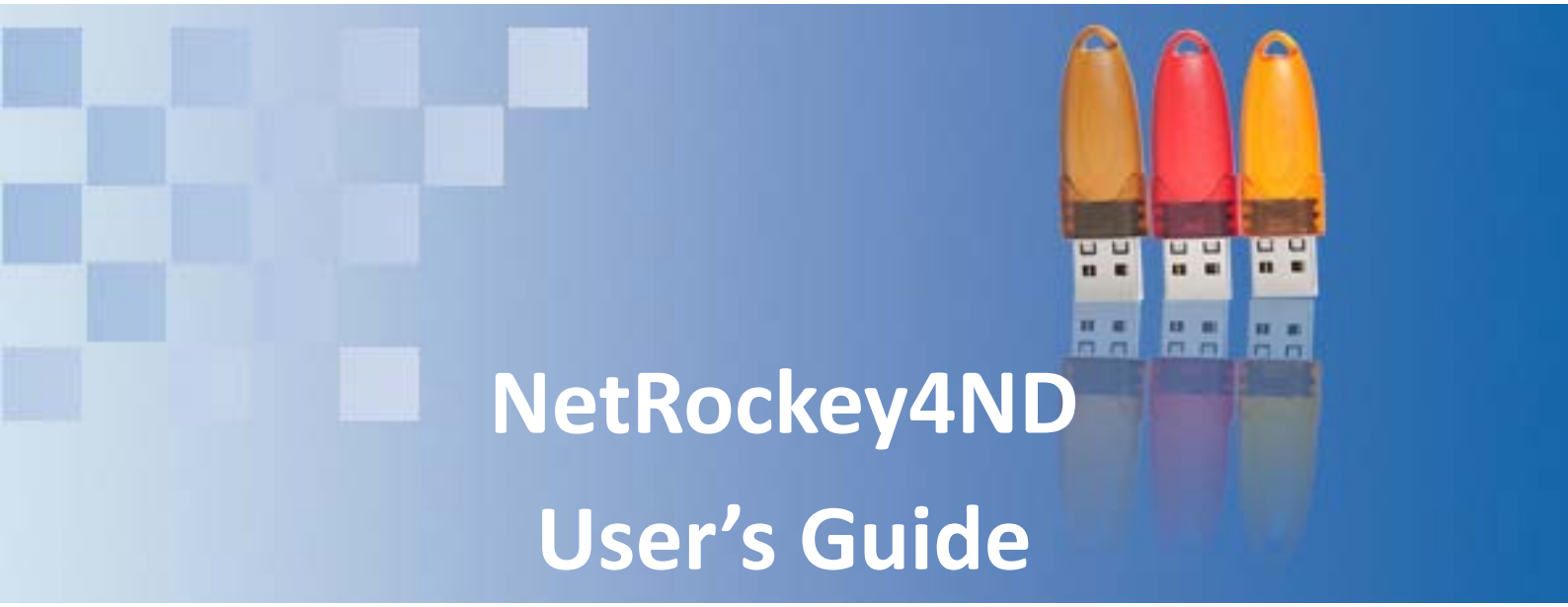
FEITIAN



NetRockey4ND User's Guide

V1.1

Feitian Technologies Co., Ltd.

Website: www.FTsafe.com

Revision History:

Date	Revision	Description

Software Developer's Agreement

All Products of Feitian Technologies Co., Ltd. (Feitian) including, but not limited to, evaluation copies, diskettes, CD-ROMs, hardware and documentation, and all future orders, are subject to the terms of this Agreement. If you do not agree with the terms herein, please return the evaluation package to us, postage and insurance prepaid, within seven days of their receipt, and we will reimburse you the cost of the Product, less freight and reasonable handling charges.

Allowable Use – You may merge and link the Software with other programs for the sole purpose of protecting those programs in accordance with the usage described in the Developer's Guide. You may make archival copies of the Software.

Prohibited Use – The Software or hardware or any other part of the Product may not be copied, reengineered, disassembled, decompiled, revised, enhanced or otherwise modified, except as specifically allowed in item 1. You may not reverse engineer the Software or any part of the product or attempt to discover the Software's source code. You may not use the magnetic or optical media included with the Product for the purposes of transferring or storing data that was not either an original part of the Product, or a Feitian provided enhancement or upgrade to the Product.

Warranty – Feitian warrants that the hardware and Software storage media are substantially free from significant defects of workmanship or materials for a time period of twelve (12) months from the date of delivery of the Product to you.

Breach of Warranty – In the event of breach of this warranty, Feitian's sole obligation is to replace or repair, at the discretion of Feitian, any Product free of charge. Any replaced Product becomes the property of Feitian.

Warranty claims must be made in writing to Feitian during the warranty period and within fourteen (14) days after the observation of the defect. All warranty claims must be accompanied by evidence of the defect that is deemed satisfactory by Feitian. Any Products that you return to Feitian, or a Feitian authorized distributor, must be sent with freight and insurance prepaid.

EXCEPT AS STATED ABOVE, THERE IS NO OTHER WARRANTY OR REPRESENTATION OF THE PRODUCT, EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Limitation of Feitian's Liability – Feitian's entire liability to you or any other party for any cause whatsoever, whether in contract or in tort, including negligence, shall not exceed the price you paid for the unit of the Product that caused the damages or are the subject of, or indirectly related to the cause of action. In no event shall Feitian be liable for any damages caused by your failure to meet your obligations, nor for any loss of data, profit or savings, or any other consequential and incidental damages, even if Feitian has been advised of the possibility

of damages, or for any claim by you based on any third-party claim.

Termination – This Agreement shall terminate if you fail to comply with the terms herein. Items 2, 3, 4 and 5 shall survive any termination of this Agreement.

Contents

Chapter 1. Introduction 1

1.1 Basic Concepts 1
1.2 NetROCKEY4ND Developer's Kit 3
1.3 Configuration Files and Tool 4

Chapter 2. NetRockey4ND Tools 7

2.1 Configuration File Editor 7
2.2 NetROCKEY4ND Service Program 9
2.3 NetROCKEY4ND Monitor 10
2.4 Client Test Utility 13

Chapter 3. NetROCKEY4ND APIs 18

3.1 Function Prototype and Definition 18
3.2 APIs 20

Chapter 4. NetROCKEY4ND Error Codes 28

4.1 Normal error codes: 28
4.2 Extended Error Codes for UDP/TCP and IPX 29
4.3 Extended Error Codes for NetBios 31

Chapter 5. Quick Test 33

Chapter 1. Introduction

NetROCKEY4ND is a network aware software protection system designed to limit the number of simultaneous users who can access a software application. It combines all the functionality of the standard ROCKEY4ND system, with the ability to work seamlessly in LAN/WAN environments that support the UDP/TCP, IPX or NetBIOS protocols.

NetROCKEY4ND was engineered to support stand alone or redundant server environments on both the Windows and Linux platforms. The system includes powerful and intuitive network monitoring and testing tools that ease the implementation effort. NetRockey4ND can be used to limit the number of software instances running over a network.

1.1 Basic Concepts

1.1.1 Configuration Files

There is a configuration file for the Service program (SvrCfg.ini), and one for the Client program (CliCfg.ini). The service and client programs take their network settings from the configuration files. The developer may use a text editor or tools provided by Feitian to edit such configuration file parameters as: protocol type, time-to-live, server address and other information required for the dongle to attach to the network.

1.1.2 Log Files

The NetROCKEY4ND log file (svrlog.txt) records the running status of the service program. It can be helpful if you encounter a problem with the service program. The path and name of the log file may be configured in the SvrCfg.ini file.

1.1.3 Ports and Group

The UDP/TCP and IPX protocols require the specification of a port number. Port numbers range from 0 to 65535. The default port number for NetROCKEY4ND is 3152. 3152 is registered with IASA and should be available on most networks. If it is occupied though, the NetROCKEY4ND service program will report a "bind" error. If a bind error occurs you may want to move NetROCKEY4ND to an available port; the port number can be changed in the SvrCfg.ini file. The NetBIOS protocol does not use a port number. It uses a group name. The group name is a character string that may be a maximum of 16 characters in length. Each server in a NetBIOS network has both a computer name and a group name. The group name for the service program may be altered in the SvrCfg.ini file

and the group name for the clients in the CliCfg.ini file. All clients and servers that need to communicate in a NetBIOS network need to have the same group name.

Note: It is not recommended to start network dongle services of different kinds on the same machine. If you must do that, change the port numbers for the services to make them different. But for the same network dongle service, the server and the client should share the same port number.

1.1.4 Network Address

Each computer in network has a unique address. A UDP/TCP network (IPv4) uses IP addresses that may look like: 192.168.0.1. An IPX network uses a MAC address that may look like: 00-35-4f-20-00-32. A NetBIOS network uses 14 letters.

1.1.5 Search Mode

The NetROCKEY4ND client program will search for the address of the service program at start-up. The client program will look to the CliCfg.ini file for the search "mode". There are three search modes that may be set in the client configuration file: Automatic, Custom and Semi-automatic. Automatic mode means that the client will broadcast to locate the service program. Custom mode requires that you enter a search list of the service addresses in the CliCfg.ini file. The client program will not issue a broadcast message but will use the search list to find the service programs. Automatic mode has the advantage of being easy to configure, but the drawbacks of slow response and added network overhead. Custom mode is faster than automatic but requires that you know the addresses of the service programs. Semi-automatic mode attempts to overcome the drawbacks of both the custom and automatic modes. In semi-automatic mode, the client will first go to its search list. If it finds one or more service programs it will quit searching. If it does not find a service program, it will broadcast to find the service program.

