OrCAD® Network Licensing

Network Licensing Guide

OrCAD®

OrCAD®

9300 SW Nimbus Ave.
Beaverton, OR 97008 USA

Contents

Contents

Tables

Before you begin

OrCAD offers a total solution for your core design tasks: schematic- and VHDL-based design entry; FPGA and CPLD design synthesis; digital, analog, and mixed-signal simulation; and printed circuit board layout. What's more, OrCAD's products are a suite of applications built around an engineer's design flow—not just a collection of independently developed point tools. OrCAD network licensing is just one element in OrCAD's total solution design flow.

OrCAD network licensing allows users in a network configuration to use OrCAD programs, such as PSpice, Layout, Express, Capture, or Capture CIS, on any Novell, Windows 95, or Windows NT computer (or *client*) on the network without being locked to a specific computer.

The number of users who can run OrCAD programs at the same time is limited to the number of program licenses purchased.

OrCAD network licensing is handled by the License Manager, which administers licenses to clients.

Product configurations

When you install OrCAD network licensing, you can choose between two types of license managers: NetHASP, or FLEXlm.

NetHASP works with TPC/IP, IPX, and NetBIOS, so system administrators can use a wide variety of configurations for client computers.

FLEXlm gives administrators more control and flexibility than NetHASP in administering OrCAD licenses, but requires that the license server and all of the client machines be configured with TCP/IP.

How to use this guide

This guide is designed so you can quickly find the information you need to use network licensing. To help you learn and use network licensing efficiently, this manual is organized by tasks, beginning with installation and configuration basics (chapter one), and moving on to more advanced concepts for the NetHASP and FLEXlm license managers (chapters two and three).

Symbols and conventions

OrCAD printed documentation uses a few special symbols and conventions.

Notation	Examples	Description
[Ctrl]+[R]	Press [Ctrl]+[R]	Means to hold down the [Ctrl] key while pressing [R].
[Alt], [F], [O]	From the File menu, choose Open ([Alt], [F], [O])	Means that you have two options. You can use the mouse to choose the Open command from the File menu, or you can press each of the keys in parentheses in order: first [Alt], then [F], then [O].
Monospace font	In the Part Name text box, type PARAM.	Text that you type is shown in monospace font. In the example, you type the characters [P], [A], [R], [A], and [M].
	`.MODEL MLOAD NMOS` `+ (LEVEL=1 VTO=0.7 CJ=0.02pF)`	Examples of syntax, netlist output, and source code are displayed in monospace font. The example shows an example of the syntax for the PSpice .MODEL statement.
UPPERCASE	In Capture, open CLIPPERA.DSN.	Path and filenames are shown in uppercase. In the example, you open the design file named CLIPPERA.DSN.
Italics	In Capture, save *design_name*.DSN.	Information that you are to provide is shown in italics. In the example, you save the design with a name of your choice, but it must have an extension of .DSN.

Related documentation

In addition to this guide, you can find technical information in online help files, online books, OrCAD's technical web site, as well as other books. The table below describes the types of technical documentation provided with the Network Licensing Guide.

This documentation component . . .	Provides this . . .
This guide— OrCAD Network Licensing Guide	Basic and advanced information on installing and configuring network licensing for OrCAD products.
NetHASP settings online help (NETTEST.HLP)	Testing information for the NetHASP License Manager.
HASP device driver installation online help (HINSTALL.HLP)	Installation, troubleshooting, and command line information for the HASP device driver.
NetHASP License Manager online help (NHSRVW32.HLP)	Information about using the NetHASP License Manager.
NetHASP Monitor online help (MONITOR.HLP)	Data on the loaded NetHASP License Manager, including set and installed protocols, the number of NetHASP License Managers on the network, and the number of users that have logged in to the License Manager.
Online *OrCAD Network Licensing Guide*	An online, searchable version of this guide.
ODN—OrCAD Design Network www.orcad.com/odn	An internet-based technical support solution. ODN provides a variety of options for receiving and accessing design and technical information. ODN provides:
	• A Knowledge Base with thousands of answers to questions on topics ranging from schematic design entry and VHDL-based programmable logic design to printed circuit board layout methodologies.
	• A Knowledge Exchange forum for you to exchange information, ideas, and dialog with OrCAD users and technical experts from around the world. A list of new postings appears each time you visit the Knowledge Exchange, for a quick update of what's new since your last visit.
	• Tech Tips that deliver up-to-the-minute product information in your email box. Stay informed about the latest advances, tips, and announcements on your OrCAD product.
	• Online technical support via the Tech Support Connection. Use this service to submit technical support incidents online. Create submissions, upload files, track your incidents and add comments directly into OrCAD's support database.

Getting started

1

Overview

The following sections provide information for setting up your network for network licensing:

- OrCAD network licensing on page 1-3 describes the network licensing and the License Manager.

- Setting up network licensing for OrCAD programs on page 1-4 gives guidelines for selecting and setting up a server.

- Troubleshooting your network licensing setup on page 1-10 provides information on troubleshooting the License Manager, client crashes, or your network setup.

- Setting up the License Manager on a Novell Netware 386/486 file server on page 1-13 describes setting up the License Manager on a Novell server.

- Setting up Windows 95 or NT clients for Novell server on page 1-15 describes setting up a network with Windows clients and a Novell server.

- describes setting up a network of Windows computers using the IPX protocol.

- describes setting up a network of Novell computers.

- explains how to update or regenerate your license information.

OrCAD network licensing

Network licensing allows users running in a network configuration to use OrCAD programs, such as PSpice, Layout, Express, or Capture, on any Novell, Windows 95, or Windows NT computer (or *client*) on the network without being locked into a specific computer.

The number of users who can run OrCAD programs at the same time is limited to the number of program licenses purchased.

What is the License Manager?

The License Manager is the OrCAD program that administers licenses to clients. It runs on the server and accesses the hardware lock for licensing information. Depending on your configuration, the License Manager is called `NHSRVW32.EXE`, `HASPSERV.NLM`, or `LMGRD.EXE`.

Once the hardware lock and License Manager are installed on the server, each license is checked in and checked out by the License Manager. The License Manager administers the licenses throughout the network. This process is completely transparent.

See **Chapter 2,** NetHASP, for more information about NetHASP licensing.

See **Chapter 3,** FLEXlm, for more information about FLEXlm licensing.

Setting up network licensing for OrCAD programs

There are three major steps in setting up network licensing for OrCAD programs. They must be performed in the following order:

1   Configure the designated license server by installing the License Manager program.

Note   *You must install the License Manager prior to installing OrCAD programs.*

2   Install OrCAD programs on a file server (or on individual clients).

3   If the OrCAD programs are installed on a file server (not a client), configure network clients to access the OrCAD programs installed on the file server.

Server guidelines

Additional requirements

The computer must have a parallel port and must be stable, which means it should not be subject to frequent restarting.

Selecting a server

The computer you choose to be the server will depend on your network configuration. The server can be any Novell, Windows 95, or Windows NT computer on the network. However, you must designate one computer as the server. This computer then becomes the license server that will handle the requests for individual licenses from clients by monitoring the number of users or licenses.

Supported protocols

OrCAD products support IPX, NETBIOS, and TCP/IP protocols on Novell, Windows NT and Windows 95.

Choosing a license server

If you are running the TCP/IP protocol, OrCAD recommends configuring your server with FLEXlm; however, you have the option to use NetHASP.

FLEXlm

Pros	Cons
FLEXlm gives administrators more control and flexibility than NetHASP in administering OrCAD licenses.	FLEXlm requires the license server and all of the client machines to be configured with TCP/IP.
With FLEXlm you can:	

- Selectively allow access to licenses for a defined group of people.
- Check in licenses that are not being used.
- Get information about what licenses are being used and by whom.

See **Chapter 3,** FLEXlm, for more information about the features and benefits of FLEXlm.

NetHASP

Pros	Cons
NetHASP works with TCP/IP, IPX and NetBIOS, so system administrators can use a wider variety of configurations for client computers.	NetHASP provides only minimal administrative tools.
NetHASP Monitor provides remote monitoring of the License Manager.	

See **Chapter 2,** NetHASP, for more information about the features and benefits of NetHASP.

Step one: Configuring the designated license server

The license server and the file server can have different operating systems. For example, if you are using Novell on your file server, you can designate a Windows 95 workstation as the license server.

Installing the hardware lock

The license server requires a hardware lock. The red network hardware lock attaches to the parallel port (the plug does not affect the operation of printers or other devices).

Note   *The machine that hosts the hardware lock is designated as the license server.*

Supported operating systems

The computer you choose for the license server depends on your network configuration. The server can be any Novell, Windows 95, or Windows NT computer on the network.

During the Novell Netware file server installation, the installation program requests that you attach the hardware lock to the client. After the installation, remove the hardware lock and attach it to the server.

Novell Netware file server

To install the License Manager on a Novell Netware file server you must start the installation process on a client so you can remotely access the appropriate files through the Novell console.

1   Select a client (Windows 95 or NT) that has permission to copy files to an area that can be accessed from the Novell console.

2   Insert the OrCAD CD into the CD-ROM drive on the
    client.

3   From the Start menu of the Windows task bar, choose
    Run.

4   Enter: drive:\SETUP.EXE (where drive is the
    CD-ROM drive).

5   When the Installation Setup screen appears, select
    Network Server Installation. Follow the instructions
    on how to install the License Manager program.

Windows 95 or NT 4.0

1   Insert the OrCAD CD into the CD-ROM drive.

2   From the Start menu of the Windows task bar, choose
    Run.

3   Enter: drive:\SETUP.EXE (where drive is the
    CD-ROM drive).

4   When the Installation Setup screen appears, select
    Network Server Installation. Follow the instructions
    on how to install the License Manager program.

Note   *Press F1 to access help during the License Manager installation
       process.*

Step two: Installing OrCAD programs on a file server

Once the License Manager has been installed and tested on the server, you are now ready to install OrCAD programs. These programs can be installed on a file server, or on individual clients.

Note   *Press F1 to access help during the OrCAD program installation process.*

Step three: Configuring remote or local clients

Note   *Configuring remote clients is only possible if you have installed OrCAD programs on a designated file server.*

Windows 95 or NT 4.0

1   Insert the OrCAD CD into the CD-ROM drive on the client.

2   From the Start menu of the Windows task bar, choose Run.

3   Enter: drive:\SETUP.EXE (where drive is the CD-ROM drive).

4   Select Configure Client on the Installation Setup screen.

5   Select Remote on the following screen to create a local program group on the client, with references to program files that are installed on a remote server.

Alternatively, select Local to install the program files on the local client.

Note   *Press F1 to access help during the configure client process.*

Troubleshooting your network licensing setup

If the license manager becomes unavailable for any reason while an OrCAD program is running, you may be prompted to remedy the situation before you will be allowed to resume operation. The program makes every attempt to provide uninterrupted service.

Note   *If you need to restart the License Manager or the license server, the clients will lose their licenses, and their applications will need to be restarted.*

License Manager

Run the built-in diagnostic program to verify the License Manager is working properly. To run the built-in diagnostic program, do the following:

1   Insert the OrCAD CD into the CD-ROM drive.

2   From the Windows Start menu, choose Run.

3   Enter: drive:\SETUP.EXE (where drive is the CD-ROM drive).

4   On the Installation Setup dialog, select Network Server Installation and choose Next.

5   Select Test License Manager and choose Next.

6   Enter the path to the Server ID file, then choose Next.

7   Enter the directory path to where the License Manager is installed, or accept the default path provided, then choose Next.

8   In the Activate the License Manager dialog, select either Activate License Manager or Do not activate, it is already running, depending on the activation status of the License Manager. Choose Next to run the diagnostic program.

If the test fails on this…		Verify this…
Novell server	1	License Manager version 6.0 is loaded and running on Novell file server.
	2	Current workstation has IPX installed and configured.
	3	Current workstation can communicate with the Novell server. (This can be done by doing something as simple as mounting a drive from the Novell server.)
Win 95/NT server	1	The current HASP drivers are correctly installed. (Do this by running HINSTALL -INFO on the license server. Version 3.61 or later should be installed.
	2	License Manager is loaded and running.
	3	The license server is configured to work on a network.
Win 95/NT clients	1	License Manager Server is loaded and running.
	2	Current workstation is configured to work on a network.
	3	Current workstation can communicate with the License Server.

Clients

When a client running an OrCAD program fails, the license may remain checked out from the server. To correct this, you have three options:

- If no one else is using OrCAD programs, you can restart the License Manager program. Note that if users have licenses checked out, their sessions may be interrupted.

- After the client computer has restarted, restart and exit the program that was running at the time of the failure. This makes the license available for use by others.

- The license automatically checks itself back in after 36 hours.

Server or client setup

If you have this…	With this protocol…	Read this…
Win 95 or NT clients on Novell network	IPX	Setting up Windows 95 or NT clients for Novell server on page 1-15
Win 95 or NT clients	IPX	Setting up the License Manager on Windows 95 or NT on page 1-16
Win 95 or NT clients	NetBIOS	Setting up Windows 95 or NT for NetBIOS on page 1-18
Win 95 or NT clients	TCP/IP	Setting up Windows 95 or NT for TCP/IP on page 1-19
License access restrictions under Novell	IPX	Setting up local networks under Novell on page 1-20

Also, see **Chapter 2,** NetHASP, to learn more about the features and benefits of using NetHASP, and see **Chapter 3,** FLEXlm, to learn more about the features and benefits of using FLEXlm.

Setting up the License Manager on a Novell Netware 386/486 file server

The NetHASP License Manager for Novell Netware file servers is `HASPSERV.NLM`. The install program will not set up the NetHASP License Manager automatically.

To load the NetHASP License Manager

1 Attach the NetHASP hardware lock to the parallel port of the server.

2 Copy `HASPSERV.NLM` to the SYSTEM directory.

3 Load the NetHASP License Manager by entering:

   `LOAD HASPSERV`

   The NetHASP License Manager screen appears and shows the protocol that is being used.

To load the NetHASP License Manager automatically

1 Add the line LOAD HASPSERV to the AUTOEXEC.NCF file in the SYS:SYSTEM directory.

To remove the NetHASP License Manager

1 Enter: UNLOAD HASPSERV

To set up a client for a remote installation

1 Start the OrCAD installation program.

2 Select Configure Client on the Installation Setup screen, then select Remote on the following screen.

3 Enter the path to the directory on the network drive where the OrCAD programs are installed.

   This will do the following:

   • create a local set of OrCAD program icons

   • create a local copy of `ORCAD.LIC`

   • configure and create a local backup directory

   • if PSpice is part of your purchased package, configure PSpice to use a local temp directory

   • copy the appropriate `.DLL` files for the operating system to the local hard drive

Setting up clients

Client computers can run any combination of Windows 95 and Windows NT, regardless of the license server's platform.

Before setting up a client, verify the following:

• The client computer has a protocol installed that matches a protocol listed as loaded on the License Manager.

• For a remote client installation, the client computer has access to the OrCAD program files loaded on the file server.

13

To set up a client for a local installation

1   Start the OrCAD installation program.

2   Select Configure Client on the Installation Setup
     screen, then select Local on the following screen.

3   Specify the location of the Server ID file (SERVER.INI).

4   Enter the appropriate key code and authorization
     code information.

Setting up Windows 95 or NT clients for Novell server

OrCAD programs only support a Novell server when used with the IPX protocol. The client machines must also have the IPX protocol loaded.

Setup for Novell server is done automatically by the install program.

Setting up the License Manager on Windows 95 or NT

The NetHASP License Manager for Windows 95 or NT file servers is NHSRVW32.EXE. The install program automatically sets up the NetHASP License Manager.

If OrCAD programs are located on a shared drive, it is recommended that you place NHSRVW32.EXE in the OrCAD directory.

Windows 95 server

To run the NETHASP License Manager (NHSRV32.EXE)

Note   *The HASP drivers must be installed before you can run the License Manager. Verify the state of the drivers by running HINSTALL -INFO. Version 3.61 or later should be installed.*

To remove the NetHASP License Manager

1   Close the License Manager window.

1   Attach the hardware lock to the parallel port of the license server.

2   Start NHSRV32.EXE. A window appears with a message informing you that the NetHASP License Manager is loaded. It also shows which protocols are loaded and available.

Windows NT server

To run the NETHASP License Manager (NHSRVW32.EXE)

Note   *The HASP drivers must be installed before you can run the License Manager. Verify the state of the drivers by running HINSTALL -INFO. Version 3.61 or later should be installed.*

1   Attach the hardware lock to the parallel port of the license server.

2   Start NHSRVW32.EXE. A window appears with a message that the NetHASP License Manager is loaded. It also shows which protocols are loaded and available.

Automatic activation for NetHasp License Manager

Under Windows NT, you can load the NetHASP License Manager automatically before user logon. This is implemented with HINSTALL.EXE.

Hinstall installs a HASP NT Loader implemented as a Windows NT service. This loader automatically loads the NetHASP License Manager. With the HASP NT Loader, you do not have to log on or interact with the Windows NT system.

To install the HASP NT Loader

When you install the NT Loader, the HASP Device Driver is also installed.

Because the NetHASP License Manager loads automatically before user logon, the HASP NT Loader and the NetHASP License Manager messages do not appear on the screen until after logging on. For example, if you load the NetHASP License Manager without a connected HASP, you do not see the error message Hasp not found until after logging on.

