Security File

Specify the full path name of the Security File to be used by Winsock RSHD/NT to enforce security (allow and deny users and hosts). If you do not specify a Security File, **all** users and hosts will be granted access to execute commands and transfer files to and from the system, unless you enable the option that requires remote user names to exist as Windows NT users or enable the option to reject all **rsh** commands.

If you do specify a Security File and it does not exist, ***no*** users or hosts will be granted access. If you do not wish to enforce any security, do not specify a filename.

Default: Blank - No Security File

WINSOCK RSHD/NT SECURITY FILE

You create the Security File using a text editor. In the Winsock RSHD/NT Control Panel applet, you can click on the **Edit Security** button to run the Windows Notepad editor to edit the security file specified in the Security File configuration option.

The Security File consists of lines that specify who may or may not access the PC using Winsock RSHD/NT. Its format is similar to the Unix **.rhosts** file, but there are a few differences. The following are the options available in the Security File:

#

Any line beginning with # is treated as a comment and is ignored.

+

A plus sign (+) on a line by itself specifies that ALL hosts and users are granted permission. This is useful if you wish to allow many hosts and users, but deny only a few. Use the deny options on subsequent lines.

host

You can specify a host that is granted permission by entering the name of the host on a line by itself. **All** users on that host are granted permission, unless you specifically deny those users on subsequent lines.

You may also use the IP address of the host instead (the dotted-decimal representation). If you specify the name of the host, that name must appear in the *hosts* file used by your TCP/IP package.

!host

You can specify a host that is denied permission by entering an exclamation point (!) followed by the name of the host name of the host on a line. **All** users on that host are denied permission, regardless of subsequent lines.

You may also use the IP address of the host instead (the dotted-decimal representation). If you specify the name of the host, that name must appear in the *hosts* file used by your TCP/IP package.

+user

You can specify a user name that is granted permission by entering a plus sign (+) followed by the user name on a line. Do not put any spaces between the plus sign and the user name. That user will be granted permission regardless of the host (as long as the host is not specifically denied).

See below for an explanation of the source of the user name and how it is validated.

-user

You can specify a user name that is to be denied permission by entering a minus sign (-) followed by the user name on a line. Do not put any spaces between the plus sign and the user name. That user will be denied permission on all hosts.

See below for an explanation of the source of the user name and how it is validated.

+user@host

You can specify a user name and a host that is granted permission by entering a plus sign (+) followed by the user name, an at-sign (@), followed by the host name on a line. Do not put any spaces between the plus sign and the user name or before or after the at-sign. That user on the specified host will be granted permission, but only from that host.

You may also use the IP address of the host instead (the dotted-decimal representation). If you specify the name of the host, that name must appear in the *hosts* file used by your TCP/IP package.

-user@host

You can specify a user name and a host that is denied permission by entering a minus sign (-) followed by the user name, an at-sign (@), followed by the host name on a line. Do not put any spaces between the minus sign and the user name or before or after the at-sign. That user on the specified host will be denied permission, but only when coming from that host.

You may also use the IP address of the host instead (the dotted-decimal representation). If you specify the name of the host, that name must appear in the *hosts* on the Windows NT system.

If the request is coming from a Unix system, the user name is the login name of the user. If the request is coming from another Windows PC, the method of specifying the user name is determined by the implementation of the **rsh** or **rcp** command you are using.

Note that the standard Unix **rsh** command (and the Winsock **rsh** command available from Denicomp Systems) allows a "**-l**" option to specify an alternate user name. The "**-l**" option has meaning on a Unix system, but is not especially useful with Winsock RSHD/NT. However, if you do use the "**-l**" option to specify an alternate user, as with Unix, that user must be granted permission through the Security File in addition to the login name (Unix) or the name specified in your particular TCP/IP implementation (Windows/DOS).

USING THE SECURITY FILE

To effectively use the Security File, you must first understand how it is viewed by Winsock RSHD/NT.