1.1.6 Opening Mode

The NetROCKEY4ND client programs issue an "open" command to the service program. This open command is equivalent to a network login and it is the means by which the service program limits the number of users that can attach to the application. There are two operating modes for the open command: private and share. The default setting is private mode. In private mode operation the service program adds "1" each time a user attaches to the application. If the calculated quantity reaches the maximum set by the developer, the open command will fail, the service program will issue an error message and the user will not be allowed to access the application. In share mode, all programs in the same computer share one user number. No matter how many times the computer accesses the service program, it is considered to be one user. Share mode is appropriate if the number of computers, rather than the number of users that attach to an application, need to be limited. The open mode is

set with the lp2 parameter in the open operation (See the ROCKEY4 API section for an explanation of the open operation.) The low byte of the lp2 parameter sets the NetROCKEY4ND module number that will store the maximum number of simultaneous users (see item 8 below), and the high byte sets the open mode.

1.1.7 Timeout

Each time the client sends data to the server it will wait a time period defined by the "time out" parameter. If the client does not receive a response after the time out period, it will quit and return an error code. The unit of time for the time out parameter is seconds and the default is two. In automatic search mode, the time out is also the period that the client program will wait for a response to its broadcast message. The time out parameter can be changed in the CliCfg.ini file.

1.1.8 Maximum Number of Simultaneous Users

The maximum number of simultaneous users that will be allowed to access an application will be set by a value stored in a NetROCKEY4ND module. For example, if you write "5" to module 0, up to 5 users can log into module 0 simultaneously. The writing operation can be completed using Rockey4ND Editor.

1.1.9 Client Time to Live (TTL)

This parameter can be set in the configuration file.The client program automatically sends a "blank" message to the service program every 1.5 minutes. If the service program does not receive an idle message from a client during the TTL period, it will delete the client handle, terminating the connection. This parameter is useful in the event the client is shutdown abnormally, the user forgets to close the handle, or the network connection is lost, to prevent logon by 2 or more users.

1.1.10 Limitation on Opened Module

A module can only be opened once in a single process with security in mind. You may set the handle as a global variable to use it in every thread.

1.2 NetROCKEY4ND Developer's Kit

The developing tools for NetROCKEY in directory "Net":

<Client> DLLs and configuration files for the NetROCKEY4ND client program

<Server> Executable and configuration file for the NetROCKEY4ND service program.

<Tools> Developing tools for NetROCKEY4ND

<Samples> Sample program files

1.3 Configuration Files and Tool

The configuration files and tool is located under Net\Tools\NrConfig directory. They are:

NrConfig.exe - Configuration files editor. Use to edit configuration files for service program and client program.

SvrCfg.ini - Configuration file for the NetROCKEY4ND service program. NrClient.dll API library for the NetROCKEY4ND client program (You should not change the name of this file).

CliCfg.ini - Configuration file for the NetROCKEY4ND client program.

There is a configuration file for the Service program (SvrCfg.ini), and one for the Client program (CliCfg.ini). The configuration files will configure the settings of the network (All the characters in the configuration files are case sensitive). Below is the template of Client Program Configuration File (CliCfg.ini):

```
[Header]

Sign=RockeyClientHeader
        ;Indicates the header of the server configuration file
 [Common]
Timeout=2
        ;Timeout setting, for waiting for response under all protocols (in seconds) SearchFlag=0
        ;Search flag, 0 for auto, 1 for manual, 2 for semi-auto
        ;for manual search, the list of searched servers is required, see SearhList below
ProtoFlag=0
        ;Whether to choose the speediest protocol or not: 1 for yes, 0 for no
        ; (it will take some time to detect initially)
 [TCPUDP]
bUsedTCP=1
bUsedUDP=1
        ;Whether to use TCP/UDP protocol or not, 1 for yes, 0 for no
TCPPort=3152
        ;TCP port, which can be changed, but it must be consistent with the port used by the server

UDPPort=3152
        ;UDP port, which can be changed, but it must be consistent with the port used by the server

SearchList=127.0.0.1
        ;Search sequence for manual search, it includes the IP addresses of servers,
        ;which are separated by ','(machine names can also be used)

[IPX]
```

```
bUsed=0
        ;Whether to use IPX protocol or not, 1 for yes, 0 for no

IPXPort=3152
        ;IPX port, must be consistent with the port used by the server

SearchList=00-A0-0C-13-0E-D2,00-00-B4-B2-ED-7B
        ; Search sequence for manual search, it includes the IP addresses of server network cards, which are separated by ','
        ;the network card addresses can be obtained using Windows tool nbtstat –a machine name.

[NetBios]

bUsed=0
        ;Whether to use NetBios protocol or not, 1 for yes, 0 for no

RegGrpName=FTNetServer
        ;The name of the server group, must be consistent with the settings of the server

SearchList=FTNetServer001,FTNetServer002,FTNetServer003
;Search sequence for manual search, it includes service names separated by ','
        ;The service names are the registered names in the server configuration file.
```

Below is the template of the server configuration file:

```
[Header]
Sign=RockeySvrHeader
        ;Indicates the server configuration file, do not change it
 [common]

Timeout=2
        ; Timeout setting, for waiting for response under all protocols (in seconds)

IdleTime=3
        ;If client idle time exceeds this upper limit, the server will close
        ;this customer automatically. This is used in the event the client forgets ;to execute Close command. Note: Do not specify a value lower than 2 (in min)

LogFile=svrlog.txt
        ;Name of log file, recording server output, it may include path information

[TCPUDP]

bUsed=1
```

```
        ;Whether to start the TCP/UDP service or not, 1 for yes, 0 for no

TCPPort=3152
        ;TCP port, a different port can be attempted if it has been used,
        ;must be consistent with the setting of the client

UDPPort=3152
        ;UDP port, a different port can be attempted if it has been used,
        ;must be consistent with the setting of the client

[IPX]

bUsed=0
        ;Whether to start the IPX service, 1 for yes, 0 for no

IPXPort=3152
        ;IPX port, a different port can be attempted if it has been used,
        ;must be consistent with the setting of the client

[NetBios]
bUsed=0
        ;whether to start the NetBios service, 1 for yes, 0 for no

RegName=FTNetServer
        ;Whether to register a server name or not, the first default value is RegName001
        ;Or, you can specify a name yourself, if the name is not unique,
        ;postfix it with 002,003…

RegGrpName=FTNetServer

            ;Name of server group, it can be replaced with the name of your company,

            ;but you must make sure that this entry is the same for all server and

            ;client configuration files
```

Chapter 2. NetRockey4ND Tools

2.1 Configuration File Editor

The Configuration File Editor is a graphical program that may be used to edit SvrCfg.ini and CliCfg.ini. The screen pictured in Figure 2.1 will appear if neither SvrCfg.ini nor CliCfg.ini is found in the current directory. Click on either or both of the check boxes to create the configuration file(s) in the current directory with default settings.

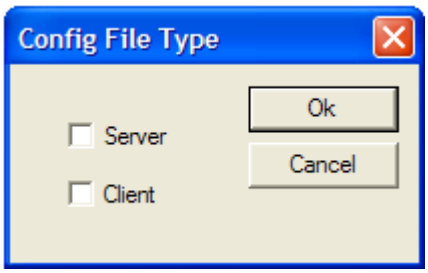


Figure 2.1

The editor may edit SvrCfg.ini and CliCfg.ini files in the current directory, the screen is pictured below:

