

# **Managing Software Products with iFOR/LS**



**HP Part No. B2355-90064  
Printed in USA January 1995**

**E0195**

The information contained in this document is subject to change without notice.

GRADIENT TECHNOLOGIES MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Gradient shall not be liable for errors contained herein or for any direct or indirect, incidental, special, or consequential damages in connection with the furnishing, performance or use of this material.

Copyright © Gradient Technologies, Inc. 1991, 1992, 1993, 1994.

The software to which this document refers is derived, in part, from materials supplied by Hewlett-Packard Co.

iFOR is a trademark and Gradient is a registered trademark of Gradient Technologies, Inc. NetLS is a trademark of Apollo Computer, Inc., a subsidiary of Hewlett-Packard Co.

This document contains proprietary information which is protected by copyright. All rights reserved. No part of this document may be photocopied, reproduced or translated to another language without the prior written consent of Gradient Technologies, Inc.

Reprinted with permission from Gradient Technologies, Inc.

Use, duplication or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (i) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013

Gradient Technologies, Inc.  
5 Mt. Royal Avenue  
Marlboro, MA 01752, USA  
Tel:(508) 624-9600  
FAX:(508) 229-0338  
E-Mail: info@gradient.com

# Contents

---

<b>1. Preface</b>	
Target Audience . . . . .	1-1
How to Use This Manual . . . . .	1-2
Administrator's Runtime Kit Release Documentation . . . . .	1-2
Documentation Conventions . . . . .	1-3
<b>2. Introduction</b>	
Current Licensing Methods and iFOR/LS . . . . .	2-1
Product Overview and Requirements . . . . .	2-2
End User Components of iFOR/LS . . . . .	2-3
Walkthrough of a License Acquisition Transaction . . . . .	2-4
Types of Licenses . . . . .	2-6
Concurrent-Access Licenses . . . . .	2-6
Use-Once Licenses . . . . .	2-6
Nodelocked Licenses . . . . .	2-6
License Annotations . . . . .	2-7
Vendor-Specific Administrative Considerations . . . . .	2-7
What You Get with the ARK . . . . .	2-8
Summary of the Administrator's Tasks . . . . .	2-9
The ARK Administrative Tools and the Global Location Broker . . . . .	2-9
<b>3. Setting Up The ARK</b>	
The ARK Prerequisites . . . . .	3-1
TCP/IP . . . . .	3-2
The Network Computing Kernel . . . . .	3-2
The License Server Daemon . . . . .	3-3
Choosing Nodes to Run the License Server . . . . .	3-3
Starting the License Server . . . . .	3-3
Verifying License Servers . . . . .	3-4
Starting the Client Ally . . . . .	3-4
User Files . . . . .	3-4
A Sample User File . . . . .	3-5

Writing and Integrating User Files . . . . .	3-7
Maintaining User Files . . . . .	3-7
User Names For Personal Computers . . . . .	3-7
Adding Passwords to License Servers . . . . .	3-7
A Sample i4admin Session . . . . .	3-8
Nodelocked Licenses . . . . .	3-12
Types of Passwords Specifying Nodelocked Licenses . . . . .	3-12
Single-license Passwords . . . . .	3-12
Compound Passwords . . . . .	3-12
Installing Passwords for Nodelocked Licenses . . . . .	3-13
Installing Single-license Passwords . . . . .	3-14
In Case of a Problem . . . . .	3-15
<b>4. Monitoring The ARK</b>	
Getting Status Information With i4stat . . . . .	4-1
A Sample i4stat Session - Getting Status Information . . . . .	4-4
License Server Reports . . . . .	4-7
<b>A. The ARK Command Reference</b>	
i4admin - Graphic Interface . . . . .	A-2
i4admin - Command-Line Interface . . . . .	A-5
i4report - Command-Line Interface . . . . .	A-9
i4stat - Graphic Interface . . . . .	A-12
i4stat - Command-Line Interface . . . . .	A-15
i4lmd - Command-Line Interface . . . . .	A-17
<b>B. Format of the User File</b>	
<b>C. Troubleshooting The ARK</b>	
i4tv . . . . .	C-2
<b>D. Glossary</b>	
<b>Index</b>	

## Figures

---

2-1. Three Software Products Installed with the ARK . . . . .	2-4
2-2. Invoking a Licensed Product . . . . .	2-5
3-1. A Sample User File . . . . .	3-5
3-2. Example of the i4admin Startup Display . . . . .	3-9
3-3. Modern Solutions, Inc. Added to the Vendor List for cobweb . . . . .	3-10
3-4. Kwik-Draw Added to the Product List for Modern Solutions . . . . .	3-11
3-5. A Sample nodelock File . . . . .	3-15
4-1. Using i4stat, Step 1 . . . . .	4-2
4-2. Using i4stat, Step 2 . . . . .	4-5
4-3. Using i4stat, Step 3 . . . . .	4-6
4-4. Using i4stat, Step 4 . . . . .	4-7
B-1. Format of the User File . . . . .	B-1
C-1. Example output from i4tv . . . . .	C-3

## Tables

---

A-1. Event Types . . . . .	A-10
A-2. Information Filters . . . . .	A-11
A-3. Event Types . . . . .	A-18
C-1. Common License Server Problems . . . . .	C-4



## Preface

---

The *i*FOR Licensing Suite (*i*FOR/LS) consists of two major components: the Application Developer's Kit (ADK), used by software developers to license software products, and the Administrator's Runtime Kit (ARK), used at end-user sites to manage products licensed with the ADK.

This book describes using the ARK to manage software products that are licensed with the ADK component of *i*FOR/LS.

---

**Note**        The contents of this manual are as provided by Gradient Technologies, Inc. The *i*FOR/LS product runs on multiple hardware platforms, and thus this manual may not be completely applicable to an HP-UX system.

---

---

### Target Audience

There are two audiences for this manual:

1. System administrators at end-user sites can use this manual to manage *i*FOR/LS, the Administrator's Runtime Kit, and the licensed software products that it controls.
2. Programmers and system administrators at application development sites can use this manual to set up environments for testing the licensed software products and for creating licenses for those products.

---

## How to Use This Manual

System administrators at end-user sites should read all of this manual. Programmers and system administrators at software development sites should read the *iFOR/LS Programmer's Guide* (available from Gradient Technologies, Inc.) before reading this manual.

---

## Administrator's Runtime Kit Release Documentation

The *iFOR/LS* Administrator's Runtime Kit is available in several operating-system-specific binary formats and runs on several different kinds of computers. A separate installation guide and a set of release notes accompanies each release of each ARK binary. Refer to it for information on the following topics:

- ARK version compatibility
- Installing the ARK software
- Applicable ARK user documentation
- Ordering user documentation
- Known problems and restrictions
- Reporting methods for bugs found in the software or this manual

If you use more than one binary version of the ARK software, check all applicable release documents: the information and procedures are different for different ARK binaries.



---

## Documentation Conventions

Unless otherwise noted in the text, this manual uses the following symbolic conventions.

<code>literal values</code>	Words or characters that appear in formats and command descriptions using this typeface represent commands or keywords that you must use literally. Path names are also in this typeface.
<b>new term</b>	The first use of a new term appears in bold.
<i>user-supplied values</i>	Italic words or characters in formats and command descriptions represent values that you must supply.
<code>output/source code</code>	Information that the system displays appears in this typeface. Examples of source code also appear in this typeface.
[     ]	Square brackets enclose optional items in formats and command descriptions.
{     }	Braces in formats and command descriptions enclose a list from which you must choose an item.
	A vertical bar separates items in a list of choices.
<code>key</code>	The name of a key on the keyboard.
<code>CTRL/ ^</code>	The notation <code>CTRL/</code> or the <code>^</code> followed by the name of a key indicates a control character sequence. Hold down <code>CTRL</code> while you press the key.
...	Horizontal ellipsis points indicate that you can repeat the preceding item one or more times.



## Introduction

---

This chapter presents an introduction to the Administrator's Runtime Kit component of *i*FOR/LS. The ARK gives users access to software products licensed with the ADK, the other component of *i*FOR/LS.

The Administrator's Runtime Kit is layered on top of the Network Computing System (NCS) and enables the uniform management of software licenses in a heterogeneous network environment.

---

### Current Licensing Methods and *i*FOR/LS

Current methods used in licensing software products within a network environment are usually limited to:

- Nodelocking (also known as CPU locking) - a software product is installed so that it can be used only at a specific node.
- Site licensing - a vendor sells a site-wide license for the product; the number of users who may use the product at the same time is theoretically limited only by the number of nodes at the site.

Neither of these schemes is appropriate for all products or for all customers. Nodelocking is often inconvenient for end users: anyone who wants to use a nodelocked product must work at the node to which it is locked. Furthermore, if the node is down or being used for another purpose, no one can use the product.

Site licensing, on the other hand, provides end users with unrestricted access to software. Because this flexibility makes it difficult for vendors to price software in proportion to its actual use, they often price according to availability instead. As a result, customers may be dissatisfied with paying for availability they do not use.

Despite the drawbacks of nodelocking and site licensing, there are products and customer environments for which these licensing mechanisms are entirely appropriate and satisfactory. For this reason, *iFOR/LS* supports these licensing methods. However, *iFOR/LS* also extends the range of licensing options available to software vendors. Two of these options are:

- Concurrent-access licensing.
- Single-execution licensing (called use-once licensing).

---

## Product Overview and Requirements

The HP-UX version of the *iFOR/LS* server (*iFOR/LS* ARK) is supported on HP PA-RISC processors running HP-UX version 10.0 or higher. The *iFOR/LS* ARK allows you to administer a network of license servers and also provides a license server on the local machine.

For *iFOR/LS* to operate correctly in a networked environment, you must have the Network Computing System (NCS) running on your system. NCS is included in HP-UX 10.0 and higher, and provides Remote Procedure Call libraries and local and global naming services. At least one copy of the global name server *glbd* must also be running. For more information, refer to *Managing NCS Software*, the *glbd*(1M) manual reference page, and the configuration file `/etc/rc.config.d/ncs`.

---

**Note**            In HP-UX, the NCS *llbd* daemon has been replaced by the DCE *rpcd* daemon. Refer to the *rpcd*(1M) manual reference page.