When Winsock RSHD/NT receives a request, it sequentially processes the lines in the Security File to determine whether or not the host and user are granted or denied access. It looks at each line in the Security File until it determines that either the host or the user is specifically denied permission.

Winsock RSHD/NT begins by assuming that permission is denied for the request. It then examines the lines in the Security File to see if any of the lines pertain to this request.

Once Winsock RSHD/NT finds a line that denies access to either the user or the host, it stops looking and denies permission.

If it finds a line that grants permission to the user and/or host, permission is tentatively granted, but it continues to process the lines in the Security File.

If it processes the entire Security File and does not find a line that grants permission to the user and/or the host, permission is denied. If security was tentatively granted at some point and not denied subsequently, permission is granted.

For example, let's say that the following is the contents of the Security File:

```
jetty
booey
eib
192.56.42.3
rs6000
+fred@mars
-gary@booey
-jackie
```

```
+robin
```

In this example, if any user on the host "jetty" makes a request, permission will be granted, unless the user is "jackie", since "jackie" is denied access from all hosts (-jackie).

If "jackie@jetty" makes a request, Winsock RSHD/NT reads through the Security File and finds the host name "jetty", and tentatively grants permission. However, it continues and finds that the user "jackie" is denied from all hosts, so permission is denied.

Also, if any user on the host "booey" makes a request, they are granted permission unless the user is "gary", since "gary@booey" is specifically denied permission (-gary@booey). All other users on the host "booey" are granted permission except "jackie" (-jackie).

The user "fred" on the host "mars" is granted permission because of the line "+fred@mars". However, since the host "mars" does not appear on a line by itself, no other users on the host "mars" are granted permission except the user "robin", who is granted permission from *any* host (+robin).

Must Remote Users be Valid Users on this System?

If this option is unchecked, user login names sent to Winsock RSHD/NT by **rcp** and **rsh** do **not** need to be valid users on this system. Security is enforced solely through the Security File.

If this option is checked, user login names from **rcp** and **rsh must** be valid users on the this system. If the user is not valid, access will be denied. This is the standard behavior of a *rshd* daemon. However, if the **rcp** and **rsh** commands are being executed from another operating system such as Unix, the user login names may not be the same between systems. If user logins are the same, you can then enable this option for security. If they are different, you should not enable this option.

For example, if this option is enabled and you are logged in as “john” on a Unix system and you issue an **rsh** command to this system, a user account named “john” must exist on this system or access will be denied.

Default: Not Checked

Message File

Specify the full path name of a file where any messages from Winsock RSHD/NT should be stored. The message file is **optional.** You should only enable the message log when you are trying to find the source of a problem, since the message log can become quite large on an active system.

This option is used in conjunction with the **Message Level** option. If **Message Level** is set to a value greater than zero (0), Winsock RSHD/NT will output messages that provide information about its operation. These messages are mostly useful for problem determination.

The message file created is a text file that you can examine at any time using utilities such as TYPE or MORE, or editors such as Notepad. You can clear the message log at any time by simply deleting it.

Default: Blank - No Message File

Message Level

Specifies the level of detail of the messages stored in the file specified in the **Message File** option. The default level is **0**, which will not write any messages to the message log file. Levels 1 through 4 will product increasing amounts of detail (level 1 provides the least detail, level 4 provides the most).

Default: 0

Request Log

This option allows you to log all requests (commands to be executed) in a file you specify. Each time someone attempts to execute a command through Winsock RSHD/NT, the date and time, the user name, the host name, and the command will be written to this file.

Default: Blank - No Request Log

Deny Log

This option allows you to log all permission violations in a file you specify. Each time someone is denied permission to execute a command through Winsock RSHD/NT, the date and time, the user name, the host name, and the command will be written to this file.

Default: Blank - No Deny Log

Error Log

This option allows you to log all command execution errors in a file you specify. Each time someone receives an error trying to execute a command through Winsock RSHD/NT, the date and time, the user name, the host name, the command, and error message will be written to this file. These are errors that occur after the user has been granted permission to execute the command. For example, an error would be logged if a program was to be run that did not exist.

