IBM SPSS Statistics Server Administrator's Guide

Note

Before using this information and the product it supports, read the information in “Notices” on page 57.

Product Information

This edition applies to version 28, release 0, modification 0 of IBM® SPSS® Statistics Server and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Chapter 1. Overview

The IBM SPSS Statistics server technology is a **distributed architecture**, and coupled with key data management optimizations, it supports scalable analysis. The technology is client/server based. It distributes client requests for resource-intensive operations to powerful server software. When the client and server work together like this, it is referred to as **distributed analysis mode**. Distributed analysis allows end users to perform analyses that their desktop computers cannot support.

For maximum flexibility, client applications that use the server technology can also be configured to run solely on the end user's desktop computer—this is referred to as **local analysis mode**. End users can easily switch modes.

Products and Operating Systems

The server technology supports the IBM SPSS Statistics client application, and the server software runs on several operating systems (see the installation instructions for specifics). You can install multiple versions of server software at your site, on the same server computer, or on different server computers.

Architecture

The server software has a two-tier, distributed architecture. It distributes software operations between the client and the server computers. Memory-intensive operations, such as accessing a large database or analyzing a large data file, are done on the server computer without downloading the data to the client computer.

Tier 1. The **client** application. It is installed and runs on the end user's desktop computer. The client application provides the graphical user interface to data access and analysis. It presents the results of the end user's analyses.

Tier 2. The **server** software. It is installed and runs on a networked server computer. The server software provides the framework necessary to handle multiple clients, the algorithms used in statistical analysis, and data access.

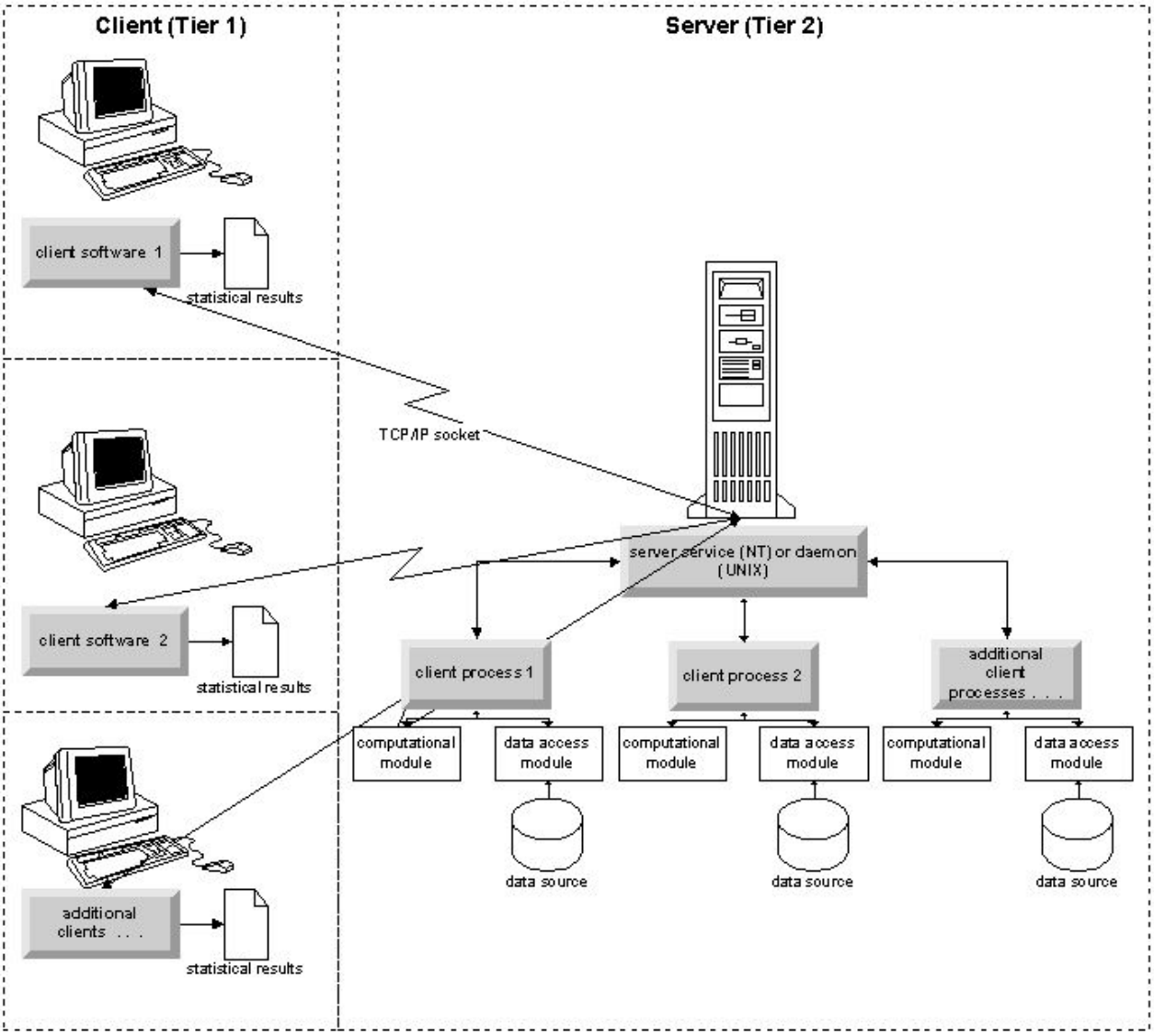

Figure 1. Distributed analysis mode

For analyses that don't require intensive data access or numeric processing, the client software can be used as a standard standalone desktop application. When in local analysis mode, all data access and statistical processing are handled on the end user's desktop computer.

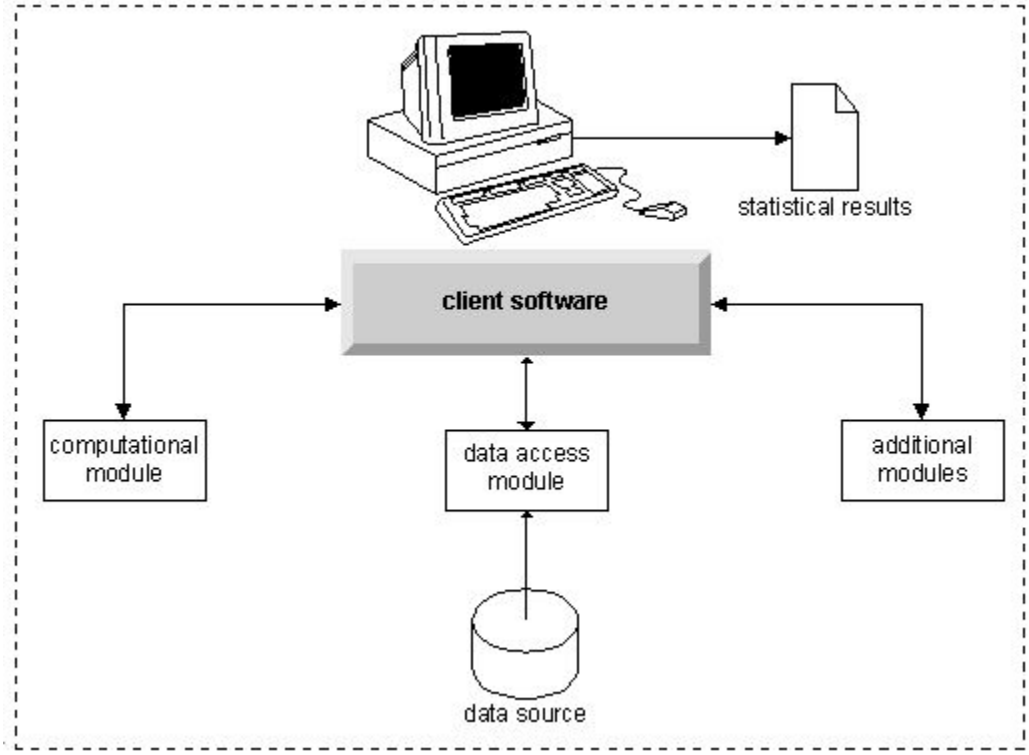

Figure 2. Local analysis mode

Software Components

As shown in the figure "Distributed analysis mode," the components of the server technology are the following. (See Figure 1 on page 2.)

Client software. The client application is a complete installation of the end user's IBM Corp. product (e.g., IBM SPSS Statistics). When it is used for distributed analysis, only the graphical user interface and the editing capabilities are used. When it is used for local analysis, its data access and statistical processing capabilities are also used.

Server software. The server software is made up of sub-components: a framework that manages client/server communication, client processes that manage client requests, and modules that access data and perform analyses.

- **Framework**. The framework of server software is a service (on Windows) or daemon (on UNIX). It handles all communication between the client application and the modules. The framework runs continually on the server computer, waiting for client connections. When a client connects, the framework launches a process that handles requests for that client.
- **Client process**. A client process is effectively a session for the client. There is one process for each client. A process is launched when the client connects, and it is terminated when the client disconnects. The process manages its client's requests for data and analyses. It loads the modules that are needed to access and analyze data. It unloads modules when they are no longer required.
- **Modules**. A module is an executable, DLL, or shared library that accesses data and runs analytic procedures. The analytical server software has several modules. Modules are loaded on demand. Modules may load other modules.

Using Distributed Mode

The following steps occur when an end user runs a product in distributed analysis mode:

1. **Launch the client application**. The end user launches the client software on his or her desktop computer. The client application presents a complete user interface.