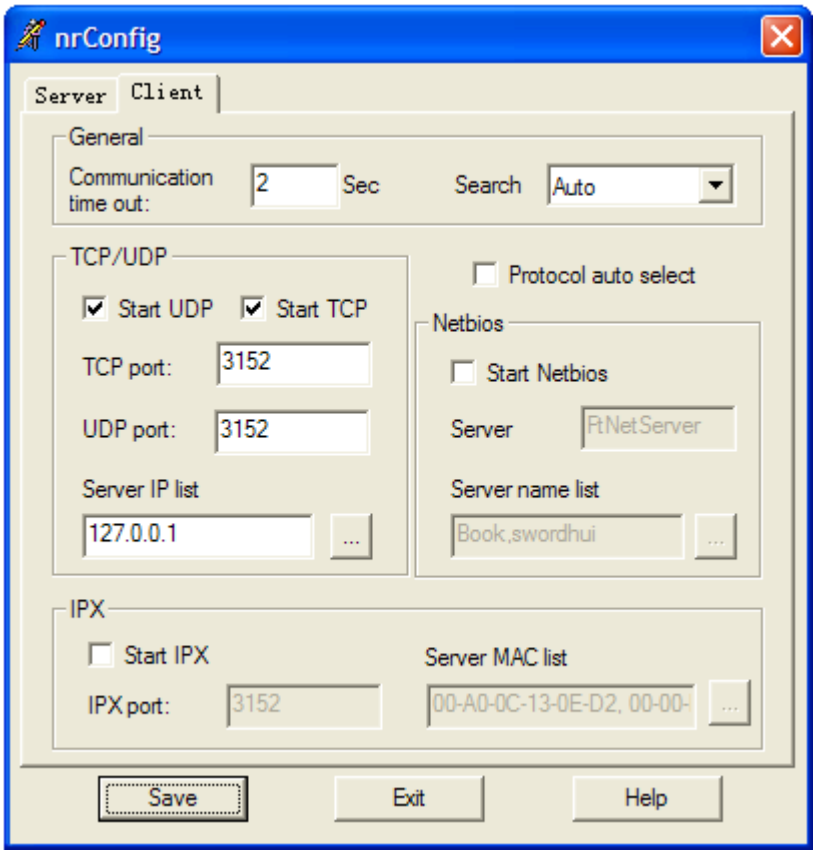


Figure 2.2

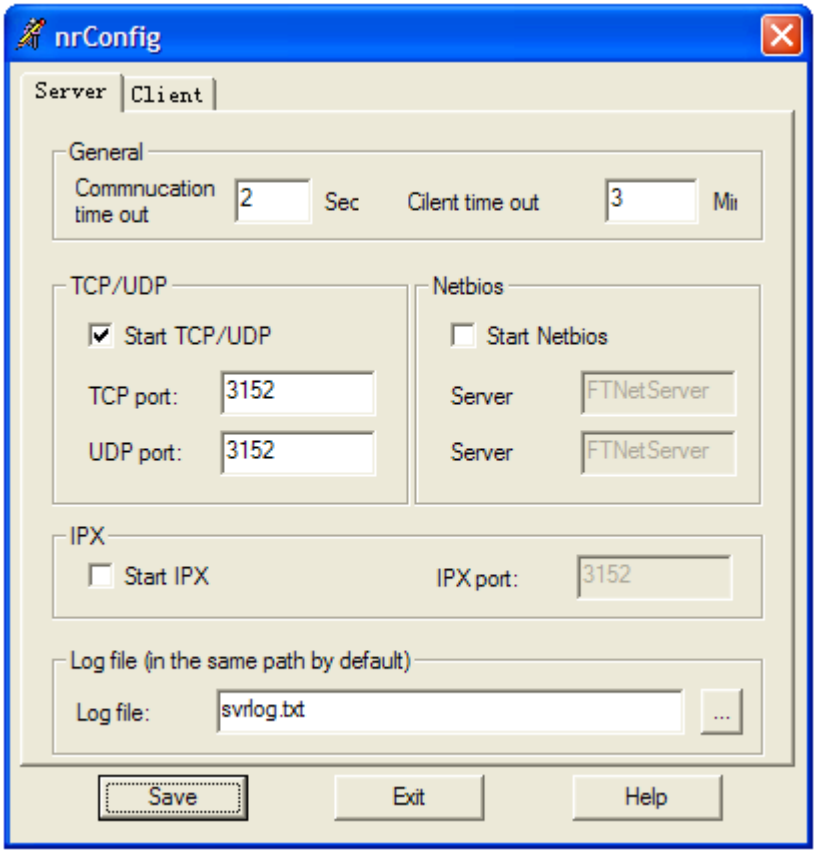


Figure 2.3

Hold the mouse pointer on a particular field for a couple of seconds for a helpful tip. An Editor screen with a "tip" caption is shown in Figure 2.4:

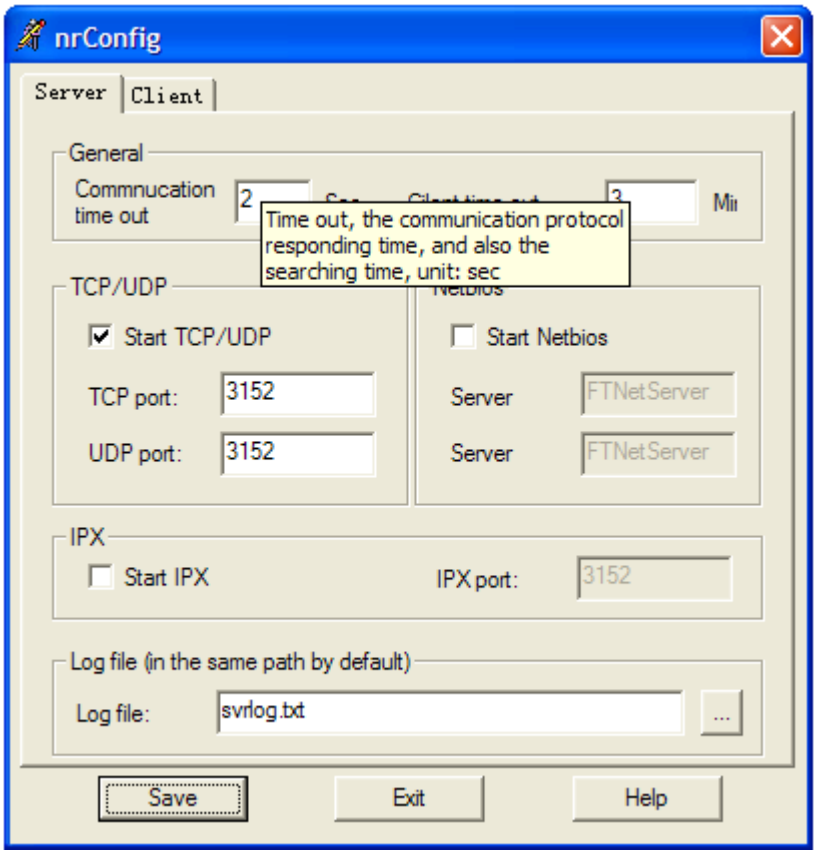


Figure 2.4

Note: If only SvrCfg.ini or CliCfg.ini is in the current directory, the Configuration file Editor will allow you to edit the file that it finds. You can also extract the other .ini files by clicking on the icon in the upper left portion of the screen. A pop-up menu will then appear that will allow you extract the file you need.

2.2 NetROCKEY4ND Service Program

The service program (nrSvr.exe) is under directory Net\Server. The NetROCKEY4ND dongle is detected by the client program only after the service program is started on the computer to which the NetROCKEY4ND dongle is attached.

After the service program is run for the first time it will automatically register itself as the service program, it will run automatically every time you start your computer, unless you uninstall it. After it is started the service program will look in the current directory for the service configuration file (SvrCfg.ini) and take the configuration information, if it can not find the configuration file it will use the default configuration. The status of the Service program will be recorded in a log file specified in the SvrCfg.ini file. See Figure 2.4 above. When the service program is started, an icon will appear in the system tray. See Figure 2.5 (The service program icon is the one furthest to the left of the tray.)

Figure 2.5

Double click or right click the Service program icon to open the screen pictured in Figure 2.6.

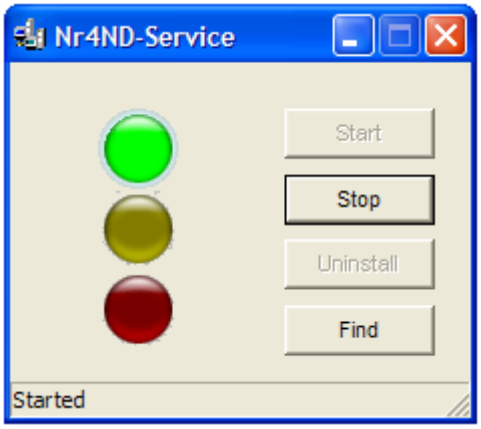


Figure 2.6

The Service program screen may be used to stop, start or uninstall the Service program. Right click the Service program icon to open a menu to start, stop, uninstall or exit the program.

Note: The Service program does require a driver and a NetROCKEY4ND, however, the client program does not require any driver and NetROCKEY4ND.

2.3 NetROCKEY4ND Monitor

The monitor program (NrMon.exe) is under the directory Net\Tools\Monitor. The monitor will function on any LAN attached PC; it does not require a ROCKEY4 driver and Dll. It was designed to monitor the activities of all NetROCKEY4ND devices on the network, if it is installed on the PC running the Service program, it can also be used to start and stop the Service functions or kill a client. If SvrCfg.ini or CliCfg.ini is under the executing directory of NrMon.exe, NrMon.exe will automatically take the port information of these files to connect the network.

The Monitor program will first search for all network attached service programs and clients, as shown in Figure 2.7. You can specify the protocol for the search operation. Select the search protocol from the "Setting" pull-down menu or from the tool bar. Click the button to invoke the application.

