CHAPTER

35

PC Spawner Program

Starting the PC Spawner Program **471**
Examples of Starting and Connecting to the PC Spawner **477**
Ending the PC Spawner Program **477**

Starting the PC Spawner Program

The PC spawner program is stored on the remote host in the !*sasroot*\CONNECT\SASEXE directory. Start the spawner program from the !*sasroot* directory.

The syntax for the command to invoke the PC spawner program is

SPAWNER<-ADAPTER *n*>
<-AUTHSERVER *NT-domain-or-NT-server-name*> (Windows NT only)
<-COMAMID *access-method-id*>
<-DELETE> (Windows NT only)
<-DRIVER *device-driver*>(OS/2 only)
<-FILE>
<-FILEPROMPT>
<-HELP>
<-INHERITANCE>
<-INSTALL> (Windows NT only)
<-NETENCRALG *algorithm*>
<-NETENCRYPT YES | NO>
<-NETENCRKEY *n*>
<-NETMAC | -NONETMAC>
<-NETNAME *network-name*>
<-NOCLEARTEXT>
<-NOOPTION>
<-NOSCRIPT>
<-PATH *pathname*>
<-SECURITY | -NOSECURITY> (Windows NT only)
<-SERVICE *n*>

-ADAPTER *n*
specifies which adapter number the spawner should use when communicating through the NetBIOS API. The default is adapter 0. This option is useful if you are running NetBIOS and you want to use an adapter other than adapter 0.

Note: This option applies only to the NetBIOS access method. △

-AUTHSERVER *NT-domain-or-NT-server-name*
specifies the location of the user authentication database. Specify the name of either an NT domain or an NT server at which the database resides.

For Version 8, you are not limited to specifying a single NT domain by means of the -AUTHSERVER option. Instead, you may bypass this option and specify the domain name in the form *domain\username* when you supply your username to the Windows NT environment. Here are examples of how you might specify this information:

```
%let tcpsec=apex\bass.time2go  #Versions 6 and 7
signon user=apex\bass password=time2go; #Version 8
```

Domain name **apex** identifies the location of the user authentication database. Username **bass** and password **time2go** will be verified against those in the identified domain's username and password database.

Note: This option is valid for Windows NT only. △

-COMAMID *access-method-id*
specifies the access method to use. Valid values for -COMAMID are based on the operating system of the remote node that is running the spawner program and on the communications software that is installed on that node. If multiple connections will be made to this node with more than one type of access method, you can specify multiple -COMAMID options and values, as defined in Table 32.1 on page 457.

-DELETE
calls the Service Control Manager to remove the spawner service.

Note: Be sure to stop the spawner program service before you delete it. △

Note: This option is valid for Windows NT only. △

-DRIVER *device-driver*
specifies the driver that is used by the spawner program when communicating with the MNetBIOS access method. The default is to use the first driver listed in the name table. This option is needed only if you have more than one NetBIOS network device driver installed on the node that is running the spawner program.

Note: The Novell requestor for OS/2 does not support the -DRIVER option. △

Note: This option applies only to the MNetBIOS access method and is valid on OS/2 only. △

-FILE
specifies the file that invokes SAS when a request for a connection to a remote node is received. The default behavior is to try to invoke the file from the directory from which the spawner is invoked.

To invoke SAS from a directory that is not the default location, to specify different SAS invocation options, or to execute other statements before invoking SAS, use the -FILE or -FILEPROMPT option (the -FILEPROMPT option is discussed later).

For Windows NT and Windows 95, the following options are supplied by default when you invoke SAS:

```
-DMR -COMAMID access-method -NOLOGO -ICON
```

For OS/2, the following options are supplied by default when you invoke the spawner:

```
-DMR -COMAMID access-method -NOLOGO -ICON
```

For Windows NT or Windows 95, the alternate file will be a batch file that is signified by the .BAT extension. For OS/2, the alternate file will be a command file that is signified by the .CMD extension.

Your batch file must contain the following two lines:

```
cd \sas  /*or other path to the sas.exe file */
sas.exe %1 %2 %3 %4 %5 %6 %7 %8 %9  /* and any additional options */
```

The first line changes to the directory where the SAS executable is stored. The second line invokes SAS. Add options as needed at this SAS invocation.

With OS/2, you can use a REXX command file that contains statements in the following format:

```
/* Invoke remote SAS session */
parse arg parameters
'drive:\path\file-name.ext SAS-options'
     parameters
```

where

- The first line must be a comment line to indicate to the OS/2 command processor that the file is a REXX file.
- The second line parses the arguments that are passed to the REXX file.
- The third line specifies the location of the file that is to be invoked. Enclose the SAS options that you want in quotation marks on this line. You do not need to specify the -DMR option or the -COMAMID option; they are set automatically.