1   Log on as a user with Administrator privileges.

2   Run HINSTALL.

3   Display the License Manager Loader tab, verify the server path for the License Manager, and click Install.

4   Restart the computer. The HASP NT Loader will automatically load the NetHASP License Manager.

For Windows 95 or NT clients (IPX)

1   In the Windows directory, create a file called NETHASP.INI that includes the following (not case-sensitive):

```
[nh_common]
nh_ipx=enabled
```

To remove the HASP NT Loader

1   Log on as a user with Administrator privileges.

2   Run HINSTALL -RS.

3   Restart the computer.

Note *This procedure does not remove the HASP Device Driver.*

17

Setting up Windows 95 or NT for NetBIOS

If OrCAD programs are located on a shared drive, OrCAD recommends placing NHSRVW32.EXE in the OrCAD directory.

1   Create a file called NETHASP.INI file with two lines:

```
[nh_common]
nh_NetBIOS=enabled
```

2   Copy the NETHASP.INI file into the Windows directory of each client computer.

3   Run NHSRVW32.EXE.

Setting up Windows 95 or NT for TCP/IP

1   Create a `NETHASP.INI` file with five lines:

```
[nh_common]
nh_tcpip=enabled

[nh_tcpip]
nh_server_addr=<addr>
nh_tcpip_method=tcp
```

where <addr> is of the form 192.114.176.65, and is the IP address of the license server.

2   Copy the `NETHASP.INI` file into the Windows directory on each client computer.

3   Run NHSRVW32.EXE.

Windows 95: double-click Network in the Control Panel, double-click TCP/IP, and note the IP address.

Windows NT: double-click Network in the Control Panel, double-click the IP protocol, and note the IP address.

Setting up local networks under Novell

An internetwork is a group of two or more local networks linked together.

A local network is a single cabling scheme identified by a unique network number, to which one or more workstations are connected. When you load the NetHASP Server program, it advertises its name to the entire internetwork (by default) and serves all of the internetwork's local networks and workstations (by default).

If you want the NetHASP Server program to serve the entire network, use the defaults. If the defaults do not suit your needs, you can instruct the NetHASP Server program to serve only a specific local network. For more information on how to do this, see one of the following:

- [Specifying local networks with HASPSERV.NLM on page 1-20](#)

- [Specifying local networks with other servers on page 1-24](#)

- [Setting timeout length on page 1-24](#)

Specifying local networks with HASPSERV.NLM

- To instruct `haspserv.nlm` to serve only those local networks connected to the file server on which `HASPSERV.NLM` is running, load it with the LOCALNET switch:

  ```
  load haspserv -localnet
  ```

- To instruct a workstation to communicate only with the `HASPSERV.NLM` that serves the workstation's local network (and not with other servers loaded on the internetwork), add the following lines (not case-sensitive) to the [nh_ipx] section of the `NETHASP.INI` file:

  ```
  nh_search_method=localnet
  ```

- To instruct HASPSERV to advertise its address with a specific local network number, load it with the NET switch:

  ```
  load haspserv -net m [net n...]
  ```

  where m and n are IPX network numbers in the internetwork.

- To find out the network numbers of a NetHASP Server, enter `config` on the file server console. The information displayed on the screen contains the network numbers.

  Although `HASPSERV.NLM` advertises its address with specific network numbers, it continues to serve all of the workstations in the entire internetwork. HASPSERV serves the specified local networks only when it is loaded with the LOCALNET switch and with the NET switch, concurrently.

Examples

Example one: serving only one network in an internetwork

An internetwork consists of local networks A, B, C, etc. `HASPSERV.NLM` is running on a file server to which local networks A and B are connected directly. You want OrCAD programs to run only on computers belonging to local network A.

Initially, `HASPSERV.NLM` was loaded as follows:

```
load haspserv
```

Loaded this way, `HASPSERV.NLM` serves all the local networks in the entire internetwork. This allows computers (from local networks other than local network A) to access the NetHASP plug, and therefore, to run OrCAD programs that are not assigned to them. Additionally, computers from local network A search for the NetHASP plug throughout the entire internetwork.

Solution

1   Load `HASPSERV.NLM` as follows:

    ```
    load haspserv -net A -localnet
    ```

    The LOCALNET switch prevents HASPSERV.NLM from serving all the computers in the entire internetwork. Moreover, the NET switch ensures that `HASPSERV.NLM` serves only local network A.

2   To ensure that the computers on local network A communicate with the above `HASPSERV.NLM`, and not with others in the internetwork, add the following lines (not case-sensitive) to the [nh_ipx] section of the `NETHASP.INI` file on each computer in its Windows directory:

    ```
    nh_search_method=localnet
    nh_server_name=<name1>,<name2>
    ```

Example two: Serving specifc segments of an internetwork

An internetwork has two file servers: File Server 1 and File Server 2. File Server 1 has two segments with network numbers 111 and 222. File Server 2 has two segments with network numbers 333 and 444.

By default, the NetHASP License Manager loaded on either File Server 1 or File Server 2 serves the entire internetwork. This means that when the NetHASP License Manager is loaded with no switches, it serves network segments 111, 222, 333, and 444.

Solution: the NetHASP License Manager serves only workstations on the File Server 1 local network

1   On File Server 1, load HASPSERV.NLM with the -localnet switch:

    ```
    load haspserv -localnet
    ```

    The License Manager serves workstations on segments 111 and 222, but not on 333 or 444.

Solution: the NetHASP License Manager serves only workstations in segment 222

2   On File Server 1, load `HASPSERV.NLM` with the
    -localnet switch and an additional switch, -net:

```
load haspserv -localnet -net 222
```

The License Manager serves workstations on segment
222, but not on 111, 333, or 444.

Example three: adapting the NetHASP client

By default, an OrCAD program searches for the NetHASP
License Manager throughout the internetwork.

To have the OrCAD program search only its local network for a NetHASP License Manager

3   Edit the `NETHASP.INI` file on each computer in its
    Windows directory (not case-sensitive):

```
[NH_COMMON]
NH_IPX=enabled

[NH_IPX]
NH_SEARCH_METHOD=localnet
```

If the OrCAD program is running on a workstation in
segment 222, the program finds the netHASP License
Manager on segment 222 only. The program does not
search for a NetHASP License Manager that is active on
other local networks.

Specifying local networks with other servers

To instruct `NHSRVW32.EXE` to serve a local network:

If you load the License Manager without using the LOCALNET switch, it serves all of the networks and computers in the entire internetwork.

1   Use the following switch when loading (not case-sensitive) :

```
nhsrvw32 -localnet
```

To instruct the computer to access only the NetHASP Server program (that advertised its name with the computer's local network), do the following:

1   Add the following lines to the `NETHASP.INI` file, in the [NH_IPX] section:

```
NH_SEARCH_METHOD=Localnet
```

Setting timeout length

The License Manager cannot serve more than one request at a time. The timeout length determines how long an OrCAD program repeatedly tries to access the License Manager before giving up.

In almost all networks, the default timeout values are sufficient, so you only need to override the default values in networks that have a NetHASP connected to a slow or busy workstation or network. Typically, this type of problem is reported as an Error 3008.

To set the timeout length

By default, <m> is 2 seconds and <n> is 5 seconds. Increasing these values will not slow down license access, but it will increase the time it takes for license access to fail, so error reporting will take longer.

1   In the [NH_COMMON] section of the file `NETHASP.INI`, enter (not case-sensitive):

```
NH_SESSION=<m>
NH_SEND_RCV=<n>
```

where <M> and <N> are measured in seconds.

Update or regenerate license information

Update license information

Update the license information to add additional OrCAD program licenses to an existing license server. A new authorization code (or pair of codes) is required to add additional licenses.

- Repair your license file if it has been corrupted, lost or tampered with.

- Update your license (plug and license file) if you are adding new programs, features, or licenses.

Regenerate license information

If you are using FLEXlm licensing and are experiencing difficulties locating or accessing the LICENSE.DAT file, it is possible to regenerate this file.

Similarly, if you are using NetHASP, you can regenerate the license information for NetHASP licensing.

To update or regenerate

Your original key code and authorization code (or pair of codes) are required to update or regenerate the license information.

1   Attach the hardware lock to the parallel port on your Windows 95 or NT license server or client computer.

2   Start the OrCAD installation program and select Update Licensing on the Installation Setup screen.

3   Depending on the type of license you need to update or regenerate, select either Local License File, Client License File, or License Manager.

4   Follow the instructions on the screen.

Note   *Press F1 to access help during the Update Licensing installation process.*

NetHASP

2

Overview

This chapter describes the theory of operation and the general capabilities of the NetHASP License Management software. Although most installations can be completed during the installation of the OrCAD programs, some complex network configurations require additional information to adapt the NetHASP software to your particular network.

This chapter includes the following sections:

- Getting started on page 2-29 lists what you need for NetHASP licensing.

- The NetHASP License Manager on page 2-32 provides instructions for installing the License Manager on different platforms.

- NetHASP Automatic Search on page 2-36 describes the search mechanism used to locate the License Manager in the absence of a `nethasp.ini` file.

- describes how to customize the NetHASP search mechanism.

- describes additional switches and keywords that you can use to adjust NetHASP to your network.

- lists the switches that you can use in the License Manager.

- lists the keywords that you can edit in the `nethasp.ini` file.

- lists and describes the codes displayed when errors occur in network licensing.

Getting started

To use NetHASP licensing, you need the following (including OrCAD software):

- A red NetHASP hardware lock. (Provided with your OrCAD products.)

- Two NetHASP License Managers: `nhsrvw32.exe` for Windows 95 and Windows NT, and `haspserv.nlm` for Novell Netware servers.

  The License Manager is the program that communicates with OrCAD programs. The NetHASP hardware lock administers licenses.

- The HASP Device Driver, which is the interface between the NetHASP License Manager and the NetHASP hardware lock.

- The NetHASP Monitor, which is a utility that monitors the NetHASP License Managers on the network and the programs that are using NetHASP licenses.

- `Nethasp.ini`, necessary for most configurations, which is the NetHASP configuration file that you can edit to modify the NetHASP default settings.

Networks and protocols supported by NetHASP

NetHASP supports the following networks:

- 3Com
- Banyan Vines
- D-Link
- DEC Pathworks
- Netware Lite
- Novell 3.x, Novell 4.x
- LANtastic
- LanManager
- OS/2 Lan Server
- Any other network that uses the protocols listed below.

NetHASP supports the IPX, TCP/IP, and NetBIOS network communication protocols.

IPX support

NetHASP under IPX supports the following advertising mechanisms:

- Broadcast—while workstations broadcast to the network, the NetHASP License Manager constantly listens in.
- NetHASP Address File—advertises the NetHASP License Manager.

TCP/IP support

In environments that support TCP/IP, NetHASP operates in two modes:

- TCP, UDP—the address of NetHASP License Manager is specified.

- UDP Broadcast—while workstations broadcast to the network, the NetHASP License Manager constantly listens in.

NetBIOS support

NetHASP supports various types of NetBIOS, including Microsoft NetBEUI.

The NetHASP License Manager

You can connect the NetHASP hardware lock to any computer on the network. This computer does not necessarily have to be the network file server. OrCAD programs will communicate with the NetHASP, providing you activate the correct NetHASP License Manager.

The NetHASP License Manager is not dependent on the NetHASP or the OrCAD program; rather, it functions as a link between the two. Regardless of the environment the OrCAD program is started in—Windows 95 or Windows NT—you can activate any one of the NetHASP License Managers to communicate with the OrCAD program and the NetHASP.

The number of programs you can run depends on the number of licenses you have purchased. One NetHASP License Manager can communicate with up to 250 programs running in any environment. To communicate with more than 250 under Windows, see .

If you have products from vendors other than OrCAD that use NetHASP hardware locks, we recommend that you daisy-chain all hardware locks to a single computer. Alternatively, you can connect the hardware locks to different computers, but then you need to activate the NetHASP License Manager on each of these computers.

Note   *The NetHASP License Manager must remain active as long as any workstation is running a protected program, whether it is from OrCAD or from another vendor.*

The NetHASP License Manager should comply with the operating system of the computer it is installed on. The table below lists the various NetHASP License Managers, under which operating system you activate them, and which protocols they support.

NetHASP License Manager	IPX	TCP/IP	NetBIOS
Windows 95 and Windows NT (`NHSRVW32.EXE`)	✓	✓	✓
Netware 386 and 486 file servers (`HASPSERV.NLM`)	✓		

Note   *By default, the NetHASP License Manager for Windows 95 and Windows NT listens to all three protocols at once.*

The following sections describe how to install the NetHASP License Manager on various workstations and file servers in the network.

NetHASP on a Windows NT workstation

The NetHASP License Manager for Windows NT is `nhsrvw32.exe`.

To run the NetHASP License Manager

1   Connect the NetHASP hardware lock to the computer.

2   Enter `hinstall -i,` or run HINSTALL.EXE and click the Install button.

3   Start `nhsrvw32.exe`. A window appears with a message informing you that the NetHASP License Manager is loaded. It also shows which protocols are being used.

To remove the NetHASP License Manager

1   Exit the NetHASP License Manager program.

Automatic activation of NHSRVW32.EXE

Under Windows NT, you can start the NetHASP License Manager automatically before users log on. You implement this with `hinstall.exe`.

Hinstall installs a HASP NT Loader implemented as a Windows NT service. This program automatically starts the NetHASP License Manager. With the HASP NT Loader, you do not have to log on or interact with the Windows NT system.

To run the HASP NT loader

1   Log on as a user with Administrator Privileges.

2   Run HINSTALL.EXE.

3   Click the Network License Manager tab, confirm the server path, and click Install.

4   Restart the system. The HASP NT Loader will automatically start the NetHASP License Manager.

Note   *Since the NetHASP License Manager starts automatically before the user logs on, the HASP NT Loader and the NetHASP License Manager messages do not appear on the screen until you log on. For example, if you try to start the NetHASP License Manager without a connected HASP, you do not see the "HASP not found" error message until you log on.*

To remove the HASP NT Loader

1   Log on as a user with Administrator Privileges.

2   Run `hinstall -rs,` or run HINSTALL.EXE and click the Remove button.

3   Restart the system.

Viewing the HASP NT Loader entry in the SCManager

If the HASP NT Loader is installed properly, you can see its entry in the Windows NT SCManager (Service Control Manager).

To see the entry in the SCManager

1   In the Control Panel, double-click the Services icon. The HASP NT Loader entry appears in the Services list.

2   Check that the Status field is empty, and that the Startup field is set to Automatic.

NetHASP Automatic Search

The most critical stage in NetHASP system installation is ensuring that the NetHASP client (the workstation activating the protected program) finds the NetHASP License Manager. A communication session can begin only after the NetHASP License Manager is found.

The NetHASP Automatic Search lets you connect the NetHASP, start the NetHASP License Manager, and start the OrCAD program, without any settings or manual changes to the NetHASP system, or the environment it is installed in. Nevertheless, the search mechanism allows fine-tuning of the NetHASP system with simple client-server adjustments. This is implemented by the NetHASP Custom Search.

The NetHASP Automatic Search starts when the NetHASP client does not find a NetHASP configuration file (`nethasp.ini`). The OrCAD program then follows the following algorithm to search for the NetHASP License Manager.

Before following the automatic search algorithm, the NetHASP system first checks which protocols are installed, and performs only the relevant steps. For example, if only the TCP/IP protocol is detected, only step 2 in the algorithm is performed. It will, however, be performed up to three times, doubling the value of n each time.

Automatic search algorithm

Begin

Detect the active communication protocols

Loop up to 3 times on the procedure below

1   Obtain a license by using the IPX protocol, allowing n seconds for success

> IF successful, THEN continue using IPX
>
> ELSE
>
> 2   Obtain a license by using the TCP/IP protocol, allow n seconds for success
>
> > IF successful, THEN continue using TCP/IP
> >
> > ELSE
> >
> > 3   Obtain a license by using the NetBIOS protocol
> >
> > > IF successful, THEN continue using NetBIOS
> > >
> > > ELSE
> > >
> > > 4   Return to the beginning of the loop, and double the value of n

If all fails after the third loop, return a NetHASP error

End

Note   *When beginning the NetHASP search loop initially, n=2 seconds.*

Automatic search under IPX

Under IPX the NetHASP system uses a SAP broadcast mechanism to automate the search for the NetHASP LM. That is, the NetHASP client broadcasts, while all IPX active NetHASP License Managers listen in. The first one to answer the client, enabling a NetHASP LOGIN, will provide the NetHASP services. With the automatic search under IPX, clients and NetHASP License Managers on separate segments can communicate.

Automatic search under TCP/IP

Under TCP/IP the NetHASP system uses a UDP broadcast mechanism to automate the search for the NetHASP LM. That is, the NetHASP client broadcasts, while all TCP/IP active NetHASP License Managers listen in. The first one to answer the client, enabling a NetHASP LOGIN, will provide the NetHASP services.

Note   *With the automatic search under TCP/IP, NetHASP clients cannot access NetHASP License Managers which are located in other segments. To cross segments you need to customize the search mechanism; see* .

Automatic search under NetBIOS

Under NetBIOS

- Does not limit the search to *n* seconds, because the search mechanism takes longer than the specified search time.

- Determines which lana numbers (communication channels) are in operation and uses them for communication.