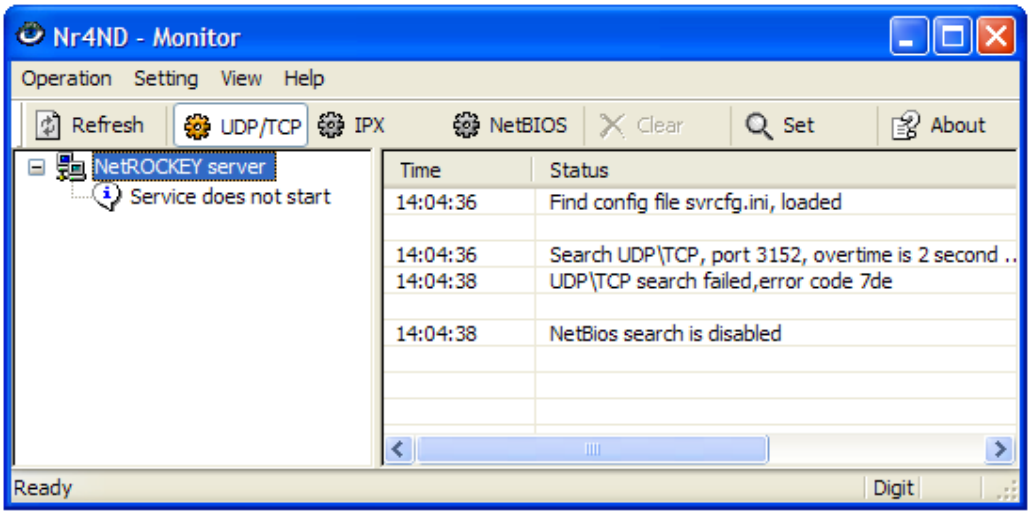


Figure 2.7

Return code will appear if there is anything wrong with any protocol, and you may refer to section Return Codes to find out the reason.

The search results will be displayed below:

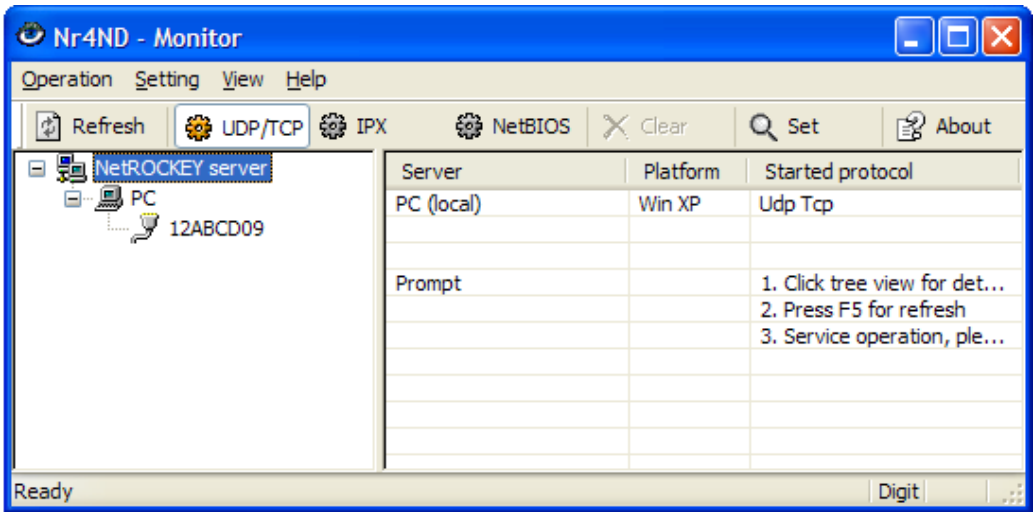


Figure 2.8

The server names appear on the left portion of the window. The NetROCKEY4ND hardware ID (HID) will appear if users are logged into the server. Server information, including server platforms and opened protocols, appears in the right portion of the screen. If the Monitor program is installed on the same computer as the service program, the word "Local" will appear next to the computer name, and you may control the service via monitor, including starting and stopping the Service functions or killing a client. You may invoke these operations from operation menu or tool bar.

In Figure 2.9, the user has selected the HID of a server.

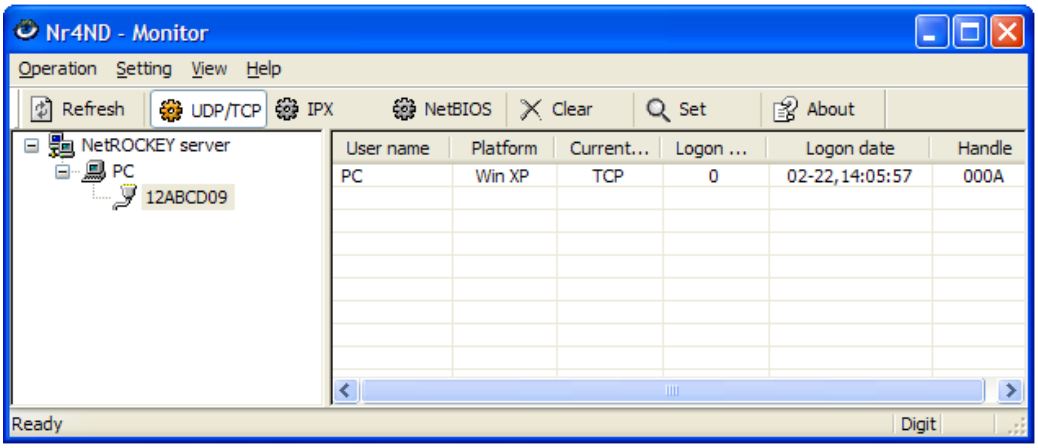


Figure 2.9

Client information, including computer name, platform, opened protocol, log-in module, log-in time, and handle, appears in the right portion of the screen. If a user logs into the local server you can delete the connection by clicking on the client and then pressing the "Clear" key, or clicking the "Kill" button on the toolbar.

Pressing F5 key or choosing the "Refresh" button on toolbar can refresh the current screen. Auto-refresh mode may be activated from the toolbar or pull-down menu. Please see the auto-refresh screen pictured below:

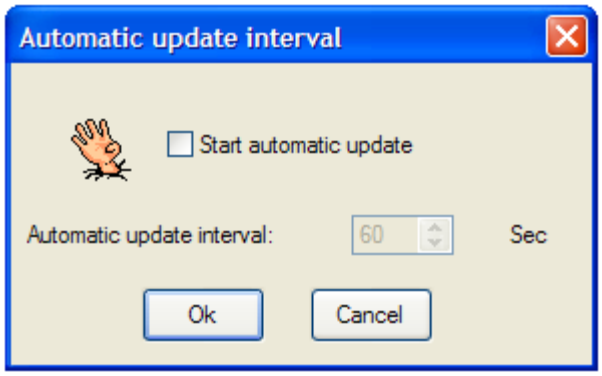


Figure 1.10

Simply click the "Auto Refresh" option, enter a time interval (seconds) and then click "OK". Press Refresh button or Auto Refresh only refreshed the current screen. Click root, server or NetROCKEY in the left part of the screen to display and refresh the corresponding information.

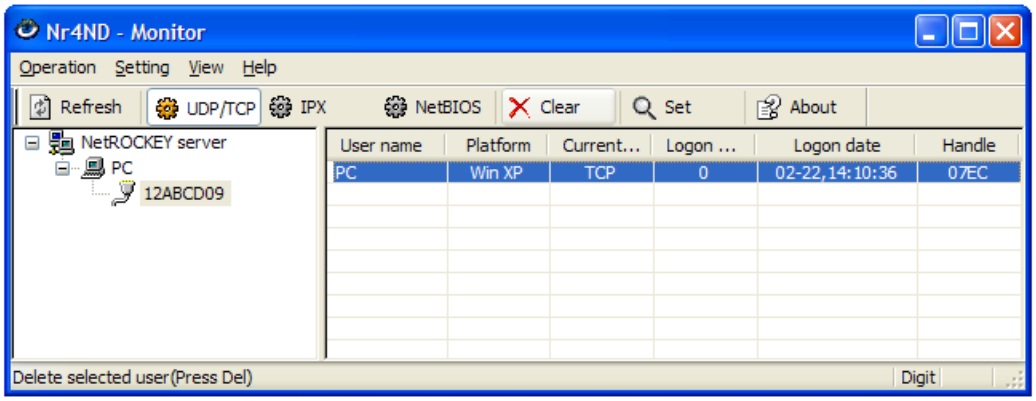


Figure 2.11

You can stop or start specific protocols from the "Operation" pull down menu. Protocols can only be stopped or started from the Monitor program if it is running on the same machine as the Service program.

Choose the clients logged onto local computer, move the mouse pointer to Clear button, the button will get highlighted. Clicking the "Clear" button on the toolbar or pressing the "Del" key will remove these clients by force. See Figure 2.11.

2.4 Client Test Utility

The client test utility (NrTest.exe) is under the directory Net\Tools\NrTest. it was designed to test the functions of NetROCKEY4ND. NrTest.exe may take the client configuration file in current directory to test NetROCKEY4ND system. It requires both NrClient.dll and CliCfg.ini in the current directory.

To use the utility, the NetROCKEY4ND passwords are required. See Figure 2.12 below.