---

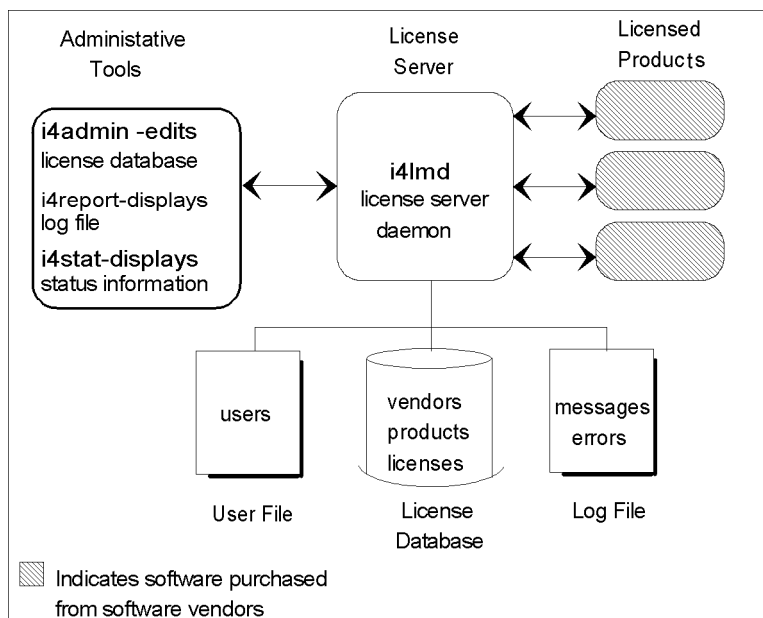
---

## End User Components of iFOR/LS

When distributed with a licensed software product and the ARK, the end-user *i*FOR/LS environments consists of the following components:

- The licensed software product, which requests licenses from the license management daemon.
- The license management daemon (*i4lmd*), which processes requests from the product by referring to information about licenses and users.
- The Client Ally (*i4ally*), which provides license server access to licensed products running on personal computers.
- The license database (*lic\_db*), which contains such data as vendor and product identifiers, the number of licenses for each product, and their expiration dates.
- The user file (*user\_file*), which contains the names of users who are authorized, or not authorized, to use the licensed product.
- Administrative tools (*i4admin*, *i4stat*, *i4report*), which system administrators use to manage *i*FOR/LS.
- The log file (*log\_file*), where the license management daemon keeps a history of license server events.

Figure 2-1 shows the components of a typical ARK installation and indicates how they communicate with one another.

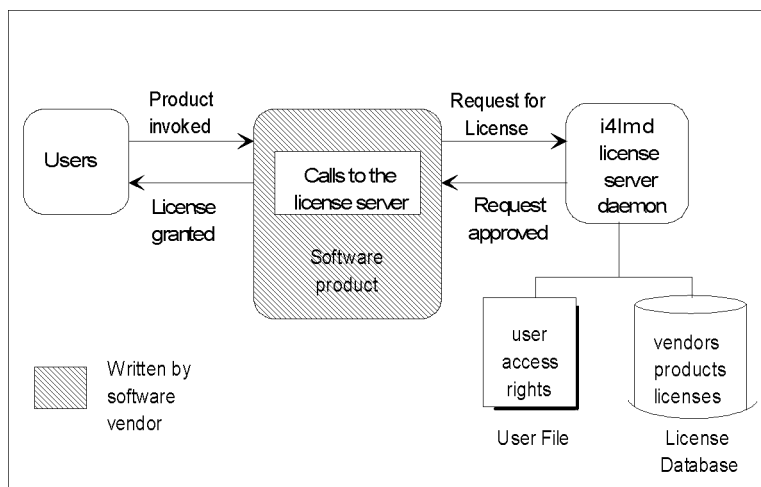


**Figure 2-1. Three Software Products Installed with the ARK**

---

## Walkthrough of a License Acquisition Transaction

Figure 2-2 shows the events that take place when a user invokes a software product that is administered by the license management daemon. Note that all license types other than nodelocked licenses are administered by the daemon.



**Figure 2-2. Invoking a Licensed Product**

The following is a simplified summary of the sequence of events associated with using a licensed product:

1. A user invokes the product.
2. The product requests a license from a license management daemon.
3. The license server checks the license database to see if a valid license is available, and the user file describing user-access rights, to find out whether the user is allowed to use the product.
4. If a license is available and the user is allowed to use the product, the server approves the request.
5. When the user receives a license from the license management daemon, the product begins executing.
6. When the user exits the product, the product releases the license back to the license server.

If the software product is executing on an IBM compatible personal computer, this path is slightly different. The request for a license is first directed to the Client Ally which establishes a connection with the i4lmd on behalf of the client software product. Likewise, when the response from the license server is returned it first goes through the Client Ally. The communications between the PC and the Client Ally do not employ the NCS remote procedure calls to

keep the overhead demands on the PC to a minimum. However, the bilateral authentication process is preserved to ensure security over this link.

---

## **Types of Licenses**

This section explains how *i*FOR/LS licenses work, and gives some examples of how software vendors use them.

### **Concurrent-Access Licenses**

Concurrent-access licenses allow as many users to use a software product at the same time as there are licenses for the product, until the licenses expire. For as long as someone is using the product, a concurrent-access license remains flagged as being in use and is therefore unavailable to other users. This is a popular type of license, especially for medium-to-large size networks.

### **Use-Once Licenses**

A use-once license allows a user to use a product only once within the period for which the license is valid. Each time the product is used, the license server decrements the number of installed use-once licenses by one.

A use is considered to begin when the software product is initially invoked. The completion of a use is left to the developer to determine.

Software vendors can provide use-once licenses as the primary license type for a product, as free-sample or demonstration licenses, or as backup licenses for products that are primarily licensed on a concurrent-access basis.

### **Nodelocked Licenses**

A nodelocked license allows the use of a product at the single node for which the license was created, until the license expires. The license server does not administer nodelocked licenses, so nodelocked licensed products are available to users whether or not a license server is available. Therefore, with nodelocked licenses, you do not need license servers nor any of the ARK administrative tools.



---

**Note**            There is a case—dynamic nodelocking—for which the license management daemon and the `i4admin` tool are required for installation of, but not use of, nodelocked licenses. Dynamic nodelocking is explained in Chapter 3.

---

## License Annotations

A license annotation is data defined by the software vendor and included as part of the license information when the license is created. A license annotation modifies the manner in which an application may be used.

Suppose an interactive graphics application includes optional features A, B, and C, all of which are bundled with the product, but sold separately. The vendor can use license annotations to specify which options are available to a given customer, since different customers buy different options. Therefore, if you buy options A and C, you would receive licenses with an annotation specifying that options A and C, but not B, are accessible. If no options are purchased, no annotation would be required.

## Vendor-Specific Administrative Considerations

A software vendor may offer different kinds of licenses for the same product. Sometimes a product that is usually licensed for concurrent access might also be offered on a nodelocked basis. Demonstration licenses for a new version of a product might be of the use-once type, whereas older versions of the product might be licensed concurrently. Usually when a vendor offers different kinds of licenses for the same product, one of the license types is the primary licensing mechanism, and other types are offered to suit special circumstances.

A vendor might also provide **hybrid** licenses for a product. For example, if all concurrent-access licenses for a product are in use, the product might look for use-once licenses in order to give access to more users. This type of product requires that you install and manage both types of licenses.

Under *iFOR/LS*, software vendors are not constrained to enforce the terms of their licenses. For example, a product may allow more users to use it than there are valid concurrent-access licenses. Or you may be able to use a product even after all of its licenses have expired. In these cases, the number and term of the licenses are not reliable indicators of how many users can use the product, nor for how long.

With the flexibility of *i*FOR/LS, software vendors can implement many different licensing and license-enforcement mechanisms - too many to describe in this manual. In the vendor's documentation, look for any administrative details that are relevant to the manner in which *i*FOR/LS was implemented for that product. Also take care to look for information about how the vendor uses license annotations and any information about how the terms of the licenses are enforced. Information of this type makes it easier to understand and administer the licenses.

---

## What You Get with the ARK

The vendor of a licensed product always provides:

- The licensed software product(s).
- A vendor ID, a vendor password, and one or more product passwords for each licensed product.

If the following items are not a standard part of your operating system, your software vendor will also provide:

- The Administrator's Runtime Kit product.
- This manual, which describes the Administrator's Runtime Kit.
- The Network Computing Kernel (NCK), which is the run-time component of the Network Computing System (NCS), and is used by *i*FOR/LS for communication between components running across the network.

In networks where only nodelocked licenses are used, the software vendor might provide only the software product, vendor ID, and product passwords. This is because the Administrator's Runtime Kit and the Network Computing Kernel are not needed to use only nodelocked licenses.

---

## Summary of the Administrator's Tasks

The job of the administrator of an ARK installation consists of the following tasks:

1. Ensuring that the ARK prerequisites are met.
2. Starting the license management daemon (`i4lmd`) on one or more nodes.
3. Installing the vendor's software product.
4. Creating and maintaining a user file to specify who may use the product (optional).
5. Setting the permissions on the ARK tools and databases to protect them from unauthorized access.
6. Using `i4admin` to enter the passwords for the vendor and product, thereby creating a license database at each server node. (Also, editing `nodelock` files at end user nodes, if `nodelocked` licenses are among the license types to be administered.)
7. Using `i4stat` and `i4report` to get status information and generate reports on license use (optional).
8. Using `i4admin` to monitor the expiration dates of all licenses, and the number of remaining use-once licenses.
9. Installing and starting the Client Ally.

Chapter 3 and Chapter 4 explain these tasks in detail.

If you administer a small network where only `nodelocked` licenses are used, you can choose to install the passwords at each node manually. In this case you do not need to run the license management daemon or perform any of the other tasks listed above.

---

## The ARK Administrative Tools and the Global Location Broker

The ARK uses the NCS Global Location Broker (GLB) to propagate information on the ARK databases throughout a network. As the system administrator, it is your responsibility to monitor the location brokers and to clean up unused or invalid entries. (See *Managing NCS Software* for information on how to do this.)



## Setting Up The ARK

---

This chapter explains how to set up the Administrator's Runtime Kit and provides a few examples to illustrate the process. For reference information on the commands and files discussed in this chapter, refer to the appendixes.

---

### The ARK Prerequisites

The following are prerequisites for using the ARK:

- A network transport such as TCP/IP.
- You must install the Network Computing Kernel (NCK) if this software is not standard on nodes in your network.
- You must install the Administrator's Runtime Kit if this software is not standard on nodes in your network.
- You must also make certain that all node clocks (not just server node clocks) are synchronized to within 12 hours. If this is not done, users whose node clocks differ by more than 12 hours from the server node clock may be unable to acquire licenses.

Note that these are prerequisites for using the license management daemon and its tools. If you have a network too small to warrant running a license server and its associated tools, your vendors may offer you nodelocked licenses for their software products.

## TCP/IP

If you have a heterogeneous or homogeneous network of nodes, install TCP/IP and run TCP servers, following the instructions in your TCP/IP documentation.

---

**Note** TCP daemons must be running before you can run NCS daemons and brokers.

---

## The Network Computing Kernel

The Network Computing Kernel (NCK) handles communication between license servers and the software products they regulate. NCK provides run-time support for distributed computing and is a component of NCS, a set of tools for heterogeneous distributed computing. Software products can use NCS to distribute both program execution and data among different types of processors and across one or more heterogeneous networks.

Software products licensed under *i*FOR/LS use an NCK component called the Location Broker to locate the license servers that hold their licenses. Two daemons, the Local Location Broker daemon (llbd) and the Global Location Broker daemon (glbd) are essential to Location Broker operation and hence to the operation of license servers. Because they use the Location Brokerage services, license servers depend on the following:

- The Network Computing Kernel must be present on all nodes in your licensing environment.
- The llbd process must run on every node where an *i*FOR/LS license server is running.<sup>1</sup>
- The glbd process must run on at least one node in the network where the license servers are running.
- The node that is running the glbd must also run an llbd.<sup>2</sup>

---

<sup>1</sup> If DCE is running on the server node, the DCE Remote Procedure Call Daemon (rpcd) must run in place of the llbd process.