Default: Blank - No Error Log

Reject All Incoming RSH Commands?

If you check this option, all incoming **rsh** commands will be rejected, effectively disabling the **rsh** serving capability of RSHD/NT. This is useful if you only want to use RSHD/NT as an **rcp** server.

If a remote user attempts to issue an **rsh** command to this system, an error will be returned to the remote user stating that **rsh** has been disabled.

Default: Not Checked

Attempt Stdout/Stderr Capture on Every Command?

When you execute a command through Winsock RSHD/NT, it assumes that the command is a Windows program (not a Console or MS-DOS program) and that there is no redirection of standard output/standard error back to the remote system, **unless** you use the special "<[CON]>", "<[CON2]>", "<[DOS]>", or "<[DOS2]>" indicators in the **rsh** command.

If you mostly execute Windows NT Console programs and/or MS-DOS programs via **rsh** through Winsock RSHD/NT, you can check this option and Winsock RSHD/NT will assume that each command is a Console/MS-DOS program and attempt to send its standard output/standard error back to the remote system. It treats every **rsh** request as if the "<[CON]>" option was specified. You do not need to specify the "<[CON]>" indicator; it is assumed.

If you check this option, you still may execute Windows programs via **rsh** and they will operate properly. However, unless you specify the special "<[WIN]>" indicator for Windows programs in the **rsh** command, there are a few downsides: First, there will be slightly more overhead when executing Windows programs because Winsock RSHD/NT will attempt to capture the standard output/standard error. Also, Winsock RSHD/NT will wait for the Windows program to complete before closing the connection. Again, these downsides can be overcome by specifying the "<[WIN]>" indicator when executing Windows programs.

Default: Not Checked

Execute All Commands through Command Shell?

If you check this option, Winsock RSHD/NT will automatically prefix every command you execute with the default command shell (usually **cmd /c**).

This is useful if you commonly execute batch files (.BAT or .CMD) or other command scripts for the shell you are using and you do not want to have to specify the shell command in every **rsh** command.

The default shell command is used as the prefix. This command can be specified in the **Default Shell Command** field; if no default shell command is specified there, **cmd /c** is used.

For example, if this option is enabled and you execute the following command from a remote system:

```
rsh ntsystem xyz.bat
```

RSHD/NT will execute the command as:

```
cmd /c xyz.bat
```

Default: Not Checked

Disable Detection of Internal Commands

When you execute a command through Winsock RSHD/NT, it examines the command to determine whether or not it is a command internal to the default command shell (interpreter). If it is, it automatically prefixes the command with the default command shell (specified in the **Default Command Shell** field).

Some commands are not actually programs; they are interpreted and executed internally by the command shell. In the Windows NT command interpreter (CMD.EXE), some examples of these are DIR, SET, and COPY. If you look on your hard drive, you will not find a DIR.EXE or COPY.EXE. They are part of the NT command interpreter, CMD.EXE.

So, if you tried to execute a DIR command through RSHD/NT, it would not find the program since it doesn't exist. You would have to tell it to use the command interpreter by executing the command "CMD /C DIR".

By default, RSHD/NT examines the command and if it determines that the command is an internal command, it adds the shell command for you. You can disable this by checking this option. All commands will be executed as they are specified in the **rsh** command.

NOTE: If you check the option **Execute All Commands through Command Shell?**, this option is irrelevant, since the command shell will be added to the command in all cases.

Default: Not Checked

Default Window Type for Commands

This specifies the default window type to be used when executing commands through Winsock RSHD/NT using the **rsh** command. The default window type is used when the special window type indicators are not specified in the **rsh** command (<[NORMAL]>, <[MINIMIZE]>, <[MAXIMIZE]>, <[HIDE]>, etc.).

The options available are:

Normal: The window for the command will display at its normal size.

Minimized: The window for the command will be minimized (without focus).

Maximized: The window for the command will be maximized.

Hidden: The window for the command will be hidden.