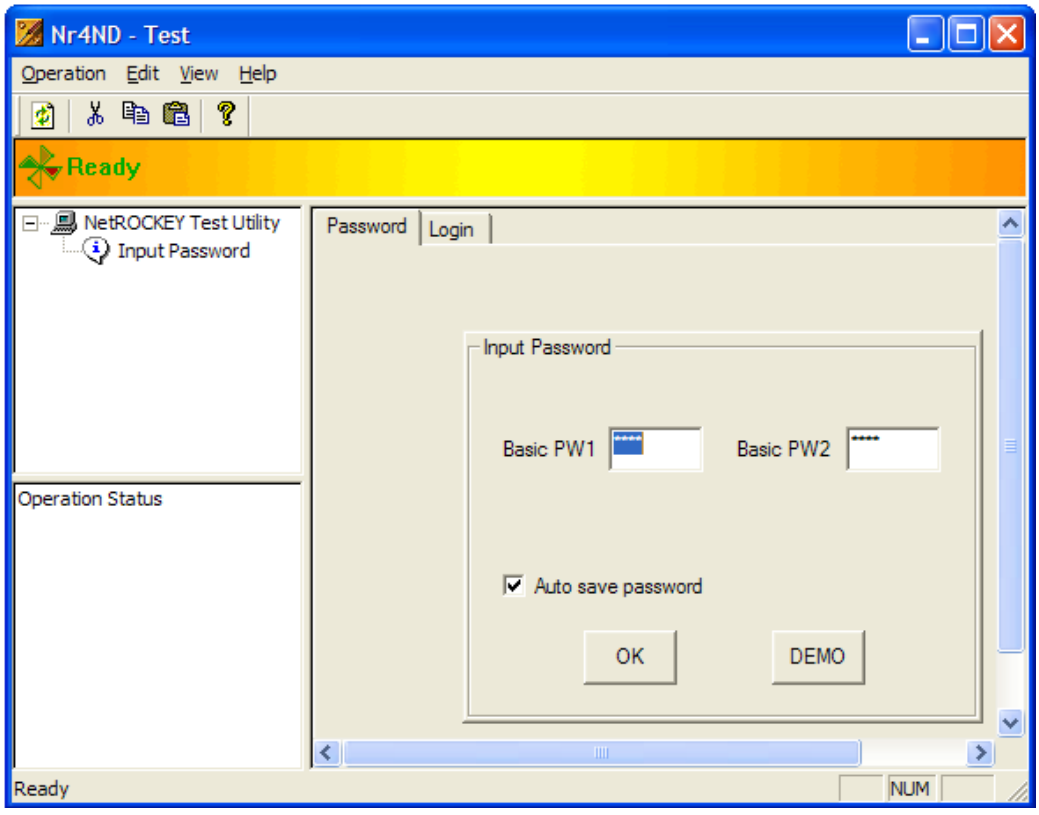


Figure 2.12

The Basic passwords alone allow full functionality of the Test program, keeping security in mind NetROCKEY4ND server does not accept Write operation with advanced passwords. If you need to change the content of NetROCKEY4ND, you must edit the content of it with the Editor of ROCKEY Standalone. The "Auto save password" option will save the entered passwords in encrypted form in the system registry. This feature is handy if you do

not want to reenter the passwords each time you work with the Test utility. If you are working with a Demo NetROCKEY4ND dongle, click the "DEMO" button. No password entry is required. The next step is to search for NetROCKEY4ND network clients. This action requires the NrClient.dll and CliCfg.ini files. The Test utility will show error information if NrClient.dll is not found. See Figure 2.13.

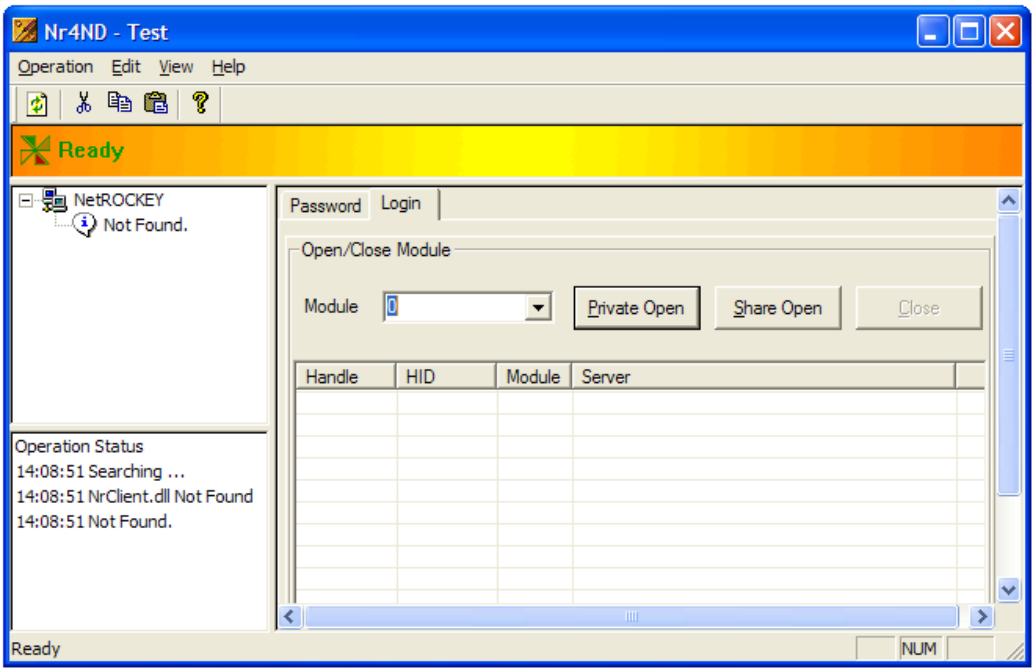


Figure 2.13

If this problem occurs, copy the NrClient.dll and CliCfg.ini files to current directory. Test utility will automatically take the network configuration information in file CliCfg.ini, such as protocol and ports. And then search with the function in NrClient.dll, if CliCfg.ini is not found, the system will use the default configuration.

The searching screen is pictured below:

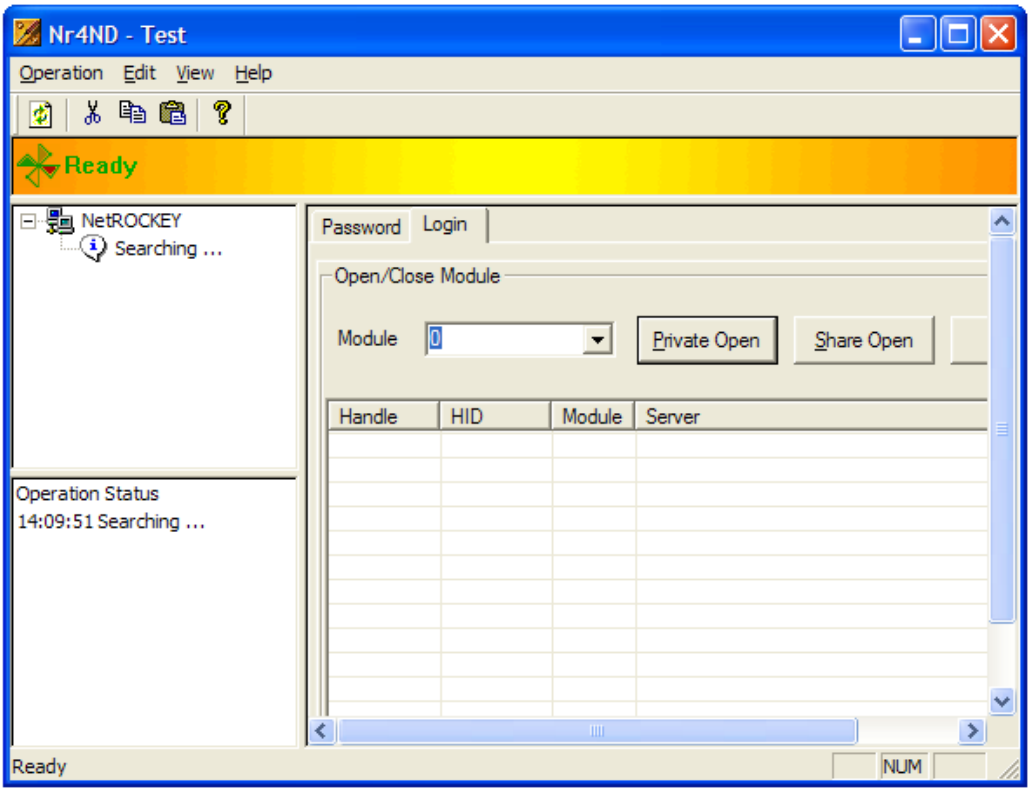


Figure 2.14

The results of a successful search will look like Figure 2.15, all servers and NetROCKEY4ND dongles are displayed here.