<sup>2</sup> If DCE is running on the system where the glbd process is running, the DCE Remote Procedure Call Daemon (rpcd) must run in place of the llbd process.

### 3.2 Setting Up The ARK

---

**Note** NCS daemons and brokers must be running before you can start license servers. In HP-UX, the NCS 11bd daemon has been replaced by the DCE rpcd daemon.

Refer to *Managing NCS Software* for more details on setting up location brokers. Also refer to the *rpcd(1M)* and *glibd(1M)* manual reference pages, and to the configuration file `/etc/rc.config.d/ncs`.

---

---

## The License Server Daemon

Each license management daemon maintains its own license database at the node on which it runs. This database contains the licenses belonging to that server, along with other information needed to process license requests from users. To avoid a situation in which a problem at a single server node entirely disables user access to a product, you should configure multiple servers and distribute the concurrent-access or use-once licenses for the product among them.

To create passwords, the ID of each node used as a license server is needed by the software suppliers. Use the `i4target` utility program to determine the ID of a node.

### Choosing Nodes to Run the License Server

The criteria for choosing license server nodes are similar to those for choosing nodes for any server process:

- Do not use diskless nodes to run the license server.
- In an Internet environment, plan on at least one server node per network.

### Starting the License Server

The license server can be manually started by invoking `i4lmd`, which resides in the `/opt/ifor/ls/bin` directory.

---

**Note** See your release notes for details on each specific server platform.

---

The license server also can be invoked automatically at boot time by executing the steps described in the release notes. The next time the node is booted, the license server will start automatically.

For detailed information on `i4lmd`, consult Appendix A.

---

**Note** License server daemons must be running before the ARK tools can be used.

---

## Verifying License Servers

After starting the license servers, use `i4tv`, the network license server test and verification tool, to verify that they are running properly. The `i4tv` program resides in the `/opt/igor/ls/bin` directory. A message describing a completed license transaction, and a list of all license servers will be displayed. For complete information on the `i4tv` command, refer to Appendix C.

## Starting the Client Ally

If PC applications are to be supported, a Client Ally must be started on at least one machine in the network that is able to access a license server. Multiple machines can run Client Allies simultaneously to provide redundancy.

---

## User Files

License server user files are optional—they are necessary only if you want to prohibit some users from accessing certain products or have different access priorities.



## A Sample User File

Figure 3-1 shows a sample user file. In this example, the keywords are lowercase; they may be uppercase or lowercase, but mixed-case keywords are not allowed. Comments are preceded by a %.

```
% This line is a comment
crawl 10
% *****
vendor "Modern Solutions, Inc." "The New Math"
allow fritz -p 2 harry -p 1 monique -p 1 penny
% *****
vendor "Grafix, Inc." all
disallow heather jason
% *****
vendor "Unique Applications, Inc." dynamath
disallow bob
vendor "Unique Applications, Inc." versitex
disallow bob
vendor "Unique Applications, Inc." minigraph
disallow bob
```

**Figure 3-1. A Sample User File**

Following are explanations of the keywords and their uses in this file.

```
crawl 10
```

This keyword causes all users listed in the file to move up one level in priority after the product polls the queues of users waiting for licenses a specified number of times (in this case, 10 times). Using `crawl` makes sense only when different users have different priorities. The `crawl` keyword and user priorities apply only to products that use wait queues.

```
vendor "Modern Solutions, Inc." "The New Math"
allow fritz -p 2 harry -p 1 monique -p 1 penny
```

Here the user file declares the vendor `Modern Solutions` and its only installed product, `The New Math`. The vendor name and the product name contain spaces, and so are delimited by quotes. Four users may use `The New Math`: Fritz is a priority 2 user; Harry and Monique, priority 1 (the highest); Penny's priority is unspecified, and so defaults to 3 (the lowest).

When a high-priority user requests a license for which low-priority users are already waiting, the high-priority user moves ahead of the lower-priority users in the queues. In this case, if Penny and Fritz are unsuccessful in obtaining a license and are willing to wait, `crawl` eventually changes their priority to 1. At that time, the declared priority 1 users (Harry and Monique) can no longer displace Penny and Fritz in the user queues.

```
vendor "Grafix, Inc." all
disallow heather jason
```

The keyword `all` means that all licensed products of this vendor (whatever those products may be) have the same user authorizations. In this case, everyone except Heather and Jason may use all Grafix software products.

```
vendor "Unique Applications, Inc." dynamath
disallow bob
```

```
vendor "Unique Applications, Inc." versitex
disallow bob
```

```
vendor "Unique Applications, Inc." minigraph
disallow bob
```

These lines specify three products of Unique Applications, Inc.: Dynamath, Versitex, and Minigraph. (The product names are not delimited because the strings contain no spaces.) Note that the keyword `vendor` takes exactly one vendor and one product declaration. The keyword `disallow` excludes only Bob from using the three products of Unique Applications, Inc. All other users are priority 3 by default.

That this user file specifies the products of Unique Applications individually probably indicates that Unique Applications has other installed products that anyone, including Bob, may use. If Dynamath, Versitex, and Minigraph were the only installed products of Unique Applications (and Bob was prohibited from using them) the user file would more likely use the keyword `all`, as follows,

```
vendor "Unique Applications, Inc." all
disallow bob
```

instead of specifying each product individually.

You can use the `i4admin` command to check that the format of a user file is correct—see Appendix A for more information.

### 3-6 Setting Up The ARK

## **Writing and Integrating User Files**

If you decide to have a user file, it can be created by the use of any text editor. Put one copy on each license server node in the `/var/opt/ifor` directory. Although you can write different user files for different server nodes, or put user files on some nodes and not on others, such a policy might cause access to a product to vary depending upon which server is granting the license. The same user file should be placed on all server nodes in the interest of a consistent user authorization policy.

## **Maintaining User Files**

When adding a new product or changing user priorities for existing products, remember to update the user files at all license server nodes accordingly. See Appendix B for detailed reference information on user files.

## **User Names For Personal Computers**

The user name associated with a personal computer will be the network node name of the PC itself. Therefore, individuals who share a PC will not be distinguished from one another. The node name will be that of the node at which the Client Ally is running. The group name will be PC-NetLOK.

---

## **Adding Passwords to License Servers**

When you purchase a licensed product, the vendor will ask for the target IDs and target types of the license server nodes and how the licenses are to be distributed among them. Using this information, the vendor creates a product password for each server node, and a vendor password. The product password is a coded string that specifies license information for the server it is created for, including the product, number of licenses, their type and duration, and so forth. The vendor password, along with the vendor ID (which is also supplied by the vendor) is used to establish the vendor in the license database at each server.

Usually, the vendor delivers these passwords and the vendor ID at about the same time as the product, either by mail or telephone. This section explains how to use the `i4admin` command to add the vendor ID and password as well as the product passwords to the license databases so that users can begin using the product.

For reference information on `i4admin`, see Appendix A.

## A Sample `i4admin` Session

Suppose you want to add concurrent-license passwords for a new product to two license server databases. The decision has been made to have the 50 licenses divided equally between servers `cobweb` (target ID `a5ef`) and `apollo` (target ID `876b61`), and the vendor, Modern Solutions, Inc. is so informed. Modern Solutions then sends the following information:

- Vendor: Modern Solutions
- Vendor ID: 3750e6cc9000.0d.00.00.5c.d3.00.00.00
- Vendor password: dasghgxt3yhk2
- Product/version: Kwik-Draw, Version 2.1
- Total licenses: 50 concurrent-access licenses from 12/10/90 to 12/9/91
- Server `a5ef` password: 53g55mqt5xmxcgiaacu89f2 (specifies 25 licenses)
- Server `876b61` password: 53g55mqt5xmxcgiaabgqh2 (specifies 25 licenses)

Assume license servers are already running on `apollo` and `cobweb`. The vendor ID and passwords can be added either before or after installing Version 2.1 of `KwikDraw`.

---

### Note

The following sample session is intended as a model for using `i4admin`, not as an exercise. If you attempt to duplicate this sample session using the data supplied, *it will not work*.

Also note that this sample session assumes you have the graphic interface version of `i4admin`. Although the ARK for some platforms supports the graphic interface, the ARK for other platforms may require use of the command-line interface. Check the release document for each implementation of the ARK to find out if it supports the graphic interface.

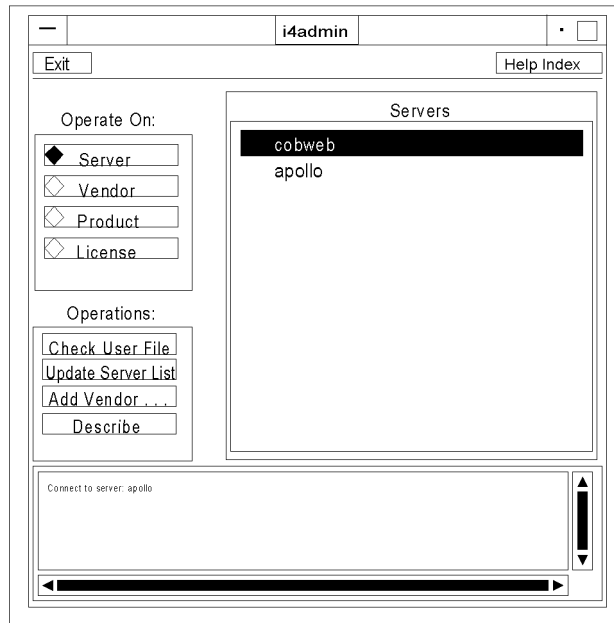
---

First, start up `i4admin`.

1. At the server node `cobweb`, change directory to the `/opt/ifor/ls/bin` directory. Type `i4admin` and a display like the one in Figure 3-2 would appear. (Depending on the type of node you are using, your display may look slightly different; but it works the same way.) Notice that the local server (`cobweb`) is selected by default. (If you want to see the node ID of

## 3-8 Setting Up The ARK

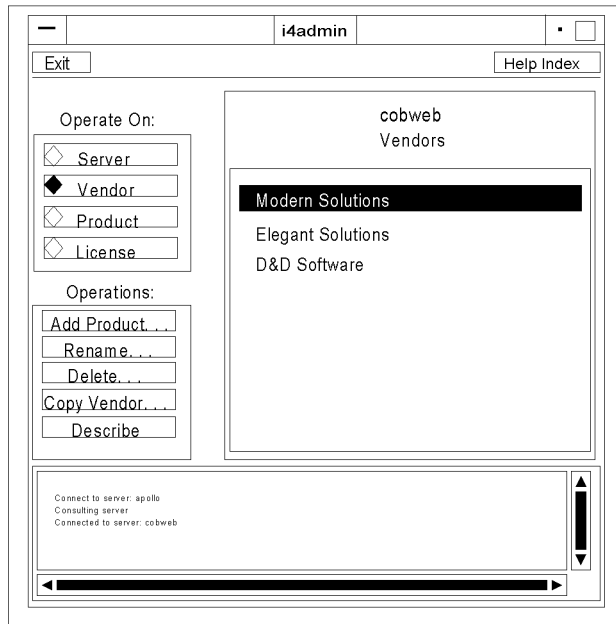
the selected license server node in the data display area below the i4admin window, select Describe.)