There are a few points you must consider when selecting the default window type:

The **Minimized** or **Hidden** options are useful when the system running RSHD/NT is actively used and is not a standalone server. With the **Normal** or **Maximized** options, the person using the system will see a window appear each time a command is executed through **rsh**.

You ***cannot*** send keystrokes to commands that are minimized or hidden. So if you select the **Minimized** or **Hidden** option, you must override it in the **rsh** command when you want to send keystrokes. That is, you will need to add the <[NORMAL]> option to the command.

Using the **Hidden** option can cause administrative problems. When a program's window is hidden, it does not appear in the Task List, so there is not an easy way to stop a hidden program or to tell if any are running. There are utilities that allow you to see the hidden programs (such as the PView program that comes with Microsoft Visual C/C++).

List of Commands to Allow (File)

This option allows you to specify the name of a file that contains a list of commands that users are permitted to execute on this system through **rsh**. This allows you to provide strict control over the commands users can execute.

If no filename is specified here, all commands are permitted.

The file must be a plain text file, with each permitted command on a line by itself. Commands in the file should not contain any spaces. Comparison of commands is done only up to the first space or tab character.

When a user executes a command on this system through **rsh**, RSHD/NT will extract the first part of the command, up to the first space or tab character, and compare that to the lines in the file specified. If it does not exist in the file, the **rsh** command will be rejected.

Default: Blank

Environment Variable File

This allows you to specify the name or names of files that contain environment variables that should be made available when commands are executed by RSHD/NT through **rsh**.

Normally, the environment for commands executed through RSHD/NT comes from the **System Environment Variables** specified in the System applet in the Control Panel. Those variables are inherited from the Windows NT Service Manager, so if the System Environment Variables are changed, you must reboot the system for them to be propagated to RSHD/NT (if you are running RSHD/NT as a service).

Alternatively, you can create a custom environment for RSHD/NT by entering the environment variables and values in a plain text file. Each line in the text file should have the format:

VARIABLE=VALUE

Each time a command is executed through RSHD/NT by **rsh**, a custom environment is built from the file or files specified in this parameter, based on the lines in those files.

You can specify a single filename or multiple filenames, with each separated by semi-colons (;). Each file is read in sequence and added to the System Environment Variables inherited by RSHD/NT to create a new environment for the command to be executed. If a variable name appears in multiple files, the last value read will be used.

You may reference previously set environment variables as you do in Windows NT batch files using %VAR%. For example:

PATH=%PATH%;C:\MYPROGS

The filenames should be full path names. There are three special keywords that you can use in the filenames if you wish:

%ruser% - Substitute the login name of the remote user
%luser% - Substitute the login name of the local user
%rhost% - Substitute the host name of the remote host

The **%ruser%** substitutes the login name of the remote user. This will be the login name the user used to log into the remote host from which the **rsh** command is being issued, unless the **-l** option of the **rsh** command was used to specify a different user; then that user will be substituted.

The **%luser%** substitutes the login name of the local user on the remote host. Normally, it is the same as **%ruser%**, unless the **-l** option of the **rsh** command was used. Then, this will contain the actual user login used at the remote host.

For example, if you are logged in as “john” on a remote host and you issue the command “rsh -l

mary winpc xyz", the %ruser% will substitute "mary" and the %luser% will substitute "john".

The **%rhost%** substitutes the host name of the remote host, if it is available. That is, RSHD/NT must be able to find the name of the remote host based on its IP address, either by using the HOSTS file or DNS. If it is not found, the IP address will be substituted.

These special keywords allow you to have different environment files for different users if necessary. For example, if you specify the environment variable file:

c:\env\%ruser%.env

When "john" issues an **rsh** command, RSHD/NT will get the environment from the file "c:\env\john.env". When "mary" issues an **rsh** command, RSHD/NT will get the environment from the file "c:\env\mary.env".

Also, using the capability to specify multiple files, you can have a single "master" environment, and then only modifications to it by user. For example, you can have a standard set of environment variables in the file "c:\env\master.env" and user-specific modifications in the file "c:\env\%ruser%.env". Your environment variable file field would read:

c:\env\master.env;c:\env\%ruser%.env

First the variables in master.env would be read, then those for the user in %ruser%.env.