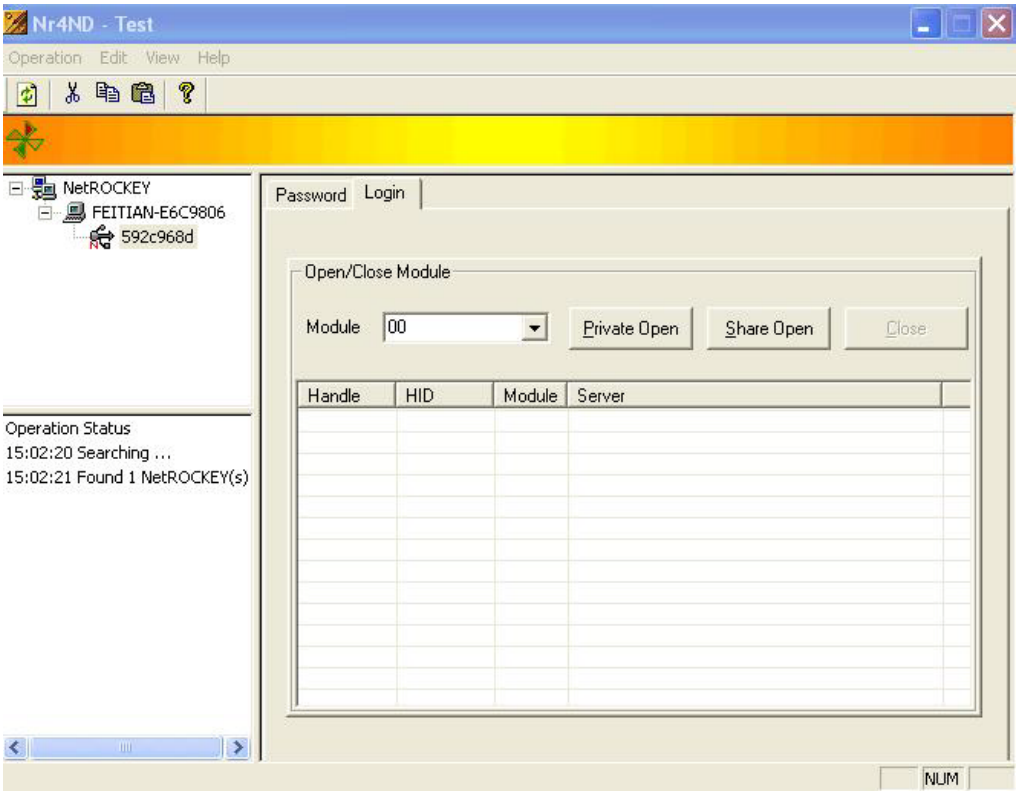


Figure 2.15

Select a NetROCKEY4ND dongle and the log-in screen will appear, choose the module you would like to open, and then click on either the "Private Open" or "Share Open" to open it and you will get a return handle for future operations. See Figure 2.16.

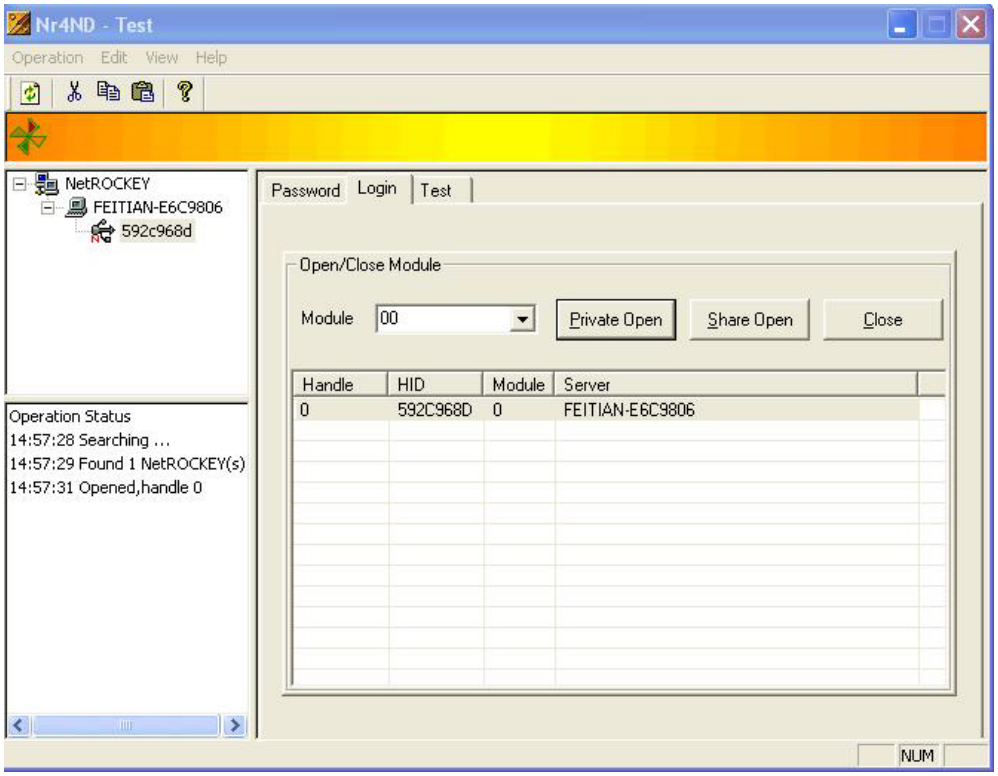


Figure 2.16

Here the returned handle is 0, and click the "Test" tab to operate on handle 0. See Figure 2.17.

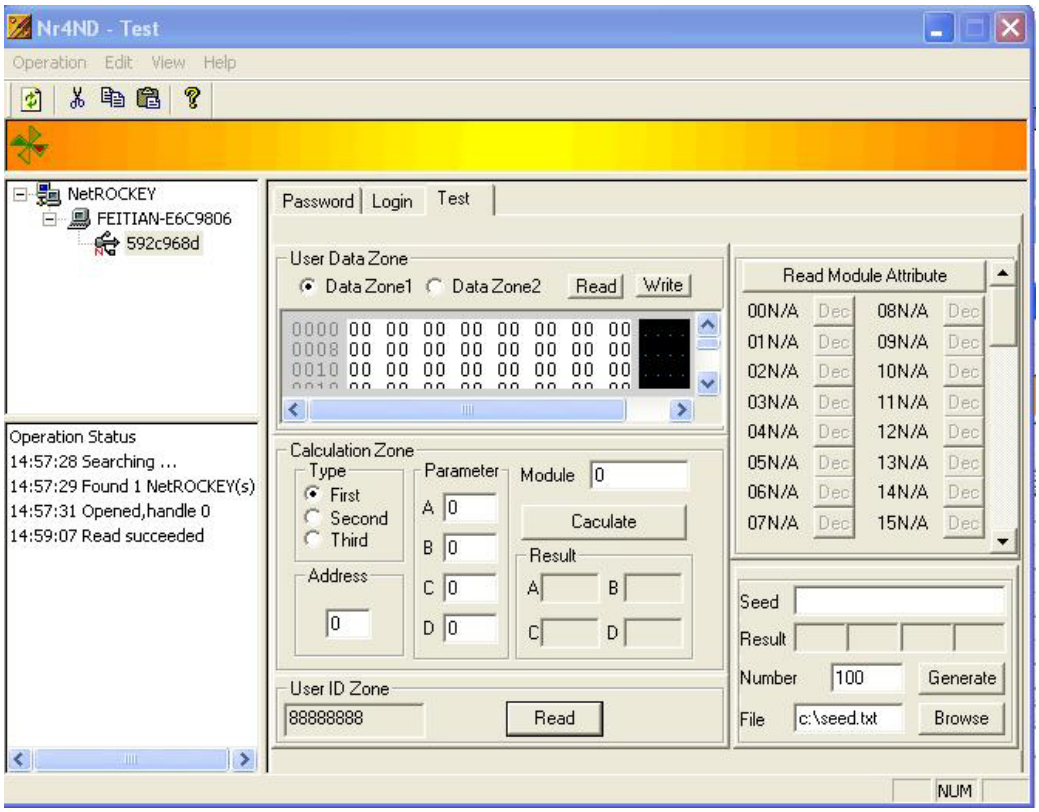


Figure 2.17

You may test the functions, such as read from and write to UDZ, read user ID and the status of the module, calculate, generate seed random number.

All input must be in hexadecimal, except the seed generation number and the character string. After the test please return to the log-in screen to close the handle. If you do not close the handle the service program will kill this client several minutes later (refer to Time To Live). The Test and Monitor programs together can be used to verify and trouble shoot the status of the entire NetROCKEY4ND system.

Chapter 3. NetROCKEY4ND APIs

3.1 Function Prototype and Definition

3.1.1 Function Prototype

NrClient.dll supports NetROCKEY4ND function call. There are the followings 4 functions:

```
  DWORD WINAPI NetRockey
(
    WORD  function,
    WORD  *handle,
    DWORD  *lp1,
    DWORD  *lp2,
    WORD   *p1,
    WORD   *p2,
    WORD   *p3,
    WORD   *p4,
    BYTE   *buffer
  );
DWORD WINAPI NrGetLastError();
DWORD WINAPI NrGetVersion();
DWORD WINAPI SetIniPathName(LPCTSTR iniName);
```

Note: NetRockey is the main function for NetROCKEY4ND.

3.1.2 Parameters

NetRockey provides Net ROCKEY4 function call. A return code of "0" indicates the operation succeeded, all other return codes indicate an error (refer to Chapter 4).

3.1.2.1 function

function is a 16-bit number, it indicates the specific function of NetRockey, and it is defined below:

#define RY_FIND 1 Find NetROCKEY4ND

#define RY_FIND_NEXT 2 Find next NetROCKEY4ND

#define RY_OPEN 3 Open NetROCKEY4ND

#define RY_CLOSE	4	Close NetROCKEY4ND
#define RY_READ	5	Read NetROCKEY4ND
#define RY_WRITE	6	Write NetROCKEY4ND
#define RY_RANDOM	7	Generate Random Number
#define RY_SEED	8	Generate Seed Code
[*]#define RY_WRITE_USERID	9	Write User ID
#define RY_READ_USERID	10	Read User ID
[*] #define RY_SET_MOUDLE	11	Set Module
#define RY_CHECK_MOUDLE	12	Check Module
[*] #define RY_WRITE_ARITHMETIC	13	Write Algorithm
#define RY_CALCULATE1	14	Calculate 1
#define RY_CALCULATE2	15	Calculate 2
#define RY_CALCULATE3	16	Calculate 3
#define RY_DECREASE	17	Decrease Module Unit

Note: function Parameters 9, 11 and 13 are not valid for NetROCKEY4ND because they require the Advanced passwords. The Service program will not recognize these parameters. The Editor for the standard (stand alone)ROCKEY may be used to update these functions in the NetROCKEY product. So these NetROCKEY functions only require basic passwords.

lp1 and *lp2* are long integer parameter pointers (32 bits).