- Uses the default NetHASP NetBIOS name, AladinHaspV012.0.

Under NetBIOS, the automatic search mechanism operates according to the program type running on the NetHASP client:

- Win32 programs search on all detected lana numbers for a NetHASP hardware lock. That is, the search will not end when a NetHASP License Manager answers, but rather when a NetHASP License Manager with the right NetHASP hardware lock answers.

- The Win32 NetHASP License Manager listens on all detected lana numbers.

- Because communication on all detected lananums takes time, an attempt will be made to communicate on a single lananum per iteration session. If communication fails, the next iteration session will use the next detected lananum.

Note   *If more than three lananums are detected, when the algorithm loop terminates its three iterations, session three is performed again using the remaining unchecked lananums.*

NetHASP custom search

Under various environments you might want or need to customize the NetHASP search mechanism. This is done by setting keywords in the NetHASP configuration file (`nethasp.ini`). The `nethasp.ini` file is read by the NetHASP client (the OrCAD program) to customize the search mechanism. If the NetHASP client finds a `nethasp.ini` file, it reads it and uses the information. If not, the above NetHASP Automatic Search will take effect.

The NetHASP client searches for the `nethasp.ini` file in the following locations:

- Current directory
- Windows directory
- Windows system directory
- Protected executable file directory (directory from where the program is started)
- Path
- Netware search map

The following sections describe the `nethasp.ini` keywords used for basic custom of the NetHASP search mechanism. Not all `nethasp.ini` keywords are described in these sections.

describes its keywords, which you can use for additional fine-tuning of the NetHASP system.

General custom search information

To change the initial value of *n*—the first search period in the NetHASP algorithm loop—from its default value of 2 seconds, perform the following:

```
[NH_COMMON]
NH_SESSION=<seconds>
```

where `<seconds>` is the desired initial search period.

Setting the NH_SESSION keyword in the NH_COMMON section, applies to all protocols. To set the session value for each protocol independently, set the value of NH_SESSION in each protocol section separately.

Custom search under IPX

Custom search under IPX is program-dependent. The following sections describe two custom options for Windows 95 and Windows NT programs.

Broadcast

1 Start the NetHASP License Manager with the `-ipx` switch.

2 Edit the `nethasp.ini` file and copy it to a location accessible by the program:

```
[NH_COMMON]
NH_IPX=enabled
```

```
[NH_IPX]
NH_USE_BROADCAST=enabled
```

This option is the default. It is also used by the Automatic Search (session (1) in the algorithm loop). It instructs the NetHASP system to use the SAP broadcast mechanism to search for the NetHASP License Manager.

Address file

1   Start the NetHASP License Manager with the -ipx switch.

2   Edit the `nethasp.ini` file and copy it to a location accessible by the program:

```
[NH_COMMON]
NH_IPX=enabled
```

```
[NH_IPX]
NH_USE_BROADCAST=disabled
```

This option instructs the NetHASP system to search for the NetHASP License Manager using the address file mechanism (looks for the NetHASP LM address in the `HASPADDR.DAT` file).

General notes for IPX

By default, a NetHASP address file (`HASPADDR.DAT`) is created in the directory of the NetHASP LM. When using `HASPSERV.NLM`, the address file is created in the server root directory. The NetHASP client searches for the address file in the following order:

- current directory
- Windows directory
- Windows system directory
- protected executable file directory (directory from where the program is activated)
- path
- Netware search map

Adding to the Address File Mechanism options described above, you can specify the location path for the address file.

```
NH_DATFILE_PATH=<path>
```

Custom search under TCP/IP

The following sections describe three custom options for TCP/IP.

UDP broadcast

1   Start the NetHASP License Manager with the `-tcpip` switch.

2   Edit the `nethasp.ini` file as follows:

```
[NH_COMMON]
NH_TCPIP=enabled
[NH_TCPIP]
NH_TCPIP_METHOD=udp
NH_USE_BROADCAST=enabled
```

This option is the default. It is also used by the Automatic Search (session two in the algorithm loop). It instructs the NetHASP system to use the broadcast mechanism to search for the NetHASP License Manager.

3   Copy the file to a location accessible by the program.

Note that with this setting, NetHASP clients cannot access NetHASP License Managers which are located in other segments, nor can they cross firewalls.

UDP

1   Start the NetHASP License Manager with the `-tcpip` switch.

2   Edit the `nethasp.ini` file and copy it to a location accessible by the program:

```
[NH_COMMON]
NH_TCPIP=enabled

[NH_TCPIP]
NH_TCPIP_METHOD=udp
NH_SERVER_ADDR=<IP address of the NetHASP
LM>
NH_USE_BROADCAST=disabled
```

43

This option instructs the NetHASP client to search for a NetHASP License Manager with the specified IP address. This option does not resolve Host names using DNS. This option lets the NetHASP cross segments.

TCP

1   Start the NetHASP License Manager with the -tcpip switch.

2   Edit the `nethasp.ini` file and copy it to a location accessible by the program:

```
[NH_COMMON]
NH_TCPIP=enabled

[NH_TCPIP]
NH_TCPIP_METHOD=tcp
NH_SERVER_ADDR=<IP address of the NetHASP
LM, or NetHASP LM Host name>
```

This option instructs the NetHASP client to search for a NetHASP License Manager with the specified IP address or Host name. This option lets NetHASP cross segments.

Note   *This option triggers automatic dialing on computers configured for dial-up networking. This is a known side effect of initiating a TCP session on such a computer. If your only use of TCP/IP is for dial-up networking, you should use one of the two options above, or a different protocol.*

General notes for TCP/IP

NetHASP will not cross most firewalls or packet filters. To support UDP most firewalls use packet filtering, TCP connections can be supported with either proxies or packet filtering. NetHASP uses port 475 for both TCP and UDP. If NetHASP is configured for UDP, both the client workstation and the license manager will initiate connections. If NetHASP is configured for TCP, only the client workstation will initiate connections. To configure your firewall to allow NetHASP to work, please consult your network administrator.

The NetHASP License Manager defaults to port number 475.

If you want it to listen to another port

1   Start the NetHASP License Manager with both the `-tcpip` and `-portnum=<number>` switches.

2   Make the following additional change to the `nethasp.ini` file:

```
[NH_TCPIP]
NH_PORT_NUMBER =<number>
```

General note for NetBIOS

The NetBEUI protocol does not cross segments. If you are starting the protected program in a NetBEUI environment, where the NetHASP License Manager and the NetHASP clients are on separate segments, the NetHASP clients will not find the NetHASP LM. In this case, you need to select the "IPX/SPX compatible transport with NetBIOS" and set it as the default protocol to enable the communication to cross segments.

Adapting NetHASP to the network

This section describes additional NetHASP License Manager switches and `nethasp.ini` keywords that you can use to further adjust the NetHASP system to your network environment.

Defining the range of workstations under IPX

To allow only some of the workstations to access NetHASP on an IPX-based network:

1   Start the NetHASP License Manager with the `-ipxnosap` switch. This ensures that the address of the NetHASP License Manager is advertised in the NetHASP address file, `HASPADDR.DAT`, and that the NetHASP License Manager will not respond to SAP broadcast requests.

2   Edit the `nethasp.ini` file:

```
[NH_COMMON]
NH_IPX=enabled

[NH_IPX]
NH_USE_BROADCAST=disabled
NH_USE_BINDERY=disabled
```

This instructs the protected program running on workstations in the range to search for the address file and read the address of the NetHASP License Manager.

3   Copy the protected program and the `nethasp.ini` file to the same directory. Ensure that each workstation in the range, and only the workstations in the range, start the program from this directory.

Defining the range of workstations under TCP/IP

This capability is not currently available.

Defining the range of workstations under NetBIOS

To allow only some of the workstations to access the hardware lock in a NetBIOS-based network

1   Start the NetHASP License Manager with the
     `-nbname` switch set to a name of your choice (up to 8
     characters, not case-sensitive). This defines the
     NetBIOS name of the NetHASP License Manager.

     For example, to start `haspserv.exe` and define the
     NetBIOS name of the workstation as `firstsrv`, enter:

     ```
     haspserv -nbname=firstsrv
     ```

2   Edit the `nethasp.ini` file:

     ```
     [NH_COMMON]
     NH_NETBIOS=enabled

     [NH_NETBIOS]
     NH_NBNAME=firstsrv
     ```

     This links the workstations to the NetBIOS name.

3   Copy the protected program and the nethasp.ini file to
     the same directory. Make sure that each workstation
     in the range, and only workstations in the range, start
     the program from this directory.

Note   *You cannot have two workstations with the same NetBIOS name. If
        you want to start the NetHASP License Manager on more than one
        workstation, you must define a different NetBIOS name for each
        NetHASP License Manager.*

Adapting the timeout length

The NetHASP License Manager cannot serve more than one request at a time. The timeout length determines how long an OrCAD program repeatedly tries to access the NetHASP License Manager before giving up.

In almost all networks, the default timeout values are sufficient, so you only need to change the default values in networks that have a NetHASP connected to a slow or busy workstation.

To define the timeout length

If you are experiencing timeouts (error 3008), with only one particular protocol, then add these options to that protocol's section (i.e., for IPX, add them to the [NH_IPX] section). If you are experiencing timeouts more generally, then add these options to the [NH_COMMON] section.

1   In the appropriate section of the `nethasp.ini` file, set:

```
NH_SESSION=<m>
NH_SEND_RCV=<n>
```

where `m` and `n` are measured in seconds. By default, m is 2 seconds and n is 5 seconds.

Defining the number of served protected programs

NetHASP License Manager for Windows 95 and Windows NT

The NetHASP License Manager for Win32 can serve more than the default 250 protected programs. This is done by starting the LM with a switch:

```
nhsrvw32 -userlist=n
```

where `n` is the number of protected programs it serves.

NetHASP in Novell local networks and internetworks

This section describes Novell networks that consist of several interconnected networks using the IPX protocol.

With the NetHASP system, *local network* is a flexible term. The NetHASP License Manager and NetHASP client (the OrCAD program) refer to the local network differently. The NetHASP License Manager refers to the local network as all the segments directly connected to the NetHASP License Manager. The NetHASP client refers to the local network as the current segment.

An *internetwork* is a group of two or more local networks linked together.

The example below shows how you can set the NetHASP client and NetHASP License Manager to work on either a local or internetwork.

Adapting the NetHASP License Manager

The internetwork in this example has two file servers: File Server 1 and File Server 2. File Server 1 has two segments with network numbers 111 and 222. File Server 2 has two segments with network numbers 333 and 444.

By default, the NetHASP License Manager started on either File Server 1 or File Server 2 serves the entire internetwork. That means that if you start the NetHASP License Manager with no switches, it serves network segments 111, 222, 333, and 444.

If you want the NetHASP License Manager to serve only workstations in the File Server 1 local network (workstations on segments 111 and 222):

On File Server 1, start `haspserv.nlm` with the `-localnet` switch as follows:

```
load haspserv -localnet
```

Now the NetHASP License Manager will not serve workstations on segments 333 and 444.

49

If you want the NetHASP License Manager to serve only workstations in segment 222:

On File Server 1, start the NetHASP License Manager with the additional switch, -net, as follows:

```
load haspserv -localnet -net 222
```

Now the NetHASP License Manager will serve workstations on segment 222, but not those on segments 111, 333, or 444.

Adapting the NetHASP client

By default, a protected program searches for the NetHASP License Manager throughout the internetwork.

If you want the protected program to search only its local network for a NetHASP License Manager:

Edit the `nethasp.ini` file as follows:

```
[NH_COMMON]
NH_IPX=enabled

[NH_IPX]
NH_SEARCH_METHOD=localnet
```

If the protected program is running on a workstation in segment 222, the program finds the NetHASP License Manager on segment 222 only. The program does not search for a NetHASP License Manager that may be active in other local networks.

Setting the default protocol on a network workstation

Windows 95 workstation

1   In the Control Panel, select Network to display the Network dialog box.

2   Double-click the protocol name on the Configuration tab.

3   Choose the Advanced tab.

4   Select Set this protocol to be the default protocol, and then click OK.

5   Restart the system.

Windows NT workstation

1   In the Control Panel, select Network.

2   Select the Services tab.

3   Double-click the NetBIOS Interface to display the NetBIOS Configuration dialog box.

4   Edit the lana numbers so that the default protocol you want is associated with lana number 0000, and then click OK.

5   Restart the system.

Note   *Nbf is NetBEUI.*
NwlnkNb is IPX/SPX-compatible with NetBIOS.
NetBT is TCP/IP.

NetHASP License Manager switches

The table below lists all the switches available for the different NetHASP License Managers.

Table 1   *NetHASP License Manager switches*

Switch	HASPSERV.NLM	NHSRVW32.EXE
-?	✓	
-addrappend	✓	✓
-addrpath	✓	✓
-help	✓	
-ibm	✓	✓
-ibm36	✓	✓
-ipx		✓
-ipxnosap	✓	✓
-localnet	✓	✓
-maxlananum		✓
-nbname		✓
-nec	✓	
-net	✓	
-netbios		✓
-nethaspnbname		✓
-openmin		✓
-portnum		✓
-saptofile	✓	
-srvname	✓	✓
-tcp/ip		✓
-uselananum		✓
-userlist		✓

The table below describes the usage of the NetHASP
License Manager switches.

Table 2   *NetHASP License Manager switch descriptions*

Switch	Explanation
`-?`	Displays a list of available switches.
`-addrappend`	Adds the NetHASP License Manager address to the already existing addresses in previously created address files. Does not overwrite the `HASPADDR.DAT` file.
`-addrpath=` `<path>`	Specifies a location for saving `HASPADDR.DAT`. By default, the NetHASP License Manager places `HASPADDR.DAT` in the directory where it is started.
`-broadcast`	Uses the IPX Broadcast mechanism. With the Broadcast mechanism, while the clients broadcast to the network, the NetHASP License Manager constantly listens in.
`-help`	Displays a list of available switches.
`-ibm`	Default. Searches IBM PCs for standard NetHASP hardware locks only.
`-ibm36`	Search on (IBM PCs) first for a NetHASP36 and then for a standard NetHASP.
`-ipx`	Instructs the NetHASP system to use the IPX protocol with SAP.

Table 2   *NetHASP License Manager switch descriptions*

Switch	Explanation
-ipxnosap	Instructs the NetHASP system to use only the IPX protocol (without SAP). (With the NetHASP License Manager for Win32, you can still start other protocols by using either the -tcpip or -netbios switch or both.)
	When you use this switch, the NetHASP License Manager creates a file called HASPADDR.DAT. This file contains the node address of the workstation running the NetHASP License Manager. When you start the NetHASP License Manager with this switch, only protected programs with access to HASPADDR.DAT can communicate with the NetHASP License Manager.
-localnet	If you want the NetHASP License Manager to serve only requests from workstations in the local network.
	If requests come from workstations that are not part of the local network, the NetHASP License Manager returns error 140.
-maxlananum=<number>	When you start the NetHASP License Manager, it automatically searches the lana numbers (communication channels) for the NetBIOS protocol. By default, it searches lana numbers 0 to 6. Use this switch to change this range. The NetHASP License Manager searches the lana numbers from 0 to <number>-1.
-nbname=<name>	Assigns a NetBIOS name to the NetHASP License Manager. This switch is identical to the -nethaspnbname switch.

Table 2   *NetHASP License Manager switch descriptions*

Switch	Explanation
`-nec`	Search NEC computers to search for NetHASP36 hardware locks only.
`-net`	Advertises the NetHASP License Manager address with a specific local network number.
`-netbios`	Forces the NetHASP system to listen only to the NetBIOS protocol. (With the NetHASP License Manager for Win32, you can still start other protocols by using either the `-tcpip` or `-ipxnosap` switch or both.)
`-nethaspnbname= <name>`	Assign a NetBIOS name to the NetHASP License Manager. This switch is identical to the `-nbname` switch.
`-openmin`	Starts the NetHASP License Manager program with its window minimized. By default the windows is maximized.
`-portnum= <number>`	Instructs the NetHASP License Manager to listen to the port you specify, if you are using the TCP/IP protocol. The default Port Number is 475.
`-saptofile`	Creates the `HASPADDR.DAT` file. This file contains the node address of the workstation running the NetHASP License Manager.
`-srvname= <name>[,name]`	Assigns one or more names to the NetHASP License Manager. You can assign up to six names.

Table 2   *NetHASP License Manager switch descriptions*

Switch	Explanation
`-tcpip`	Forces the NetHASP system to listen only to the TCP/IP protocol. (With the NetHASP License Manager for Win32, you can still start other protocols by using either the `-ipxnosap` or `-netbios` switch or both.)
`-uselananum=` `<x>[,<x>]`	Instructs the NetHASP License Manager to listen to specific lana numbers.
`-userlist`	Limits the maximum number of protected programs.

NetHASP configuration file template

The following are keywords used in the [NH_COMMON] section of the `nethasp.ini` file.

Table 3   *[NH_COMMON] keywords*

Keyword	Value	Meaning
Section-specific keywords		
NH_IPX	enabled or disabled	use the IPX protocol
NH_NETBIOS	enabled or disabled	use the NETBIOS protocol
NH_TCPIP	enabled or disabled	use the TCP/IP protocol
NH_MACHINE	IBM or NEC default: IBM	the computer used to run the protected program
General keywords		
NH_SESSION	<Num>	See <u>Adapting the timeout length on page 2-48</u>.
NH_SEND_RCV	<Num>	

The table below lists keywords used for the IPX protocol, in the [NH_IPX] section of the `nethasp.ini` file.