THE SPECIAL "new" KEYWORD

The format of the environment variable files must be:

VARIABLE=VALUE

But, with one exception. If you specify the word "new" on a line by itself in an environment variable file, it will purge all environment variables set up to that point.

The primary purpose of this would be to remove all variables inherited from the System Environment Variables. It allows you to start with a "clean slate" and set all environment variables from scratch.

Default Command Shell

This option allows you to specify the default command shell to be used when RSHD/NT detects an internal command or the command shell to be used if the option to Execute All Commands through Command Shell is checked.

You should use this option only if you are using an alternate command shell. By default, RSHD/NT uses the Windows NT command shell CMD.EXE.

You must specify all necessary options to the command shell so that it can be prefixed to any command (internal or external).

For example, if you were using a Windows NT implementation of the Unix Korn Shell, you might specify “ksh -c” here.

Default: Blank - Use the Windows NT command shell (CMD /C)

Internal Command List

If you specified a Default Command Shell that has different internal commands than those of the standard Windows NT command shell CMD.EXE, you can specify the internal commands for that shell here. Separate each command with a comma (,). Do not include any spaces.

If you checked the option to Disable Detection of Internal Commands, this is not necessary and will have no effect.

You only need to specify this list if you want RSHD/NT to recognize commands internal to your command shell and automatically prefix the command with the appropriate shell command.

Default: Blank - Use a known list of internal commands for NT's command shell

Buffer Stdout/Stderr Until End of Command?

Check this option if you want RSHD/NT to buffer (store in a file) the standard output and standard error output of the commands you execute, and then send all of the output when the command completes.

Prior to Version 2.0, this was the standard behavior of RSHD/NT. With Version 2.0, the standard output and standard error is sent as it occurs (although it may be buffered by Windows NT or the program executed). If you check this option, RSHD/NT will operate as it did in previous versions.

Default: Not Checked

Reject All RCP Copies to This System

If this option is checked, all attempts to copy files to this system with the **rcp** command will be rejected with an error message stating that incoming copies are disabled. This allows you to make the system "read only" when using the **rcp** command.

You can also reject copying from the system, essentially disabling the **rcp** capability of RSHD/NT.

Default: Not Checked

Reject All RCP Copies to From System

If this option is checked, all attempts to copy files from this system with the **rcp** command will be rejected with an error message stating that outgoing copies are disabled. This allows you to make the system "write only" when using the **rcp** command.

You can also reject copying to the system, essentially disabling the **rcp** capability of RSHD/NT.

Default: Not Checked

Preserve Case in Multi-File Copies

Specifies whether Winsock RSHD/NT should preserve the case of filenames when files are copied **from** this system by **rcp** using wildcards or recursive copies. By default, when the remote system uses a wildcard or recursive copy to get files from this PC, Winsock RSHD/NT will convert all directory and filenames to **lowercase** letters before sending them to the remote system.

Although the Windows NT filesystem is not case sensitive (ABC and abc are the same file), it can store the case of the filename. When copying files via **rcp** to operating systems that are case sensitive, such as Unix, it is usually most useful to convert all of the names to lowercase letters.

If you do not wish to have all of the names converted to lowercase letters, check this option. The **rcp** command will then create files in exactly the same case as the names appear in the directory under Windows NT.

Note that this affects **only** wildcard and recursive copies. When copying individual files, the files will be created in the case you specify in the **rcp** command.

Default: Not Checked

Automatic End-of-Line Conversion

Specifies whether or not RSHD/NT should perform any end-of-line conversions on files transferred to or from the NT system using **rcp**.

Under MS-DOS, Windows 3.x, Windows 95, and Windows NT, lines of text are delimited by carriage return and newline pairs (ASCII 13 and ASCII 10). Under Unix, lines of text are delimited by only newlines (ASCII 10). Often, when copying text files between the two operating systems, it is necessary to convert the end-of-line delimiters to the proper method. RSHD/NT provides a way to automatically do this.