Please refer to APIs section for details.

3.1.2.2 SetIniPathName

IniName is the string name of file, such as "c:\CliCfg.ini".

3.1.2.3 NrGetLastError

Get the last error code from NetROCKEY4ND. This function will also return the error code description.

3.1.2.4 NrGetVersion

Get the version number of NrClient.dll. The high WORD of return value is major version number, and the low WORD is the minor version number.

3.2 APIs

The NetROCKEY4ND API function parameters are defined in details below.

The passwords 3 and 4 should be set to 0 in the final products you offer to your end users.

The functions marked with "*" require advanced passwords.

3.2.1 Find a NetROCKEY4ND dongle (RY_FIND)

Objective: Find Net-ROCKEY4 client and service programs according to parameters set in the configuration file.

Input parameters:

function = RY_FIND

*p1 = Password 1

*p2 = Password 2

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. A successful operation will write the NetROCKEY4ND Hardware ID (HID) to *lp1 and the server name to the buffer.

3.2.2 Find the Next NetROCKEY4ND dongle (RY_FIND_NEXT)

Objective: To check if another NetROCKEY4ND dongle is attached to the network.

Input parameters:

function = RY_FIND_NEXT

*p1 = Password 1

*p2 = Password 2

*lp1 =Hardware ID of last found dongle

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. A successful operation will write the NetROCKEY4ND Hardware ID (HID) to *lp1 and the server name to the buffer.

3.2.3 Open the NetROCKEY4ND dongle (RY_OPEN)

Objective: To log into a specified NetROCKEY4ND module and get a handle number to enable other operations.

Input parameters:

function = RY_OPEN

*p1 = Password 1

*p2 = Password 2

*lp1 = Hardware ID

*lp2 = High word is open mode (0 = private mode, 1 = shared mode); low word is module number that is logged in (0 to 63)

For example,

**lp2=1* means to log into module 1 in private mode.

**lp2 = 0x11* means to log into module 1 in shared mode.

You can also use the VC MAKELPARAM macro, such as lp2 = MAKELPARAM(1,1);

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. A successful operation will return the correct NetROCKEY4ND handle.

Note: You must use the same *p1, *p2 values with the RY_FIND and RY_FIND_NEXT functions. When the operation is successful, *handle is the handle of the dongle and the buffer contains the seed return value (if a seed has been specified).

3.2.4 Close the NetROCKEY4ND dongle (RY_CLOSE)

Objective: To close a NetROCKEY4ND service and logout.

Input parameters:

function = RY_CLOSE

*handle = The NetROCKEY4ND's handle

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error.

3.2.5 Read from a NetROCKEY4ND dongle (RY_READ)

Objective: To read the contents of the User Data Zone (UDZ).

Input parameters:

function = RY_READ

*handle = NetROCKEY4ND's handle

*p1 = offset of UDZ

*p2 = length (unit is byte)

buffer = pointer to the buffer

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. A successful operation will result in the contents of the UDZ written to the memory buffer.

3.2.6 Write to a NetROCKEY4ND dongle (RY_WRITE)

Objective: To write data to the User Data Zone. (UDZ)

Input parameters:

function = RY_WRITE

*handle = NetROCKEY4ND's handle

*p1 = offset of UDZ

*p2 = length (unit is byte)

buffer = pointer to the buffer

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error.

3.2.7 Generate a Random Number (RY_RANDOM)

Objective: To get a random number.

Input parameters:

function = RY_RANDOM

*handle = NetROCKEY4ND's handle

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. A successful operation will result in the *p1 address populated with the random number.

3.2.8 Generate Seed Code Return Values (RY_SEED)

Objective: To get return codes from the input of a seed code.

Input parameters:

function = RY_SEED

*handle = NetROCKEY4ND's handle

*lp2 = Seed Code

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. A successful operation will result in the following addresses populated with seed code return values:

*p1 = Return Code 1

*p2 = Return Code 2

*p3 = Return Code 3

*p4 = Return Code 4

3.2.9 Write the User ID (RY_WRITE_USERID)*

This function is not supported by NetROCKEY4ND. This function may be achieved in the Editor of standalone NetROCKEY4ND before the product is released.

3.2.10 Read User ID (RY_READ_USERID)

Objective: To read the user defined "User ID" from the User ID Zone. (UIZ)

Input parameters:

Function = RY_READ_USERID

*handle = NetROCKEY4ND's handle

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. If the operation is successful, **lp1* is the user ID.

3.2.11 Set a NetROCKEY4ND Module (RY_SET_MODULE)*

This function is not supported by NetROCKEY4ND. This function may be achieved in the Editor of standalone NetROCKEY4ND before the product is released.

3.2.12 Check a NetROCKEY4ND Module (RY_CHECK_MODULE)

Objective: To read the attributes of a specific NetROCKEY4ND module.

Input parameters:

function = RY_CHECK_MODULE

*handle = NetROCKEY4ND's handle

*p1 = Module Number

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. If the operation is successful, "*p2 = 1" means the module is valid; and "*p3 = 1" means the module can be decremented.

3.2.13 Write Algorithm (RY_WRITE_ARITHMETIC)*

This function is not supported by NetROCKEY4ND. This function may be achieved in the Editor of standalone NetROCKEY4ND before the product is released.

3.2.14 Calculate 1 (RY_CALCULATE1)

Objective: To return the results of calculation 1 performed in NetROCKEY4ND.

Input parameters:

function = RY_CALCULATE1

*handle = NetROCKEY4ND's handle

*lp1 = Start point of calculation

*lp2 = Module number

*p1 = Input value 1

*p2 = Input value 2

*p3 = Input value 3

*p4 = Input value 4

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. A successful operation will result in the following addresses populated with the results of the instruction:

*p1 = Return value 1

*p2 = Return value 2

*p3 = Return value 3

*p4 = Return value 4

3.2.15 Calculate 2 (RY_CALCULATE2)

Objective: To return the results of calculation 2 performed in NetROCKEY4ND.

Input parameters:

function = RY_CALCULATE2

*handle = NetROCKEY4ND's handle

*lp1 = Start point of calculation from the UAZ

*lp2 = Seed Code (32-bit)

*p1 = Input value 1

*p2 = Input value 2

*p3 = Input value 3

*p4 = Input value 4

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. A successful operation will result in the addresses p1, p2, p3 and p4 populated with the results of the instruction.

*p1 = Return value 1

*p2 = Return value 2

*p3 = Return value 3

*p4 = Return value 4

3.2.16 Calculate 3 (RY_CALCULATE3)

Objective: To return results of calculation 3 performed in NetROCKEY4ND.

Input parameters:

function = RY_CALCULATE3

*handle = NetROCKEY4ND's handle

*lp1 = Start point of calculation from UAZ

*lp2 = Module number

*p1 = Input value 1

*p2 = Input value 2

*p3 = Input value 3

*p4 = Input value 4

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error. A successful operation will result in the addresses p1, p2, p3 and p4 populated with the results of the instruction.

*p1 = Return value 1

*p2 = Return value 2

*p3 = Return value 3

*p4 = Return value 4

3.2.17 Decrease Module Unit (RY_DECREASE)

Objective: To decrease the value in a specified NetROCKEY4ND module by "1".

Input parameters:

function = RY_DECREASE

*handle = NetROCKEY4ND's handle

*p1 = Module Number

Return value:

A return value = "0" indicates that the function worked correctly. Any other return value indicates an error.

Chapter 4. NetROCKEY4ND Error Codes

There are two kinds of error codes: Normal and Extended. Normal error codes are the error values of the NetROCKEY4ND API. Extended error codes are values returned from the NrGetLastError function. Extended error codes are related to network issues.

4.1 Normal error codes:

Error Code	No.	Description
ERR_SUCCESS	0	Success
ERR_NO_PARALLEL_PORT	1	No parallel port on the computer
ERR_NO_DRIVER	2	No driver installed
ERR_NO_ROCKEY	3	No NETROCKEY4ND dongle
ERR_INVALID_PASSWORD	4	NETROCKEY4ND dongle found, but base password is incorrect
ERR_INVALID_PASSWORD_OR_ID	5	Wrong password or NETROCKEY4ND HID
ERR_SETID	6	Set NETROCKEY4ND HID wrong
ERR_INVALID_ADDR_OR_SIZE	7	Read/Write address is wrong
ERR_UNKNOWN_COMMAND	8	No such command
ERR_NOTBELEVEL3	9	Inside error
ERR_READ	10	Read error
ERR_WRITE	11	Write error
ERR_RANDOM	12	Random error
ERR_SEED	13	Seed Code error
ERR_CALCULATE	14	Calculate error
ERR_NO_OPEN	15	Ry_Open must precede this operation
ERR_OPEN_OVERFLOW	16	Too many open dongles (>16)
ERR_NOMORE	17	No more dongle
ERR_NEED_FIND	18	No Find before FindNext
ERR_DECREASE	19	Decrease error
ERR_AR_BADCOMMAND	20	Arithmetic instruction error
ERR_AR_UNKNOWN_OPCODE	21	Arithmetic operator error
ERR_AR_WRONGBEGIN	22	A constant. cannot be in the first instruction
ERR_AR_WRONG_END	23	A constant. cannot be in the last instruction
ERR_AR_VALUEOVERFLOW	24	Const number > 63
ERR_NET_LOGINAGAIN	1001	A module can only be opened once by the same process.
ERR_NET_NETERROR	1002	Network error.