2. **Connect to the server**. The end user connects to the server software by logging in from the client application. The server framework's service or daemon is always running, waiting for connection requests. When a connection is made, the server software launches a process to handle the end user.
3. **Access data.** The end user accesses data as usual from the client application, except that his or her view of database drivers, data files, directories, and drives represents the remote server computer, not the desktop computer. The server process loads the appropriate data access modules and retrieves the data. A small segment of the data is sent to the client application so that the end user can refer to it when selecting an analysis. Most of the data remain on the server. You can also configure the server software to prevent any data from being sent to the client application. You can use the administration application (IBM SPSS Statistics Administration Console, which is installed as part of IBM SPSS Deployment Manager) to prevent the data from being sent to all clients. See the topic Users in the *Deployment Manager User's Guide* (included in the help for IBM SPSS Collaboration and Deployment Services) for more information. You can also configure access for each user or group. See the topic IBM SPSS Statistics Server User Profiles and Groups, in the *Deployment Manager User's Guide*, for more information.
4. **Analyze the data**. Using the client application's user interface, the end user selects the data and requests the type of analysis he or she wants. The request is sent to the server process, which loads the appropriate data analysis packages and processes the analysis. All data-related tasks, such as reading data, transforming data, computing new variables, and calculating statistics, are done on the server computer.
5. **Review the results**. The server software sends the output from the client's request back to the client application. Only the results are sent, the data remain on the server. The end user can then use the client application to refine and edit the results.

Administering Server Software

This guide is intended primarily for system administrators who are responsible for integrating server technology into a networked environment in which client applications are run in distributed analysis mode. Administrative tasks include:

Installation. The server software is designed to run continuously and respond to logins and requests from end-user desktop computers. Select an appropriate server computer for the server software—one that has little downtime, is configured for end-user access, and is networked to the appropriate desktop computers. The more memory and processing power the server computer has, the faster client requests are handled. The client application must be installed on the end user desktop computers. Client installation can be done from a network location. Chapter 2, "Installation," on page 7 provides an overview of how to install the server software and the client application. Detailed installation instructions are included on the product DVD.

Data access. If you need to provide end users access to data on a remote server while they are working in distributed mode, the server software needs to be able to access that data. IBM Corp. products can access data from a variety of data file types, including databases. To make your job easier, IBM Corp. products are distributed with DataDirect Connect ODBC for accessing data from a database. Chapter 3, "Data Access," on page 9 introduces data access for IBM Corp. products. Additional documentation is included on the product DVD.

Configuration and maintenance. Because the server software is intended for continuous operation, it should be monitored at regular intervals by a system administrator. There are several configuration options that give you control over how the server software operates. Chapter 4, "Configuring, Monitoring Usage, and Maintenance," on page 13 discusses configuring and monitoring the server software.

Supporting end users. End users require information about server names, user accounts, and where to find data. You may also need to assist them in solving problems. Chapter 5, "Supporting End Users," on page 21 discusses the kind of support that end users require.

Performance. Chapter 6, "Analyzing and Improving Performance," on page 43 provides strategies for improving the performance of the server software.

Troubleshooting. "Server Software" on page 47 provides troubleshooting tips.

IBM SPSS Statistics Batch Facility (IBM SPSS Statistics Server only). The IBM SPSS Statistics Server product includes IBM SPSS Statistics Batch Facility, which is intended for automated production of statistical reports. If you are running IBM SPSS Statistics Server at your site, read Appendix B, "The IBM SPSS Statistics Batch Facility," on page 49, which describes the IBM SPSS Statistics Batch Facility and the tasks that you may need to perform to support it.

Using This Document

This guide is intended primarily for system administrators who are responsible for installing and maintaining the server software in a networked environment in which client applications are run in distributed analysis mode.

Chapter 2. Installation

Products that use the server technology are packaged on multiple media—one DVD for the server software and one DVD for the client application.

To deploy the server technology, you:

- Install the server software on a networked server computer.
- Install, or supervise the installation of, the client application so that it is accessible from end-user desktop computers.

This chapter provides an overview of the installation process. Detailed installation instructions are available on your product DVD in the */Documentation/<language>/InstallationDocuments* directory.

Refer to "Products and Operating Systems" on page 1 for a complete list of server products and their associated client applications.

Installing the Server Software

Install the server software on a networked server computer. The server computer must be running the appropriate version of the operating system. If possible, use a server computer configured for, and dedicated to, rapid numeric processing and data access. Additional processing power and memory enhance the server software's performance. Detailed hardware and software requirements, including operating system requirements, appear in the installation instructions.

Installing the server technology installs software that manages access to data and performs the computations required for statistical analysis. It also installs a service (on Windows) or daemon (on UNIX) that listens for incoming end-user login requests and launches a process to handle each end user.

To install the server software, follow the instructions in the */Documentation/<language>/InstallationDocuments* directory on the server DVD.

Installing the Client Application

Installing the client application installs software that handles the user interface and the presentation of results. You must install, or supervise the installation of, the client application on the desktop computer of each end user. The desktop computer must be running Windows and must meet minimum hardware and operating system requirements. Detailed requirements appear in the installation instructions, which are in the */Documentation/<language>/InstallationDocuments* directory on the client DVD.

Before reading any other installation documents, see `Getting Started with Installation and Licensing.pdf`.

Chapter 3. Data Access

If you want your end users to be able to access data on remote servers, including data from databases, you must plan, install, and configure data access. To do this, you need to understand how the application decides where to look for data. You also need to decide if you want to use the data access technology that offers DataDirect Connect ODBC. You can also use OLE DB data sources.

View of the Data

Before you begin planning data access for the end users, it is important to understand how the application decides what data are available to the end user. The view of the data that is presented to end users depends on how they are running the program—locally or in distributed mode.

Local analysis mode. In local analysis mode, in which all data access and processing occur on the end user's desktop computer, the view of data files, ODBC data sources, directories, and drives is from the perspective of the desktop computer—that is, when the end user tries to open a data file, he or she sees the data files, directories, and network drives on his or her desktop computer.

Distributed analysis mode. In distributed analysis mode, in which data access and processing occur on a remote server, the view of data files, ODBC data sources, directories, and drives is from the perspective of the server computer—that is, when the end user tries to open a data file, he or she sees the data files, directories, and mounted drives on the server computer.

Your job is to configure data access in either local analysis mode or distributed analysis mode, as required by the end user.

Data Access Technology

A brief description of Connect ODBC follows. For more information about how the data access technology works with IBM Corp. products and for links to detailed documentation for specific databases, see *IBM SPSS Data Access Pack Installation Instructions* in the */Documentation/<language>/InstallationDocuments* directory on the product DVD.

Connect ODBC

Connect ODBC is a comprehensive set of individual, database-specific drivers that use ODBC to deliver connectivity to all major data stores, from relational databases to flat-file data.

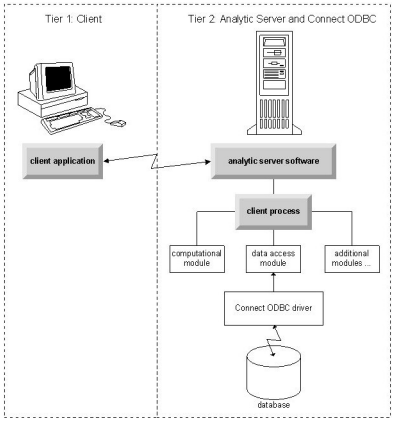

Figure 3. Connect ODBC in distributed analysis mode

Accessing Data

As you set up data access, consider the following:

Data access technology. Decide if you want to use one of the data access technologies distributed with your IBM Corp. product. See the topic "Data Access Technology" on page 9 for more information. A more detailed discussion of how to choose a technology appears in the *IBM SPSS Data Access Pack Installation Instructions*(in */Documentation/<language>/InstallationDocuments* on the product DVD).

Analysis mode. The end user's analysis mode determines what data he or she can access. See the topic "View of the Data" on page 9 for more information.

File system performance (Windows only). If most of your data are in an proprietary format from IBM Corp. (for example, *.sav* files) rather than in a database, we recommend that you store your data on a networked Windows NTFS drive for the best performance.

File format. The software handles opening and reading files in UNIX format automatically—you and your end users do not need to take any action to tell the software that a file is in UNIX format.

Referencing Data

Some client software allows the end user to save references to data and other files. These references must be written from the perspective of the computer that will access the data. For example, if the end user is running in local analysis mode, a reference to *C:\mydata\mydata.sav* causes the software to try to access the file on the local *C* drive of the *desktop computer*. If the end user is running in distributed analysis mode, the same reference to *C:\mydata\mydata.sav* causes the software to try to access the file on the local *C* drive of the *server computer*, possibly resulting in an error.

Windows. If you are administering a Windows system, you may decide to store data on the same computer as the server software. If you do, we recommend that users refer to the location of the data from the perspective of the server computer (for example, *C:\ServerData\mydata.sav*). Performance is faster because the network isn't used to locate the file. If your data are on another networked computer, we recommend that your users use UNC file references (for example, *\\mydataserver\ServerData\mydata.sav*). Note that UNC names can be used only when the referenced locations contain the name of a *shared resource* on the network. End users who frequently switch from distributed to local analysis mode are encouraged to use UNC file references because they work regardless of the mode.

UNIX. If you are administering a UNIX version of the server software, you may decide to put files on a UNIX server. End users can reference files on a UNIX server—tell them to use the full file specification and forward slashes (for example, */FILE = '/public/data/ourdata.txt'*.) Avoid using the backslash character in the UNIX directory and in filenames used with the server software.