**Figure 3-2. Example of the i4admin Startup Display**

Now you need to add the vendor, Modern Solutions, to the server's vendor list. (If you had previously installed a product from this vendor, skip this step and just select Vendor to display the Vendor list, and then the vendor name, before continuing to Step 3.)

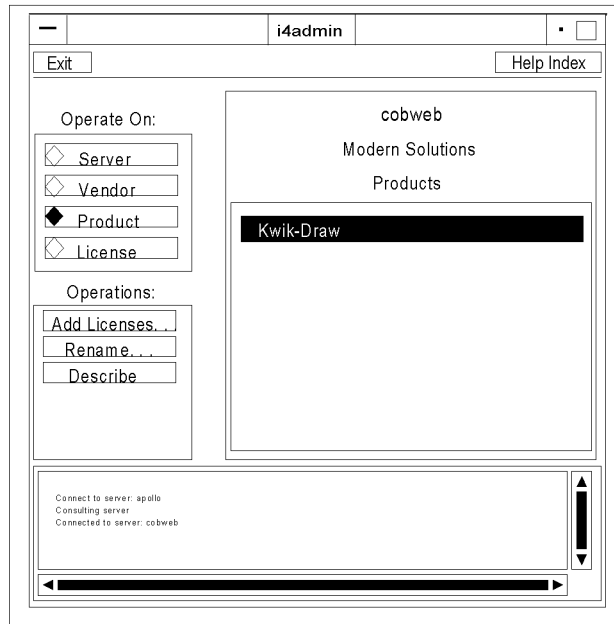
2. Use the left mouse button to select Add vendor from the Operations menu. Add the vendor name in the first field of the pop-up that appears, and then press **(RETURN)** to enter the data. Add the vendor ID and vendor password in the same manner, pressing **(RETURN)** after entering data in each field. Check to be sure that the vendor ID and password are correct and then select the Add vendor button on the pop-up. When the i4admin display re-appears (Figure 3-3) notice that Vendor has become the selected object type and that the vendor Modern Solutions appears in the list of vendors at cobweb.



**Figure 3-3. Modern Solutions, Inc. Added to the Vendor List for cobweb**

Now that you have added the vendor, and the vendor's name is selected in the Vendors list, the next step is to add the product, Kwik-Draw, and the password specifying the licenses, to the vendor's product list.

3. Select Add product from the Operations: menu. On the pop-up that appears, add the product name, version, and password for server cobweb, pressing **RETURN** after completing each field. (For this sample session, there is no license annotation, so skip the annotation field.) Then select Add product. Now when the i4admin display re-appears (Figure 3-4) notice that Product has become the selected object type and that the product Kwik-Draw is selected on the list of products for the vendor Modern Solutions at cobweb.



**Figure 3-4. Kwik-Draw Added to the Product List for Modern Solutions**

At this point, you have added the vendor Modern Solutions, its product Kwik-Draw Version 2.1, and a password specifying 25 licenses to the license database at cobweb. The next step is to copy the vendor to the other server, apollo.

4. Select Vendor from the Operate on menu, and select the vendor Modern Solutions, Inc. from the list of vendors.
5. To add the vendor data to the server apollo, select Copy vendor from the Operation menu, then select apollo from the list of servers that appears.

Now that Modern Solutions has been added to apollo's vendor list, you complete your task by adding the product Kwik-Draw and a password specifying 25 licenses to apollo's license database.

6. Select Server from the Operate on menu, and select the server apollo from the list of servers at the right of the display.
7. Select Vendor from the Operate on menu, and select the vendor Modern Solutions from the list of vendors.

8. Select **Add product**. Add the product name, version, and password for server 876b61, then select **Add product** on the pop-up.

---

## **Nodelocked Licenses**

In certain circumstances, you may have nodelocked licenses for a software product. Such circumstances might include any of the following:

- The product is licensed only on a nodelocked basis.
- Your site has a small number of nodes. For such sites, nodelocked licenses are usually more convenient to administer.
- An end user of a software product does not want to compete with other users of the product for a concurrent-access license.
- One or more server nodes at a site have become unavailable (perhaps because they are being serviced). In this case, end users do not have access to software products, even though the licenses have not expired. Since you may not be able to move the concurrent-access licenses to a server node that is available, you may receive temporary nodelocked licenses from the vendors to use until your server nodes are working.

## **Types of Passwords Specifying Nodelocked Licenses**

There are two kinds of passwords that specify nodelocked licenses:

1. Single-license passwords—each password specifies a single nodelocked license.
2. Compound passwords—each password specifies parameters for the creation of several single-license passwords that in turn specify nodelocked licenses.

### **Single-license Passwords**

Vendors usually deliver nodelocked licenses as a set of single-license passwords that are installed individually by editing the nodelock file at each node to be licensed.

### **Compound Passwords**

In some cases, nodelocked licenses are specified as compound passwords, rather than as single-license passwords. Compound passwords are usually issued for



products that are licensed on a nodelocked basis and that are to be installed on a large network.

If only a few nodes are to be licensed, it does not take long to edit the nodelock file at each node manually, adding a single-license password at each node. If many nodes need to be licensed, compound passwords can automate the otherwise tedious task of editing perhaps hundreds of nodelock files. This is known as dynamic nodelocking.

Dynamic nodelocking works like this:

1. When a user invokes the nodelocked product, the product looks for a nodelocked license in the file `nodelock` located in the `/var/opt/ifor` directory.
2. If there is a nodelocked license, the software product is made available to the user. If there is no nodelocked license, the product requests one from the license management daemon and passes the target ID of the node on which the product is running.
3. Assuming there is a compound password installed at the license server for the requesting product, the server creates (and passes back to the requesting node) a single-license password, the target of which is the target ID of the requesting node. The license server then decrements by one the number of nodelocked licenses specified by the compound password.
4. The product automatically installs the single-license password in the nodelock file of the node on which it is running.

This procedure is repeated until all nodes have installed nodelocked licenses, or until all nodelocked licenses specified by the compound password are used up. Although you need the license server and the `i4admin` tool to install a compound password, once all the single-license passwords it specifies have been installed, you no longer need the license server to run the product.

## **Installing Passwords for Nodelocked Licenses**

The software vendor should inform you of whether you are getting single-license passwords or compound passwords for your nodelocked licenses. Then do one of the following:

- To install single-license passwords for nodelocked licenses, use the procedures described in the following subsections.

- To install compound passwords for nodelocked licenses, follow the instructions in Adding Passwords to License Servers earlier in this chapter. Then invoke the product at all nodes that are to be licensed.

### Installing Single-license Passwords

For example, if a software vendor has given you the following vendor ID and nodelocked license password for `apollo` (target ID 876b61):

```
Vendor ID:  
38241e850000.0d.00.00.da.b4.00.00.00  
Password for node 876b61:  
87thcmbc4474eeamascqaa
```

Following is the procedure for adding this password to node `apollo` (target ID 876b61):

1. Use any text editor to edit the file `nodelock`.
2. At the first blank line of the file, type the vendor ID:  

```
38241e850000.0d.00.00.da.b4.00.00.00
```
3. *On the same line*, type a space after the vendor ID and then type the password:

```
87thcmbc4474eeamascqaa
```

4. If there is a license annotation, type a space after the password, then type the annotation on the same line. Annotations containing spaces must be enclosed in quotation marks ("). If there is no annotation, continue.
5. Write and close the file.

If more than one nodelocked license password is to be installed at a node, add the vendor ID and password for the node to the node's `nodelock` file exactly as described above.

You can also add comments to a `nodelock` file by preceding the comments with a `#` character, either at the end of a line occupied by a vendor ID and password, or on a separate line by itself.

---

**Note** It is recommended that you add as comments to the `nodelock` file the name of the product the license is for and its expiration date, since that information is not otherwise available.

---

```
# Dynamath licenses - Expire 08/25/97
68b4b677a609.02.c0.5c.6e.19.00.00.00 4ya29c4e84bbrdxtafjsua3X "Product 4""2.0"

# Minigraph licenses - Expire 05/31/99
3ffa59166000.0d.00.01.4e.4f.00.00.00 4imh242g3ib9d8ak9kumdhi kmn """"3.0"

# Versitex licenses - Expire 12/31/98
5gje93849dd3.0z.00.7c.21.2m.00.00.00 xibpuujhiwb4qqqkixifyjz6naa "" "1.2"
```

**Figure 3-5. A Sample nodelock File**

### **In Case of a Problem**

If a node with a nodelocked license won't allow an end user to use the corresponding product, check to be sure that:

- The nodelock file is named `nodelock` and is located in the `/var/opt/ifor` directory.
- The vendor ID and password are correct—it is always a good idea to double check this data.
- The format of the file is correct—the vendor ID and password must appear on the same line, the ID followed by the password, and the two separated by a space. If there is a license annotation, it follows the password, separated by a space. Each additional nodelocked license installed on the node must be in the same format.



## Monitoring The ARK

---

This chapter discusses the `i4stat` (license server status) and `i4report` (license server report) commands, which enable checking of the license server installation status and displays the history of the ARK events. See Appendix A for complete reference information on these commands.

---

### Getting Status Information With `i4stat`

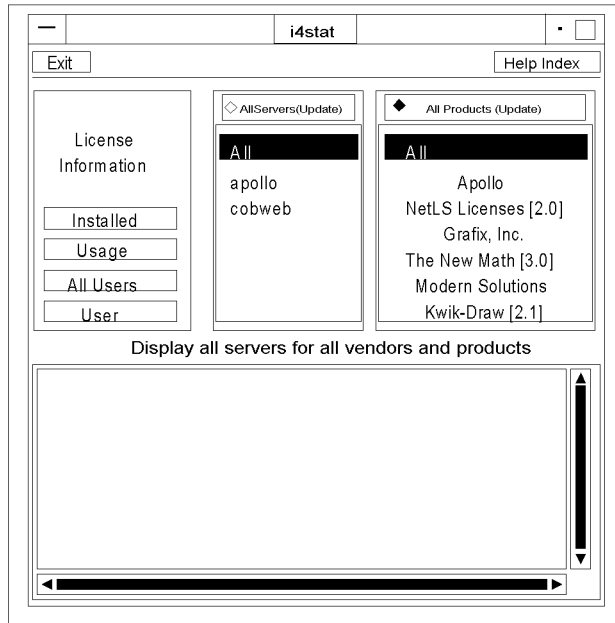
The `i4stat` command provides both users and administrators with status information on product licenses. The information `i4stat` provides includes:

- Lists of the `i4lmd` servers on the network
- Lists of licensed products at all servers or at selected servers
- Information, listed by vendor, product and server, about current users of licensed products, including user ID, node name, group, number of licenses held, and start time
- Information, listed by vendor, product and server, about product licenses installed at selected servers, including number of active licenses, their start and end dates, their type, the number of licenses currently in use, and the length of the queue of users waiting for licenses.
- Information, listed by vendor, product and server, about the usage of products, including licenses in use, total number of licenses, and licenses available.

This chapter introduces the `i4stat` graphical interface; Appendix A provides reference information on the graphical interface and on the alternative command-line interface.