Table 4   *[NH_IPX] keywords*

Keyword	Value	Meaning
Section-specific keywords		
NH_USE_BINDERY	enabled or disabled default: disabled	Use IPX with bindery. Ignored under Win32 API. This switch replaces older switch named `NH_USE_SAP`.
NH_USE_BROADCAST	enabled or disabled default: enabled	Use IPX Broadcast mechanism.
NH_BC_SOCKET_NUM	<Number> default: 7483H	Broadcast socket number (HEX).
NH_USE_INT	2F_NEW or 7A_OLD default: 2F_NEW	2F_NEW means that IPX protocol will use interrupt 2Fh ONLY. 7A_OLD means that IPX protocol will use interrupt 7Ah ONLY.

Table 4   *[NH_IPX] keywords (continued)*

Keyword	Value	Meaning
General keywords		
NH_SEARCH_METHOD	Localnet or Internet	Specfies how much of the network to search when looking for a License Manager. See <u>NetHASP in Novell local networks and internetworks on page 2-49</u>.
NH_SERVER_NAME	<Name1> [, Name2]	Communicate with the NetHASP Server with the specified name. Maximum: 6 names, up to 7 case-insensitive characters each.
NH_SESSION	<Num>	See <u>Adapting the timeout length on page 2-48</u> .
NH_SEND_RCV	<Num>	
NH_DATFILE_PATH	<path>	Specify the location of the NetHASP License Manager's address file.

The table below lists keywords used for the NetBIOS protocol, in the [NH_NETBIOS] section of the `nethasp.ini` file.

Table 5   *[NH_NETBIOS] keywords*

Keyword	Value	Meaning
Section-specific keywords		
NH_NBNAME	\<Name\>	Assign a name to the NetHASP License Manager. One name limit, up to eight case-insensitive characters.
NH_USELANANUM	\<Num\>	Assign a lana number to be used as a communication channel.
General keywords		
NH_SESSION	\<Num\>	See <u>Adapting the timeout length on page 2-48</u>.
NH_SEND_RCV	\<Num\>	

The table below lists keywords used for the TCP/IP protocol, in the [NH_TCPIP] section of the `nethasp.ini` file.

Table 6   *[NH_TCPIP] keywords*

Keyword	Value	Meaning
Section-specific keywords		
NH_SERVER_ADDR	\<Addr1>, \<Addr2>	IP addresses of all the NetHASP License Managers you want to search.
		You can use multiple lines and unlimited addresses.
		Examples: IP address: 192.114.176.65 Local Hostname: ftp.aladdin.co.il
NH_PORT_NUMBER	\<Num> The default number is 475.	Set the TCP/IP port number. (Optional.)
NH_TCPIP_METHOD	TCP or UDP Default: UDP	Send a TCP packet or UDP packet.
NH_USE_BROADCAST	Enabled or Disabled Default: Enabled	Use TCPI/IP Broadcast mechanism.

Table 6   *[NH_TCPIP] keywords (continued)*

Keyword	Value	Meaning
General Keywords		
NH_SERVER_NAME	<Name1> [,Name2]	Communicate with the NetHASP Server with the specified name.
		Limit: 6 names, up to 7 case-insensitive characters each.
NH_SESSION	<Num>	See Adapting the timeout length on page 2-48.
NH_SEND_RCV	<Num>	

NetHASP status codes

The table below lists and describes the codes displayed during network licensing.

Table 7   *NetHASP server status codes*

Code	Description
3001	The IPX or NetBIOS protocols have not been installed properly.
	Act accordingly.
3002	Communication error; unable to get the socket number. Applies to IPX and TCP/IP.
	Verify that the protocol is installed properly.
3003	Communication error.
	NetBIOS: unable to establish the session.
	IPX: unable to get the immediate address of the NetHASP License Manager. Verify that the protocol is installed properly.
	TCP/IP: unable to connect to the server. Verify that the server address is correct.
3004	No NetHASP server was found. Check whether your software has a path to, and read permission for, the address file.
	TCP/IP: the network is busy. See Adapting the timeout length on page 2-48.
3005	Cannot read the NetHASP Server address file. See General notes for IPX on page 2-42.
	TCP/IP: the network is busy. See Adapting the timeout length on page 2-48.
3006	Cannot close the NetHASP Server address file.
	TCP/IP: the network is busy. See Adapting the timeout length on page 2-48.
3007	Communication error.
	IPX, NetBIOS: failed to send packet.
	TCP/IP: the network is busy. See Adapting the timeout length on page 2-48.

Table 7   *NetHASP server status codes*

Code	Description
3008	No answer from the NetHASP License Manager.
	NetBIOS: the network is busy.
	IPX: the network is busy, or incorrect address files were found. In the latter case, delete all copies of haspaddr.dat and HASPADDR.DAT. See General notes for IPX on page 2-42.
3010	Internal software error. This error sometimes occurs when the License Manager restarts.
3011	NetBIOS: communication error. Verify that the protocol is installed properly.
3015	No active NetHASP License Manager was found.
	See NetHASP custom search on page 2-40.
3018	Cannot log in because the SetServerByName call was not successful.
3019	Syntax error in the configuration file, or a NetHASP environment variable has an invalid setting.
	If LastStatus displays 0 in Par2, there is an environment variable with an invalid setting.
3020	Error handling the configuration file.
3021	NetHASP did not succeed in allocating memory under Windows. Try to free DOS memory.
3022	NetHASP did not succeed in freeing DOS memory under Windows.
3023	Invalid NetHASP memory address.
3024	Invalid NetHASP service.
3025	Failed to load winsock.dll. Applies only to TCP/IP.
3026	Failed to unload winsock.dll. Applies only to TCP/IP.

Table 7   *NetHASP server status codes*

Code	Description
3129	The appropriate NetHASP hardware lock is not connected to the NetHASP Server.
3130	The license you are trying to obtain is not in the list of configured options.
3131	Error reading from NetHASP memory.
3132	Error writing to NetHASP memory.
3133	The current license request exceeds the number of licenses purchased.
3134	The current license request exceeds the number of authorized activations for the software.
3135	Internal software error. This error sometimes occurs when the License Manager restarts.
3136	The server is busy. This may occur if your NetHASP system is not well adapted to the network. See Adapting the timeout length on page 2-48.
3137	There is no space in the NetHASP Log Table.
3138	NetHASP internal error. The number of authorized workstations in larger than the maximum number designated by the NetHASP model.
3139	The NetHASP Server failed and was restarted, or and internal NetHASP error occurred.
3140	The NetHASP Server does not serve your workstation's network. See NetHASP in Novell local networks and internetworks on page 2-49.
3141	Invalid service.

Table 7   *NetHASP server status codes*

Code	Description
3142	Cannot find the NetHASP Server matching the name specified in the NetHASP configuration file.
3150	No NetHASP Server with the assigned name was found. This error is displayed by the SetServerByName service.
3151	Two or more different NetHASP Servers with the assigned name were found. This error is displayed by the SetServerByName service.

FLEXlm

3

Overview

This chapter includes the following sections:

- explains the basics of floating (network) licensing, describes the components of FLEXlm, and explains when license administrators have control and when users have control.

- tells both license administrators and users how to start managing FLEXlm.

- describes the contents of FLEXlm license files.

- explains the use of the License Finder.

- describes how to administer licenses from more than one vendor or multiple products from the same vendor.

- describes your options of which nodes to use as license server nodes.

- describes utilities for the license administrator to help manage the licensing activities on the network.

- explains FLEXlm informational messages.

- explains FLEXlm messages pertaining to problems in configuration.

- explains FLEXlm messages pertaining to daemon software errors.

- documents areas of FLEXlm that have given customers difficulty in the past.

- describes ways to find out the hostids of different types of machines.

Overview of FLEXlm

FLEXlm is best known for its ability to allow software licenses to be available (or float) anywhere on a network, instead of being tied to specific computers.

Floating licensing benefits both users and license administrators. Users can make more efficient use of fewer licenses by sharing them on the network. License administrators can control who uses the licensed program, and the node(s) where the licenses will be available.

See Types of license files on page 3-91 for details about the different licensing models supported by FLEXlm.

FLEXlm components

The four main components of FLEXlm are:

*   license manager daemon
*   vendor daemon
*   license file
*   FLEXlm program

The License Manager daemon (lmgrd)

The license manager daemon (lmgrd) handles the initial contact with the client programs, passing the connection on to the appropriate vendor daemon. It also starts and restarts the vendor daemons.

FLEXlm permits multiple redundant license manager daemons on different server nodes, allowing you to make your license available if any two out of the three server nodes is running. With redundant servers, you don't have to split up your licenses among multiple servers, or rely on any one machine.

lmgrd is not present on Netware systems.

The vendor daemon

In FLEXlm, licenses are handled by running processes. There is one process for each vendor who has a FLEXlm-licensed product on the network. This process is called the vendor daemon. The vendor daemon keeps track of how many licenses are checked out and who has them. If the vendor daemon terminates for any reason, all users lose their licenses. Users normally regain their license automatically when lmgrd restarts the vendor daemon.

Client programs communicate with the vendor daemon through TCP/IP or UDP/IP sockets. The client node (where the program runs) and the daemon processes (the license server) can run on separate nodes on your network. Also, the traffic between the client and the license manager daemon is machine-independent, allowing for heterogenous networks. This simply means the license server and the workstation running a program can be either different hardware platforms or different operating systems.

The license file

Licensing data is stored in a text file called the license file. The license file is created by the license administrator. It contains information about the server nodes and vendor daemons, and at least one line of data (called FEATURE or INCREMENT lines) for each licensed product. Each FEATURE line contains a license plug based on the data in that line, the hostids specified in the SERVER lines, and other vendor-specific data.

In some environments, the licensing information for several vendors may be combined into a single license file. See the table below for more information.

Table 8 *Default location of the FLEXlm license file*

On this platform...	The default location is this...
Windows 95 or Windows NT	`C:\flexlm\license.dat`
Netware	`SYS:\SYSTEM\flexlm\license.dat`

Installing to another location

Users can usually install to another location by setting the environment variable LM_LICENSE_FILE to point elsewhere, or by following instructions supplied with the licensed program. If your site has software from multiple vendors with incompatible license files (due to different sets of servers), you can keep the data in separate files and set the LM_LICENSE_FILE variable to reference multiple files. For details, see The license file on page 3-77.

The program

The program using FLEXlm is linked with the program module (called the FLEXlm client library) that provides the communication with the license server.

During execution, the program communicates with the vendor daemon to request a license.

The license request process

When you run a FLEXlm-licensed program, the following occurs:

1 The license module in the client program finds the license file, which includes the host name of the license server node and port number of the license manager daemon, lmgrd.

2 The client establishes a connection with the license manager daemon (lmgrd) and tells it what vendor daemon it needs to talk to.

3 lmgrd determines which machine and port correspond to the master vendor daemon and sends that information back to the client.

4 The client establishes a connection with the specified vendor daemon and sends its request for a license.

5   The vendor daemon checks in its memory to see if any licenses are available and sends a grant or denial back to the client.

6   The license module in the program grants or denies use of the feature, as appropriate.

Configuring FLEXlm

Most of the parameters of FLEXlm are configurable by the license administrator. The license administrator can set:

- the location of the license file

- the location of all executables

- the location of all log files

- the TCP/IP port number used by the license manager process, lmgrd

In addition, the license administrator can reserve licenses for specific users, nodes, or groups, and control other license-related options. Changing parameters is discussed in .

Refer to your vendor's documentation before attempting to change file names, locations, or contents.

Getting started checklist

The following sections provide a quick overview of how to set up and use licensing for FLEXlm-licensed products. By scanning the list, you should be able to quickly find the areas of interest. Cross-references point to more details in other parts of this manual.

Installing licensed software

As a license administrator, you are responsible for setting up licensing on your system or network. This section tells you how to do that. If you are a user of the program who is not involved in installing it, then see Notes for end users on page 3-76.

Remember that the installation guide for your program is the final word on installing and configuring FLEXlm.

Generally, however, installing FLEXlm licensing requires the following steps:

1   Selecting your license server nodes and get their hostids. See Hostids for FLEXlm-supported machines on page 3-149.

2   Giving the hostids to your software vendor and get a license file (or the data to enter in the license file) in return.

3   Determining how the new license file relates to any other license files that may already be on your system, and install it appropriately. Combining license files on page 3-96.

4   Determining if an options file is desired, and if so, set it up.

5   Determining where to install the FLEXlm utility programs such as lmgrd, lmstat, and lmdown, and then installing the programs (the program's installation setup usually does that for you).

6   Start lmgrd (the license daemon) manually; optionally, you may also want to set it up to start automatically at boot time. See <u>Setting the path to the license file at start-up on page 3-79</u>.

These steps are discussed briefly below, with cross-references to the appropriate locations for more detail.

License servers and hostids

Before running any FLEXlm-licensed program using floating licenses, you will need to set up your license server node (or nodes). You must select which node or nodes to run your license servers on, and provide the hostid of those machines to your software vendor. For pointers on selecting your server machine, see <u>Selecting server nodes on page 3-102</u>.

You can get the hostid of the server machine by running FLEXlm's lmhostid utility on that machine. If you don't have lmhostid, you can get the hostid of your machine by using the appropriate command as described in <u>Hostids for FLEXlm-supported machines on page 3-149</u>.

After giving the hostid of your server machines to your software vendor, the vendor will send you a license file that enables their program.

lmgrd and license files

Once you have received a license file from your vendor, you must install it on your system and start up the license manager daemon, lmgrd.

If you have multiple FLEXlm-licensed products, take care to avoid licensing conflicts. For details, see <u>Combining license files on page 3-96</u>.

Your software vendor may have selected a default location for your license file. If not, you can use any location you wish. For more details, see <u>The license file on page 3-77</u>.

Some vendors provide special scripts to start up the license daemon. If not, you can run the lmgrd program directly. To start lmgrd automatically at boot time, you will have to use the FLEXlm Control Panel.

For details, see Setting the path to the license file at start-up on page 3-79.

Administration tools

Globetrotter Software supplies administration tools to your software vendor. The vendor usually includes them with their product. The recommended location for the tools is described below, but you can install them in a different location (or not at all). See License administration tools on page 3-121 for more information.

Table 9 *Where to install FLEXlm administration tools*

On this platform...	Install the tools here...
Windows 95 or NT	C:\flexlm

Options files

The options file controls various options such as reservations and timeouts of licenses. Most users run without an options file, but you may decide you want to use some options. For example, many administrators use an option to limit the quantity and content of logged messages. To set up an options file, see The options file on page 3-108.

Notes for end users

As a user of a FLEXlm-licensed program, you may need to know a few things to use the system effectively. The main things you need to know are:

- How to tell an program which license file to use.

- How to query the system to find out who is using a license.

How to specify a license file

The license file determines what features are available to a program. It also contains information telling the program how to connect to the license server.

For information about the standard way of specifying a license file for an program, see The license file on page 3-77.

Getting information about licenses

To find out who is using a license run lmstat, described in License administration tools on page 3-121.

The license file

The license file contains all site-specific information required by FLEXlm. This information includes:

- Server names and hostids

- Vendor names and paths to vendor daemon executables

- Feature information

In general the license file, or a copy of it, must be accessible to every machine that runs a FLEXlm-licensed program, and each machine designated as a license server. Before you can use the program, you have to start the license manager daemon (lmgrd). See the table below for the correct syntax.

Table 10   *Starting the License Manager daemon*

On this platform...	Use this command...
Windows 95 or NT	`lmgrd -app [-c license_file_path]` `[-t connect_timeout]`

On Netware systems, there is no lmgrd and the vendor daemon is run directly.

See and for information on how programs find an appropriate license file.

Syntax options

`[-c license_file_path]` is the full pathname to the license file. Only use this option if the license file is not installed in the default location. See for information about using alternate locations.

`[connect_timeout]` is the timeout, in seconds, for `connect' calls to other daemons if operating in redundant-server mode.

Specifying location of the license file

If your software vendor recommends a specific location for your license file, or if the default is not practical for you, use one of the following two methods to put the license file in another location:

- Set the path when you start lmgrd

- Set the path with an environment variable

If you are running the program on multiple nodes, you have three options for making your license file available on all the machines:

- Place the license file in a partition which is available to all nodes in the network that need the license file.

- Copy the license file to all of the nodes where it is needed.

- Read the license file data from lmgrd by specifying LM_LICENSE_FILE as `port@host`, where host is the server hostname and port is the port number from SERVER line the license file. Note that `port@host` is only supported in FLEXlm version 2.4 or later versions.

Since the vendor daemon keeps track of license usage, and since the license file contains encrypted data to protect it against modification, you can move and copy the license file as much as necessary.

You can only start lmgrd on the server node specified in the license file.

If you are running redundant servers, you should have one copy of the license file (as well as the binaries for lmgrd and the vendor daemons) on each server node.

If you do not do this, you lose all the advantages of having redundant servers, since the node holding these files becomes a single point of failure.

No matter which option you choose, you must first install lmgrd and the vendor daemon.