Note: Make sure that the options and parameters that are used in the second and third lines are identical. △

Example:

```
/*  */
parse arg parameters
'd:\sas\sas.exe -config config.sas'
      parameters
```

-FILEPROMPT

causes the spawner program to prompt the SAS/CONNECT user in the local SAS session for the name of the file to invoke in order to run the remote SAS session. The -FILEPROMPT option allows you to override the -FILE option on a per-user basis. To specify the file identified in the -FILE option, press the ENTER key. If the -FILE option is not specified, the spawner program uses the default filename SAS.EXE.

If the user supplies the name of an OS/2 command file, the file must be a REXX file that follows the guidelines described for the -FILE option.

Note: This option can be used only with the NetBIOS and SPX access methods. △

-HELP

prints a list of valid parameters.

-INHERITANCE

causes the SAS session that is spawned to inherit the socket that was created when the spawner accepted the initial connection from the local SAS session . This

option is useful if your configuration involves a firewall and you want to minimize the number of ports that you define to the firewall for use by SAS/CONNECT.

If you start a spawner with the -INHERITANCE option, you then define the port that the spawner is listening on to the firewall and map it to the server machine's port. This will enable any number of SAS/CONNECT clients to connect through this single port and SIGNON to a remote host on the inside of the firewall. Each client just opens a unique socket on the defined port. This eliminates the need to define an individual port for each client that may need to come in through the firewall. In this configuration you set your REMOTE= value to a two-level name where the first level is the name of the host running the firewall and the second level is the well-known service name of the port that you have enabled for connections.

-INSTALL

causes the spawner to install itself as a Windows NT service. After the spawner is installed, the spawner is started in secured mode (unless the -NOSECURITY option is specified).

Use the following syntax to install the spawner from the SAS root directory :

```
connect\sasexe\spawner -install -comamid access-method
```

For example:

```
C:\SAS> CONNECT\SASEXE\SPAWNER -I -C TCP
```

Note: The -INSTALL option is valid for Windows NT only. △

-NETENCRALG *algorithm*

If you specify more than one algorithm, enclose the algorithm names in parenthesis and use commas to separate the names. If there are embedded blanks in the algorithm name, enclose each algorithm in quotation marks.

Set this option at the remote host and, optionally, at the local host to specify one or more encryption algorithms to use in a SAS/CONNECT session. However, the local host and the remote host must share an encryption algorithm in common. If you specify the option in the remote host session only, the local host attempts to select an algorithm that was specified at the remote host. If you also set the option at the local host and specify an algorithm that is not specified at the remote host, the local host's attempt to connect to that remote host fails when the local host assigns a library.

Valid values for this option are

RC2

RC4

DES

TripleDES

SAS Proprietary.

See the *SAS/CONNECT User's Guide* or the *SAS/SHARE User's Guide* for more information about the -NETENCRALG option.

If the spawner is not installed as a service, use the -PATH option to point to the SECUREWIN\SASEXE subdirectory.

-NETENCRYPT

Set this option at both the local host and the remote host. At the remote host, this option specifies that encryption is required for each connection from a local host SAS session. At the local host, this option specifies that the local host must connect only to a remote host that supports encryption.

The default for this option is that encryption is used if the -NETENCRALG option is set and if both the local host and the remote host are capable of

encryption. If encryption algorithms were specified but either the local host or the remote host is incapable of encryption, then encryption will not be performed.

Encryption may not be supported at the local host or at the remote host for the following reasons:

- You are running a release of SAS (prior to Version 7) that does not support encryption.
- Your site has not purchased a SAS/SECURE license for a specific platform.
- You specified encryption algorithms in the local host and the remote host SAS sessions that are incompatible.
- You do not have a cryptographic service provider installed on your Windows system.

See the *SAS/CONNECT User's Guide* or the *SAS/SHARE User's Guide* for more information about the -NETENCRYPT option.

-NETENCRKEY *n*

You set this option in either the local host or the remote host SAS session. It specifies the key length to be used by the encryption algorithm.

Valid values for this option are

128	specifies 1024-bit RSA and 128-bit RC2 and RC4 key algorithms.
40	specifies 512-bit RSA and 40-bit RC2 and RC4 key algorithms.
0	no value is set. This is the default.

If you require extra security, then set the -NETENCRKEY option to 128. If you prefer to save CPU, then set the -NETENCRKEY option to 40.

By default, if you try to connect a host that is capable of only a 40-bit key length with a host that is capable of both a 40-bit and a 128-bit key length, then the connection is made using the lesser key length. If both hosts are capable of 128-bit key lengths, then a 128-bit key length is used.

See the *SAS/CONNECT User's Guide* or the *SAS/SHARE User's Guide* for more information about the -NETENCRKEY option.

-NETMAC | -NONETMAC

Set this option to control the use of Message Authentication Codes (MACs) on network communications. A Message Authentication Code is the equivalent of a checksum that is used to ensure that the original message has not been modified.

This option may be set at either the local host or the remote host. The default is -NETMAC.

See the *SAS/CONNECT User's Guide* or the *SAS/SHARE User's Guide* for more information about the -NETMAC option.

-NETNAME *network-name*

specifies the network name that NetBIOS or SPX can use for accessing the spawner program. This name is the value that the user specifies for the REMOTE= option on the local host. The name can contain up to eight characters. It is recommended that you use a name other than the node name of the host that is running the spawner because the operating system often uses the node name. The default is SAS$CONN.