Controlling Data Access

You can control access to data by using the operating system to set permissions by user IDs and groups. The end user connects to the server software by logging in from the client application. The server software uses the operating system to enforce the permissions for that user.

Note: Additional data security may be enforceable with your database software—the server software's data access modules prompt for IDs and passwords when the database requires them.

Windows. How you set access permissions on Windows depends on where the data are stored.

- If the files reside on a networked computer other then the computer running the server software, assign permissions to shared resources.
- If the files reside on the server computer, and on an NTFS drive, use security settings. You cannot control file access for data on the server computer on a FAT drive.

See "File Properties" on page 51 for information about setting sharing and security permissions on Windows.

UNIX. When the end user connects to the server software by logging in from the client application, the server software passes the user's login ID and password to the operating system and launches a process for the user. The launched process has the file access rights of the end user's login account.

Data Sources

ODBC

The IBM SPSS Statistics server software uses ODBC to access most data that are not in a proprietary format, including data that are stored in databases. ODBC requires an ODBC data source. An ODBC data source is the combination of:

- A descriptive name
- A specific driver
- A reference to a database or other type of data file

To access most data, you must configure, or help end users to configure, the ODBC data sources that they need.

The location of the configured data source is critical. It must be configured on the computer that is accessing and processing the data—so configure the ODBC data source on the *server computer* for distributed analysis and on the *desktop computer* for local analysis. For example, compare the location of the ODBC drivers in the figures shown in "Connect ODBC" on page 9 .

If you are just starting to use the data access technology (introduced in "Data Access Technology" on page 9), you need to accomplish some additional tasks before you can configure a data source. Refer to the *IBM SPSS Data Access Pack Installation Instructions* (in /Documentation/<language>/*InstallationDocuments* on the product DVD). Data source configuration is discussed again in Chapter 4, "Configuring, Monitoring Usage, and Maintenance," on page 13 of this guide.

Configuring the UNIX Environment for Data Access

For the data access technology to work on UNIX systems, the server software's startup script must be configured.

Open the Startup Script

1. Change to the */bin* subdirectory in the server software's installation directory. For example, at the UNIX prompt type:

   ```
   cd /usr/local/serverproduct/bin
   ```

   where */usr/local/serverproduct/bin* is the */bin* subdirectory of the directory in which the server software is installed.
2. Open *statsenv.sh* with a text editor.

Specify the DataDirect Script

1. Search for the first comment that contains the text:

   MERANT_ENVIRONMENT_SCRIPT
2. Find the line that defines the location of **odbc.sh**.
3. Edit the line so it contains the correct path to your Connect ODBC *client installation*, and remove the comment character if it has one. For example, change:

```
# MERANT_ENVIRONMENT_SCRIPT=/usr/slodbc50/5_01_00/odbc.sh
```

to:

```
MERANT_ENVIRONMENT_SCRIPT=/usr/myDataAccess/slodbc50/5_01_00/odbc.sh
```

Add odbc.ini Environment Variable

1. Add the following lines after the lines above to create an environment variable, ODBCINI, that allows IBM SPSS Statistics Server to find the *odbc.ini* file:

```
ODBCINI=ODBCDIR/odbc.ini
export ODBCINI
```

where ODBCDIR is replaced by the path to your Connect ODBC installation directory.

Add Paths to Database Libraries

1. Add lines appropriate for your database, usually the *database home directory* and, if you are *not* using the Data Direct Wire Protocol drivers, a *path to the database libraries*. For example, if you are using Oracle on Linux, add the following lines:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/bigdisk/oracle/product/8.1.6/lib
export LD_LIBRARY_PATH
ORACLE_HOME=/bigdisk/oracle/product/8.1.6
export ORACLE_HOME
```

where */bigdisk/oracle/product/8.1.6* is replaced by the path to your Oracle installation directory and LD_LIBRARY_PATH is the library path variable for your operating system.

Note that the Data Direct Wire Protocol drivers do not require the installation of database client libraries. However, other Data Direct drivers require these libraries.

Save the Startup Script

1. Save *statsenv.sh*.

Edit odbc.ini

1. Edit *odbc.ini*, the ODBC configuration file, so that ODBC data sources can be accessed from IBM SPSS Statistics Server. See the appendix "The UNIX Environment" in DataDirect's *DataDirect Connect ODBC Reference* (available if you installed the additional DataDirect documentation when you installed Connect ODBC) and the chapters for specific drivers in *odbchelp.pdf* in the *doc* subdirectory of your Connect ODBC installation directory.

The change will take effect the next time you start the server software.

Note: If you plan to use ODBC with the IBM SPSS Statistics Batch Facility, you need to modify the IBM SPSS Statistics Batch Facility startup script in the same manner.

Chapter 4. Configuring, Monitoring Usage, and Maintenance

After you install the server software, configure its environment by:

- Managing end-user accounts and files
- Configuring ODBC data sources
- Using the administration application (IBM SPSS Statistics Administration Console) to configure and monitor the server software
- Controlling service startup

These tasks are described in the following sections.

Managing End-User Accounts and Files

This section provides an overview of what you need to do to support end users at your site. See the topic Chapter 5, “Supporting End Users,” on page 21 for more information.

Accounts

End users need accounts in order to log in to the server software and access data. These accounts need to be authenticated and need to be able to read, write, and/or execute in specific folders on the server machine. For more information about file permissions, see the topic “Permissions” on page 28. For more information about authentication, see the topic “Authentication” on page 21.

Database permissions are enforced by the database software. Use your usual database administration tools to manage these accounts. If the database is restricted, the server software’s data access modules prompt the user to log in and pass that information back to the database for verification before accessing data.

Data Access

By default, each end user can see all the data when opening a file while connected to the server software. Displaying all the data can negatively impact performance and increase network traffic. You can choose to prevent the end user client computers from displaying the data by changing the global setting with the administration application (IBM SPSS Statistics Administration Console, which is installed as part of IBM SPSS Deployment Manager). See the topic Users in the *Deployment Manager User's Guide* (included in the help for IBM SPSS Collaboration and Deployment Services) for more information. You can also modify the user profile and groups settings to specify the data access for individual users or groups. See the topic IBM SPSS Statistics Server User Profiles and Groups, in the *Deployment Manager User's Guide*, for more information.

Files

Most files that end users need to save should be saved on the desktop computer; however, you may want to allow users to save data files on a networked computer. When the end user logs in to the analytic server software for the first time, the default directory for opening and saving files is the server software’s installation directory. Clearly this is not a location where you want users to write files, so set up a directory with write permission and distribute that location to the end users. Once they access that location from the user interface, the client application will store it, and it will become the default location for saved files.

Profiles

The server software also allows you to create profiles for users. A profile can specify the temporary directory, the UNIX umask setting, the CPU process priority, the client data access setting, and the maximum number of threads for each user or a group of users.

Configuring ODBC Data Sources

If your end users access data from databases while they are working in distributed analysis mode, you must configure ODBC data sources on the computer on which the server software is installed.

ODBC Data Sources and IBM SPSS Data Access Pack

If you are using the IBM Corp. data access technology, read Chapter 3, "Data Access," on page 9 in this document. Read the appropriate *Installation Instructions* for an the overview of configuring database access and links to detailed documentation on data access for specific databases (the document is in */Documentation/<language>/InstallationDocuments* on the product DVD).

Windows. Configure ODBC data sources using the ODBC Administrator. See the topic "ODBC Administrator" on page 52 for more information.

UNIX. Edit the startup environment script for the server software as described in "Configuring the UNIX Environment for Data Access" on page 11 and configure ODBC data sources using the *odbc.ini* file. See the topic "odbc.ini" on page 56 for more information.

Using a Third-Party Sort Engine

By default, the server software tries to use an external, third-party engine for sorting. To use the correct sorting engine, complete the following steps:

1. If the third-party sort engine is not installed on the server computer, install it. IBM Corp. does not provide the engines. You must purchase and license an engine from a third-party vendor.
2. Ensure the sort engine's library is on the system's execution or library path variable. On Windows, this is the PATH environment variable; on UNIX, this is LD_LIBRARY_PATH or LIBPATH, depending on the UNIX vendor. This step is required to allow the server software to load the third-party sort library.
3. Using the administration application (IBM SPSS Statistics Administration Console, which is installed as part of IBM SPSS Deployment Manager), set the Sort option to the appropriate third-party engine. See the topic Users in the *Deployment Manager User's Guide* (included in the help for IBM SPSS Collaboration and Deployment Services) for more information.

Any procedures that require sorting (e.g., SORT) will subsequently use the third-party sorting engine. Issuing the SET SORT=INTERNAL syntax command forces the server software to use the internal algorithm for sorting. An end user can also explicitly specify third-party sorting by issuing the SET SORT=EXTERNAL command. However, this is not necessary because third-party sorting is the default.

Checking the Current Sort Option

To check which sorting option is being used, you can issue the SET MESSAGES ON syntax command and run a SORT job. You can also use the SHOW SORT syntax command.

IBM SPSS Statistics Server Administration

The IBM SPSS Statistics Administration Console provides a user interface to monitor and configure your IBM SPSS Statistics Server installations. The IBM SPSS Statistics Administration Console is installed as part of IBM SPSS Deployment Manager. Complete documentation for the IBM SPSS Statistics Administration Console is included in the *Administration Consoles* section of the *Deployment Manager User's Guide* (included in the help for IBM SPSS Collaboration and Deployment Services).

Configuring the Production Facility Command Line Interface to Submit Jobs

Starting with IBM SPSS Statistics version 26, you can use the Production Facility command line interface to submit jobs to the SPSS Statistics Server. When the Production Facility command line interface is used in conjunction with the Microsoft Windows Task Scheduler/MacOS Automator for scheduling jobs, you can effectively replace IBM SPSS Collaboration and Deployment Services for processing SPSS Statistics jobs.

The SPSS Statistics **INSERT HIDDEN** command can execute jobs that generate output. When the command is used, users cannot access or view the source SPSS Statistics syntax.

Note: The **INSERT HIDDEN** command requires an SPSS Statistics Server. The command will not work on a standalone SPSS Statistics client machine.

INSERT HIDDEN feature

Administrators can enable the **INSERT HIDDEN** feature by using the SPSS Statistics Server Administration Console or by editing the `<install_path>/config/statisticsd.conf` file (**INSERT HIDDEN = Enabled**). The **INSERT HIDDEN Feature** field is located on the SPSS Statistics Server Administration Console's SPSS Statistics Server Configuration tab (under the **Users** section).

1. Select **Enabled** as the **INSERT HIDDEN Feature** value. Note that an asterisk(*) appears in the SPSS Statistics Server Configuration tab.
2. Save the change. Either click the **Save** icon on the toolbar, Control-S, or select **File** > **Save**.
3. Restart the SPSS Statistics Server.
4. After the server restarts on Windows servers, a dialog displays prompting the administrator to enter the **INSERT HIDDEN Feature** password. On Linux servers, the administrator must manually start the SPSS Statistics Server. When the server restarts, it prompts the administrator for the password.

   The password is stored in the system registry (similar to an SSL password) and all hidden syntax files are encrypted via the same password.

Optionally, the administrator can deny user access to the **INSERT HIDDEN** files via the file system access controls.

The SPSS Statistics Server daemon process requires read access to the **INSERT HIDDEN** files. An OMS command can be used to wrap syntax to ensure that output is not sent to the Output Viewer.