When files are copied from the NT system through RSHD/NT, carriage returns will be removed from all carriage return/newline pairs (i.e. converted to Unix format). When files are copied to the NT system through RSHD/NT, carriage returns will be added to every newline character that is not already prefixed by a carriage return (i.e. converted to NT format).

There are four options available. The option you select affects all **rcp** copies to this system and all **rcp** copies from the system. It does not affect the operation of the **rcp** command on the NT system. It only affects the result of **rcp** commands that access files on this system from other systems.

- Never - Copy all files as binary
  No end-of-line conversion will be done. All files will be transferred or received unmodified.

- Always - Convert all files
  Convert the end-of-line characters on every file copied from or to this system through RSHD/NT.

- Convert based on list of file extentions
  Only convert end-of-line characters in files ending with the specified list of file extensions. You must then enter the list of file extensions. Separate each file extension by a comma. Do not include any spaces. You must include the "dot" (.). For example: .TXT,.C,.H,.PRN,.MAK

  Any file not ending in one of these extensions will be copied without modification.

- Convert based on contents of first block
  RSHD/NT will examine the contents of the first block of the file to be sent or received and determine whether or not an end-of-line conversion is necessary. If the first block contains only text characters (letters, numbers, spaces, tabs, carriage returns, newlines, backspaces, escapes, and form feeds), RSHD/NT will perform an end-of-line conversion on the file. If the first block contains any other non-text data, it will be copied without modification.

  The size of the first block is specified in the **RCP Block Size** field.

Default: Never - Copy all files as binary

RCP Home Directory

Specifies the starting directory where files will be copied from or to when a relative path name is used in an **rcp** command (no initial slash or backslash).

This directory must exist if specified. The directory name specified can contain the following special keywords:

%ruser% - Substitute the login name of the remote user
%luser% - Substitute the login name of the local user
%rhost% - Substitute the host name of the remote host

The **%ruser%** substitutes the login name of the remote user. This will be the login name the user used to log into the remote host from which the **rcp** command is being issued, unless the **@** option of the **rcp** command was used to specify a different user; then that user will be substituted (e.g. user@host:filename).

The **%luser%** substitutes the login name of the local user on the remote host. Normally, it is the same as **%ruser%**, unless the **@** option of the **rcp** command was used. Then, this will contain the actual user login used at the remote host.

For example, if you are logged in as “john” on a remote host and you issue the command “rcp xyz mary@winpc:”, the %ruser% will substitute “mary” and the %luser% will substitute “john”.

The **%rhost%** substitutes the host name of the remote host, if it is available. That is, RSHD/NT must be able to find the name of the remote host based on its IP address, either by using the HOSTS file or DNS. If it is not found, the IP address will be substituted.

Default: Blank - Use the Initial Working Directory if specified or the Installation Directory

RCP Block Size

Specifies the number of bytes in a block of data that the Remote Copy (**rcp**) service of Winsock RSHD/NT processes at one time. When files are copied to the PC, it reads from the network and writes to the disk in blocks of this size. When files are copied from the PC, it reads from the disk and writes to the network in blocks of this size. Note that this is an internal block size only; it does not change any TCP/IP parameters.

Default: Blank - 512 bytes

RCP Spoofing Prefix

Specifies the first characters of the **rcp** command send by the remote host that Winsock RSHD/NT should use when "spoofing" the **rcp** protocol. With its roots in Unix, the **rcp** command actually internally executes an **rsh** command to start **rcp** on the remote host before transferring files. Winsock RSHD/NT "spoofs" or looks for **rcp** commands executed through **rsh** by the remote host and services the **rcp** transfer.

By default, Winsock RSHD/NT looks for the command prefixes of:

rcp -
/usr/bin/rcp -
/usr/lib/sunw,rcp -
set vms_rcp = 1 ; rcp -

Some **rcp** commands (especially those on non-Unix and non-Windows systems) may send other commands to initiate the **rcp** protocol. If yours does, you should enter the command prefix (up to and including the first hyphen) here. The last character should **always** be a hyphen with nothing after it, including spaces.