Setting the path to the license file at start-up

To start lmgrd on Windows NT systems

1   Add lmgrd to the list of available NT services. Check the Control Panel's Services icon to verify the status of the server.

Some programs do not recognize the LM_LICENSE_FILE environment variable.

Setting the path with an environment variable

Use the environment variable LM_LICENSE_FILE to set the location of the license file.

For Windows 95, add the following line to C:\AUTOEXEC.BAT:

```
SET LM_LICENSE_FILE=LICENSE_FILE_PATH
```

On Windows NT, use the System Control Panel to change the global environment, adding LM_LICENSE_FILE to LICENSE_FILE_PATH.

LICENSE_FILE_PATH is the full pathname to the license file. This can also be a PORT@HOST setting, where port and host are the port number and hostnames from the SERVER line in the license file.

See Using separate license files on the same server node on page 3-99 for more information about LM_LICENSE_FILE.

\ characters as continuation characters are only legal syntax in FLEXlm version 3.0 and later. Previous versions required that each line in the license file fit on a single line.

License file format

License files usually begin with a SERVER line (or three lines for redundant servers) followed by one or more DAEMON lines, followed by one or more FEATURE lines. In some cases the license file requires no SERVER line and no DAEMON line. See Counted vs. uncounted licenses on page 3-107 for more information.

You can modify five data items in the license file:

- node names on the SERVER line(s)
- port numbers on SERVER line(s)
- pathnames on the DAEMON line(s)
- options file pathnames on DAEMON line(s)
- values in name=value pairs on FEATURE lines, if name is all lowercase

Everything else is used to compute the license plug, and should be entered exactly as supplied by your software

vendor. All data in the license file is case-sensitive, unless otherwise indicated.

In the following sections, options modifiable by the license administrator are *italicized*.

SERVER lines

The SERVER line specifies the node name and hostid of the license server, and the port number of the license manager daemon (lmgrd). Typically, a license file has one SERVER line. Three SERVER lines means that you are using redundant servers. The absence of a SERVER line means every FEATURE or INCREMENT line in the license file is uncounted. For more information about uncounted features, see FEATURE or INCREMENT lines on page 3-82.

License administrators do not have the option of deleting SERVER lines from a license file because the hostids from the SERVER lines are encrypted into the passwords on every FEATURE and INCREMENT line. For more information about redundant servers, see Selecting server nodes on page 3-102.

The format of the SERVER line:

```
SERVER nodename id [port-number]
```

Where:

[nodename] is the system hostname. On Windows NT, IPCONFIG/ALL returns the hostname. On Windows 95, WINIPCFG/ALL returns the hostname.

[id] is the string returned by the lmhostid command.

[port-number] is the TCP port number to use.

The port number is optional, but if not specified, then the FLEXlm TCP service must be present in the network services database. The port number is usually specified.

Example:

```
SERVER enterprise 0122345 1701
```

DAEMON lines

The DAEMON line specifies the daemon name and path. The format of the DAEMON line:

```
DAEMON daemon-name daemon_path
[options_file_path]
```

Where:

`daemon-name` is the name of the vendor daemon used to serve some feature(s) in the file. This name cannot be changed by the administrator.

`path` is the pathname to the executable for this daemon. Generally the license administrator is free to install the daemon anywhere. (It is recommended however, that it be installed on the same license server node as the lmgrd.)

FLEXlm does not require an options file.

`[options_file_path]` is the full pathname of the end-user specified options file for this daemon. (See The options file on page 3-108.)

FEATURE or INCREMENT lines

INCREMENT is not available for pre-version 2.61 FLEXlm clients or servers.

A FEATURE line describes the license to use a product. An INCREMENT line can be used in place of a FEATURE line, as well as to incrementally add licenses to a prior FEATURE or INCREMENT line in the license file.

Only the first FEATURE line for a given feature will be processed by the vendor daemon. If you want to have additional copies of the same feature (for example, to have multiple node-locked counted features), then you must use multiple INCREMENT lines.

INCREMENT lines form license groups based on the feature name, version, and node-lock hostid. If the feature name, version, and node-lock hostid (and optionally, the vendor string, if the vendor specified this) match a prior INCREMENT or FEATURE line, the new number of licenses is added to the old number. If any of the three do not match, a new group of licenses is created in the vendor daemon, and this group is counted independently from others with the same feature name. A FEATURE line does

not give an additional number of licenses, whereas an INCREMENT line ALWAYS gives an additional number of licenses.

There is a rarely used option in FLEXlm which causes FEATURE lines to function as INCREMENT lines. This option is called `ls_use_all_feature_lines`. You will have to ask your vendor if they use this option. If they do, then all FEATURE lines behave exactly as INCREMENT lines.

A FEATURE line placed after another FEATURE or INCREMENT line will be ignored, unless `ls_use_all_feature_lines` is set.

FEATURE or INCREMENT syntax format

The format for the FEATURE line changed in FLEXlm version 3.0. The old format is compatible with new clients and servers, but the new format is more flexible, and you are likely to see it more often in the future.

Old format:

```
FEATURE|INCREMENT name daemon version
exp_date #lic key `vendor_string' [hostid]
```

New format:

```
FEATURE|INCREMENT name daemon version
\exp_date #lic key
\[HOSTID=hostid]
\[VENDOR_STRING='vendor-string']
\[vendor_info='...'] [dist_info='...']
\[user_info='...']
\[asset_info='...'] [ISSUER='...']
\[NOTICE='...'] [ck=nnn]
\[OVERDRAFT=nnn]
\[DUP_GROUP=NONE|SITE|[UHDV]]
```

Nothing in a FEATURE/INCREMENT line can be edited, except for values in the `name=value` pairs, where `name` is all lowercase.

<name> is the name given to the feature by the vendor.

<daemon> is the name of the vendor daemon; also found in the DAEMON line. The specified daemon serves this feature.

<version> is the version of this feature that is supported by this license.

<exp_date> is the expiration date, for example, 7-may-1996.

<#lic> is the number of concurrent licenses for this feature. If the number of users is set to 0, the licenses for this feature are uncounted and no lmgrd is required but a hostid on the FEATURE line is required.

<key> is the license plug for this FEATURE line. Two identical looking FEATURE or INCREMENT lines may have different license plugs if the start dates are different.

Optional fields

The following fields are all optional (except for vendor-string in the old format). For optional fields of the name=value syntax, if the name is lowercase, it can be modified and the license will remain valid.

[vendor_string] is the vendor-defined string, enclosed in double quotes. This string can contain any 64 characters except a quote. (White space is ignored for the purpose of the encryption algorithm, but may be significant to your software vendor.)

[hostid] is the string returned by lmhostid. Used only if the feature is to be bound to a particular host, whether its use is counted or not. (Numeric hostids are case insensitive).

[vendor_info=...] is additional information provided by the software vendor. Not encrypted into the feature's plug.

[dist_info=...] is additional information provided by the software distributor. Not encrypted into the feature's plug.

`[user_info=...]` is additional information provided by the software end-user's license administrator. Not encrypted into the feature's plug.

`[asset_info=...]` is additional information provided by the software end-user's license administrator for asset management. Not encrypted into the feature's plug.

`[ISSUER=...]` is the issuer of the license.

`[NOTICE=...]` is for intellectual property notices.

`[ck=nnn]` is a checksum, useful with the lmcksum utility, which will verify that the license has been entered correctly by the end-user. Not encrypted.

`[OVERDRAFT=nnn]` FLEXlm v4.0 and later. The OVERDRAFT policy allows your vendor to specify a number of additional licenses which users will be allowed to use, in addition to the licenses they have purchased. This allows your users to not be denied service when in a temporary overdraft state. Usage above the licensed limit will be reported by the FLEXadmin reporting tool.

`[DUP_GROUP=...]` is the duplicate Grouping parameter, which can be specified in the license in FLEXlm v4.0 and later as follows:

```
DUP_GROUP=NONE|SITE|[UHDV]
\U = DUP_USER
\H = DUP_HOST
\D = DUP_DISPLAY
\V = DUP_VENDOR_DEF
```

Any combination of UHDV is allowed, and the DUP_MASK is the OR of the combination. For example `DUP_GROUP=UHD` means the duplicate grouping is `(DUP_USER|DUP_HOST|DUP_DISPLAY)`, so a user on the same host and display will have additional uses of a feature, but not consume additional licenses.

FEATURE and INCREMENT line examples

```
FEATURE xyz_app xyzd 2.300 31-dec-1997 20
\1234567890 `xyz'
\INCREMENT f1 demo 1.000 1-jan-0 5
\12345678901234567890
\HOSTID=INTERNET=195.186.*.*
\NOTICE='Licensed to XYZ corp'
```

FEATURESET lines

The FEATURESET line is used to prevent FEATURE lines from being added to or removed from the license file.

Nothing in a FEATURESET line can be edited. Use the FEATURESET line exactly as it comes from your vendor.

FEATURESET format

```
FEATURESET daemon-name key
```

<daemon-name> is the name of the vendor daemon used to serve some feature(s) in the file.

<key> is the license plug for this FEATURESET line. This plug encrypts the plugs of all features this daemon supports, so no FEATURE lines can be removed, added to, or rearranged in this license file.

FEATURESET example

```
FEATURESET sampled 12345678
```

<sampled> is the name of the vendor daemon.

<12345678> is the plug generated by the vendor.

PACKAGE lines

The purpose of the PACKAGE line is to support two different licensing needs:

- to license a product SUITE, or

- to provide a more efficient way of distributing a license file that has a large number of features, which largely share the same FEATURE line arguments.

A PACKAGE line, by itself, does not license anything—it requires a matching FEATURE/INCREMENT line to license the whole PACKAGE.

A PACKAGE line can be shipped by your software vendor with a product, independent of any licenses. Later, when you purchase a license for that package, one or more corresponding FEATURE/INCREMENT licenses will enable the PACKAGE.

PACKAGE example

```
PACKAGE pkg_name vendor pkg_version pkg_key
COMPONENTS=pkg_list \
[ OPTIONS=pkg_options ]
```

`<pkg_name>` name of the PACKAGE. The corresponding FEATURE/INCREMENT line must have the same name.

`<vendor>` name of the vendor daemon that supports this PACKAGE.

`<pkg_version>` version of the PACKAGE. The enabling FEATURE/INCREMENT line must have the same version.

`<pkg_key>` 20-character license plug.

`<pkg_list>` list of components, as follows:

```
feature[:version[:count]]
```

PACKAGEs must consist of at least one COMPONENT. Version and count are optional, and if left out, their values come from the corresponding FEATURE/INCREMENT line. Count is only legal if OPTIONS=SUITE is not set; in this case the resulting number of licenses will be the count on the COMPONENTS line multiplied by the number of licenses in the FEATURE/INCREMENT line.

PACKAGE examples

```
COMPONENTS='comp1 comp2 comp3 comp4'
```

```
COMPONENTS='comp1:1.5 comp2 comp3:2.0:4'
\OPTIONS=pkg_options
```

Currently the only supported option is SUITE. This is what distinguishes a suite PACKAGE from a PACKAGE used to ease distribution.

With OPTIONS=SUITE, the corresponding FEATURE of the same name as the package name is checked out in addition to the component feature being checked out.

If OPTIONS=SUITE is not set, then the corresponding FEATURE of the same name as the package is removed once the PACKAGE is enabled; and it is not checked out when a component feature is checked out.

PACKAGE examples

```
PACKAGE suite demo 1.0 20CHARKEYXXXXXXXXXX
\COMPONENTS='comp1 comp2' OPTIONS=SUITE
\FEATURE suite demo 1.0 1-jan-0 5
\20CHARKEYXXXXXXXXXX
```

This is a typical SUITE example. You will have 2 features: comp1 and comp2, which are each version 1.0, with 5, non-expiring licenses available. When comp1 or comp2 are checked out, suite will also be checked out.

```
PACKAGE suite demo 1.0
\COMPONENTS='comp1:1.5:2 comp2:3.0:4 comp3'
\FEATURE suite demo 1.0 1-jan-95 3
\20CHARKEYXXXXXXXXXX
\ISSUER=distrib1
```

In this example, the component versions override the FEATURE versions, and the number of licenses available for any component is the product of the 3 licenses for suite and the number of licenses for that component. The result is equivalent to:

```
FEATURE comp1 demo 1.5 1-jan-95 6
\20CHARKEYXXXXXXXXXX
\ISSUER=distrib1
\FEATURE comp2 demo 3.0 1-jan-95 12
\20CHARKEYXXXXXXXXXX
\ISSUER=distrib1
\FEATURE comp3 demo 1.0 1-jan-95 3
\20CHARKEYXXXXXXXXXX
\ISSUER=distrib1
```

UPGRADE lines

```
UPGRADE name daemon fromversion version
\exp_date #lic key `string'
\[hostid] ck=nnn
```

All the data is the same as for a FEATURE or INCREMENT line, with the addition of the fromversion field. An UPGRADE line removes up to the number of licenses specified from any old version (>= fromversion) and creates a new version with that same number of licenses.

UPGRADE example

```
FEATURE f1 demo 1.000 1-jan-94 5
\9BFAC03164EDB7BC0462 ``

UPGRADE f1 demo 1.000 2.000 1-jan-94 2
\1B9A30316207EC8CC0F7 ``
```

would result in 3 licenses of version 1.0 of f1 and 2 licenses of version 2.0 of f1.

UPGRADE will operate on the most recent FEATURE or INCREMENT line (i.e., closest preceding FEATURE or INCREMENT line) with a version number that is >= fromversion, and < version.

Note   *UPGRADE does not work for node-locked, uncounted licenses. A new FEATURE line should be issued in this case, since the license count is irrelevant.*

89

The anatomy of a license file

The following example is a license file for single vendor with two features.

```
SERVER excellent_server 17007ea8 1700

DAEMON xyzd C:\xyzd

FEATURE xyz_app1 xyzd 1.000 01-jan-1993 10
\1EF890030EABF324 ""

FEATURE xyz_app2 xyzd 1.000 01-jan-1993 10
\0784561FE98BA073 ""
```

The license file above would allow the license server excellent_server with the hostid 17007ea8 to serve 10 floating licenses for xyz_app1 and xyz_app2 to any user on the network.

Parameters of license file fields

This section lists the limits on end-user editable FLEXlm license file parameters.

Table 11   *License file parameter length limits*

This parameter...	Has this length limit...
host names	32 characters (pre-FLEXlm v2.4)
	64 characters (FLEXlm v2.4 or later)
license file lines	200 characters
	2048 character, with ``\' as a continuation character (v3.0 or later)
user names	20 characters

Types of license files

License files are created by the software vendor. License files can specify any of the following policies:

- floating (concurrent) usage
- node-locked usage
- floating usage on a limited set of hosts
- mixed node-locked and floating usage

Floating (concurrent) licenses

A Floating license means anyone on the network can use the licensed software, up to the limit specified in the license file. (Also referred to as concurrent usage or network licensing.) Floating licenses have no hostids on the individual FEATURE lines. This configuration requires an lmgrd daemon because the daemon is going to count the concurrent usage of the licenses.

Example

```
SERVER lulu 17001234 1700
DAEMON xyzd C:\xyzd
FEATURE f1 xyzd 1.00 1-jan-95 2 key1 ""
FEATURE f2 xyzd 1.00 1-jan-95 6 key2 ""
FEATURE f3 xyzd 1.00 1-jan-95 1 key3 ""
```

This license file specifies that two licenses for feature f1, six licenses for feature f2, and one license for feature f3 are available anywhere on the network that can access the license server *lulu.*

Node-locked licenses

Node locking means the licensed software can only be used on one node. A node-locked license has a hostid on any FEATURE line that is node-locked to a particular host. There are two types of node-locked licenses; uncounted and counted.

91

If the number of licenses is set to 0, then the licenses are uncounted and unlimited use is permitted on the specified node. This configuration does not require an lmgrd daemon because it is not going to count the concurrent usage of the features.

The following license file allows unlimited usage of feature f1 on the nodes with hostids of 12001234 and 1700ab12:

```
FEATURE f1 xyzd 1.000 1-jan-95 0 key1 ""
\12001234

FEATURE f1 xyzd 1.000 1-jan-95 0 key2 ""
\1700ab12
```

If these were the only FEATURE lines in this license file, no lmgrd daemon would be necessary and you should not start one.

The following license file allows three licenses for feature f1 to be run, but only on the node with hostid 1300ab43. (In this case, the daemons should be run on the same node that runs the software, since there is no reason to run the daemons on another node.)

```
SERVER lulu 1300ab43 1700

DAEMON xyzd C:\xyzy

FEATURE f1 zyzd 1.00 1-jan-95 3 key ""
1300ab43
```

Mixed node-locked and floating licenses

Uncounted node-locked and concurrent usage licenses can be mixed in the same license file.

The following license file allows unlimited use of feature f1 to be used on nodes 17001111 and 17002222, while allowing two other licenses for feature f1 to be used anywhere else on the network:

```
SERVER lulu 17001234 1700

DAEMON xyzd C:\flexlm\xyzd.exe

FEATURE f1 xyzd 1.00 1-jan-93 0 key1 ""
\17001111
```

```
FEATURE f1 xyzd 1.00 1-jan-93 0 key2 ""
\17002222
```

```
FEATURE f1 xyzd 1.00 1-jan-93 2 key3 ""
```

This configuration does require an lmgrd daemon because it is going to count the concurrent usage of the two licenses on the third FEATURE line.

