location of the shell in the filesystem hierarchy of UnixWare 7.

Any conversions added to the SCO OpenServer Release 5 */etc/ftpconv* file should be added to the UnixWare 7 */etc/ftpconversions* file, changing the pathname of the conversion utility where appropriate.

The syntax of some entries in */etc/ftpaccess* has changed:

- In SCO OpenServer Release 5, the **private** keyword is followed by the pathname of the group access file. In UnixWare 7, it is followed by **yes** or **no** and the group access file is */etc/ftpgroups*.
- In SCO OpenServer Release 5, the **upload** keyword only applies to the anonymous user. In UnixWare 7, it is followed by an additional argument which defines the home directory of the user to whom the upload applies.

Upgrading NFS

NFS in UnixWare 7, SCO OpenServer Release 5 and SCO UnixWare 2.1 is based on Version 2. Configuration of NFS and the automounter in UnixWare 7 is similar to SCO UnixWare 2.1 but substantially different from SCO OpenServer Release 5.

NFS in UnixWare 7 and SCO UnixWare 2.1 does not include the spongy mount or transport over TCP features of NFS in SCO OpenServer Release 5.

automount in SCO OpenServer Release 5 automatically consults the NIS *auto.master* map unless the **-m** option is specified on the command line. It does not consult the */etc/auto.master* file unless this is also specified using the **-f** option. **automount** in UnixWare 7 and SCO UnixWare 2.1 reads the */etc/auto.master* file unless you override the pathname using the **-f** option. It does not consult the NIS *auto.master* map unless the following line is included in the */etc/auto.master* file on the client:

```
+auto.master
```

Configuring NFS in UnixWare 7

A filesystem is made available for mounting by NFS clients by adding **share**(1Mnfs) entries to the */etc/dfs/dfstab* file. You can invoke the entries in this file by executing the following command:

. /etc/dfs/dfstab

You can mount NFS filesystems on NFS clients using the **Filesystem Manager**.

Files to migrate

The following table shows approximate equivalences between NFS configuration files in SCO OpenServer Release 5 and UnixWare 7:

SCO OpenServer Release 5	UnixWare 7	Description
/etc/default/filesys	*/etc/vfstab*	Used by client to define filesystem to be mounted
/etc/exports	*/etc/dfs/dfstab*	Used by server to define filesystems that clients can mount
/etc/auto.master	*/etc/auto.master*	Lists initial automount configuration. The information may also be obtained as a map from an NIS server
/etc/auto.direct */etc/auto.indirect*	*/etc/auto.home*	List direct and indirect automount configuration. The information may also be obtained as map(s) from an NIS server

If migrating from SCO OpenServer Release 5, use the information in the configuration files to configure your UnixWare 7 system. Do not copy the */etc/default/filesys* and */etc/exports* files to their equivalents as the format of these files is not the same as in UnixWare 7. The following options which are supported by **mount** in SCO OpenServer Release 5 are not supported in UnixWare 7: **exec**, **noexec**, **trunc**, **notrunc**, **tcp**, and **spongy**. It is recommended that you enter the information in */etc/default/filesys* using the **Filesystem Manager**.

The information in the */etc/exports* file can be added to */etc/dfs/dfstab* as follows:

1. Edit a copy of */etc/exports*. Each line, other than comment lines that start with a "#" character, should start off with the following format:

   pathname -***options*** # ***comment***

   Change each line so that it has the following format:

   share -Fnfs -o "***options***" [-d "***comment***"] ***pathname***

   The description specified by the **-d** option is optional. The **access** option in SCO OpenServer Release 5 is not supported by UnixWare 7. Replace each **access** option with **ro** (read-only) or **rw** (read and write) to define the read permissions for each client explicitly. Note that *netgroup* entries are supported. For example, consider the following lines in a copy of the */etc/exports* file from an SCO OpenServer Release 5 system:

   ```
   /usr       -access=clients             #export to netgroup clients
   /usr/local                             #export to the world
   /usr2      -access=hermes:zip:tutorial #export to only these machines
   /usr/sun   -root=hermes:zip            #give root access only to these
   /usr/new   -anon=0                     #give all machines root access
   /usr/bin   -ro                         #export read-only to everyone
   /usr/stuff -access=zip,anon=-3,ro      #several options on one line
   ```

   This would be converted to:

   ```
   share -Fnfs -o "rw=clients"            /usr
   share -Fnfs -d "export to the world"   /usr/local
   share -Fnfs -o "rw=hermes:zip:tutorial" /usr2
   share -Fnfs -o "root=hermes:zip"       /usr/sun
   share -Fnfs -o "anon=0"                /usr/new
   share -Fnfs -o "ro"                    /usr/bin
   share -Fnfs -o "rw=zip,anon=-3,ro"     /usr/stuff
   ```

2. Copy this file to the end of */etc/dfs/dfstab* on the UnixWare 7 NFS server.

3. Run the following command to make the filesystems available for clients to mount:

   . /etc/dfs/dfstab

The **automount** configuration files may be copied across but you may have to edit them to fix compatibility differences. UnixWare 7 has a single file *auto.home* (and correspondingly named map) which combines the function of *auto.direct* and *auto.indirect*. It is possible to configure separate NIS maps by editing */var/yp/Makefile* but you may find it simpler to combine *auto.direct* and *auto.indirect*. You will also need to change any NIS map entries such as

"+auto.direct" and "+auto.indirect" to "+auto.home" if clients obtain these maps using NIS.