When `i4stat` is invoked, a display such as the one shown in Figure 4-1 appears. (Depending upon the type of node you are on, the display may look slightly

different, but it works the same way.) If not all of the display is visible, use the mouse to enlarge the window.



**Figure 4-1. Using i4stat, Step 1**

At the top are the `i4stat` configuration windows. Beneath the `i4stat` configuration windows is a larger transcript window in which the data requested is displayed. (Messages are also displayed in this window.) The `i4stat` configuration windows contain the following objects:

- |                          |  |
|--------------------------|--|
| Exit Button              | Select (click the left mouse button on) this button to exit from <code>i4stat</code> .   |
| Help Index ...           | The Help Index ... button provides access to the on-line help system.  |
| License Information Menu | This menu contains four buttons: <code>Installed</code> , <code>Usage</code> , <code>All Users</code> , and <code>Users</code> .. Select these buttons to display information about users, installed licenses, and usage of the selected server/product. |

## 4-2 Monitoring The ARK

Server List Box	This list box, directly to the right of the License Information menu, displays the server list. Use the scroll bar at the left side of this box to scroll the list. The All Servers (Update) button at the top of this box is explained below.
Product List Box	This list box, directly to the right of the Server List box, displays the product list. Use the scroll bar at the left side of this box to scroll the list. The All Products (Update) button at the top of this box is explained below.
All Servers (Update) Button	<p>This button polls the network and updates the server list. When this button is selected, a check mark appears in the box at its left. A check mark in this box indicates that:</p> <ul style="list-style-type: none"> <li>■ All existing servers are displayed in the Server List box.</li> <li>■ The vendors and products listed in the Product List box are the vendors and products existing at the server currently selected in the Server List box.</li> </ul>
All Products (Update) Button	<p>This button polls the network and updates the product list. When you select this button, a check mark appears in the box at its left. A check mark in this box indicates that:</p> <ul style="list-style-type: none"> <li>■ All existing vendors and products are displayed in the Product List box.</li> <li>■ The servers listed in the Server List box are the servers that hold licenses for the product or vendor currently selected in the Product List box.</li> </ul>
Status Message Field	This field, across the bottom of the window, describes the information currently displayed in the Server List box and the Product List box.

---

## A Sample i4stat Session - Getting Status Information

Suppose that, having installed licenses for Kwik-Draw, Version 2.1, we decide to check the license databases at `apollo` and `cobweb` to verify the installation and to find out if anyone is using that product. To do this, we run the license server status program `i4stat`.

---

**Note**            The following sample session is intended as a model for using `i4stat`, not as an exercise. If you attempt to duplicate this sample session using the data supplied, it will not work.

                  Also note that this sample session assumes you have the graphic interface to `i4stat`. Although the ARK for some platforms supports the graphic interface, the ARK for other platforms may require use of the command- line interface. Check the release document for each implementation of the ARK to find out if the graphic interface is supported.

---

First, invoke `i4stat`.

1. Run `i4stat` to display the `i4stat` window (see Figure 4-1).

Notice that, by default, all servers and all products are listed. The display shows three vendors, each with one installed product: Apollo and its product, NetLS Licenses, Version 2.0; Grafix, Inc. and its product, The New Math, Version 3.0; and Modern Solutions and its product, Kwik-Draw, Version 2.1.

---

**Note**            Products are distinguished from vendors by the square brackets that enclose the product version text.

---



Now check to see if the licenses for Kwik-Draw have been correctly installed.

2. Use the left mouse button to select Kwik-Draw [2.1] from the product listing, then select Installed. The information in the transcript pad (See Figure 4-2) shows that 25 licenses were installed at apollo, and 25 at cobweb, and one license (at cobweb) is in use. No users are waiting in the queue at either apollo or cobweb to get licenses for this product.

Notice that the status message field has changed: only servers having licenses for Kwik-Draw are displayed.

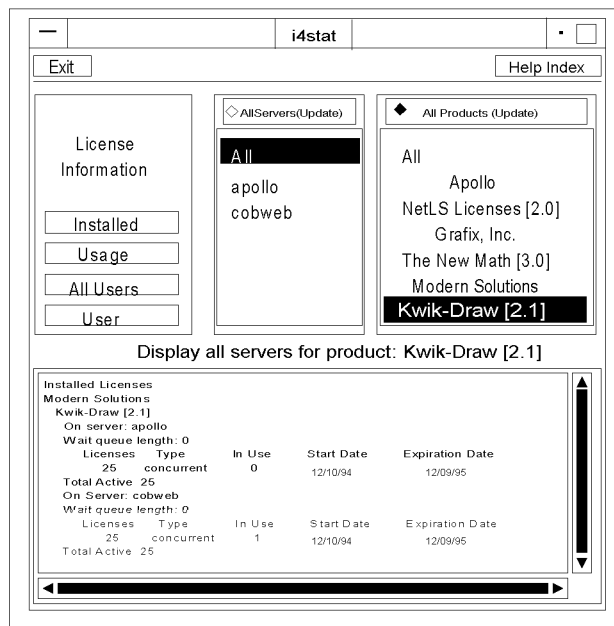
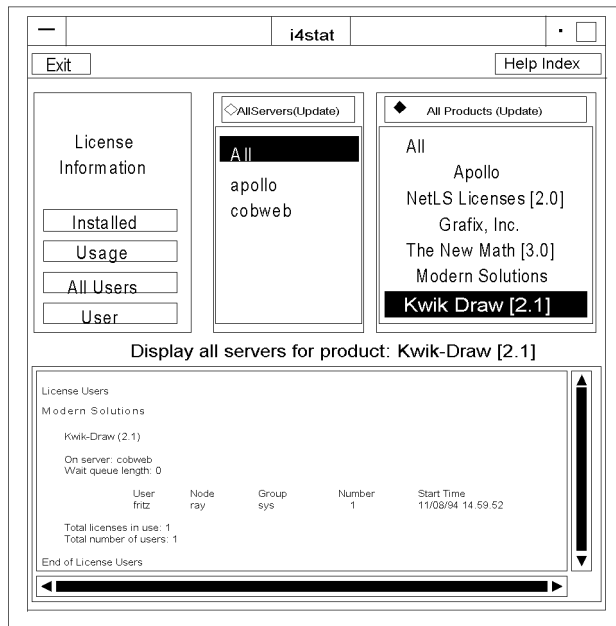


Figure 4-2. Using i4stat, Step 2

Next, see who is using the product.

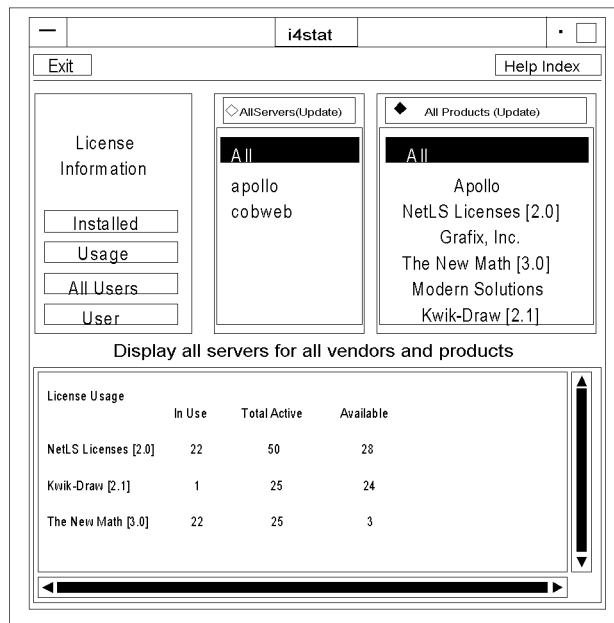
3. To find out who has the license, select **Users**, and then select **All** on the pop-up that appears. As shown in the transcript pad (Figure 4-3), the user is fritz at node ray.



**Figure 4-3. Using i4stat, Step 3**

## 4-6 Monitoring The ARK

- To find out usage of all installed products, select **All** from the **Servers** list and **All** from the **Products** list. Then select **Usage**. The display will look like Figure 4-4:



**Figure 4-4. Using i4stat, Step 4**

---

## License Server Reports

The **i4report** command generates reports on all kinds of license server activities. Different kinds of events can be reported (as well as all events) and use dates, software vendors, products, and users as information filters. Besides monitoring license server activities, license server reports can track demand for software products. This kind of information is useful for planning future purchases of licenses for those products. For complete information on **i4report**, see Appendix A.



# A

## **The ARK Command Reference**

---

This appendix contains descriptions of the ARK administration tools `i4admin`, `i4report`, and `i4stat`, and of the license management daemon `i4lmd`. The descriptions appear in alphabetical order. The ARK tools reside in the `/opt/igor/ls/bin` directory.

---

## i4admin - Graphic Interface

i4admin - Display and edit the license server database

### Format

i4admin

The `i4admin` command allows you to edit and examine license databases. This command description explains the graphic interface of `i4admin`. The command-line interface is explained separately in this appendix.

### Menus And Buttons

Exit Button	Exits from <code>i4admin</code> .
Operate On: Menu	This menu lists the license server objects you can operate on: Server, Vendor, Product, and License.
Server	Select <b>Server</b> to display a list of servers. After you select a server, you can select <b>Vendor</b> to display a list of vendors for that server, or you can select an operation to perform on the selected server from the <b>Operations:</b> menu.
Vendor	Select <b>Vendor</b> to display a list of vendors for the server selected in the <b>Servers</b> list. After you select a vendor, you can select <b>Product</b> to display a list of products for that vendor, or you can select an operation to perform on the selected vendor from the <b>Operations:</b> menu.
Product	Select <b>Product</b> to display a list of products for the selected server and vendor. After you select a product, you can select <b>License</b> to display a list of licenses for that product, or you can select an operation to perform on the selected product from the <b>Operations:</b> menu.
License	Select <b>License</b> to display a list of license records for the selected server, vendor, and product. Each record shows the number, type, and terms of the licenses. Select a license record and select an operation from the <b>Operations:</b> menu.

## i4admin - Graphic Interface

`Operations: Menu` This menu lists the license database operations you can perform. The contents of this menu vary depending on the object (Server, Vendor, Product, or License) selected in the `Operate On:` menu.

### Operations On Servers

`Check user file` Verifies that the format of the file `user_file`, located in the `/var/opt/ifor` directory, is valid.

`Update server list` Updates server and/or license database information. The information displayed is current, so it is generally unnecessary to use `Update server list` unless a communications failure has been repaired or a new server has been started since you invoked `i4admin`, or another user is currently editing a license database with `i4admin`.

`Add vendor` Adds a vendor to the selected license database. Enter the vendor name, vendor ID and vendor password on the pop-up; then select `Add vendor`.

`Describe` Provides detailed information about the selected server, including socket information, target type and target ID.

### Operations On Vendors

`Add product` Adds a product to the selected vendor at the selected server. Enter the product name, version, product password, and **license annotation** (if there is one) on the pop-up; then select `Add vendor`. If you add a product to more than one server, be sure to use exactly the same product name at all servers. Note that `Add product` performs two functions: it establishes a new product, and it adds licenses for the product. To add more licenses for an existing product, select the product and then use `Add licenses`.