ERR_NET_LOGIN	1003	Too many users are logged on.
ERR_NET_INVALIDHANDLE	1004	Invalid handle, this handle might have been closed.
ERR_NET_BADHARDWARE	1005	Defective hardware
ERR_NET_REFUSE	1006	Client dll modified, service refused request.
ERR_NET_BADSERVER	1007	Nrsvr.exe modified, service is invalid.
Below are network error codes, NrGetLastError function can return extended error codes.		
ERR_INIT_SOCK	2001	Error when initializing.
ERR_NOSUCHPROTO	2002	No such protocol.
ERR_UDPSOCKCREATE	2003	UDP error when creating socket.
ERR_UDPSETBROADCAST	2004	UDP error when setting broadcast.
ERR_UDPBINDFAILED	2005	UDP error when binding.
ERR_SVRCALLBACKNULL	2006	Server call back null.
ERR_TCPSOCKCREATE	2007	TCP error when creating socket.
ERR_TCPBINDFAILED	2008	TCP error when binding.
ERR_TCPLISTENFAILED	2009	TCP error when listening.
ERR_NOSUCHSEARCH	2010	No such search mode.
ERR_UDPSEND	2012	UDP error when sending.
ERR_UDPTIMEOUT	2013	UDP timeout error when waiting.
ERR_UDPRECV	2014	
ERR_TCPCONNECT	2015	TCP error when connecting to server.
ERR_TCPSENDTIMEOUT	2016	TCP time out error when sending.
ERR_TCPSEND	2017	TCP error when sending.
ERR_TCPRECVTIMEOUT	2018	TCP time out error when receiving.
ERR_TCPRECV	2019	TCP error when receiving.
ERR_IPXSOCKCREATE	2020	IPX error when creating socket.
ERR_IPXSETBROADCAST	2021	IPX error when setting broadcast.
ERR_IPXSEND	2022	IPX error when sending data.
ERR_IPXRECV	2023	IPX error when receiving data.
ERR_IPXBIND	2024	IPX error when binding.
ERR_NBSRESET	2025	NetBIOS error when initializing.
ERR_NBSADDNAME	2026	NetBIOS error when adding name.
ERR_NBSSEND	2027	NetBIOS error when sending data.
ERR_NBSRECV	2028	NetBIOS error when receiving data.

4.2 Extended Error Codes for UDP/TCP and IPX

Run the NrGetLastError function after receiving any network related return code. The result will be an error code constant that you can use to look up more detailed error information in the TCP/UDP or IPX specifications.

Definition atement	Regular Berkeley Error Constants
#define WSABASEERR	10000
#define WSAEINTR	(WSABASEERR+4)
#define WSAEBADF	(WSABASEERR+9)
#define WSAEACCES	(WSABASEERR+13)
#define WSAEFAULT	(WSABASEERR+14)
#define WSAEINVAL	(WSABASEERR+22)
#define WSAEMFILE	(WSABASEERR+24)
#define WSAEWOULDBLOCK	(WSABASEERR+35)
#define WSAEINPROGRESS	(WSABASEERR+36)
#define WSAEALREADY	(WSABASEERR+37)
#define WSAENOTSOCK	(WSABASEERR+38)
#define WSAEDESTADDRREQ	(WSABASEERR+39)
#define WSAEMSGSIZE	(WSABASEERR+40)
#define WSAEPROTOTYPE	(WSABASEERR+41)
#define WSAENOPROTOOPT	(WSABASEERR+42)
#define WSAEPROTONOSUPPORT	(WSABASEERR+43)
#define WSAESOCKTNOSUPPORT	(WSABASEERR+44)
#define WSAEOPNOTSUPP	(WSABASEERR+45)
#define WSAEPFNOSUPPORT	(WSABASEERR+46)
#define WSAEAFNOSUPPORT	(WSABASEERR+47)
#define WSAEADDRINUSE	(WSABASEERR+48)
#define WSAEADDRNOTAVAIL	(WSABASEERR+49)
#define WSAENETDOWN	(WSABASEERR+50)
#define WSAENETUNREACH	(WSABASEERR+51)
#define WSAENETRESET	(WSABASEERR+52)
#define WSAECONNABORTED	(WSABASEERR+53)
#define WSAECONNRESET	(WSABASEERR+54)
#define WSAENOBUFS	(WSABASEERR+55)
#define WSAEISCONN	(WSABASEERR+56)
#define WSAENOTCONN	(WSABASEERR+57)
#define WSAESHUTDOWN	(WSABASEERR+58)
#define WSAETOOMANYREFS	(WSABASEERR+59)
#define WSAETIMEDOUT	(WSABASEERR+60)
#define WSAECONNREFUSED	(WSABASEERR+61)
#define WSAELOOP	(WSABASEERR+62)
#define WSAENAMETOOLONG	(WSABASEERR+63)
#define WSAEHOSTDOWN	(WSABASEERR+64)
#define WSAEHOSTUNREACH	(WSABASEERR+65)
#define WSAENOTEMPTY	(WSABASEERR+66)
#define WSAEPROCLIM	(WSABASEERR+67)

#define WSAEUSERS	(WSABASEERR+68)
#define WSAEDQUOT	(WSABASEERR+69)
#define WSAESTALE	(WSABASEERR+70)
#define WSAEREMOTE	(WSABASEERR+71)
#define WSASYSNOTREADY	(WSABASEERR+91)
#define WSAVERNOTSUPPORTED	(WSABASEERR+92)
#define WSANOTINITIALISED	(WSABASEERR+93)
#define WSAEDISCON	(WSABASEERR+101)
#define WSAHOST_NOT_FOUND	(WSABASEERR+1001)
#define WSATRY_AGAIN	(WSABASEERR+1002)
#define WSANO_RECOVERY	(WSABASEERR+1003)
#define WSANO_DATA	(WSABASEERR+1004)

4.3 Extended Error Codes for NetBios

Return Code	Return Code Definition
0x00	good return, also returned when ASYNCH request accepted
0x01	illegal buffer length
0x03	illegal command
0x05	command time out
0x06	message incomplete, issue another command
0x07	illegal buffer address
0x08	session number out of range
0x09	no resource available
0x0a	session closed
0x0b	command cancelled
0x0d	duplicate name
0x0e	name table full
0x0f	no deletions, name has active sessions
0x11	local session table full
0x12	remote session table full
0x13	illegal name number
0x14	no callname
0x15	cannot put * in NCB_NAME
0x16	name in use on remote adapter
0x17	name deleted
0x18	session ended abnormally
0x19	name conflict detected
0x21	interface busy, IRET before retrying
0x22	too many commands outstanding, retry later
0x23	ncb_lana_num field invalid

0x24	command completed while cancel occurring
0x26	command not valid to cancel
0x30	name defined by anther local process
0x34	environment undefined. RESET required
0x35	required OS resources exhausted
0x36	max number of applications exceeded
0x37	no saps available for netbios
0x38	requested resources are not available
0x39	invalid ncb address or length > segment
0x3B	invalid NCB DDID
0x3C	lock of user area failed
0x3f	NETBIOS not loaded
0x40	system error
0xff	synchronous command is not yet finished

Chapter 5. Quick Test

Determine the network protocols that you will need (IPX, TCP/UDP and/or NetBIOS). Verify that you have all the contents of the developer's kit: a NetROCKEY4ND dongle, a CD-ROM and a user manual. You can choose a protocol from TCP/UDP, IPX, and NetBios. The following description is based on the TCP/UDP protocol.

Test physical network connectivity by running a "ping" test between the client and server machines.

Install Feitian dongle software on both server side and client side. Run Setup.exe from the CD-ROM to install all components on the server side, including the driver, and to install the NetRockey4ND component on the client side (do not install the driver on the client side).

Attach the NetROCKEY4ND dongle into server. Check if both the driver and the dongle work properly using the Editor program. If there is no problem with then, write the maximum number of simultaneous users to a NetROCKEY4ND module. For example, write "2" to module 0 to allow a maximum of two users to access the software at one time.

Start the NetROCKEY4ND service by running the *nrsvr.exe* program. The service program will use the default configuration file (*SvrCfg.ini*). After you change the configuration file, you have to exit and restart the service program. Use the NrMon to view the status of the service program.

Run the *nrtest.exe* program from a client computer. Attempt to log into the module where you wrote the maximum number of users. In the example given above, you should only be able to log into module number "0" two times. Use the NrMon to view the status of the service and client programs. By default, automatic search is done under the UDP protocol. Thus, it is not necessary to specify the address of server in the client configuration file.