Upgrading NTP

SCO UnixWare 2.1 included version 2.3 of NTP which conforms to RFC 1119 and retains compatibility with RFC 1059. SCO OpenServer Release 5 included version 3.2 of NTP and UnixWare 7 includes version 3.5f of NTP. These conform to RFC 1305 and retain compatibility with RFC 1119 and RFC 1059.

NTP configuration

NTP clients may be configured using the **Network Client Manager**.

NTP servers are configured by editing the file */etc/inet/ntp.conf*. Configuration of NTP servers does not differ substantially between SCO UnixWare 2.1, SCO OpenServer Release 5 and UnixWare 7 except for the following points:

- In UnixWare 7, servers can be specified by their domain name rather than IP address without needing to specify a resolver helper program in *ntp.conf*.
- In UnixWare 7, the syntax for using the host's internal clock has changed slightly. See **xntpd**(1Mtcp) for details.

Files to migrate

The default NTP configuration file in SCO OpenServer Release 5 is */etc/ntp.conf*. The default NTP configuration file in SCO UnixWare 2.1 is */etc/inet/ntp.conf* as in UnixWare 7. In addition, you will need to copy over files containing authentication keys. You should also create any log files such as those used for writing drift measurements and other statistics. The pathnames of these files will be defined in the *ntp.conf* file.

Upgrading NetWare and IPX/SPX

IPX/SPX in SCO OpenServer Release 5 is based on NWU Version 3.1. IPX/SPX in SCO UnixWare 2.1 is based on NWU Version 4.10. IPX/SPX in UnixWare 7 is based on Netware 4.10a.

> **NOTE** Configuration of networking stacks in UnixWare 7 should only be performed using the **Network Configuration Manager**.

Gemini supports NetWare over IP (NWIP) by tunneling IPX/SPX packets over IP. At least one NetWare server must be configured to run as a Domain SAP/RIP Server (DSS). See "NWIP configuration parameters" in help for more information.

Configuring IPX/SPX stacks

Use the **Network Configuration Manager** to configure IPX/SPX. Using **nwcm** or editing the configuration files by hand is not recommended.

IPX/SPX files to be migrated

From SCO OpenServer Release 5, configuration information in the file */etc/ipx.d/NPSConfig* may need to be migrated.

From SCO UnixWare 2.1, configuration information in the file */etc/netware/nwconfig* may need to be migrated.

The information in these files should be migrated to */etc/netware/nwconfig* on your UnixWare 7 system.

Migrating files to UnixWare 7

The configuration file */etc/netware/nwconfig* contains configuration information for all NetWare components including the IPX/SPX stack. This section refers only to IPX/SPX stack configuration.

The contents of the SCO OpenServer Release 5 configuration file */etc/netware/nwconfig* differ significantly from the file */etc/ipx.d/NPSConfig* in UnixWare 7, whereas the contents of the */etc/netware/nwconfig* file in SCO UnixWare 2.1 are very similar.

> **WARNING** Do not copy the stack related sections of the config file from one system to another. For example, do not copy lines such as the following:
>
> ```
> lan_N_adapter = "/dev/netn"
> ```

Upgrading NetBIOS

A in-kernel implementation of NetBIOS was not originally available in SCO UnixWare 2.1 from SCO. The version of NetBIOS in UnixWare 7 is based on the in-kernel NetBIOS in SCO OpenServer Release 5 with the following enhancements:

- support for DNS resolver name lookup
- support for WINS lookup (WINS server capability is provided as part of the AFPS package)
- support for multiple network interfaces
- support for up to 1100 connections

Configuring NetBIOS in UnixWare 7

NetBIOS in UnixWare 7 is not configurable using the **Network Configuration Manager**. It is necessary to edit the file */etc/inet/nb.conf* instead. However, as the default behavior of NetBIOS is to run over all available interfaces, this is not usually necessary unless you want to configure name resolution via nominated WINS servers.

NetBIOS files that must be migrated

The only NetBIOS file that needs to be migrated is */etc/default/netbios* to */etc/inet/nb.conf*.

Procedure for migrating NetBIOS configuration files

There are several differences in the parameters that can be configured in */etc/default/nbconf* and */etc/inet/nb.conf*. The following parameter has been enhanced for UnixWare 7:

NB_ADDR
: This allows you to specify the IP addresses of the interfaces to be used. Set this to the null string ("") if all available interfaces are to be used.

The following parameters are new for UnixWare 7:

NB_NAMESEARCH
: Specify name resolution methods and order.