The license finder

The license finder provides license file path information to FLEXlm clients so that you do not have to configure LM_LICENSE_FILE environment variables for all your users. The license finder provides a single license file path to all clients.

The license finder is available in FLEXlm version 4.0 and later.

Only programs linked with FLEXlm version 4.0 (or later) will be able to use the license finder.

A FLEXlm program determines the name of the license file using the following algorithm:

1   Looks for LM_LICENSE_FILE; if set, uses this path.

2   Looks for the license finder if step (1) fails.

   - Uses port FLEXLM from the services map; if not present, it uses port 744 (the official FLEXlm port).

   - attempts to connect to this port on flexlm_license_finder, flexlm_license_finder2, flexlm_license_finder3 in order. Connect timeout is fixed at 2 seconds. This is done with TCP/IP. No broadcasts are involved. (For debugging, the connect timeout can be modified by setting FLEXLM_FINDER_TIMEOUT to an integer). If the lookup of flexlm_license_finder fails, this step terminates.

   - If it can connect to any license finder, it gets the license file path from the finder.

3   Uses the default license file path if (1) and (2) fail.

Starting the license finder

To start the license finder, execute the following command on the node flexlm_license_finder:

```
lmgrd -f finder_file
```

Where:

`<finder_file>` is the license finder file.

lmgrd, when it is operating as a license finder (-f finder_file option), will only process license finder request messages, and sends the license file path back to the client. This lmgrd will not start any vendor daemons or process any other client request. This functionality is delivered in lmgrd so that you do not need a separate executable.

Finder file format

The finder file format:

```
DATA lfpath license_file_path
```

Example

```
DATA lfpath
C:\flexlm\license.dat:1234@host2
```

The license finder is not available on Netware systems.

Combining license files

Since many vendors use FLEXlm as their license manager, you may have to administer licenses from more than one vendor or multiple products from the same vendor.

Overview of combining license files

When you run FLEXlm-licensed products from multiple vendors, you may need to take steps to prevent licensing conflicts during installation. There are three ways you can accomplish this:

- Multiple license server nodes, each running one lmgrd and one license file.

- One license server node running one lmgrd and one license file.

- One license server node running multiple lmgrds and multiple license files.

Note *Each lmgrd can only read a single license file. While running separate lmgrds and license files simplifies initial installation in some ways, it probably increases the day-to-day administration of your licenses. In the first option mentioned above, you will have more license servers to monitor; in the third option you have only one server but multiple lmgrds to administer.*

Your product's license file (or files) define the license server (or servers) by hostname and hostid in the SERVER line (or lines) in the license file. If the license files for two or more products contain identical hostids on the SERVER line, then these files can be combined. See Using the same server node on page 3-97.

If the license files for two products contain different hostids on a SERVER line, then the license servers for those products will be running on different nodes.

Using the same server node

If you have two or more products whose license servers run on the same node (as specified by the SERVER lines in the license files), you may be able to combine the license files into a single license file. If the SERVER lines in those files have identical hostids, then you can combine the files into a single file. If the SERVER lines have different hostids, then you must keep the license files separate.

More precisely, you can combine two license files under the following conditions:

- The number of SERVER lines in each file is the same.

- The hostid field of each SERVER line in one file exactly matches the hostid field of each SERVER line in the other file.

Why license files may not be compatible

- License files are set up to run on different server nodes, so hostids are different.

- One file is set up for single server (has only one SERVER line), the other is set up for redundant servers (has multiple SERVER lines).

- One vendor uses a custom hostid algorithm, so the hostids on the SERVER lines are different even though the files are for the same machine.

If your license files are compatible as described above, then you have the option of combining license files and running a single lmgrd, as described below in Combining license files from multiple vendors on page 3-98.

If the license files are not compatible, then you must keep the license files separate and run separate copies of lmgrd for each license file, as described in Using separate license files on the same server node on page 3-99.

Note *You are not required to combine compatible license files; you always have the option of running separate lmgrds as described on page* **3-99***.*

Combining license files from multiple vendors

If your license files are compatible (i.e., use the same server nodes), you can combine them with any text editor. To combine license files, read all of the compatible license files into one file, then edit out the extra SERVER lines so that only one set of SERVER lines remains. Write out this data, and you have your combined license file. If you write the data to `license.dat` in the default location, you will generally not need to set the LM_LICENSE_FILE environment variable. If you write the data out elsewhere, or if you were able to combine some but not all of your license files, then you will need to set the LM_LICENSE_FILE variable (unless your program uses another method to find the license file).

FLEXlm License Manager component compatibility

When you combine license files for two different FLEXlm-licensed products, it may be the case that those products do not use the same version of FLEXlm.

FLEXlm is designed to handle this situation. There are two basic compatibility rules for FLEXlm:

- A newer lmgrd can be used with an older vendor daemon, but a newer vendor daemon might not work properly with an older lmgrd.

- A newer vendor daemon (or lmgrd) can be used with an older client program, but a newer client program might not work properly with an older vendor daemon.

From these two compatibility rules come the rules for selecting which version of administration tools to use:

- Always use the newest version of lmgrd and the newest version of each vendor daemon.

- Use the newest FLEXlm utilities if they are FLEXlm version 2.4 or later; otherwise use the oldest version of the utility programs (such as lmstat) you have.

For specific programs, you can use either the new or the old version (with the assumption that the vendor daemon

that goes with that program is at least as new as the program).

There are a couple of details and exceptions to go with the above rules:

- If you combine a product using version 2.1 or later of FLEXlm with a product using version 1.5 or earlier of FLEXlm, you must use the -b (for backwards compatibility) command line option to lmgrd. You can always use a version of lmgrd that is newer than the one provided by your software vendor, as long as you are careful to use the -b switch if replacing version 1.x lmgrd with a version 2.1 or later lmgrd.

- Products using version 2.0 of FLEXlm (none are known at this time) may not be combined in one license file with products using version 1.5 or earlier of FLEXlm. In this case, you need to use separate license files (see).

FLEXlm version 2.4 and later versions of lmgrd default to -b. You do not need to use the -b option if you are using FLEXlm version 2.4 or later.

Using separate license files on the same server node

If you have incompatible license files, you must run a separate copy of lmgrd for each license file. (You can also do this even if your license files are compatible.)

When you run lmgrd separately for each license file, there are two details to remember:

- The port number on the SERVER line of each license file must be unique. You can use a standard text editor to change the port number in each license file so that they are all different.

- You must make sure that you are using a compatible version of lmgrd when you start it up for a particular license file. This can be done by either using an explicit path to lmgrd, or by renaming lmgrd to something specific for that product (some vendors do this to

make it easier to identify the lmgrd they ship with their product).

When running client programs (such as a licensed program), you can set the LM_LICENSE_FILE environment variable to point to multiple license files.

Example

You may have a license file from vendor ABC and a license file from vendor XYZ with incompatible servers. You can place the license file from vendor ABC into:

```
C:\flexlm\abc.dat
```

and the license file from vendor XYZ into:

```
C:\flexlm\xyz.dat
```

and then set the LM_LICENSE_FILE environment variable to point to both of them. Each name in LM_LICENSE_FILE should be separated by a comma (,) on Windows 95 and Windows NT systems.

```
C:\flexlm\abc.dat,C:\flexlm\xyz.dat
```

Using a different server node

If products use different license server nodes, each set of license servers requires separate license files.

When multiple software vendors use the same set of license server nodes, the technique described in <u>Using the same server node on page 3-97</u> can be used to combine license files. The resulting (multiple) license files can be installed in convenient locations, and you would set the LM_LICENSE_FILE environment variable as follows:

```
set LM_LICENSE_FILE=
lfpath1,lfpath2,....,lfpathN
```

Where:

<lfpath1> is the path to the first license file

<lfpath2> is the path to the second license file.

.

.

.

<lfpathN> is the path to the last (Nth) license file

Use a comma (,) to separate the license file names.

Each program will query each license file in the order it is listed in the LM_LICENSE_FILE path. If the license server serving the license file listed in lfpath1 was unreachable, perhaps due to an NFS problem, changing the LM_LICENSE_FILE would allow a user to obtain a license from another server.

For more information about LM_LICENSE_FILE and setting the location of the license file, see <u>Specifying location of the license file on page 3-78</u> or <u>The license finder on page 3-94</u>.

Selecting server nodes

This section describes your options of which nodes to use as license server nodes.

Resources used by the server

This section describes the resources used by the license server. When you select a server node, you may need to take into account the system limits on these resources. For small numbers of licenses (under 100), most of these items should not be a problem on any workstation.

Processes

When you run lmgrd, it automatically starts one copy of each vendor daemon specified in the DAEMON line(s) in the license file. Each time a TCP client connects to the server, it uses a process file descriptor. If the maximum number of file descriptors is exceeded, additional vendor daemons are started to accommodate the overflow requests. Globetrotter Software has found that this is a rare occurrence, even on large networks.

See your operating system's documentation to determine what the maximum number of file descriptors is for your computer.

Sockets

When using TCP, a single vendor daemon can support as many users as the system limit for file descriptors. When files are exhausted, additional vendor daemons are spawned to allow for extra file descriptors. When using UDP, there is no limit to the number of end-users per vendor daemon process, since they can share a single socket in the vendor daemon.

Each client connected to a license server uses one socket. The total number of sockets used by the license server

programs is slightly larger than the total number of simultaneous clients.

If you have a very large number of licenses (a few hundred), you should confirm that the system limit on the number of sockets and file descriptors is adequate to handle all of the licenses.

CPU time

For small numbers of clients, the license servers use very little CPU time. The servers might have only a few seconds of CPU time after many days.

For a large number of clients (who are each exchanging heartbeat messages with the server), or for high checkout/checkin activity levels (hundreds per second), the amount of CPU time consumed by the server may start to become significant although, even here, CPU usage is normally not high. In this case, you may need to ensure that the server machine you select will have enough CPU cycles to spare.

Globetrotter Software has never encountered a situation where CPU cycles were an issue.

Disk space

The only output files created by the license servers are the debug and report log files. These log files contain one line for each checkout and one line for each checkin. If you have a lot of license activity, these log files will grow very large. You need to consider where to put these files and how often to delete or prune them. The license administrator can opt not to log messages to the debug log file if disk space is at a premium. See NOLOG on page 3-114 and REPORTLOG on page 3-114.

Note that the log files should be local files on the server machine. See Diskless nodes and remote mounted disks on page 3-104 for a discussion of remote file systems.

Memory

The FLEXlm daemons use little memory. Typically, lmgrd uses approximately 100 KB, and the vendor daemons use approximately 120 KB each.

Network bandwidth

FLEXlm sends relatively small amounts of data across the network. Each transaction, such as a checkout or checkin, is typically satisfied with less than 1Kbyte of data transferred. This means that FLEXlm licensing can be effectively run over slow networks (such as dial-up SLIP lines) for small numbers of clients.

For large number of clients (hundreds), each of which will be exchanging heartbeat messages with the vendor daemon, the network bandwidth use may start to become significant. In this case you should run client and server on the same local area network.

In high-traffic networks you may also want to avoid setting LM_LICENSE_FILE to a `port@host address`. Instead, the license administrator should place a copy of the license file in a filesystem local to the program.

See and .

Diskless nodes and remote mounted disks

Globetrotter Software recommends that you do not use remote mounted disks when you run the license server. lmgrd, the vendor daemons, the license file, and the log file should all be on locally mounted disks. If any of these files is on a remote mounted disk, you will be doubling the points of failure which could lead to a loss of all of your licenses. When all files are mounted locally, the licenses will be available as long as the server machine is up, but when the files are on a different machine, then the loss of either the license server machine or the file server machine will cause the licenses to not be available.

Diskless nodes are the extreme case of not using local disks. We recommend that you do not use diskless nodes as license servers, since the files would necessarily be accessed from a remote disk. In addition, FLEXlm sometimes (at the option of the vendor) makes a security check that fails on a diskless node. If you find that you are having problems with a lock file, one possibility is that you are attempting to run on a diskless node. The best solution is to select a different node for your license server.

Redundant servers

With FLEXlm you can set up redundant license servers to operate as a single logical license server. This feature is controlled solely by the SERVER lines in the license file; the software vendor does not need to do anything in his program or in his vendor daemon to enable this capability.

Redundant servers help ensure the availability of your licenses in the event of a system crash. The cost is some additional administration. If you are willing to work with the extra overhead, and if you want to ensure that all of your licenses are available even if one machine crashes, then you may want to use redundant servers.

Alternatively, working with your software vendor you may be able to split up your licenses among multiple independent servers. This approach is simpler to administer and it still allows you to have at least some of the licenses available even if one machine goes down.

Setting up redundant servers

To set up redundant servers, you must provide the hostid for three machines to your software vendor, who will in return give you a license file with three SERVER lines. You need to make sure that each license server has lmgrd, the vendor daemon program and the license file on a local file system. You then start lmgrd on each license server.

For caveats on remote file systems, see Diskless nodes and remote mounted disks on page 3-104.

Quorums and redundant servers

In a redundant server configuration, no licenses are available until there is a quorum of servers. A quorum of servers is defined as a strict majority of the servers listed in the license file, so a quorum in a three-server configuration is two. In other words, if only one server in a three server configuration is running, then no licenses will be available. As soon as two of the three servers are

running and communicating with each other, then all of the licenses will be available.

Counted vs. uncounted licenses

The license file will determine whether a license server is needed. If all the FEATURE (or INCREMENT/UPGRADE) lines have a license-count of 0 (UNLIMITED), then no server is needed. This type of license can be called uncounted. Alternatively, if any FEATURE lines have a non-zero license count, then a server is required to count those licenses. If a vendor wants to use FLEXlm without a server, they must issue uncounted licenses.

The options file

The options file allows the license administrator to control various operating parameters of FLEXlm. Specifically the license administrator can:

- allow the use of features based on user, hostname, or display name.

- deny the use of features based on user, hostname, or display name.

- reserve licenses based on user, hostname, or display name.

- control the amount of information logged about license usage.

Options files allow you, as the license administrator, to be as secure or open with licenses as you like.

FLEXlm version 4.0 allows the \ character as a continuation character in options file lines.

Prior to FLEXlm version 3.0, lines in the options file were limited to 200 characters. In version 3.0 and later, the line length is 2048 characters.

Creating an options file

To create an options file

1   Use the appropriate options listed in to create the options file using any text editor.

    You can put the options file anywhere; however, we recommend that the option file for vendor xyz be placed in

    ```
    C:\flexlm\options\xyz.opt
    ```

2   Add the pathname to the options file in the license file. The fourth field on the DAEMON line for the program's vendor daemon. For example,

    ```
    DAEMON xyzd C:\xyzd
    C:\flexlm\options\xyz.opt
    ```

would enable the `xyzd` vendor daemon to look at the specified options file.

Customizing the options file

This section is an overview of the options file syntax. See Understanding the options file on page 3-116 for examples and additional information.

Each line of the file controls one option.

Table 12   *FLEXlm options file syntax*

Use this option...	To do this...
EXCLUDE	deny a user access to a feature
EXCLUDEALL	deny a user access to all features served by this vendor daemon
GROUP	define a group of users for use with any options
HOST_GROUP	define a group of hosts for use with any options
INCLUDE	allow a user to use a feature
INCLUDEALL	allow a user to use all features served by this vendor daemon
LINGER	cause licenses to be held by the vendor daemon for a period after the program checks them in or exits
NOLOG	turn off logging of certain items
REPORTLOG	specify that a logfile be written suitable for use by the FLEXadmin End-User Administration Tool
RESERVE	reserve licenses for a user
TIMEOUT	specify idle timeout for a feature, returning it to the free pool for use by another user

You can include comments in your options file by starting each comment line with a pound sign `#`.

Everything in an options file is case-sensitive, so ensure that user names and feature names, for example, are entered correctly.

EXCLUDE

Excludes a user, host, display, or group from the list of who is allowed to use the feature. Excluded users will not be allowed to use the feature.

```
EXCLUDE featurename type name
```

Where:

<featurename> is the name of the feature being affected

<type> is one of USER, HOST, DISPLAY, or GROUP (see GROUP on page 3-111 and HOST_GROUP on page 3-112)

<name> is the name of the user or group to exclude

Example

To exclude the user Hank from the list of users able to use feature f1:

```
EXCLUDE f1 USER hank
```

EXCLUDEALL

Excludes a user, host, display, or group from the list of who is allowed to use all features served by this vendor daemon.

```
EXCLUDEALL type name
```

Where:

<type> is one of USER, HOST, DISPLAY, or GROUP (see GROUP on page 3-111 and HOST_GROUP on page 3-112)

<name> is the name of the user or group to exclude

Example

To exclude any user on the server Chaos from using all features served by this vendor daemon:

```
EXCLUDEALL HOST chaos
```

GROUP

Defines a group of users for use in INCLUDE, INCLUDEALL, EXCLUDE, EXCLUDEALL, and RESERVE option lines.

```
GROUP groupname usernamelist
```

Where:

<groupname> is the name of the group being defined

<usernamelist> is the list of user names in that group

In FLEXlm version 3.0, multiple GROUP lines will add all the users specified into the group. Pre-version 3.0 FLEXlm daemons do not allow multiple GROUP lines to concatenate. In fact, the second GROUP line would redefine the GROUP.