`Rename` Renames the selected vendor. Enter the new vendor name on the pop-up. If you rename a vendor at one server, you should also rename it (using the same name) at all servers where that vendor is listed.

## **i4admin - Graphic Interface**

Delete	Deletes the selected vendor at the selected server. Select the <b>Delete?</b> pop-up to confirm the operation. Move the cursor off the pop-up to cancel the operation. You cannot delete a vendor that has active licenses for its products.
Copy vendor	Copies the selected vendor to another server's license database. Select the server to which you want the vendor copied from the pop-up that appears.
Describe	Provides detailed information about the selected server and vendor, including the vendor ID.

## **Operations On Products**

Add licenses	Adds licenses to the selected product. Enter the license password on the pop-up. (Use <b>Add licenses</b> only to add more licenses for an existing product. If you are both establishing a new product and adding licenses for the product, use <b>Add product</b> rather than <b>Add licenses</b> .)
Rename	Renames the selected product. Enter the new product name on the pop-up. If you rename a product at one server, you should also rename it (using the same name) at all servers where that product is listed.
Describe	Provides detailed information about the selected server, vendor, and product. Product information displayed includes the ID, annotation string, and the number, type, and date of existing licenses for the product.

## **Operations On Licenses**

Delete	Deletes the selected license record. This enables you to get rid of expired licenses. Select the <b>Delete?</b> pop-up to confirm the operation, or move the cursor off the pop-up to cancel.
Describe	Provides detailed information about the selected server, vendor, product, and license record. License information displayed includes the number, type, date, and time stamp.



---

## i4admin - Command-Line Interface

i4admin - Display and edit the license server database

### Format

```
i4admin [-n node_name] [-r] [-l | -z | -h | -usage | -version ]
{-a | -s | -d | -f } {-v | -p} argument1 argument2...
```

The `i4admin` command allows you to examine and edit license databases. This command description describes the command-line interface to `i4admin`. The graphic interface is explained separately in this appendix.

### Options

Default options are indicated by *(D)*.

- `-n node_name` Indicates the server at which the license database to be edited or displayed resides. (Optional; the default value is the name of the license server node at which the command is executed.)
- `-r` Specifies a version of a product to be operated upon.
- `-l` Specifies the license annotation.
- `-z` Debugging flag. (Prints RPC debugging information.)
- `-h` Displays command usage information. (Same as `-usage`.)
- `-usage` Displays command usage information. (Same as `-h`.)
- `-version` Displays command version information.
- `-a (D)` Adds a new vendor, a new product (and licenses), or more licenses for an existing product to the license database.  
  
If adding a vendor, specify (as arguments to the `-v` option) the vendor name, vendor ID, and vendor password.  
  
If adding a new product and licenses, specify (as arguments to the `-p` option) the vendor name, product name, product password, version text, and license annotation (if there is one) as arguments. (Do not use the `-r` option in this case.)  
  
You must have previously added the vendor in order to add its product, and you may not establish a vendor and product

## **i4admin - Command-Line Interface**

licenses simultaneously in a single command line. If adding new licenses for an established version of a product, you may not specify a license annotation unless the established version had an annotation.

The same annotation must be used in all licenses for a given product (identified by the product ID and version).

The options `-a`, `-d`, and `-s` are mutually exclusive.

- `-s` Shows information about the specified license server, vendor or product. To show information about a license server, use the `-n` option with the node name as the argument. To show information about a vendor, use the `-v` option with the name of the vendor as the argument. To show information about all vendors at a license server, use the `-v` option without an argument. To show information about a product version, use the `-r` option with the version text as the argument followed by the `-p` option with the vendor name and product name as arguments. To show information about all versions of a product use the `-r` option without an argument, followed by the `-p` option with the vendor name and product name as arguments. To show information about all versions of all products of a vendor, use the `-p` option, giving the vendor name as the only argument.

The options `-a`, `-d`, and `-s` are mutually exclusive.

- `-d` Deletes a vendor or product from the license database. To delete a vendor, use the `-v` option with the vendor name as the argument. You may not delete a vendor unless you have previously deleted all versions of all products of the vendor at the current server, nor may you delete more than one vendor at a time. To delete a product, use the `-p` option with the vendor name and product name as arguments, followed by the license time stamp. You may not delete use-once licenses nor compound passwords that have not expired, nor may you delete more than one version of a product at a time. Use the `-s` and `-p` options to get the time stamp of the specified product licenses. The options `-a`, `-d`, and `-s` are mutually exclusive.
- `-f` Copies a vendor (specified with the `-v` option) from the server specified in the `-f` option to the server specified in the `-n` option, or to the default server if no `-n` server is specified.

## i4admin - Command-Line Interface

- v Specifies the vendor to be operated upon. -v or -p and their arguments must appear last on the command line.
- p Specifies the product to be operated upon. -p or -v and their arguments must appear last on the command line.

### Example Formats

In the following examples, argument items represented by terms such as *vendor\_name* and *product\_name* must appear in the command line separated by spaces. If a given argument item contains spaces, it must be enclosed in double quotes (""). For example, a *vendor\_name* like Acme Firmware, Inc. must appear in the actual command line as "Acme Firmware, Inc." Also, vendor and product names must be case-correct.

To add a vendor:

```
i4admin [-n node_name] -a -v vendor_name vendor_id vendor_password
```

To add a product or additional licenses:

```
i4admin [-n node_name] -a [-l annotation] -p vendor_name product_name  
product_password product_version
```

---

**Note** The -l *annotation* parameter must be included for those products having annotations.

---

To show servers:

```
i4admin [-n node_name] -s
```

To show vendors:

```
i4admin [-n node_name] -s -v [vendor_name]
```

---

**Note** If *vendor\_name* is not specified, this command shows all vendors at the specified server. If no server is specified, all vendors at the default server (the one on the node the command is run from) are displayed.

---

To show all products for a single vendor at the specified server:

```
i4admin [-n node_name] -s -p vendor_name
```

## **i4admin - Command-Line Interface**

To show all licenses for all versions of a specified product of a specified vendor:

```
i4admin [-n node_name] -s -p vendor_name product_name
```

To show a specified version of a specified product of a specified vendor:

```
i4admin [-n node_name] -r version -s -p vendor_name product_name
```

To copy a vendor from another server:

```
i4admin -f node_name -v vendor_name
```

To delete a vendor:

```
i4admin [-n node_name] -d -v vendor_name
```

---

**Note**            You cannot delete a vendor who has products listed (that is, you must delete all the products first).

---

To delete a product:

```
i4admin [-n node_name] -d -p vendor_name product_name time_stamp
```

---

**Note**            Products must be deleted one at a time and are distinguished by their time stamps.

---

## i4report - Command-Line Interface

i4report - Reports on network license server events

### Format

```
i4report [ [-n node_name] [-c] [-z] [event_type_list]
[information_filter_list] | [ -h | -usage | -version] ]
```

The `i4report` command generates reports on license server events. There is no graphic interface for this command.

### Options And Arguments

Default options are indicated by *(D)*.

-n	Specifies the server node about which the report is to be generated. If you do not specify a node, <code>i4report</code> reports on the current server node.
-c	Lists data in 80-column format.
-z	Debugging flag. (Prints RPC debugging information.)
-h	Displays command usage information. (Same as <code>-usage</code> .)
-usage	Displays command usage information. (Same as <code>-h</code> .)
-version	Displays command version information.

## i4report - Command-Line Interface

*event\_type\_list*            You may specify any combination of the following event types. Specify -a to specify all event types.

**Table A-1. Event Types**

<b>Flag</b>	<b>Event Type</b>
-a	Lists all log messages.
-l ( <i>D</i> )	Lists all license-related events (product received license, product released license to server, user entered license queue, user exited queue).
-e	Lists all error events.
-s	Lists all server start/stop events.
-m	Lists all messages that were logged by a software product or a license server.
-f	Lists any fatal error events.
-d	Lists all license database modification messages.

## i4report - Command-Line Interface

*information\_filter\_list* You may choose any combination of the following information filters. If no filters are specified the default is all dates, all vendors, all products, all users.

**Table A-2. Information Filters**

<b>Flag</b>	<b>Filters This Information</b>
<code>-b mm/dd/yy</code>	Lists events that occurred beginning at the specified date.
<code>-t mm/dd/yy</code>	Lists events that occurred up to the specified date.
<code>-v vendor_name</code>	Lists events related to the specified vendor(s).
<code>-p product_name</code>	Lists events related to the specified product(s).
<code>-u user_name</code>	List events related to the specified user(s).
<code>-r 1</code>	Lists for the specified product the number of requests for licenses, the number of licenses granted, and the percent of rejected requests.
<code>-r 2</code>	Lists the same information as <code>-r 1</code> , but also includes user names and number of licenses installed.
<code>-x mm/dd/yy</code>	Deletes log file entries written on and before the specified date.

### Examples

List license events on the local server node:

```
i4report
```

List errors and fatal errors occurring between August 31 and September 30, 1995 on the server node plums:

```
i4report -n plums -e -f -b 08/31/95 -t 09/30/95
```

List all messages logged at mars by the vendor XYZ:

```
i4report -n mars -m -v xyz
```

---

## **i4stat - Graphic Interface**

`i4stat` - Displays the status of the license server system

### **Format**

`i4stat`

The `i4stat` command displays, in the transcript pad below the `i4stat` configuration windows, status information on licenses. Note that end users as well as system administrators may find `i4stat` useful for finding out the status of licenses in an *i*FOR/LS system. This command description explains the graphic interface of `i4stat`. The command-line interface is explained separately in this appendix.

### **Menus And Buttons**

Exit Button	Select this button to exit from <code>i4stat</code> .
License Information Menu	This menu contains three buttons: Installed, Usage, All Users, and User. After you have selected a server and product from the Server and Product lists, select these buttons to display information about users, installed licenses, and usage of the selected server and product.
Installed Button	Displays information, listed by vendor, product and server, about product licenses installed at selected servers, including number of active licenses, their start and end dates, their type, the number of licenses currently in use, and the length of the queue of users waiting for licenses.
Usage Button	Displays information, listed by vendor, product and server, about the usage of products, including number of licenses in use, total number of licenses, and number of licenses available.
All Users Button	Displays information, listed by vendor, product and server, about current users of licensed products, including user ID, node name, group, number of licenses held, and start time.