At the remote host, start the spawner:

```
spawner -NETNAME rmthost
```

Note: These commands are issued in the local host session. △

Note: This option applies only to the NetBIOS and SPX access methods. △

-NOCLEARTEXT
prevents a sign on from a local host that does not support username and password encryption. Specifying this option prevents local host SAS sessions that are running releases prior to 6.09E and 6.11 TS040 from being able to sign on to the spawner program. The default action is to accept both encrypted and clear-text usernames and passwords, thereby allowing sign on from local host SAS sessions running releases prior to 6.09E and 6.11 TS040.

-NOOPTION
indicates that no SAS/CONNECT options will be used at SAS invocation.

-NOSCRIPT
prevents a sign on from a local host that uses a script. The spawner program has been enhanced to allow sign ons without scripts to a PC with the TCP/IP access method. This option requires that the user set security in the local SAS session prior to sign on. For details about setting security (for example, by means of the USER= and PASSWORD= options in an appropriate statement), see the chapter according to the applicable platform (Windows or OS/2) and the TCP/IP access method.

-NOSCRIPT specifies that SAS.EXE be executed from the directory from which the spawner is invoked unless the -FILE option has been used to specify an alternative location. In the local SAS session, use the SIGNON NOSCRIPT option in order to undefine the RLINK fileref option.

Note: This option applies only to the TCP/IP access method. △

-PATH *pathname*
specifies the location of the SAS system directory that contains the dynamic link library (dll) files needed to access the encryption algorithms. The -PATH option is used with -NETENCRALG.

-SECURITY | -NOSECURITY

CAUTION:
Windows NT Only This option is valid for Windows NT only. △

tells the spawner program whether or not to use the native Windows NT security subsystem. If the spawner is installed as a service, the -SECURITY option is set by default. When the -SECURITY option is specified, the local host must supply both a valid Windows NT username and a password to connect to the spawner program.

To run the spawner in a secured mode, Windows NT requires that the user invoking the spawner must have administrator privileges. All users connecting to the spawner must have the following rights:

- Act as part of the operating system
- Replace a process level token
- Log on as a batch job.

If you are using the SPX access method, which is supported in Version 6 only, set the SASSECUR option. If you are using the TCP/IP access method, set the TCPSEC option. See the applicable platform and access method chapter for information about these options. If you are using the NetBIOS access method, which is supported in Version 6 and later releases, set security by means of the USER= and PASSWORD= options in the appropriate statements or use the SASUSER and SASPASS variables. For the NetBIOS and SPX access methods, the spawner program checks the userid and the password that are assigned to the variables SASUSER and SASPASS, respectively. For more information, see “Setting Security for SAS/CONNECT and SAS/SHARE” on page 389.

-SERVICE *n*
specifies an alternate port for the spawner to "listen on." The default is TELNET port 23.

Note: This option applies to the TCP/IP access method only. △

Examples of Starting and Connecting to the PC Spawner

A typical example of how to invoke the PC spawner program follows:

```
C:\SAS> connect\sasexe\spawner -comamid tcp -comamid netbios
                               -netname sasrem -nocleartext
```

Note: When typing this command, do not use the New Line key to break the line. Instead, allow the command to wrap automatically to the subsequent line. △

In this example, two -COMAMID options are used to allow both the TCP/IP and the NetBIOS access methods to connect to the spawner program. When the NETBIOS access method is used, the -NETNAME SASREM should also be specified as the value of the REMOTE= option in the local SAS session. The -NOCLEARTEXT option specifies that the spawner will accept connections only from local hosts that support username and password encryption.

From a local host, the following statements make a connection to the spawner program with the NetBIOS access method. It uses the -NETNAME SASREM:

```
options comamid=netbios;
signon sasrem;
```

From a local host, the following statements make a connection to the spawner program that runs on the node REMNODE and uses the TCP/IP access method and the TCPWIN.SCR sample script file:

```
options comamid=tcp;
filename rlink '!sasroot\connect\saslink\tcpwin.scr';
signon remnode;
```

In order to sign on to a spawner program running on a nonstandard TELNET port (a port other than port 23), you need to create a macro variable that references the port number. For example, if the spawner is listening on port number 5000 and running on the remote machine HOST.CORP.COM, submit the following statements from the local host to sign on and make the connection to the PC spawner program:

```
%LET MYNODE=host.corp.com 5000;
OPTIONS REMOTE=MYNODE;
SIGNON;
```

Ending the PC Spawner Program

To end the spawner program, type CTRL-C or double-click on the top left corner of the Windows or the OS/2 window that is running the program.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *Communications Access Methods for SAS/CONNECT and SAS/SHARE Software, Version 8*, Cary, NC: SAS Institute Inc., 1999. pp. 643.

Communications Access Methods for SAS/CONNECT and SAS/SHARE Software, Version 8

ISBN 1–58025–479–9

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, September 1999