Regardless of the spoofing prefix entered, Winsock RSHD/NT will continue to look for the above default prefixes.

Default: Blank

Execute All RCP Copies As User/Password

This option allows you to specify a separate Windows NT user account and password for all **rcp** copies. Normally, RSHD/NT reads and writes all files as the user specified in the Winsock RSHD/NT service setup under Windows NT.

However, when RSHD/NT is running as the default LocalSystem user so that it has access to the desktop, it access all files as this special LocalSystem user. Unfortunately, Windows NT does not allow the LocalSystem user to access network resources.

By specifying a user account and password here, you can still run the RSHD/NT service through the LocalSystem account, but control access to files through this user account.

When RSHD/NT services an **rcp** command, it will essentially log in as this user. It will be able to read and write any files that this user can read and write. If new files are created, they will be owned by this user (not the user on the remote system). So you can control access to files through **rcp** by modifying the permissions of this user.

The user can be a normal user account or you can create a special account just for the purpose of **rcp** copies. It is recommended that you specify that the user's password does not expire, however. If the user's password expires, **rcp** copies will not function until you manually update the RSHD/NT setup with the new password. Windows NT will not update it for you.

Default: Blank

Installation Directory

This is the directory where Winsock RSHD/NT is installed. This will be filled in by the RSHD/NT installation program. If you move RSHD/NT to another location, you must update this. RSHD/NT uses this to find its files, so if it is incorrect, RSHD/NT may not operate properly.

Default: C:\WRSHDNT

Initial Working Directory

This specifies the directory that will initially be considered the current working directory for all commands executed using **rsh**. It will also be considered the current working directory for all files copied using **rcp**, unless an RCP Home Directory is specified.

When RSHD/NT starts, it changes to this directory and remains there, unless an **rsh** request is received to execute the **cd** or **chdir** command, which will change RSHD/NT's working directory.

Note that a **cd** or **chdir** command will change RSHD/NT's working directory for **all** subsequent commands, regardless of the user or system they are executed from.

Default: Blank - Use the Installation Directory

Disable Multithreading in RSHD/NT

Multithreading allows Winsock RSHD/NT to process multiple requests simultaneously. When multithreading is disabled by checking this option, Winsock RSHD/NT will accept and complete only one request at a time. Other requests received during this time will be queued and executed in the order in which they were received. Normally, you will want multithreading enabled, but you can disable it, for example, to ensure that the system will not become bogged down with requests.

Default: Not Checked

Disable Monitoring of Registry for Changes

Normally, Winsock RSHD/NT starts a thread that monitors the Windows Registry for changes to the Winsock RSHD/NT configuration options and if any options are changed, re-reads the registry so that the new options take effect.

If this option is checked to disable the monitoring of the Registry, you must stop and start Winsock RSHD/NT manually (or reboot the system) for the Registry changes to take effect.

You may want the Registry monitoring disabled for security purposes so that no Winsock RSHD/NT options are changed while the system is in operation.

Default: Not Checked

Host IP Address (If Multi-Homed)

If your system is multi-homed (it has multiple network cards, each with its own IP address), you can specify which IP address RSHD/NT will use to listen for requests. If you leave this empty, it will accept requests from any of the IP addresses associated with the system. If you specify one of the addresses of one of the cards (in dotted-decimal format), it will only accept requests routed to that address.

Default: Blank - Listen on all IP addresses associated with this system

Listener Port

Specifies the port number that Winsock RSHD/NT listens to for connections. The standard port for the Remote Shell daemon is 514.

Default: 514

Listener Backlog

The number of requests that can be *backlogged* when Winsock RSHD/NT is listening for connections. The minimum is 1; the maximum for Windows NT 3.51 is 100.

Backlogged requests are acknowledged by RSHD/NT and are processed when resources become available. Once the backlog limit is reached, RSHD/NT will begin to refuse connections until resources become available.

Default: 100