## i4stat - Graphic Interface

User Button	Displays information, listed by vendor, product and server, about a specific user of licensed products, including user ID, node name, group, number of licenses held, and start time. After the <code>User</code> button is selected, a pop-up dialog is displayed in which you may enter a user ID.
Server List Box	This list box, directly to the right of the License Information menu, displays the server list. At the top of this box is the <code>All Servers (Update)</code> button (see below.) At the left of the box is a scroll bar that you can use to scroll the list.
All Servers (Update) Button	Select this button to poll the network and update the server list. When you select this button, a check mark appears in the box at its left. A check mark in this box indicates that: <ul style="list-style-type: none"><li>■ All existing servers are displayed in the Server List box.</li><li>■ The vendors and products listed in the Product List box are the vendors and products existing at the server currently selected in the Server List box.</li></ul> After updating the server list, select a server to display (in the Product List box) the products it administers. Next, select a product (or <code>All</code> ) from the list of products; then select <code>Users</code> , <code>Installed</code> , or <code>Usage</code> .
Product List Box	This list box, directly to the right of the Server List box, displays the server list. At the top of this box is the <code>All Products (Update)</code> button (see below). At the left of the box is a scroll bar that you can use to scroll the list.
All Products (Update) Button	Select this button to poll the network and update the product list. When you select this button, a check mark appears in the box at its left. A check mark in this box indicates that: <ul style="list-style-type: none"><li>■ All existing vendors and products are displayed in the Product List box.</li><li>■ The servers listed in the Server List box are the servers that hold licenses for the product currently selected in the Product List box.</li></ul>

## **i4stat - Graphic Interface**

After updating the product list, select a product to display (in the Server List box) the servers holding licenses for the product. Select a server (or All) from the list of servers; then select **Users**, **Installed**, or **Usage**.

Status Message Field    This field, across the bottom of the window, describes the information currently displayed in the **Server List** box and the **Product List** box.

---

## i4stat - Command-Line Interface

i4stat - Displays the status of the license server system

### Format

```
i4stat {-t | -i | -a | -u user_name} [ [-n server ] [ -v vendor ]
[ -p product [ -r version ] ] [ -z ] ] | [ -h | -usage | -version]
```

The `i4stat` command provides status information on network licenses (that is, all license types except `nodelocked`). End users as well as system administrators may find `i4stat` useful for finding out the status of licenses. This command description explains the command-line interface of `i4stat`. The graphic interface is explained separately in this appendix.

### Options And Arguments

<code>-t</code>	Displays a table of total license usage compared to installed licenses; all servers and all products are listed by default.
<code>-i</code>	Displays installed licenses; all servers and all products are listed by default.
<code>-a</code>	Displays information about all concurrent license users; all servers and all products are listed by default.
<code>-u <i>user_name</i></code>	Displays licenses being used by the specified user.

---

**Note** One of the options listed above must be included in all `i4stat` command lines.

---

<code>-n <i>server</i></code>	Displays licenses located at the specified server.
<code>-v <i>vendor</i></code>	Displays licenses of the specified vendor; if the vendor string contains spaces, it must be delimited by single or double quotes.
<code>-p <i>product</i></code>	Displays licenses for the specified product; if the product string contains spaces, it must be delimited by single or double quotes.

## **i4stat - Command-Line Interface**

<code>-r <i>version</i></code>	Displays licenses for the specified revision of a product specified by <code>-p</code> ; if the version string contains spaces, it must be delimited by single or double quotes.
<code>-z</code>	Debugging flag. (Prints RPC debugging information.)
<code>-h</code>	Displays command usage information. (Same as <code>-usage</code> )
<code>-usage</code>	Displays command usage information. (Same as <code>-h</code> )
<code>-version</code>	Displays command version information.

## **Examples**

Display all licenses installed for all products on all servers:

```
i4stat -i
```

Display licenses in use from the server park:

```
i4stat -a -n park
```

Display licenses installed and currently in use for the product Kwik-Draw, Version 2.1:

```
i4stat -a -i -p Kwik-Draw -r 2.1
```

Display licenses installed on park for the vendor Apollo.

```
i4stat -i -v Apollo -n park
```

## i4lmd - Command-Line Interface

i4lmd - (license management daemon) Starts the license server

### Format

```
i4lmd [ [ -no event_list | -o | -v | -z ] ] | -l log_name
```

The `i4lmd` command starts a license server on the local node. There is no graphic interface for this command.

---

**Note** See the release notes for information on how to automate the startup of `i4lmd` on your specific platform.

---

## i4lmd - Command-Line Interface

### Options

- no** Turns off logging of the events specified in *event\_list*. Any combination of events is valid, but items in the list of events *must not* be separated by spaces or other characters. The following table lists the event types that you may specify:

**Table A-3. Event Types**

Flag	Event Type
l	License-grant and license-release events.
c	License check-in events. (Licensed products usually check in with the license server at regular intervals while a user is using the product.)
w	Waiting events: these include wait events (a user was waiting for a license), wait-grant events (a user was waiting for and then was granted a license), and wait-remove events (a user was waiting for a license and then asked to be removed from the queues before a license was granted).
v	Vendor events: a vendor was added, renamed or deleted.
p	Product events: a product was added, renamed, or deleted.
e	Error events.
t	License timeout events. (When a licensed product fails to check in with the license server, it may stop running after it “times out.” The vendor of the product sets the timeout interval, which is how long a product may run after it has lost contact with the license server).
m	Message events.
s	License server start/stop events.

- o** Overrides the in-use flag at a license server database. While a license server is running, its database is flagged as being in use to prevent more than one server from running on the same node. When a license server stops running, the flag is reset. However, if a license server exits abnormally, the flag may not be reset, which prevents the server from restarting. The **-o** option overrides an in-use flag, allowing the server to be

## i4lmd - Command-Line Interface

restarted. Do not use `-o` unless you are sure the license server is not running.

- `-s` Secure mode — A license server running in secure mode will only permit modifications to its database from tools run locally (on the same node). Tools running on remote license servers are not permitted to modify the database.
- `-v` *i*FOR/LS library verbose mode.
- `-z` Debugging flag. (Prints RPC debugging information.)
- `-l` Redirects license server log entries to a file and location other than the default (`/var/opt/ifor/log_file`). Alternative log file specifications (*log\_name*) must be fully qualified starting from the root directory (*/*).

### Examples

Start a license server; do not log check-in, vendor, product, timeout, or message events:

```
i4lmd -no cvptm
```

Start a license server, overriding the in-use flag:

```
i4lmd -o
```





# B

## Format of the User File

---

This appendix describes the format of the file `user_file`. An example of a user file can be found in Chapter 3, at Figure 3-1.

```
[CRAWL n]

% comment

VENDOR 1st_vendor_name {ALL |
1st_product_name}

{ALLOW | DISALLOW}

[1st_user_name [-P1 | -P2]
[nth_user_name [-P1 | -P2]

VENDOR 1st_vendor_name {ALL |
nth_product_name}

{ALLOW | DISALLOW}

[1st_user_name [-P1 | -P2]
[nth_user_name [-P1 | -P2]

VENDOR nth_vendor_name {ALL |
1st_product_name}

{ALLOW | DISALLOW}

[1st_user_name [-P1 | -P2]
[nth_user_name [-P1 | -P2]
```

**Figure B-1. Format of the User File**

Figure B-1 shows the format of the file `user_file`, which resides in the `/var/opt/iform` directory. You may specify any number of vendors with associated products and user authorizations, and any number of user names (or none) may follow the `ALLOW` or `DISALLOW` keywords. Keywords may be uppercase or lowercase, but mixed-case keywords are not allowed. Single or double quotes must delimit vendor or product names that contain spaces. Following are explanations of the keywords in a user file:

<code>CRAWL</code>	Causes the priority of users to be changed to the next higher priority after the application has polled the queue <code>n</code> times. Crawling may be specified only once in a user file. This keyword only applies to products that use queuing.
<code>VENDOR</code>	Specifies a vendor of licensed product(s). Either <code>All</code> or the name of a single product must follow the declaration of a vendor.
<code>ALL</code>	Specifies that all of the vendor's products have the same user authorizations. The keyword <code>ALL</code> and the name of a single product are mutually exclusive.
<code>ALLOW</code>	Specifies that the user names following this keyword are allowed to use the product(s). If no user names follow this keyword, it means no users are allowed to use the product(s). <code>ALLOW</code> and <code>DISALLOW</code> are mutually exclusive.
<code>DISALLOW</code>	Specifies that the user names following this keyword are not allowed to use the product(s). If no user names follow this keyword, it means all users are allowed to use the product(s). <code>ALLOW</code> and <code>DISALLOW</code> are mutually exclusive.
<code>-P 1</code>	Specifies a priority-1 user (highest priority). Priority-1 users may displace lower-priority users already waiting in a queue. Default if omitted: priority 3 (lowest priority). This keyword only applies to products that use queuing.
<code>-P 2</code>	Specifies a priority-2 user. Priority-2 users may displace priority-3 users already waiting in a queue. Default if omitted: priority 3. This keyword only applies to products that use queuing.

## **B-2 Format of the User File**

# C

## Troubleshooting The ARK

---

This appendix describes `i4tv`, the network license server test and verification tool. You may run `i4tv` at any time to verify that license servers are working properly. Also included in this appendix is a table (Table C-1) that summarizes the most common license server problems and their causes, and tells how to fix them.

---

## **i4tv**

`i4tv` - Verifies that license servers are working

### **Format**

```
i4tv [ -h | -usage | -version]
```

The `i4tv` tool requests a concurrent-access license from a license server and prints a list of active license servers.

- If you can run `i4tv` successfully but are still having a problem with a license product, the problem is probably with the licenses, or possibly with the product itself: in this case, talk to the vendor of the licensed software product.
- If you cannot run `i4tv` successfully and receive one of the error messages listed below, use the explanation of the error to fix the problem. Then try running `i4tv` again.
- If you cannot run `i4tv` successfully and receive an error that's not listed below, it means there is a problem with the software on which the ARK is layered (for example, TCP or NCS), or a hardware problem.

### **Options**

- |                       |   |
|-----------------------|---|
| <code>-h</code>       | Displays command usage information. (Same as <code>-usage</code> .) |
| <code>-usage</code>   | Displays command usage information. (Same as <code>-h</code> .)     |
| <code>-version</code> | Displays command version information.                               |

## Error Messages

Error	Meaning
netls_no_svrs_found	Either no license servers are running, or someone has deleted the vendor Apollo form license servers.
netls_license_not_found	Someone has deleted the licenses for i4tv; this prohibits anyone from using the test and verification tool.
netls_not_authorized	Someone has edited the user file to restrict the use of i4tv.
netls_bad_timestamp	Node clocks have not been synchronized to within 12 hours.