NB_WINS_PRIMARY
: Specify a primary WINS server.

NB_WINS_SECONDARY
: Specify a secondary WINS server.

See **netbios**(1Mtcp) for more information.

The following parameters are no longer valid in UnixWare 7:

NB_HOST

NB_MAXPKT

NB_BROADCAST

These parameters should be deleted.

Upgrading PPP

PPP has changed extensively in UnixWare 7. It supports the following new features:

- Multilink PPP (RFC 1990)

- Extensions for name server addresses (RFC 1877)
- Compression control protocol (RFC 1962)
- PPP over ISDN (RFC 1618)
- Dynamic IP address assignement (also in SCO OpenServer Release 5)
- Support for third-party framing drivers (also in SCO OpenServer Release 5)

PPP in UnixWare 7 does not support SNMP Managed Objects for LCP or IP (RFC 1471 and RFC 1473) which were supported in SCO OpenServer Release 5 and SCO UnixWare 2.1.

Configuring PPP

Most of the PPP configuration files in SCO OpenServer Release 5 and SCO UnixWare 2.1 are replaced by a single file which should not be edited by hand. The contents of the file may be changed using the **PPP Manager** or the **ppptalk**(1M) command. You can also use the **PPP Internet Connection Manager** to set up simple outgoing PPP configurations.

Pools of available IP addresses may be configured using the **Address Allocation Manager** in UnixWare 7. The UUCP *Systems* and *Devices* files may be configured from the WAN view of the **Network Configuration Manager**. Packet filter definitions may be configured using the **Packet Filter Manager**.

Upgrading PPP

PPP configuration was very similar in SCO OpenServer Release 5 and SCO UnixWare 2.1. Both versions of PPP used configuration files whose formats were almost identical. The following table shows the equivalence between these configuration files and data definition statements that are internal to **ppptalk** in UnixWare 7:

Feature configured	SCO OpenServer Release 5 file	SCO UnixWare Release 2.1 file	UnixWare 7 definitions
PPP endpoints	*/etc/ppphosts*	*/etc/inet/ppphosts*	**bundle** **link** **protocol**
Authentication database	*/etc/pppauth*	*/etc/inet/pppauth*	**auth**
Third-party framing drivers	*/etc/pppstack*		**link**
IP address pool	*/etc/ppppool*	*/etc/addrpool*	**protocol**
Packet filters	*/etc/pppfilter*	*/etc/inet/pppfilter*	**protocol**

The following table shows equivalences between parameters in the *ppphosts* file in SCO UnixWare 2.1 or SCO OpenServer Release 5 and parameters that can be configured using **ppptalk** in UnixWare 7:

ppphosts parameter	ppptalk parameter	ppptalk definition
accm	**accm**	**protocol = lcp**
attach	*bundle_tag*	**bundle**
auth	**protocol**	**auth**
authtmout	**authtmout**	**bundle | global**
bypassframing	No equivalent	
clocal	No equivalent	
conf	**maxcfg**	**protocol = ccp | ip | lcp**
debug	**debug**	**bundle | link | protocol**
filter	**bringup** **keepup** **passin** **passout**	**protocol = ip**
flow	**flow**	**link**
forcefarip	**peeropt = force**	**protocol = ip**
forcenearip	**localopt = force**	**protocol = ip**
getfarip	**peeropt = any**	**protocol = ip**
getnearip	**localopt = any**	**protocol = ip**
idle	**maxidle**	**bundle**
local	**localaddr**	**protocol = ip**
mask	**netmask**	**protocol = ip**
maxslot	**vjmaxslot**	**protocol = ip**
mru	**mru**	**protocol = lcp**
nak	**maxfail**	**protocol = ccp | ip | lcp**
name	**peerauthname**	**bundle | global**
noaccomp	**acfc = no**	**protocol = lcp**
noslotcomp	**vjslotcomp = no**	**protocol = ip**
noipaddr	**localopt = any** **peeropt = any**	**protocol = ip**
nomgc	**magic = no**	**protocol = lcp**
noprotcomp	**acfc = no**	**protocol = lcp**
novj	**vjcompress = no**	**protocol = ip**
old	No equivalent	
providefarip	**peeropt = prefer**	**protocol =ip**
providenearip	**localopt = prefer**	**protocol =ip**
proxy	**proxyarp**	**protocol = ip**
remote	**peeraddr**	**protocol = ip**
reqtmout	**reqtmout**	**protocol = ccp | ip | lcp**
retry	No equivalent	
rfc1172addr	No equivalent	
sh_hook	**exec**	**protocol = ip**
speed	No equivalent	
staticdev	**dev**	**link**
term	**maxterm**	**protocol = ccp | ip | lcp**
uucp	**remotesys**	**bundle**

Migrating PPP configuration files

It is not feasible to migrate the PPP configuration files from SCO UnixWare 2.1 and SCO OpenServer Release 5 to UnixWare 7. You should make backup copies of the files, and refer to these for configuration information when setting up PPP on your UnixWare 7 systems.