In FLEXlm v4.0 or later, USER_GROUP is an alias for GROUP.

Example

To define the group called Hackers consisting of Bob, Howard, and Jennifer:

```
GROUP Hackers bob howard jennifer
```

HOST_GROUP

Defines a group of hosts for use in INCLUDE, INCLUDEALL, EXCLUDE, EXCLUDEALL, and RESERVE option lines.

```
HOST_GROUP groupname hostnamelist
```

Where:

> `<groupname>` is the name of the group being defined

> `<hostnamelist>` is the list of host names in that group

Example

To define the group called Hackers consisting of Bob, Howard, and Jennifer:

```
GROUP Hackers bob howard jennifer
```

INCLUDE

Includes a user, host, display, or group in the list of who is allowed to use the feature. Anyone not in an INCLUDE statement will not be allowed to use that feature.

```
INCLUDE featurename type name
```

Where:

> `<featurename>` is the name of the feature being affected

> `<type>` is one of USER, HOST, DISPLAY, or GROUP (see and)

> `<name>` is the name of the user or group to include

Example

To include user Bob in the list of users allowed to use feature f1:

```
INCLUDE F1 USER BOB
```

INCLUDEALL

Includes a user, host, display, or group in the list of who is allowed to use all features served by this vendor daemon. Anyone not in an INCLUDEALL statement will not be allowed to use these features.

```
INCLUDEALL type name
```

Where:

<type> is one of USER, HOST, DISPLAY, or GROUP (see GROUP on page 3-111 and HOST_GROUP on page 3-112)

<name> is the name of the user or group to include

Example

To allow the user Jane to use all features served by this vendor daemon:

```
INCLUDEALL USER jane
```

LINGER

```
LINGER featurename interval
```

This causes the daemon to hold on to the license for <featurename> for interval seconds after the program checks the license in or exits.

This could be useful for short-duration programs that will be used many times in a row by the same user, to ensure that the user will be able to reacquire the license repeatedly. This is only useful if the program uses duplicate grouping. Otherwise, LINGER will cause you to use extra licenses. Contact your software vendor for information about how they implemented duplicate grouping in their product.

License administrators might use this option to reduce the size of the lmgrd or debug log file.

This file is only useful with the FLEXadmin license administration utility.

NOLOG

Turns off logging of specific events by the FLEXlm daemons.

```
NOLOG what
```

Where:

<what> is what to turn off; one of IN, OUT, DENIED, or QUEUED

To turn off logging of checkins:

```
NOLOG IN
```

To turn off logging of checkouts and queued requests two separate NOLOG lines are required:

```
NOLOG DENIED
NOLOG QUEUED
```

REPORTLOG

Specifies the file which will contain the report-writer log for this vendor daemon.

```
REPORTLOG filename
```

If <filename> begins with a + character, the file will be opened for append, otherwise the file will be overwritten each time the daemon is started.

RESERVE

Reserves licenses for a specific user.

```
RESERVE numlic featurename type name
```

Where:

<numlic> is the number of licenses to reserve

<featurename> is the name of the feature reserve

<type> is one of USER, HOST, DISPLAY, or GROUP (see GROUP on page 3-111 and HOST_GROUP on page 3-112)

<name> is the name of the user or group to reserve licenses for

To reserve one license of feature f1 for user Mel:

```
RESERVE 1 f1 USER mel
```

Any licenses reserved for a user are dedicated to that user. Even when that user is not actively using the license, it will be unavailable to other users. However, a RESERVED license will not cause an overdraft to be reported by FLEXadmin if the license is not actually in use.

TIMEOUT

Sets the time after which an inactive license is reclaimed by the vendor daemon.

```
TIMEOUT featurename seconds
```

Where:

`<featurename>` is the name of the feature

`<seconds>` is the number of seconds after which inactive license is reclaimed

To set the timeout for feature f1 to one hour (3600 seconds):

```
TIMEOUT f1 3600
```

TIMEOUT removes a feature from a user if he or she has been idle for a period longer than the specified time period, and someone else wants the license. The daemon declares a process idle when it has not heard from the process (the client sends heartbeats). The program must explicitly declare itself idle for this to work, or the program must be stopped (Ctrl+Z). That is, unless the program explicitly supports this feature, it will not work. Contact your software vendor for information about how they implemented this feature in their product.

The program vendor can also disable the timeout feature, in which case the TIMEOUT option has no effect. The vendor can set a minimum value for the timeout. If you specify a timeout value smaller than the minimum, the minimum is used. The default minimum value is 900 seconds (15 minutes).

If you do not specify a timeout value in your options file, then there will be no timeout for that feature. Licenses are only freed by TIMEOUT when a new request for a license is made.

Understanding the options file

The following section is an overview of the syntax of a complete options file, followed byexamples of ways to effectively control access to your licenses.

How lmgrd uses the options file

When the vendor daemon is started by lmgrd, it is passed the location of the options file. The location is specified in the license file for that product, on the DAEMON line. If no file is listed the daemon will not use any options file.

There can only be one options file per vendor daemon, and each vendor needs a separate options file.

Rules of precedence in the options file

Before you can use options to utilize licenses effectively you must understand the options file precedence. INCLUDE and EXCLUDE statements can be combined in the same options file and control access to the same features. When doing so, keep in mind the following:

- If there is only an EXCLUDE list, everyone who is not on the list will be allowed to use the feature.

- If there is only an INCLUDE list, only those users on the list will be allowed to use the feature.

- If neither list exists, then everyone is allowed to use the feature.

The EXCLUDE list is checked before the INCLUDE list, so someone who is on both lists will not be allowed to use the feature.

Once you create an INCLUDE or EXCLUDE list everyone else is implicitly `outside' the group. This feature allows you, as an administrator, the ability to control licenses without having to explicitly list each user that you wish to allow or deny access to. In other words, there are two approaches; you can either:

- Give most users access and list only the exceptions,

  or

- Severely limit access and list only the those users who have access privileges.

Options file example

```
RESERVE  1 compile     USER lori
RESERVE  3 compile     HOST mainline
EXCLUDE compile USER robert
NOLOG QUEUED
```

This options file would:

- reserve one license for the feature Compile for the user Lori.

- reserve three licenses for the feature Compile for anyone on a computer with the hostname Mainline.

- prevent the user Robert from using the Compile feature on any node on the network.

- cause QUEUED messages to be omitted from the log file.

The sum total of the licenses reserved must be less than or equal to the number of licenses specified in the FEATURE line. In the example above, there must be a minimum of four licenses on the compile FEATURE line. If fewer licenses are available, only the first set of reservations (up to the license limit) are used.

If this data were in file `C:\flexlm\options\local.opt`, then you would modify the license file DAEMON line as follows:

```
DAEMON xyzd C:\flexlm\xyzd
C:\flexlm\options\local.opt
```

Limiting access for multiple users

Each INCLUDE, INCLUDEALL, EXCLUDE, EXCLUDEALL, and RESERVE line must have a single user name (or group) listed. To affect more than one user name create a GROUP.

For example, to exclude Bob, Howard, and Jennifer from using the feature called Toothbrush we could create the following options file:

```
EXCLUDE toothbrush USER bob
EXCLUDE toothbrush USER howard
EXCLUDE toothbrush USER jennifer
```

However, there is an easier way: create a GROUP and exclude the list of users from using the feature. Like the previous example, the following options file would exclude Bob, Howard, and Jennifer from using the feature called Toothbrush:

First define the group Hackers

```
GROUP Hackers bob howard jennifer
```

Then exclude the group

```
EXCLUDE toothbrush GROUP Hackers
```

Now when you want to allow or deny access to any feature to that group, you have an alias list to make it simple.

The GROUP function works for a list of user names prior to FLEXlm version 4.0. To control access to multiple displays (and hosts in pre-version 4.0 FLEXlm) you must use multiple option lines in your options file. For example, in pre-version 4.0 FLEXlm to exclude all users logged in on the hosts Fred and Barney from using a feature called F1, add these lines to your options file:

```
EXCLUDE f1 USER fred
EXCLUDE f1 USER barney
```

In FLEXlm version 4.0, you can use HOST_GROUP to allow, deny or reserve licenses for multiple hosts. For example, to exclude all users logged in on the hosts Fred and Barney from using a feature called F1 add these lines to your options file:

```
HOST_GROUP writers fred barney
EXCLUDE f1 HOST_GROUP writers
```

See see <u>GROUP on page 3-111</u> and <u>HOST_GROUP on page 3-112</u> for more information about defining groups.

EXCLUDE example

#First Define the group painters

```
GROUP painters picasso mondrian klee
EXCLUDE spell GROUP painters
EXCLUDE spell USER bob
EXCLUDE spell HOST bigbrush
```

This options file would:

- Prevent the users Picasso, Mondrian, and Klee from using the feature spell on any machine on the network.

- Prevent the user Bob from using the feature Spell on any machine on the network.

- Prevent any user logged into the host Bigbrush from using the feature Spell

- Allow any other user, as long as they are not on Bigbrush, not in Painters, and are not Bob, to use feature Spell (by implication.)

Note that Bob could have been added to the group Painters. However, Painters may be used for some other purpose in the future, so the license administrator chose to handle Bob as a special case here. In this case, the two EXCLUDE statements concatenate to create a list of four users.

INCLUDE example

```
INCLUDE paint USER picasso
INCLUDE paint USER mondrian
INCLUDE paint HOST bigbrush
```

This options file would:

- Allow the user Picasso to use the feature Paint on any machine on the network.

- Allow the user Mondrian to use the feature Paint on any machine on the network.

- Allow any user, as long as they are on the host Bigbrush, to use feature Paint

- Deny access to the feature Paint to anyone except Picasso, Mondrian, or users on Bigbrush

License administration tools

FLEXlm provides utilities for the license administrator to help manage the licensing activities on the network. These utilities are:

- lmcksum (v2.4 or later) prints license checksums. (See page **3-122**.)

- lmdiag (v4.0 or later) diagnoses license checkout problems. (See page **3-123**.)

- lmdown gracefully shuts down all license daemons (both lmgrd and all vendor daemons) on the license server node (or on all three nodes in the case of redundant servers). (See page **3-124**.)

- lmgrd is the main daemon program for FLEXlm. (See page **3-125**.)

- lmhostid reports the hostid of a system. (See page **3-126**.)

- lmremove removes a single user's license for a specified feature. (See page **3-127**.)

- lmreread causes the license daemon to reread the license file and start any new vendor daemons. (See page **3-128**.)

- lmstat helps you monitor the status of all network licensing activities. (See page **3-129**.)

- lmswitchr switches the report writer log file. (See page **3-130**.)

- lmver reports the FLEXlm version of a library or binary file. (See page **3-130**.)

Beginning in FLEXlm version 2.4, all FLEXlm utility programs (except lmgrd) are packaged as a single executable called lmutil. lmutil can either be installed as the individual commands (either by creating links to the individual command names, or making copies of lmutil as the individual command names), or the commands can be run as lmutil command, e.g. lmutil lmstat, or lmutil lmdown. On Windows or Windows NT systems, the

`lmutil command_name` form of the commands are recommended.

lmcksum

The lmcksum program (FLEXlm version 2.4 or later) will perform a checksum of a license file. This is useful to verify data entry errors at your customer's location.

lmcksum will print a line-by-line checksum for the file as well as an overall file checksum. lmcksum takes the `-k` switch to force the encryption plug checksum to be case-sensitive.

lmcksum will ignore all fields that do not enter into the encryption plug computation; thus the server node name and port number, as well as the daemon pathname and options file names are not checksummed. In addition, lmcksum will treat non-case sensitive fields correctly (in general, lmcksum is not case-sensitive).

lmcksum takes an optional daemon name; if specified, only license file lines for the selected daemon are used to compute the checksums.

By default, lmcksum operates on `license.dat` in the current directory. Specify `-c license_file` if you want to checksum another license file.

Example output is:

```
lmcksum - Copyright (C) 1989, 1993
Globetrotter Software, Inc.

lmcksum: using license file
`C:\flexlm\licenses\license.dat'

189: SERVER speedy 08002b32b161 2837

166: DAEMON demo C:\flexlm\demo.exe

8: FEATURE f1 demo 1.000 01-jan-99 0
3B2BC33CE4E1B8F3A0BF `` 08002b32b161

109: (overall file checksum)
```

lmdiag

lmdiag (FLEXlm version 4.0 or later) allows you to diagnose problems when you cannot check out a license.

Usage is:

```
lmdiag [-c license_file] [-n] [feature]
```

Where:

[-c license_file] is the path of the file to diagnose.

[-n] means to run in non-interactive mode; lmdiag will not prompt for any input in this mode. In this mode, extended connection diagnostics are not available.

[feature] means to diagnose this feature only.

If no feature is specified, lmdiag will operate on all features in the license file(s) in your path. lmdiag will first print information about the license, then attempt to check out each license. If the checkout succeeds, lmdiag will indicate this. If the checkout fails, lmdiag will give you the reason for the failure. If the checkout fails because lmdiag cannot connect to the license server, then you have the option of running `extended connection diagnostics'.

These extended diagnostics attempt to connect to each port on the license server node, and can detect if the port number in the license file is incorrect. lmdiag will indicate each port number that is listening, and if it is an lmgrd process, lmdiag will indicate this as well. If lmdiag finds the vendor daemon for the feature being tested, then it will indicate the correct port number for the license file to correct the problem.

lmdown

The lmdown utility allows for the graceful shutdown of all license daemons (both lmgrd and all vendor daemons) on all nodes.

Usage is:

```
lmdown [-c license_file]
```

Where:

[-c license_file] means to use the specified license file. If this switch is not used, lmgrd looks for the environment variable LM_LICENSE_FILE. If that environment variable is not set, lmgrd looks for the file C:\flexlm\license.dat on Windows 95 or NT.

You should protect the execution of lmdown, since shutting down the servers causes users to lose their licenses. See the -p or the -x options in for details about securing access to lmdown.

When shutting down redundant servers, there is a 1 minute delay before the servers shut down. Do not use kill-9 to shut down the license servers.

lmgrd

lmgrd is the main daemon program for FLEXlm. When you start lmgrd, it looks for the license file. The license file contains information about vendors and features.

Usage is:

```
lmgrd [-app] [ -c license_file ]
[ -t timeout_interval ] [-l logfile]

[ -s timestamp_interval ] [ -p ][ -2 ] -v
[-x lmdown] [-x lmremove]
```

Where:

[-app] is required for Windows NT systems.

[-c license_file] means to use the license file named. If this switch is not specified, lmgrd looks for the environment variable LM_LICENSE_FILE. If that environment variable is not set, lmgrd looks for the file C:\flexlm\license.dat on Windows 95 or NT systems.

[-t timeout _interval] sets a timeout interval, in seconds, during which redundant daemons must complete their connections to each other. The default value is 10 seconds. A larger value may be desirable if the daemons are being run on busy systems or a very heavily loaded network.

[-l logfile] writes the debug log to logfile.

[-s timestamp_interval] specifies the logfile timestamp interval, in minutes. The default is 360 minutes.

[-p] restricts usage of lmdown, lmreread, and lmremove to a FLEXlm administrator who is by default root. If there a group called `lmadmin' then use is restricted to only members of that group. If root is not a member of this group, then root does not have permission to use any of the above utilities. Requires the -2 option.

[-2] uses the FLEXlm version 2.x daemon startup protocol.

125

[-v] prints lmgrd's version number and copyright and exits.

[-x lmdown] disallows the lmdown command (no user can run lmdown)

[-x lmremove] disallows the lmremove command (no user can run lmremove)

The -x lmdown and -x lmremove options are available in FLEXlm version 4.0 and later.

The -p option is available in FLEXlm version 2.4 and later.

lmhostid

The lmhostid utility reports the hostid of a system.

Usage is:

```
lmhostid
```

The output of this command looks as follows:

```
lmhostid - Copyright (c) 1989, 1990
Globetrotter Software, Inc.

The FLEXlm hostid of this machine is
`69021c89'
```

See Hostids for FLEXlm-supported machines on page 3-149.

lmremove

The lmremove utility allows you to remove a single user's license for a specified feature. For instance, when a user is running the software on a node that crashes, the user's license sometimes remains checked out and unavailable to anyone. lmremove returns the license to the pool of available licenses.

Usage is:

```
lmremove [-c license_file] feature user
host display
```

Where:

[-c license_file] means to use the license file named. If this switch is not specified, lmgrd looks for the environment variable LM_LICENSE_FILE. If the environment variable is not set, lmgrd looks for the default license file.

[feature] is the name of the feature checked out by the user.

[user] is the name of the user whose license you are removing.

[host] is the name of the host the user is logged in to.

[display] is the name of the display where the user is working.

You should protect the execution of lmremove, since removing a user's license can be disruptive. See the -p or the -x options in for details about securing access to lmremove.

If you use lmremove to reclaim a license when that user is still using the program the license will usually be rechecked out within a couple of minutes.

lmreread

The lmreread utility causes the license daemon to reread the license file and start any new vendor daemons that have been added. In addition, all pre-existing daemons will be signaled to reread the license file for changes in feature licensing information.

Usage is:

```
lmreread [-c license_file]
```

Where:

[-c license_file] means to use the license file named. If this switch is not specified, lmgrd looks for the environment variable LM_LICENSE_FILE. If that environment variable is not set, lmgrd looks for the file C:\flexlm\license.dat.