## Examples

Run the ARK test and verification tool:

```
i4tv
```

```
LS-TV Version 2.0.1 -- IFDR/LS Test and Verification Tool
© Copyright 1991, 1992, 1993, Hewlett-Packard Company, All Rights Reserved
© Copyright 1991, 1992, 1993, Gradient Technologies, Inc. All Rights Reserved
Completed license transactions on node 1f9a4 running IFDR/LS 2.0.1
Active IFDR/LS Servers:
  altair (DomainOS SR 10.3) running IFDR/LS 2.0.1
  cobweb (SunOS) running IFDR/LS 2.0.1
```

**Figure C-1. Example output from i4tv**

**Table C-1. Common License Server Problems**

<b>Problem</b>	<b>Probable Cause</b>	<b>Solution</b>
License server won't start	No glbd running	Run glbd <sup>1</sup>
glb won't start	No llbd running	Run llbd <sup>1</sup>
glb won't run	No NCK	Install NCK
Application fails	No servers running	Start servers <sup>2</sup>
The ARK prints message " <i>status</i> (rpc_\$lock) cannot get 0 lock at <i>socket address (command)</i> cannot create socket (network computing system/RPC run time)"	TCP/IP installed and /or configured incorrectly	Reinstall or reconfigure TCP/IP <sup>3</sup>

1 See the appropriate NCS documentation.

2 Described in this manual.

3 See the appropriate TCP/IP documentation.

## Glossary

---

<b>ADK</b>	The component of <i>i</i> FOR/LS that is used by software developers to define and create licenses for software products.
<b>annotation</b>	See license annotation.
<b>ARK</b>	Administrator's Runtime Kit. The run-time environment for licensed software products. The Administrator's Runtime Kit is available in several binary representations. It consists of the network license management daemon ( <i>i4lmd</i> ) and its associated tools: <i>i4admin</i> , <i>i4report</i> , and <i>i4stat</i> .
<b>Client Ally</b>	An ARK component ( <i>i4ally</i> ) that facilitates access to a license server on behalf of personal computer-based client applications.
<b>compound password</b>	A type of product password that specifies parameters for creating license passwords. In an end-user environment, a compound password can specify multiple nodelocked licenses. The compound password is installed at a license server, and the license server then derives license passwords (each of which specifies a single nodelocked license) from the compound password. It then automatically installs them at user nodes at which the product so licensed is invoked.
<b>concurrent access license</b>	A type of license administered by the license server that can be used by different users at any node that is connected to a license server node. Concurrent access licenses allow as many users to use a software product concurrently as there are licenses.
<b>CPU lock</b>	See nodelocked license.

<code>i4admin</code>	The software program used to modify a license server database, invoked with the command <code>i4admin</code> , which is located in the <code>/opt/ifor/ls/bin</code> directory.
<code>i4lmd</code>	The string used to invoke the network license management daemon.
<code>i4report</code>	The software program that reports on the history of license server events, invoked with the command <code>i4report</code> , which is located in the <code>/opt/ifor/ls/bin</code> directory.
<code>i4stat</code>	The software program that reports on the status of licenses, invoked with the command <code>i4stat</code> , located in the <code>/opt/ifor/ls/bin</code> directory.
<code>i4tv</code>	The network license management daemon test and verification tool, invoked with the command <code>i4tv</code> , located in the <code>/opt/ifor/ls/bin</code> directory.
<b>iFOR/LS</b>	A term that refers collectively to the ARK and the ADK components of the <i>i</i> FOR Licensing Suite. Also known as <i>i</i> FOR/LS.
<b>iFOR/LS Test Product</b>	The product used by the <code>i4tv</code> tool to verify that license servers are working properly.
<b>license</b>	An instance of permission to use a licensed software product or service. Sometimes, a user needs more than one license to use a product.
<b>license annotation</b>	A string that modifies the use of a license in a manner defined by the vendor of the software product.
<b>license database</b>	The database of licenses maintained by a license server. The license database file - <code>lic_db</code> - resides in the <code>/var/opt/ifor</code> directory.
<b>license information</b>	The information that describes licenses. This information consists of the product name, the product version, the number of licenses, the license type, the start and end dates of the licenses, the target type, the target ID, and a time stamp.
<b>license management daemon</b>	A software program that administers licenses for software products, invoked with the command



	i4lmd. The i4lmd command can be found in the /opt/ifor/ls/bin directory.
<b>license password</b>	A string encoded with license information for a software product.
<b>licensed product</b>	A software product that has been licensed under iFOR/LS.
<b>log file</b>	The text file — log_file — that records messages and errors from the license server, and sometimes from licensed products as well, resides in the /var/opt/ifor directory.
<b>network license server</b>	See license management daemon.
<b>nodelocked license</b>	A type of license locked to a specific node so that the product may only be used at that node. The license server does not administer nodelocked licenses.
<b>nodelock file</b>	The text file at a user node (rather than at a license server node) where nodelocked licenses are added. The file nodelock is located in the /var/opt/ifor directory.
<b>password</b>	A string encoded with information about a software vendor (vendor password) or about a software product (product password).
<b>product ID</b>	An integer that identifies a vendor's licensed software product; by means of product IDs, the license server distinguishes among products of the same vendor.
<b>product password</b>	A string encoded with information about licenses for a software product. Product passwords are of two types: license passwords and compound passwords.
<b>revision text</b>	See version identifier.
<b>site license</b>	An authorization to use a software product on all computers at a site. In iFOR/LS, the effect of a site license is achieved by installing nodelocked licenses on all nodes, or by installing concurrent-access licenses equal to the number of concurrent users (or nodes) at the site.

<b>target</b>	The node at which a password is to be installed. If the password specifies a single nodelocked license, the target is the node licensed to run the product. If the password specifies multiple nodelocked licenses (that is, a compound password for nodelocked licenses), or licenses of any other type, then the target is a node running the license management daemon.
<b>time stamp</b>	An integer that describes the date and time at which a set of licenses was created.
<b>use-once license</b>	A type of license administered by the license server that can be used for a single instance of invoking a product or of using a service. The license server decrements the number of use-once licenses each time the product is used.
<b>user file</b>	A text file that specifies the users who may (or may not) use licensed software products.
<b>vendor ID</b>	The identifier of a vendor of licensed products. By means of vendor IDs, license servers can distinguish among any number of vendors established in a network. Vendor IDs are an <i>i</i> FOR/LS - specific usage of NCS Universal Unique Identifiers (UUIDs).
<b>vendor password</b>	A string encoded with information about a vendor that, together with a vendor ID, establishes the vendor of a licensed product in a license database.
<b>version identifier</b>	A string that identifies a version of a product; by means of version identifiers, the license server distinguishes among different versions of a product.

# Index

---

## A

adding passwords to license servers,  
3-7

Add product example, 3-10

Add vendor example, 3-9

administrative tasks, summary of,  
2-8

administrative tools, ARK, 1-2, 2-3,  
2-9

all keyword, 3-6, B-2

allow keyword, B-2

annotations

license, 3-14

annotations, licenses, 2-7

ARK

administrative tools, 2-9

Global Location Broker (GLB) and,  
2-9

NCK and, 3-1

prerequisites, 3-1

problems, C-1

TCP/IP and, 3-1

troubleshooting, C-1

walkthrough, 2-4

what you get, 2-8

ARK commands, A-1

## C

client ally

starting, 3-4

commands

i4admin, command-line interface,  
A-5

i4admin graphic interface, A-2

i4lmd command-line interface,  
A-17

i4report, command-line interface,  
A-9

i4stat, command-line interface,  
A-15

i4stat, graphic interface, A-12

compound passwords, 3-12

concurrent-access licenses, 3-3

concurrent-access licensing, 2-6

conventions, documentation, 1-2

Copy vendor example, 3-11

crawl keyword, 3-5, B-2

## D

daemon

DCE Remote Procedure Call  
Daemon (rpcd), 3-2

license server, 3-3

llbd- Local Location Broker  
Daemon, 3-2

database

license, 3-3

disallow keyword, 3-6, B-2

documentation conventions, 1-2

dynamic nodelocking, 2-6, 3-13

## **E**

error messages, C-2  
Exit Button, i4stat, 4-2

## **F**

files  
    nodelock, 3-13

## **G**

GLB and ARK, 2-9  
Global Location Broker (GLB), 2-9

## **H**

hybrid licenses, 2-7

## **I**

i4admin, 2-9  
    Add product, 3-10  
    Add vendor use of, 3-9  
    checking user file format, 3-6  
    command-line interface, A-5  
    Copy vendor button, 3-11  
    Describe button, 3-8  
    graphic interface, A-2  
    Operations startup display  
        example, 3-9  
    Sample Session, 3-8  
i4lmd, 3-3  
    command-line interface, A-17  
    directory, 3-3  
    starting, 3-3  
i4report  
    command-line interface, A-9  
    license server reports, 4-7  
i4stat, 2-9  
    All Products (Update) Button,  
        4-3  
    All Servers (Update) Button,  
        4-3, A-13  
    All Users Button, A-12  
    command-line interface, A-15

Exit Button, 4-2, A-12  
    getting status information, 4-1  
    graphical interface, 4-1, A-12  
    help index, 4-2  
    Installed Button, A-12  
    License Information Menu, 4-2,  
        A-12  
    Menus And Buttons, A-12  
    Product List Box, 4-3  
    sample session, 4-3  
    Server List Box, 4-2, A-13  
    Status Message Field, 4-3  
    Usage Button, A-12  
    User Button, A-12

i4target  
    node ID using, 3-3  
i4tv, 3-4  
    error messages, C-2  
    network license server test and  
        verification tool, C-1  
installing passwords, 3-14  
installing passwords for nodelocked  
    licenses, 3-13

## **K**

keywords  
    all, 3-6, B-2  
    allow, B-2  
    case, B-2  
    crawl, 3-5, B-2  
    disallow, 3-6, B-2  
    -P1, 3-5, B-2  
    -P2, 3-5, B-2  
    vendor, B-2

## **L**

license annotations, 3-14  
license database, 3-3  
licensed servers  
    installing passwords at, 3-14  
licenses

## **Index-2**

- annotations, 2-7
  - concurrent-access, 3-3
  - enforcement, 2-7
  - hybrid, 2-7
  - nodelocked, 2-6, 3-12
- license server
  - starting, 3-3
- license server daemon, 3-3
- license server reports using `i4report`, 4-7
- license servers
  - adding passwords, 3-7
  - verifying, 3-4
- licensing
  - concurrent-access, 2-6
  - current methods, 2-1
  - site, 2-1
  - use-once, 2-6
- 11bd- Local Location Broker Daemon, 3-2
- Local Location Broker Daemon (11bd), 3-2

**M**

- maintaining user files, 3-7
- monitoring ARK, 4-1

**N**

- NCK, Network Computing Kernel, 3-2
- Network Computing Kernel (NCK), 3-2
- Network Computing Kernel (NCK) and ARK, 3-1, 3-2
- node ID using `i4target`, 3-3
- nodelocked licenses, 2-6, 3-12
- nodelock file, 3-13
- nodelocking
  - dynamic, 2-6, 3-13
- nodelock sample file, 3-14
- nodes
  - choosing to run license server, 3-3

**P**

- P1 keyword, 3-5, B-2
- P2 keyword, 3-5, B-2
- passwords
  - adding to license servers, 3-7
  - compound, 3-12
  - installing, 3-14
  - installing for nodelocked licenses, 3-13
  - single-license, 3-12
  - specifying nodelocked licenses, 3-12
  - types of, 3-12
  - vendor, 3-14
- problems, 3-15
- problems with ARK, C-1

**S**

- sample nodelock file, 3-14
- Server List, `i4stat`, 4-2
- single-license passwords, 3-12
- starting client ally, 3-4
- starting license server, 3-3
- status information, getting, 4-3
- status information using `i4stat`, 4-1

**T**

- TCP/IP, 3-1
- test and verification tool, `i4tv`, C-1
- troubleshooting, 3-15
- troubleshooting ARK, C-1

**U**

- Use-Once licensing, 2-6
- user file format, B-1
- user files, 3-4
  - maintaining, 3-7
  - writing and integrating, 3-6
- user names

personal computers, 3-7

## **V**

vendor ID, 3-7

vendor keyword, B-2

vendor passwords, 3-14

verification and test tool, i4tv, C-1

verifying license servers, 3-4

## **W**

writing and integrating user files,  
3-6