The license administrator may want to protect the execution of lmreread, since removing a user's license can be disruptive. See the -p option in <u>lmgrd on page 3-125</u> for details about securing access to lmreread.

If you use the -c option, the license file specified will be read by lmreread, not by lmgrd; lmgrd rereads the file it read originally. Also, lmreread cannot be used to change server node names or port numbers. Vendor daemons will not reread their option files as a result of lmreread.

lmstat

The lmstat utility helps you monitor the status of all
network licensing activities, including:

- which daemons are running

- users of individual features

- users of features served by a specific DAEMON

Usage is:

```
lmstat  [-a] [-S [daemon]] [-f [ feature]]
[-i feature] [-s [server]] [-t value]
[-c license_file] [ -A ]
```

Where:

[-a] displays all information

[-A] lists all active licenses

[-c license_file] means to use the license file
named. If this switch is not specified, lmgrd looks for
the environment variable LM_LICENSE_FILE. If that
environment variable is not set, lmgrd looks for the
file C:\flexlm\license.dat.

-f [feature_name] lists users of feature(s).

-i [feature_name] prints information about the
named feature, or all features if no feature name is
given.

-s [server_name] displays status of server node(s).

-S [DAEMON] lists all users of DAEMON's features.

[-t value] sets lmstat timeout to value.

lmswitchr

The lmswitchr utility switches the report writer (REPORTLOG) log file for the specified feature.

Usage is:

```
lmswitchr feature new-file
```

Where:

[feature] is any feature this daemon supports.

[new-file] is the new file path.

lmver

The lmver utility reports the FLEXlm version of a library or binary file.

Usage is:

```
lmver [filename]
```

Where:

[filename] is the name of the executable of the product.

For example, if you have a program called Spell, type:

```
lmver spell
```

The debug log file

The daemons all generate debug log files in the following format.

```
hh:mm (DAEMON NAME) message
```

Where:

hh:mm is the time when the message was logged.

[DAEMON NAME] is either lmgrd or the vendor DAEMON name. In the case where a single copy of the daemon cannot handle all of the requested licenses, an optional _ (underscore) followed by a number indicates that this message comes from a forked daemon.

message is the text of the message.

The debug log files can be used to:

- diagnose configuration problems.
- diagnose daemon software errors.

The debug log file should not be used for usage reporting.

Informational messages

The table below describes informational messages produced by FLEXlm.

Table 13   *FLEXlm informational messages*

This message...	Means this...
`Connected to node`	This daemon is connected to its peer on `<node>`.
`CONNECTED, master is name`	The license daemons log this message when a quorum is up and everyone has selected a master.
`DEMO mode supports only one SERVER host!`	An attempt was made to configure a demo version of the software for more than one server host.
`DENIED: N feature to user`	`<user>` was denied access to N licenses of `<feature>`.
`EXITING DUE TO SIGNAL nnn` `EXITING with code nnn`	All daemons list the reason that the daemon has exited.
`EXPIRED: feature`	`<feature>` has passed its expiration date.
`IN: feature by user (N licenses)`	user has checked back in N licenses of feature.
`IN server died: feature by user (number licenses)`	user has checked in N licenses by virtue of the fact that his server died.
`License Manager server started`	The license daemon started.

Table 13   *FLEXlm informational messages*

This message...	Means this...
`Lost connection to host`	A daemon can no longer communicate with its peer on node `<host>`, which can cause the clients to have to reconnect, or cause the number of daemons to go below the minimum number, in which case clients may start exiting. If the license daemons lose the connection to the master, they will kill all the vendor daemons; vendor daemons will shut themselves down.
`Lost quorum`	The daemon lost quorum, so will process only connection requests from other deamons.
`MASTER SERVER died due to signal nnn`	The license daemon received fatal signal `<nnn>`.

`MULTIPLE xxx servers running.`
 	
`Please kill, and restart license daemon`
	The license daemon has detected that multiple licenses for vendor daemon `<xxx>` are running. The user should kill all `<xxx>` daemon processes and restart the license daemon.

`OUT: feature by user (N licenses)`	`<user>` has checked out `N` licenses of `feature`.
`Removing clients of children`	The top-level daemon logs this message when one of the child daemons dies.

Table 13   *FLEXlm informational messages*

This message...	Means this...

`RESERVE feature for HOST name`
 	
`RESERVE feature for USER name`
	A license of feature is reserved for either user name or host name.

`REStarted xxx (internet port nnn)`	Vendor daemon xxx was restarted at internet port `<nnn>`.
`Retrying socket bind (address in use)`	The license servers try to bind their sockets for approximately 6 minutes if they detect Address in Use errors.
`Selected (EXISTING) master node.`	This license daemon has selected an existing master (node) as the master.
`SERVER shutdown requested.`	A daemon was requested to shut down via a user-generated kill command.
`[NEW] Server started for: feature-list`	A new server was started for the features listed.
`Shutting down xxx`	The license daemon is shutting down the vendor daemon xxx.
`SIGCHLD received. Killing child servers.`	A vendor daemon logs this message when a shutdown was requested by the license daemon.
`Started name`	The license daemon logs this message whenever it starts a new vendor daemon.
`Trying connection to node`	The daemon is attempting a connection to `<node>`.

Configuration problem messages

The table below describes messages produced by FLEXlm that pertain to configuration problems.

Table 14   *FLEXlm configuration problem messages*

This message...	Means this...
`hostname: Not a valid server host, exiting`	This daemon was run on an invalid hostname.
`hostname: Wrong hostid, exiting`	The hostid is wrong for `<hostname>`.
`BAD CODE for feature-name`	The specified feature name has a bad license plug.
`CANNOT OPEN options file file`	The options file specified in the license file could not be opened.
`Couldn't find a master`	The daemons could not agree on a master.
`license daemon: lost all connections`	This message is logged when all the connections to a server are lost, which often indicates a network problem.
`lost lock, exiting`	Error closing lock file
`Unable to re-open lock file`	The vendor daemon has a problem with its lock file, usually because of an attempt to run more than one copy of the daemon on a single node. Locate the other daemon that is running via a `ps` command, and kill it with `kill -9`.

Table 14   *FLEXlm configuration problem messages*

This message...	Means this...
`NO DAEMON line for daemon`	The license file does not contain a DAEMON line for daemon.
`No license service found`	The TCP license service did not exist in /etc/services.
`No license data for feat, feature unsupported`	There is no FEATURE line for feat in the license file.
`No features to serve!`	A vendor daemon found no features to serve. This could be caused by bad data in the license file.
`UNSUPPORTED FEATURE request: feature by user`	The `<user>` has requested a feature that this vendor daemon does not support. This can happen for a number of reasons: the license file is bad, the feature has expired, or the daemon is accessing the wrong license file.
`Unknown host: hostname`	The hostname specified on a SERVER line in the license file does not exist in the network database.

Table 14   *FLEXlm configuration problem messages*

This message...	Means this...
`lm_server: lost all connections`	This message is logged when all the connections to a server are lost. This probably indicates a network problem.
`NO DAEMON lines, exiting`	The license daemon logs this message if there are no DAEMON lines in the license file. Since there are no vendor daemons to start, there is nothing to do.
`NO DAEMON line for name`	A vendor daemon logs this error if it cannot find its own DAEMON name in the license file.

Daemon software error messages

The table below describes error messages produced by the daemon software.

Table 15   *Daemon software error messages*

This message...	Means this...
`accept: message`	An error was detected in the accept system call.
`ATTEMPT TO START VENDOR DAEMON xxx with NO MASTER`	A vendor daemon was started with no master selected. This is an internal consistency error in the daemons.
`BAD PID message from nnn: pid: xxx (msg)`	A top-level vendor daemon received an invalid PID message from one of its children (daemon number <xxx>).
`BAD SCONNECT message: (message)`	An invalid server connect message was received.
`Cannot create pipes for server communication`	The pipe call failed.
`Can't allocate server table space`	A malloc error. Check swap space.
`Connection to node TIMED OUT`	The daemon could not connect to node.
`Error sending PID to master server`	The vendor server could not send its PID to the top-level server in the hierarchy.

Table 15 *Daemon software error messages*

This message...	Means this...
`Illegal connection request to DAEMON`	A connection request was made to `<DAEMON>`, but this vendor daemon is not `<DAEMON>`.
`Illegal server connection request`	A connection request came in from another server without a DAEMON name.
`KILL of child failed, errno = nnn`	A daemon could not kill its child.
`No internet port number specified`	A vendor daemon was started without an internet port.
`Not enough descriptors to re-create pipes`	The top-level daemon detected the death of a sub-daemon. In trying to restart the chain of sub-daemons, it was unable to get the file descriptors to set up the pipes to communicate. This is a fatal error, and the daemons must be restarted.
`read: error message`	An error in a read system call was detected.
`select: message`	An error in a select system call was detected. This is usually a sign of a system networking failure.
`Server exiting`	The server is exiting. This is normally due to an error.

FLEXlm troubleshooting guide

This section documents areas of FLEXlm that have given customers difficulty in the past.

General debugging tips

- When you start the license server (lmgrd) be sure that you direct the output into a log file where you can examine it. The log file often contains useful information. You should examine it when you have a problem, and be prepared to answer questions about it when you talk to a support person.

- If the license server appears to have started correctly (which you should be able to determine from the log file), try running lmstat -a to see if that program has the same problem as your program.

- When you talk to a support person, you should be prepared to answer the following questions:

  - What kind of machine is your license server running on?

  - What version of the operating system?

  - What machine and operating system is the program running on?

  - What version of FLEXlm does the program use? Use the lmver script. Alternatively, LMGRD -V gives the version, and this works with the vendor daemon also.

  - What error or warning messages appear in the log file?

  - Did the server start correctly? Look for a message such as:

    ```
    server xyz started for: feature1
    feature2.
    ```

  - What is the output from running lmstat -a?

- Are you running other products which are also licensed by FLEXlm?

- Are you using a combined license file or separate license files?

- Are you using redundant servers (multiple SERVER lines in your license file)?

Troubleshooting list

Problem-description-solution format

Each problem is presented in three parts:

- The Problem section describes the problem.
- The Cause section discusses possible causes of the problem.
- The Solution section gives instructions for solving the problem.

You can scan through the list of problems to find the one that relates to your concerns. To solve your problem, you may need to use more than one of the solutions listed here.

Hostid problems

Problem

When I run the license manager on my machine, it tells me it is the wrong hostid.

Cause

The vendor daemon checks the hostid listed on the SERVER line in the license file; if it does not match the hostid of the machine it is running on, this message will be printed. Possible causes include:

- You are trying to run the license server on a different machine from the machine the file was made for.

- The hostid of the machine you are running on changed (for example, the HP ID module was moved, or the CPU board was replaced).

- The hostid in the license file was modified.

Solution

Verify that the hostid of the machine on which the vendor daemon (or node-locked client program) is being run matches the hostid specified in the license file (on the SERVER line for the vendor, or on the FEATURE line for a node-locked client). You can run the lmhostid program to see what FLEXlm thinks the hostid is. You may not modify the hostid in the license file. If the hostid of your server machine changes, you will have to get a new license file from your software vendor.

Connection problems

Problem

The program (or lmstat) cannot connect to the server to check out a license.

Cause

The FLEXlm routines in the program are unable to make a TCP connection to the server and port specified in the license file. Possible reasons for this are:

- The wrong license file is being referenced by the program.

- The server machine specified in the license file is down.

- The vendor daemon specified in the license file is not running.

- The hostname in the license file is not recognized by the system.

- The network between the client machine and the server machine is down.

- You are mixing FLEXlm version 1.5 (or earlier) and version 2.1 (or later) vendor daemons in one license file, and have run lmgrd without the -b command line option.

- TCP is not running on your machine.

Solution

Verify that the program is using the proper license file.

Verify that specified server machine is up and reachable by executing another command that uses TCP, such as `rsh` or `rlogin`, from the client to the server.

Verify that the vendor daemon is running (you can use ps on the server to look for it).

Examine the license log file to see if any problems are reported, particularly messages indicating that the vendor daemon has quit.

Run lmstat -a from the server machine to verify that the vendor daemon is alive.

Run `lmstat -a` from the client machine to verify the connection from client to vendor daemon across the network.

Try using `telnet <hostname> <portnum>` where hostname and portnum are the same as on the SERVER line in your license file.

If you have the FLEXlm version 4.0 utilities, try lmdiag.

Problem

The program is able to connect from some machines but not from others.

Cause

You are running a version 1.5 license server on a machine with two ethernet boards.

There is a restriction in FLEXlm version 1.5 in which the license server only listens to the ethernet address associated with the primary hostname on the system.

Solution

The simplest solution is to get an upgraded license server from your software vendor (you can run a new license server and still keep running the old program). Alternatively, you can ask your software vendor for a different license file and run your license daemon on a different machine with a single ethernet board. If this is not possible, you may be able to modify your host tables to direct all FLEXlm requests to the active port on the machine.

Problem

The program (or lmstat) gets the error can't read data when attempting to connect to the license server.

Cause

The program is able to find the server, but it is not getting the expected data. The most likely cause for this is that you are running a newer program than the vendor daemon.

Solution

Check the version of the program.

Verify that it is using the expected license file.

Check the version of the vendor daemon program referred to by the license file (look at the log file for that daemon).

Other client problems

Problem

When I run my program (or vendor daemon), I get the error bad code or inconsistent encryption code.

Cause

Possible causes for this are

- The license file was modified (either the hostid on a SERVER line or anything on the FEATURE line was changed).

- The vendor used the wrong version of his license creation program to generate your license file (or there is a bug in that process).

Solution

You may not modify the license file (except for specific fields—see). If you need to change something in your license file, you need to get a new license file from your software vendor.

Problem

When the second user tries to check out a license, the vendor daemon prints an error concerning Parameter mismatch in the log file and refuses the license.

Cause

The most likely cause of this problem is that you are simultaneously trying to run two different versions of the program, and the software vendor has not specifically set up the new version for this kind of compatibility.

Solution

Check the license server log file for a comm version mismatch warning message; this indicates that someone is running an older client than the license server daemon, lmgrd.

Run only the new version of the program (or only the old version).

Problem

When I start up lmgrd, it says execl failed on my vendor daemon.

Cause

lmgrd uses execl to start each vendor daemon running. If there is a problem starting the vendor daemon, this message is output to the log file.

This error is typically caused by one of the following:

- There is no executable at the location referred to by the license file (and printed out in the log file).

- The executable does not have the proper permissions to be run (the file does not have the x bit set, or one of the directories in the path is not readable).

- There was an error building the executable, and it can not be run.

- The executable is for a different machine architecture.

Solution

Verify that the path to the vendor daemon is absolute (i.e. starts with a slash character, and that it points to the executable program itself, not the containing directory (for FLEXlm version 1.5).

Ensure that the file exists by doing an `ls -l` of the vendor daemon filename specified in the log file. Make sure that you do this as the same user that started lmgrd.

Verify that the file is executable.

Do a whatis on the file (if your system has that program). whatis should tell you the file is an executable for the machine you are trying to run it on.

Run the vendor daemon directly from the command line. If the vendor daemon is properly linked, it will tell you that it must be run from lmgrd; if it crashes or fails to execute, then it is not properly linked.

Note *If you are running as root and using an NFS-mounted filesystem, the relevant protection bits are the other bits (not the user bits), even if the file is owned by root.*

Problem

The license server keeps reporting `lost lock' errors in the log file and exiting.

Cause

The lockfile (normally placed in C:\FLEXLM on Windows or NT, SYS:\SYSTEM\FLEXLM on Netware) is being removed by someone else.

There could be another daemon running, or the license administrator (or a script he set up) could have deleted the file.

Solution

Check to see if there is more than one copy of the daemon running.

Check for more than one lmgrd running, as well, since it will restart your vendor daemon when it is killed. If more than one lmgrd is running, kill them all (using the Control Panel Services dialog on Windows 95 or NT), then kill any remaining vendor daemons and start one fresh copy of lmgrd.

Hostids for FLEXlm-supported machines

FLEXlm uses different machine identifications for different machine architectures. For example, all Sun machines have a unique hostid, whereas all DEC machines do not. For this reason, the ethernet address is used on some machine architectures as the `hostid`. An ethernet address is a 6-byte quantity, with each byte specified as two hexadecimal digits. Specify all 12 hex digits when using an ethernet address as a hostid. For example, if the ethernet address is `8:0:20:0:5:ac`, you would specify `0800200005AC` as the hostid.

The program lmhostid will print the exact hostid that FLEXlm expects to use on any given machine.

Index

A

automatic activation, 17
automatic search under NetHASP, 36

C

configuration
    file, 57
    problem messages, 135
custom search under NetHASP, 40

D

daemon software error messages, 138
debug log file, 131

E

error codes, 63
EXCLUDE option, 109

F

FLEXlm
    combining license files, 96
    configuring, 72
    debug log file, 131
    installing, 73
    introduction, 69
    license administration tools, 121
    license file, 70, 77
    license file types, 91
    license finder, 94
    License Manager daemon, 69
    license request process, 71
    license server, 5
    messages, 132, 135, 138
    options file, 108
    selecting server nodes, 102
    vendor daemon, 70

G

GROUP option, 109