```
OMS /SELECT ALL /DESTINATION VIEWER=NO.
     *  commands executed here will not output to the viewer.
     DESC ALL.
     FREQ ALL.
OMSEND.
```

The **INSERT HIDDEN** syntax is similar to **INSERT FILE** syntax. For example:

```
INSERT HIDDEN
     SOURCE='source specification'
     [SYNTAX = {INTERACTIVE*}]
          {BATCH }
     [ERROR = {CONTINUE*}]
          {STOP }
     [ENCODING = 'encoding specification']
```

The **INSERT HIDDEN** file author provides the hidden file paths to the SPSS Statistics client users. During runtime, the client users execute **INSERT HIDDEN SOURCE="<file_path>"** syntax.

Refer to the *IBM SPSS Statistics Command Syntax Guide* for more detailed information.

The client process on the SPSS Statistics Server sends the **SOURCE** path to the server's daemon process. The daemon process decrypts the file and then returns it to the client process for execution.

The client process disables logs and journals, executes the decrypted file, and then re-enables logs and journals. The result is that the SPSS Statistics Output Viewer does not contain any source syntax logging (it does contain output). The journal also does not contain any source syntax.

Configuring Multiple Instances

You can create multiple instances of the server software, each with its own port number. This is often used in conjunction with group authorization to assign a group of users to a specific instance. However, multiple instances can be used independently of group authorization. For more information about group authorization, see “Group Authorization” on page 29.

Creating a New Instance

To create a group instance of the server software, you must run a script.

On Windows, run the following script from the server installation directory.

```
create_group_service <group_name> <port_number>
```

On UNIX and Linux, run the following script from the `bin` subdirectory of the installation directory.

```
create_group_configuration -group <group_name> -port <port_number>
```

`<group_name>` is a unique name for the instance, and `<port_number>` is the available port number that will be used by the instance.

After you run the script, there will be a configuration folder that is specific for the instance. Look for `config_<groupname>` in server installation direction. The folder contains several configuration files, such as `statisticsd.conf` and `UserSettings.xml`. When you want to update the configuration for a specific instance, be sure to update the configuration file in the correct location.

Starting the Server Instance

On Windows, the instance is a separate service that is named *IBM SPSS Statistics NN.m*, where *NN* is the major version number and *m* is the minor version number. You can start and stop this service like any other Windows services.

On Linux and UNIX, you need to specify the group name when running the startup script:

```
./start_statistics_server -d -g <group_name>
```

where `<group_name>` is the instance group name.

Deleting a Server Instance

1. On Windows, remove the service entry:
   a. Open a cmd prompt as Administrator.
   b. Run the following command:

      ```
      sc delete "IBM SPSS Statistics NN.m Server <group_name>"
      ```

      where *NN* is the major version number, *m* is the minor version number, and *<group_name>* is the instance group.
2. Delete the `config_<groupname>` subdirectory in the server installation directory.

Controlling Service Startup

The server software has a framework component that handles all communication between the client application and the modules. On Windows, the framework component is a service. On UNIX, the framework component is an application, usually run as a daemon.

Windows

By default, the service is configured for automatic startup, which means that it will restart automatically when the computer is rebooted. When started this way, the service runs unattended and the server computer can be logged off without affecting the service. You can use the Windows Services Control Panel to change the service startup parameters. See the topic "Services Control Panel" on page 52 for more information. If you are running multiple instances, the Services Panel will include an entry for each server instance.

Note: If the server computer does not support the localhost IP address (127.0.0.1/::1), then you must create a system environment variable that is named STATS_LH_OVERRIDE and set its value to YES before you start the server. For information on starting and stopping the server, see "Starting and Stopping the Server Software" on page 18.

UNIX

A startup script, *start_statistics_server*, is included in the */bin* subdirectory of the installation directory. The script calls *statsenv.sh* to configure the environment for the server software and then starts the application. You must start the server software with this script. The startup script must be run from the */bin* subdirectory. To execute it, you must be logged in as **root** if using the default unix authentication or the Pluggable Authentication Module (PAM) . Otherwise, you must be logged in as the user who owns the server software daemon. For more information about authentication, see the topic "Authentication" on page 21. The command:

```
./start_statistics_server -d
```

will start the server software as a daemon process, which is the recommended way to run the server software.

If you are running multiple instances of the server software, this command will start the default instance. To start another instance, specify the instance group with the -g switch:

```
./start_statistics_server -d -g <groupname>
```

where <groupname> is the instance group name.

Note: If the server computer does not support the localhost IP address (127.0.0.1/::1), then you must set the environment variable STATS_LH_OVERRIDE to YES before you start the server. STATS_LH_OVERRIDE is set from statsenv.sh, which is included in the /bin subdirectory of the installation directory. For information on starting and stopping the server, see "Starting and Stopping the Server Software" on page 18.

Startup Script Command Line Parameters

The *start_statistics_server* script accepts the following command line parameters (in any order):

- **Daemon**. Run the server software as a daemon process by optionally specifying -d. If you omit the -d, the server will start as a foreground process. For example, to start the server software as a daemon, use the command:

  ```
  ./start_statistics_server -d
  ```

- **Group**. If you are running multiple instances of the server software, specify the group that is associated with the instance:

  ```
  start_statistics_server -g <groupname>
  ```

  where

  <groupname> is the appropriate group name.

- **Port**. A port number can optionally be specified on the startup script command line. For example, to specify a port number, use the command:

  ```
  start_statistics_server -p nnnn
  ```

where

nnnn is the desired port number.

Specify a port number only if you need to resolve a port number conflict. The default will work unless another application on the computer is using the same number. This command line setting overrides the value set in the administration application.

Other Maintenance

Remove unneeded files. Periodically check the temporary file location and the log file location for unneeded files and remove them. The locations are defined with the administration application.

Check running processes. If you don't regularly reboot the server computer, periodically check the processes running on the computer and end any processes that are not in use. The process names are listed in "Starting and Stopping the Server Software" on page 18 .

Starting and Stopping the Server Software

The administration application will restart the server software for you so that configuration changes can be committed; however, at times you may need to start or stop the server software from the operating system. Follow the steps below for your operating system, using the process name of your server product. On Windows, the default service name is *IBM SPSS Statistics NN.m*, where *NN* is the major version number and *m* is the minor version number. If you are running multiple instances, the service name is *IBM SPSS Statistics NN.m <groupname>*, where *NN* is the major version number, *m* is the minor version number, and *<groupname>* is the instance's group. On UNIX and Linux, the daemon name is *statisticsd*.

Scheduling note: Stopping the service or daemon disconnects end users and terminates their processes, so try to schedule configuration and maintenance tasks for a time when you expect few users to access the system (for example, early morning or late evening).

To Start the Service or Daemon

Windows. Use the Windows Services Control Panel to start the service. See the topic "Services Control Panel" on page 52 for more information.

UNIX. Start the Server with the startup script, *start_statistics_server*, which is included in the */bin* subdirectory of the installation directory. The startup script must be run from the */bin* subdirectory. To execute it, you must be logged in as **root** if using the default unix authentication or the Pluggable Authentication Module (PAM) . Otherwise, you must be logged in as the user who owns the server software daemon. For more information about authentication, see the topic "Authentication" on page 21.

To Stop the Service or Daemon

Windows. Use the Windows Services Control Panel to stop the service. See the topic "Services Control Panel" on page 52 for more information.

UNIX. Kill the server process. (See "ps and kill" on page 55 for an example.) The daemon automatically creates a file (*statisticsd.pid*) that contains the process ID for the daemon. You can use this file in conjunction with the kill command by running the following from the `config` subdirectory of the installation directory or the `config_<group_name>` for another instance of the server software:

```
kill -9 `cat statisticsd.pid`
```

Platform independent. Use the administration application (IBM SPSS Statistics Administration Console, which is installed as part of IBM SPSS Deployment Manager). See the topic Controlling the IBM SPSS Statistics Server in the *Deployment Manager User's Guide* (included in the help for IBM SPSS Collaboration and Deployment Services) for more information.

Configuration for Improving Performance

Refer to Chapter 6, “Analyzing and Improving Performance,” on page 43 for information about modifying the configuration of the server software to improve performance.

Chapter 5. Supporting End Users

Supporting end users involves making sure that they have the information that they need to run their IBM Corp. product in distributed analysis mode. To use the server software, end users need to know:

- How to connect to the server software.
- How to access data and files.
- Where to save data and files.

Authentication

You have several options for authenticating users. Some options require that the server runs with root privileges.

Table 1. Authentication Options

Method	**OS Availability**	**Server Must Run as System/Root?**
Standard OS-level authentication (Windows or UNIX account)	• Windows • UNIX	Yes
Pluggable Authentication Module (PAM)	• UNIX	Yes
Internal Authentication	• Windows • UNIX	No
unix2	• UNIX	No
Single Sign-On	• Windows • UNIX	No

Configuring OS-level Authentication

OS-level authentication is the default authentication method. Use your usual system administration tools to create and manage standard OS-level end-user accounts (see “User Manager” on page 52 for information about how to access the Windows User Manager).

If you try another authentication method and want to revert to OS-level authentication, you need to update the *userauth* element in the `statisticsd.conf` file and change the *value* parameter to `unix` or `win32`.

Configuring PAM

The server software on UNIX can use the Pluggable Authentication Module (PAM) to authenticate users. You must first configure the server software to use PAM. Then you configure PAM by following the instructions specific to your UNIX vendor. Steps follow for Linux. These may vary, depending on version and vendor.

Note: If the server software is running, you need to restart it after completing all the steps.

Configure the Server Software to Use PAM

1. Log on to the UNIX machine as *root*.

2. In the *config* subdirectory in the server software installation directory, open the configuration file (e.g., *statisticsd.conf*) in a text editor.
3. Find the *userauth* element and change the *value* parameter from `unix` to `pam`.
4. Save the file.

Configure PAM on Linux

1. Change to the PAM configuration directory (e.g., */etc/pam.d*).
2. Use a text editor to create a file named *statisticsd*.
3. Add the PAM configuration information that you want to use. For example:

```
auth       include      system-auth
account    required     pam_nologin.so
account    include      system-auth
password   include      system-auth
session    optional     pam_keyinit.so force revoke
session    include      system-auth
session    required     pam_loginuid.so
```

   Note: These lines may vary depending on your particular configuration. Consult the Linux documentation for more information.
4. Save the file.

Configuring Internal Authentication

Internal authentication allows the server software to run without root privileges. However, it limits client connections to the same disk-access. Every user who connects to the server software has the same disk-access security. Therefore, one user can delete another user's file. If this is a concern, it is recommended that you use the unix2 authentication method instead. This method does not restrict client connections because it uses the UNIX *passwd* file for authentication. See the topic "Configuring unix2 Authentication" on page 23 for more information.

Warning: Do not use internal authentication when running the daemon/service as root/SYSTEM. Doing so is the same as giving root/SYSTEM access to your server to any user that connects.

Configuring Internal Authentication on UNIX

1. Create a group for users who will connect to the server software. We recommend naming this group **statistics**.
2. A member of this group must install the server software. This user will be the owner of the server software daemon.
3. Another member of this group (different from the daemon owner and typically the user who maintains the server software users) creates a *statisticsusers* file in the *config* directory in the server software installation directory. This file should have read/write access for the user who created it. It should have read access for the users group. No other users should be able to access it. If you don't create this file manually, it is automatically created the first time you run the `statisticsuser` command line tool (see the next step). The command line tool sets the appropriate permissions.
4. In the *config* directory, use the `statisticsuser` command line tool to add users. As the user who created the *statisticsusers* file, type `statisticsuser <username>` to create a regular user (e.g., `statisticsuser jdoe`). Use the `-a` option to create an admin user (e.g., `statisticsuser -a jdoe`). The `statisticsuser` command line tool prompts you for a password. An end user enters the user name and password to connect to the server software. Be sure to distribute the user name and passwords appropriately. To delete a user, use the `-d` option (e.g., `statisticsuser -d jdoe`).
5. Logged in as the owner of the server software daemon, open the configuration file (e.g., *statisticsd.conf*) in a text editor.
6. Find the *userauth* element and change the *value* parameter from `unix` to `internal`.
7. Logged in as the owner of the server software daemon, start the server.

Configuring Internal Authentication on Windows

1. Edit the IBM SPSS Statistics Server entry to run as a specific user:
   a. Open the Windows Services Panel and double click the entry for *IBM SPSS Statistics NN.m*, where *NN* is the major version number and *m* is the minor version number.
   b. Click the **Log On** tab.
   c. Under **Log on as**, select **This account**.
   d. Enter the domain\username and password of the user that will own the server process. This user will need the *Logon as a service* privilege.
2. The same user must create a `statisticsusers` file in the `config` directory in the server software installation directory. This file should have read/write access for the user who created it. No other users should have write access. If you don't create this file manually, it is automatically created the first time you run the `statisticsuser` command line tool (see the next step).
3. In the `config` directory, use the `statisticsuser` command line tool to add users. As the user who created the `statisticsusers` file, type `statisticsuser <username>` to create a regular user (e.g., `statisticsuser jdoe`). Use the `-a` option to create an admin user (e.g., `statisticsuser -a jdoe`). The `statisticsuser` command line tool prompts you for a password. An end user enters the user name and password to connect to the server software. Be sure to distribute the user name and passwords appropriately. To delete a user, use the `-d` option (e.g., `statisticsuser -d jdoe`).
4. Logged in as the owner of the server software daemon, open the configuration file (e.g., `statisticsd.conf`) in a text editor.
5. Find the *userauth* element and change the *value* parameter from `win32` to `internal`.
6. Go to the Windows Services Panel and start the service.

Configuring unix2 Authentication

unix2 authentication allows the server software to run without root privileges and authenticates against the UNIX *passwd* file with standard user accounts. An executable file (*suauth*) installed with the server software performs the authentication. For it to work correctly, you must set the necessary permissions.

To configure unix2 authentication, complete the following steps:

1. Using either setuid and setguid or role-based access control (RBAC), change the permissions of the *suauth* executable so that the user who will run the server software daemon has the necessary root permissions. This user needs to be able to authenticate the user against the *passwd* file and to change the user ID and group ID of the spawned server process for each end user. Details about setting permissions follow. Note that you use *either* setuid/setguid *or* RBAC. Do not use both methods.
2. Open the configuration file (e.g., *statisticsd.conf*) in a text editor.
3. Find the *userauth* element and change the *value* parameter from `unix` to `unix2`.
4. Logged in as the owner of the server software daemon, start the server.

Setting Permissions with setuid and setguid

1. Create a group for the user who will run the server software. We recommend naming this group **statistics**. We also recommend that you limit group membership to only the user who will run the server software daemon.
2. A member of this group must install the server software. This user will be the owner of the server software daemon.
3. Start a terminal session as *root*.
4. Change to the *bin* directory in the server software installation directory.
5. Change the owner of the *suauth* file to be *root*.

```
chown root suauth
```

6. Add the setuid and setgid bits to *suauth*. These bits allow the user in the installer group to execute the file and run temporarily as *root*. Root privileges are required for the reasons stated earlier in this topic.

```
chmod 6550 suauth
```

7. Exit as *root* and log in as the owner of the server software daemon.

Setting Permissions with Role-based Access Control

You should also be able to use role-based access control (RBAC) to set the necessary permissions. Refer to your vendor's RBAC documentation for information. You will need to do the following:

1. Create an authorization for the *suauth* executable.
2. Create a role for this authorization.
3. Assign the owner of the server software daemon to the role.
4. Configure the authorization to allow the following permissions:
   - Read the *passwd* file.
   - Change user ID.
   - Change group ID.

Configuring Single Sign-On (SSO)

You can use single sign-on to connect to a server that is running on any supported platform. You must first configure your server and client machines. Internal authentication allows the server software to run without root privileges.

If you are using single sign-on to connect to both IBM SPSS Statistics Server and IBM SPSS Collaboration and Deployment Services, you must connect to IBM SPSS Collaboration and Deployment Services before you connect to IBM SPSS Statistics Server.

To inter-operate with most modern, secure Active Directory installations, you must install the high-strength encryption pack for Java because the required encryption algorithms are not supported by default. You must install the pack for both client and server. An error message such as `Illegal key size` is displayed on the client when a server connection fails because the pack is not installed. See “Installing unlimited strength encryption” on page 36.

Note: Before you configure your server and client machines for single sign-on, you must make sure that the machines have access to the domain controller server.

Configuring the Server for Single Sign-On

Configure the Server on Windows

1. Ensure that the Windows server machine is a member of the Active Directory (AD) domain.
2. In the IBM SPSS Statistics Server installation location, locate the folder called `config`.
3. In the `config` folder, create a subfolder called `sso`.
4. In the `sso` folder, create a `krb5.conf` file. Instructions for how to create the `krb5.conf` file can be found at http://web.mit.edu/kerberos/krb5-current/doc/admin/conf_files/krb5_conf.html. An example of a `krb5.conf` file is given below:

```
[libdefaults]
     default_realm = STATISTICSSSO.COM
     dns_lookup_kdc = true
     dns_lookup_realm = true

[realms]
     STATISTICSSSO.COM = {
          kdc = statisticssso.com:88
          admin_server = statisticssso.com:749
          default_domain = STATISTICSSSO.COM
```

```
    }

[domain_realm]
    .statisticssso.com = STATISTICSSSO.COM
```

Configure the Server on UNIX

To configure Single Sign-On for UNIX server machines, you can add the UNIX machine to the Windows AD domain, then follow the instructions for configuring Single Sign-On on Windows. Alternatively, you can perform the following steps:

1. Create a domain user account for the UNIX machine.
2. Change the host name. If you are using RedHat Linux, open the `/etc/sysconfig/network` file and modify `HOSTNAME` to the form <*name*>.<*realm*>. This enables the AD to find the server credentials.
3. To enable the DNS server to find the UNIX machine, take one of the following steps:
   - Open the `%windows%/system32/drivers/etc/hosts` file and add the IP/host mapping, for example:

     ```
     192.168.1.102 test.statisticssso.com test
     ```

     Or
   - Add a new reverse lookup zone entry. This will add an IP/host mapping on the DNS server.

   If the DNS entry for the UNIX machine is not correct, you can manually add the reverse lookup entry on the DNS server.

Configuring the Client for Single Sign-On

The steps are common to all clients except the steps that are noted specifically for Windows.

1. Ensure that the local Windows machine that is running IBM SPSS Statistics is a member of the Active Directory (AD) domain.
2. Add the domain user as an administrator on the local machine.
3. Enable Windows to access the TGT session key:
   a. From the **Start** menu, click **Run**.
   b. Enter `regedit` and click **OK** to open the **Registry Editor**.
   c. Navigate to the following registry location:

      `My Computer\HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\Kerberos\Parameters`
   d. Right click the folder and select **New** > **DWORD**. The name of the new value should be `allowtgtsessionkey`.
   e. Set the value of `allowtgtsessionkey` to a hexadecimal value of 1, that is 0x0000001.
   f. Close the **Registry Editor**.
   g. Run `kinit.exe`, which can be found in <IBM SPSS Statistics *installation location*>\jre\bin.
4. In the `config` folder of the IBM SPSS Statistics installation location, create a folder called `sso`.
5. Copy the `krb5.conf` file from the server in to the `sso` folder.
6. Restart the client machine and the server machine.

Registering the Service Principle Name (SPN)

Each server instance must register a unique *service principal name (SPN)* to identity itself, and the client must specify the same SPN when it connects to the server.

An SPN for an instance of the server software has the form:

```
statisticsserver/<host>:<port>
```

For example:

```
statisticsserver/jdoemachine.ibm.com:3023
```

Note that the host name must be qualified with its DNS domain (`ibm.com` in this example), and the domain must map to the Kerberos realm.

The combination of host name and port number makes the SPN unique (because each instance on a given host must listen on a different port). And both client and server already have the host name and port number and so can construct the appropriate SPN for the instance. The additional configuration step required is to register the SPN in the Kerberos database.

Registering the SPN on Windows

If you are using Active Directory as your Kerberos implementation, use the `setspn` command to register the SPN. To run this command, the following conditions must be satisfied:

- You must be logged on to a domain controller
- You must run the command prompt with elevated privileges (run as administrator)
- You must be a member of the Domain Admins group (or have had the appropriate permission delegated to you by a domain administrator)

For more information, refer to the following articles:

- Setspn Command-Line Reference
- Delegating Authority to Modify SPNs

For the default instance, listening on the standard port (3023 for version 23, for example) and running under the Local System account, you must register the SPN against the server computer name. For example:

```
setspn -s statisticsserver/jdoemachine.spss.com:3023 jdoemachine
```

For each subsequent server instance, listening on a custom port (for example, 3099) and running under an arbitrary user account (for example, johndoe) with the option `userauth` set to `internal` (that is, using internal authentication), you must register the SPN against the service user account name:

```
setspn -s statisticsserver/jdoemachine.spss.com:3099 jdoe
```

Note that in this case (when the service account is other than Local System), registering the SPN is not sufficient to enable a client to connect. Additional configuration steps are described in the next section.

To see which SPNs are registered to the account `jdoe`:

```
setspn -l jdoe
```

Registering the SPN on UNIX

If you are using Active Directory as your Kerberos implementation, use the `setspn` command as described in the previous Windows section. This assumes you already created the computer or user account in the directory. Or you can experiment with `ktpass`, if desired (see Ktpass Command-Line Reference).

If you are using some other Kerberos implementation, then use your favorite Kerberos administration tool to add the service principal to the Kerberos database. To convert the SPN to a Kerberos principal you must append the name of the Kerberos realm. For example:

```
statisticsserver/jdoemachine.ibm.com:3023@ibm.com
```

Add this same principal and password to the server's keytab. The keytab must contain an entry for every instance running on the host.

Configuring SSO when Running as Non-Root/System

When the server service/daemon is running as an arbitrary user (not root on UNIX and not System on Windows), you need to register the service/daemon account. You need the SPN that you created earlier.

1. Create the directory <STATISTICSSERVER>\config\sso.
2. Copy the file krb5.conf from the client SSO directory to the server SSO directory you created in step 1.
3. Use the following command to create the file krb5.keytab in the server SSO directory:

   ```
   <STATISTICSSERVER>\jre\bin\ktab -a <spn>@<realm> -k krb5.keytab
   ```

   For example:

   ```
   "..\jre\bin\ktab.exe" -a statisticsserver/
   jdoemachine.ibm.com:3023@ibm.com
        -k krb5.keytab
   ```

This will prompt you for a password. The password you enter must be the password of the service account. So if the service account is jdoe, for example, you must enter the password for the user jdoe.

The service account itself is not mentioned in the keytab, but earlier you registered the SPN to that account using setspn. This means that the password for the service principal and the password for the service account are one and the same.

For each new server instance you create, you must register the SPN for that instance (using setspn) and create a keytab file. The keytab file should be copied to the config_<group_name>/sso subdirectory in the server installation directory.. The default instance does not need a keytab file.

To verify that an instance is included in the keytab:

```
ktab.exe -l -e -k krb5.keytab
```

You may see multiple entries for each principal with different encryption types, but this is normal.

Configuring Group Membership

If you are using group authorization, you can configure IBM SPSS Collaboration and Deployment Services to query an LDAP provider to determine the group to which an authenticated user belongs. For more information about group authorization, see the topic “Group Authorization” on page 29.

Then, for group lookup to work properly, you must configure your repository first to add an LDAP or Active Directory provider and then to enable SSO using that provider:

1. Start IBM SPSS Deployment Manager client and select **File** > **New** > **Administered Server Connection...** to create an administered server connection for your repository (if you do not have one already).
2. Log on to the administered server connection and expand the **Configuration** folder.
3. Right-click **Security Providers**, choose **New** > **Security provider definition...**, and enter the appropriate values. Click **Help** in the dialog for more information.
4. Expand the **Single Sign-On Providers** folder, right-click **Kerberos SSO Provider**, and select **Open**.

5. Click **Enable**, select your security provider, and then click **Save**. You do not have to fill in any other details here unless you want to use SSO (simply having the provider enabled is sufficient to allow the group lookup).

Important: For group lookup to work properly, the Kerberos provider you configure here must be the same as the provider you configured for IBM SPSS Statistics Server. In particular, they must be working within the same Kerberos realm. So if a user logs on to the server using SSO and it identifies him as `jdoe@ibm.com` (where `ibm.com` is the realm), it will expect the security provider in IBM SPSS Collaboration and Deployment Services to recognize that user principal name and return the corresponding group membership from the LDAP directory.

Configuring SSO for Data Sources

You can connect to databases from IBM SPSS Statistics using single sign-on. If you want to create a database connection using single sign-on, you must first use your ODBC management software to properly configure a data source and single sign-on token. Then when connecting to a database in IBM SPSS Statistics, IBM SPSS Statistics will use that same single sign-on token and the user will not be prompted to log on to the data source.

However, if the data source was not configured properly for single sign-on, IBM SPSS Statistics will prompt the user to log on to the data source. The user will still be able to access the data source after providing valid credentials.

For complete details about configuring ODBC data sources on your system with single sign-on enabled, see your database vendor documentation. Following is an example of the general steps that may be involved:

1. Configure your database so it can support Kerberos single sign-on.
2. On the server machine, create an ODBC data source and test it. The DSN connection should not require a user ID and password.
3. Connect to the server using single sign-on and begin using the ODBC data source created and validate in step 2.

Permissions

If you are not using internal authentication or group authorization with single sign-on, the server software launches a process for the end user, passing the user's ID and password to the operating system. The launched process has the file access rights of the end user's account. A user connecting to server software must log in with an account that has the following permissions:

- Read and execute permissions to the server's installation directory and its subdirectories
- Read, execute, and write permissions to the directory location for temporary files

For internal authentication and single sign-on, the connecting client user has the permissions that are assigned to the user who started the service/daemon.

You can use the administration application (IBM SPSS Statistics Administration Console, which is installed as part of IBM SPSS Deployment Manager) to change the default location of the temporary files. See the topic File Locations in the *Deployment Manager User's Guide* (included in the help for IBM SPSS Collaboration and Deployment Services) for more information. You can also change the location for individual users or groups. See the topic IBM SPSS Statistics Server User Profiles and Groups, in the *Deployment Manager User's Guide*, for more information.

Administrator-Level Permissions

By default, the administrator group for the server software is the administrators group for the machine on which the server software is running. You can change the administrator group for the server software by specifying it in the Admin Group text box in the administration application (IBM SPSS Statistics Administration Console, which is installed as part of IBM SPSS Deployment Manager). See the topic Users in the *Deployment Manager User's Guide* (included in the help for IBM SPSS Collaboration and

Deployment Services) for more information. If you are using internal authentication on UNIX, you can create administrators directly. See the topic "Configuring Internal Authentication" on page 22 for more information.

Group Authorization

You can configure the server software to support group authorization. A separate instance of the service/daemon is run for each authorized group of users.

Configuring Group Authorization

1. Create a server instance for each group. For more information about creating server instances, see "Configuring Multiple Instances" on page 16.
2. Create the groups in IBM SPSS Collaboration and Deployment Services and assign users to the groups.
3. Open the administration application and update the value of **Group Authorization Service URL** to the URL for IBM SPSS Collaboration and Deployment Services. Be sure to include the port number (for example, `http://myserver.mydomain.com:9080`).

Controlling DSN access by Group

Multi-factor authentication (MFA) requires that users can be restricted in the set of ODBC data source names (DSNs) that they are allowed to access according to their group membership.

1. Open the administration application and set **Restrict Database Access** to `Yes`.
2. In the **Permitted Database Sources** field, enter a list of semicolon (;) separated DSNs that are permitted for access (for example, `Fraud - Analytic;Fraud - Operational`).

When this restriction is enabled, it has the following results:

- When a user browses for data sources in the Database Wizard, instead of being presented with all the DSNs defined on the server system, the user will only see the subset of DSNs that is defined by the administration application. Note that the path may contain DSNs that are not defined on the server. Thesee are ignored, and the user will not see those names.
- If a user modifies `GET DATA /TYPE=ODBC` syntax that specifies a DSN that is not specified by the administration application, the syntax will not run and the user will be presented with an error similar to **Access denied to data source: <X>**.

Profiles

The server software provides the ability to create profiles of individual users and groups of users. These user profiles and groups allow you to define settings for specific users.

Client and Server Versions

Beginning with version 20.0.1, the client software does not have to be at the same release level as the server software to which it is connecting. For example, the 20.0.1 client can connect to the 21 server software, and the 21 client can connect to a 20.0.1 server. Note that you can also run multiple versions of the server software on a server computer.

Mixing release levels is allowed only to simplify upgrades. Release levels can be staggered during the upgrade period, and clients do not need to be upgraded simultaneously. However, it is not recommended to maintain this configuration for an extended time. If the server is newer than the client, the server may create output that cannot be read by the client. If the client is newer than the server, syntax submitted by the client may not be recognized by the server. Therefore, you should upgrade the client or server software as soon as possible, depending on which one lags the other.

When you distribute connection information to end users, keep in mind which version of the client software they are running and be sure that they have the connection information for a matching server version.

Connecting Users through a Firewall

If you use a **firewall** to keep your network secure from intruders, you can configure your firewall and the server software so that end users outside of the firewall can connect the client to the server software. Your firewall can use **NAT** (Network Address Translation), but it isn't required.

The typical scenario for connecting end users through a firewall that uses NAT is as follows:

1. The end user connects the client application to the server software using the **masqueraded IP** (the IP address that NAT presents to the outside world) and the server's port number. For example, the end user connects with IP `10.10.10.2` and port number `3016`.
2. The firewall allows the connection because it has been configured to accept connections from the masqueraded IP.
3. The firewall redirects the masqueraded IP to the server's real internal IP. It allows the connection because the port (for example, 3016) is enabled on the firewall.
4. The server spawns a process for the end user's client connection and assigns it a port number from the list in the system environment variable `STATISTICS_CLIENT_PORTS`. For example, the process communicates through port 3287.
5. The firewall allows the communication through that port (for example, 3287) because it is enabled on the firewall.

Configuring connections through a firewall

Introduction

IBM SPSS Statistics Server re-uses the same port number for client connections. This means that only two ports need to be open through the firewall: the SPSS Statistics daemon or listening port (defaults to 3028, although it can be altered if necessary) and the response port through which SPSS Statistics clients talk to their SPSS Statistics Server child processes.

Note: Port re-use is tied to the client reconnect capability. By default, client-reconnect is enabled for 100 seconds. When client-reconnect is enabled, SPSS Statistics Server has a 1:1 ratio of connected clients to open response ports. In order to enable port re-use, you must first disable client-reconnect. This is done by editing the **`reconnect-timeout`** setting in `<Statistics Install Path>/config/statisticsd.conf`. For example:

```
<reconnect-timeout desc="The timeout in minutes that the server uses to drop disconnected
clients (default: 100)." value="0"/>
```

In an environment with many possible client-server connections, you may want to configure more than one client response port. When a SPSS Statistics client is in the process of connecting to the SPSS Statistics Server, the client port is essentially locked and can be used by only one client until the connection process has concluded. The time to connect is in the range of 1 - 3 seconds (this time can vary depending on the system load). When a second or third SPSS Statistics client attempts to connect during this time, the clients are blocked until the client port becomes available. Opening multiple client port reduces the wait time when connecting in an environment where many users simultaneously initiate SPSS Statistics sessions.

Example

Assume five client ports are listed in the **`STATISTICS_CLIENT_PORTS`** system environment variable (ports 40001 - 40005) and there are four possible client. A user initiates a connection with the SPSS Statistics Server and first contact is made through the listening port (3028). The server spawns a child process and continues communication through the first available client port (4001). If port 40001 is not locked (because communication with another client has just started), the port will be re-used. If port 40001 is locked, communication moves on to the next port (4002), assuming it's not locked, and so on.

After all four clients are connected they will most likely all use the same port number (40001). There is a small chance that one or more clients will use port 40002, a smaller chance that a client will use port 40003, and an even smaller chance that one or more clients will use port 40004. There in no chance that a client will use port 40005 because there are only four clients, and the algorithm starts at the first available port number that is listed in STATISTICS_CLIENT_PORTS system environment variable.

There are two recommended methods for configuring SPSS Statistics Server connections through a firewall.

Configure the firewall to allow processes

Using your firewall software, ensure that the following processes are allowed to accept network connections.

statisticsproc.exe
The statisticsproc.exe process opens, closes, and re-uses the response ports (or ports that are defined in STATISTICS_CLIENT_PORTS).

statisticssrvr.exe (Microsoft Windows) or statisticsd (UNIX or Linux)
The process is the main Windows service, or UNIX/Linux daemon, and manages the listening port.

Providing access to the processes effectively allows any port that the process will use.

Note: The following conditions apply when the reconnection timeout value is greater than 0 and statisticsproc.exe is allowed to accept network connections:

- STATISTICS_CLIENT_PORTS is irrelevant, except for diagnostics. Any port can be used.
- There is no limit to the number of connections unless ports are defined in **STATISTICS_CLIENT_PORTS**. The number of defined **STATISTICS_CLIENT_PORTS** ports effectively limits the ports that SPSS Statistics Server will use.

Configure the firewall by manually opening ports

To manually configure the server software and the firewall, follow these steps:

1. Install the server software as usual. You need to know IP address of the computer on which the server is installed and the port number that the server software uses for communications. For example, install the server on 202.123.456.78 at listening port 3028.
2. Configure the system environment variable STATISTICS_CLIENT_PORTS by specifying at least one port number. The environment variable lists the ports that are used to continue client connections with the server (**RESPONSE** ports). If needed, you can specify a comma-delimited list and a range of ports (for example, 4001, 4002, 4003-4005).

   Important:

   - When setting the automatic reconnection timeout to a value greater than 0, STATISTICS_CLIENT_PORTS defines the maximum number of allowed concurrent client and server connections.
   - Do not list the **LISTEN** port (3028) in the STATISTICS_CLIENT_PORTS environment variable.

   Microsoft® Windows™
   Use the Windows System properties to create and configure the environment variable. See “System Properties” on page 51 for instructions.

   UNIX
   Edit the server software’s environment script, statsenv.sh, which is included in the /bin subdirectory of the installation directory. Define the port that can be used by the client processes that the server starts. For example, add the following lines:

   ```
   STATISTICS_CLIENT_PORTS=4001
   export STATISTICS_CLIENT_PORTS
   ```

3. When you use Network Address Translation (NAT), create and map IPs. Using your firewall software, create a masqueraded IP for external use and map it to the server's internal IP. For example, create a masquerade IP `10.10.10.2` and map it to `202.123.456.78`.
4. Using your firewall software, enable port numbers on the firewall:
   - The server's **`LISTEN`** port number. For example, enable port `3028`.
   - The port numbers that you specified in the `STATISTICS_CLIENT_PORTS` environment variable. For example, enable port `4001`.
5. Distribute connection information to the users who connect to the server software from outside the firewall.
   - If used, the masqueraded IP of the computer on which the server software is installed (do not distribute the server's internal IP). For example, distribute `10.10.10.2` as the server's IP.
   - Distribute the server software's port number as usual. For example, distribute `3028` as the server's **`LISTEN`** port number.

Connecting Users with PPTP

End users can connect a remote client computer to the analytic server software with **Point-to-Point Tunneling Protocol** (PPTP). PPTP is a networking protocol that supports multiprotocol virtual private networks (VPNs). It enables remote end users to access your network securely across the Internet.

To use PPTP connections:

1. **Configure a remote access server for PPTP.** Be sure to create enough IP addresses for the clients because the server software supports multiple client connections. Each client connection requires its own IP address.
2. **Configure the client desktop computer.** Use the Windows Network control panel to add a private network connection using PPTP. Enter an IP address that the remote access server will recognize as a PPTP connection.
3. **Enable the PPTP connection on the client desktop computer.** When end users want to connect to the server software from a remote location, they enable the PPTP connection and then use the client software to connect to the server as usual.

Using SSL to secure data transfer

Secure Sockets Layer (SSL) is a protocol for encrypting data transferred between two computers. SSL ensures that communication between the computers is secure. SSL can encrypt the authentication of a username/password and the contents of an exchange between a server and client.

How SSL works

SSL relies on the server's public and private keys, in addition to a public key certificate that binds the server's identity to its public key.

1. When a client connects to a server, the client authenticates the server with the public key certificate.
2. The client then generates a random number, encrypts the number with the server's public key, and sends the encrypted message back to the server.
3. The server decrypts the random number with its private key.
4. From the random number, both the server and client create the session keys used for encrypting and decrypting subsequent information.

The public key certificate is typically signed by a certificate authority. Certificate authorities, such as VeriSign and Thawte, are organizations that issue, authenticate, and manage security credentials contained in the public key certificates. Essentially, the certificate authority confirms the identity of the server. The certificate authority usually charges a monetary fee for a certificate, but self-signed certificates can also be generated.

Enabling SSL using GSKit

Securing client/server and server-server communications with GSKit

The main steps in securing client/server and server-server communications with SSL are:

1. Obtain and install the SSL certificate and keys.
2. Enable and configure a specified configuration file located in the IBM SPSS Statistics Server installation directory.

   Note: IBM SPSS Statistics Server supports the TLSv1.2 protocol. GSKit currently doesn't support any other versions.
3. If using encryption certificates with a strength greater than 2048 bits, install unlimited strength encryption on the client computers.
4. Instruct users to enable SSL when connecting to the server.

Note: Occasionally a server product acts as a client. An example is IBM SPSS Statistics Server connecting to the IBM SPSS Collaboration and Deployment Services Repository. In this case, IBM SPSS Statistics Server is the *client*.

Obtaining and installing SSL certificate and keys

The first steps you must follow to configure SSL support are:

1. Obtain an SSL certificate and key file. There are various ways you can do this:
   - Purchase them from a public certificate authority (such as VeriSign, Thawte, or Entrust). The public certificate authority (CA) signs the certificate to verify the server that uses it.
   - Obtain the key and certificate files from a third-party certificate authority. If this approach is taken, the third-party CA's `*.pfx` root certificate must be imported into the server's keystore file (explained below).
   - Generate the key and certificate files with an internal self-signed certificate authority. The steps to do this are:
     a. Prepare a key database. See the topic "Creating an SSL key database" on page 35 for more information.
     b. Create the self-signed certificate. See the topic "Creating a self-signed SSL certificate" on page 35 for more information.
2. For certificate authority (CA) or self-signed certificates, copy the `.kdb` and `.sth` files from step 1 into a directory to which the IBM SPSS Statistics Server has access and specify the path to that directory in the `statisticsd.conf` file. The `statisticsd.conf` file is located in `<Statistics Server installation directory>/config/`; for third-party certificates, copy the `.pfx` and `.sth` files from step 1.
3. Set the following parameters in the `statisticsd.conf` file:

   For certificate authority (CA) or a self-signed certificates:

   - `<gsk desc="0=GSKSSL Disabled; 1=GSKSSL Enabled" value="<value>"/>`, where `<value>` is either `0` or `1` which indicates whether to enable GSKit.
   - `<gsk-keystore desc="GSKSSL Key store database filename." value="<filename>.kdb"/>`, where `<filename>` is the name of the key database file.
   - `<gsk-keystore-stash desc="GSKSSL Key store stash filename." value="<filename>.sth"/>`, where `<filename>` is the name of the key database password stash file.
   - `<gsk-cert-label desc="GSKSSL certificate label." value=""/>`, where `<label>` is the label of your certificate.

   For third-party certificates:

- <gsk desc="0=GSKSSL Disabled; 1=GSKSSL Enabled" value="<value>"/>, where <value> is either 0 or 1 which indicates whether to enable GSKit.
- <gsk-keystore ="<*.pfx_file_location>", where <*.pfx_file_location> is the location and name of *.pfx root certificate file.
- <gsk-keystore-stash desc="GSKSSL Key store stash filename." value="<filename>.sth"/>, where <filename> is the name of the key database password stash file.
- <gsk-cert-label desc="GSKSSL certificate label." value=""/>, where <label> is the label of your certificate.

4. For third-party certificates:
   a. Extract the root.pem file from the *.pfx file, using the following GSK command as an example:

```
gsk8capicmd_64.exe -cert -extract -db C:\SSL\<certificate_name>.pfx -stashed -label
<cert-certificate_issuing_server.com> -target C:\SSL\root.pem
```

   b. Copy the root.pem to the C:\ProgramData\IBM\SPSS\certificates folder (Windows) or /Library/Application Support/IBM/SPSS/certificates (macOS) on the client.
   c. On the client, set the connection using the fully-qualified domain name (for example, cert-certificate_issuing_server.com)" in the **Server Name** field, and enable the **SSL** option.
5. For self-signed certificates install the certificate on client systems. For purchased public CA or third-party certificates, this step is not required. Ensure that access permissions deny casual browsing of the directory that contains the certificate. See the topic "Installing a self-signed SSL certificate" on page 35 for more information.

Configuring the environment to run GSKit

The GSKCapiCmd is a non-Java-based command-line tool, and Java™ does not need to be installed on your system to use this tool; it is located in the <Statistics Server installation directory>/bin folder. The process to configure your environment to run IBM Global Security Kit (GSKit) varies depending on the platform in use.

To configure for Linux/Unix, add the shared libraries directory <Statistics Server installation directory>/lib to your environment:

```
$export <Shared library path environment variable>=<Statistics_server_install_path>/lib:<Shared
library
path environment variable>
$export PATH=$PATH:<Statistics_server_install_path>/bin
```

The shared library path variable name depends on your platform:

- Linux uses the variable name: LD_LIBRARY_PATH

For example, to set the environment on Linux, use:

```
$export LD_LIBRARY_PATH=/opt/IBM/SPSS/StatisticsServer/25/lib:$LD_LIBRARY_PATH
$export PATH=$PATH:/opt/IBM/SPSS/StatisticsServer/25/bin
```

Account access to files

Ensure that you grant the correct permissions for the accounts that will access the SSL files:

1. For all accounts that are used by IBM SPSS Statistics for connection, grant read access to the SSL files.

   Note: This also applies to the *Log on as* user that is defined in the IBM SPSS Statistics Server service. On UNIX or Linux, it applies to the user you are starting the server as.
2. For Windows, it is not enough that the accounts are in the Administrators group and that permission is given to that Administrators group when User Access Control (UAC) is enabled. In addition you must take one of the following actions:

- Give the accounts permission separately.
- Create a new group, add the accounts into the new group, and give the group permission to access the SSL files.
- Disable UAC.

Creating an SSL key database

Use the GSKCapiCmd tool to create your key database. Before using the tool, you must configure your environment; see the topic "Configuring the environment to run GSKit" on page 34 for more information

To create the key database, run GSKit and enter the following command:

```
gsk<ver>capicmd[_64] -keydb -create -populate -db <filename>.kdb -pw <password> -stash
```

where `<ver>` is the GSKit version number, `<filename>` is the name you want to use for the key database file, and `<password>` is the password for the key database.

The `-stash` option creates a stash file at the same path as the key database, with a file extension of `.sth`. GSKit uses the stash file to obtain the password to the key database so that it doesn't have to be entered on the command line each time.

Note: You should use strong file system protection on the `.sth` file.

Creating a self-signed SSL certificate

To generate a self-signed certificate and store it in the key database, use the following command:

```
gsk<ver>capicmd[_64] -cert -create -db <filename>.kdb -stashed -dn
"CN=myserver,OU=mynetwork,O=mycompany,
C=mycountry" -label <label> -expire <Number of days certificate is valid>
```

where `<ver>` is the GSKit version number, `<filename>` is the name of the key database file, `<Number of days certificate is valid>` is the physical number of days that the certificate is valid, and `<label>` is a descriptive label to help you identify the file (for example, you could use a label such as: `myselfsigned`).

Installing a self-signed SSL certificate

For the client machines that connect to your server using SSL, you must distribute the public part of the certificate to the clients so that it can be stored in their key databases. To do this, perform the following steps:

Note: Skip this step if you are using a certificate that is signed by a certificate authority. If you are using a self-signed certificate, you need to copy the trusted certificate authority to the client computers. Be aware that a server computer may also act as a client. An example is IBM SPSS Statistics Server connecting to the IBM SPSS Collaboration and Deployment Services Repository. In this case, IBM SPSS Statistics Server is the client, and therefore you need to copy the certificate for the IBM SPSS Collaboration and Deployment Services Repository server to the IBM SPSS Statistics Server.

1. Extract the public part to a file using the following command:

   ```
   gsk<ver>capicmd[_64] -cert -extract -db <filename>.kdb -stashed -label <label> -target
   root.pem
   ```

2. Distribute `root.pem` to the clients. If you have multiple trusted certificate authorities, copy them into a single `root.pem` file. Trusted certificate authorities are text files, so you can copy and paste the certificate or certificates. Copy `root.pem` to the following location on the client computers. If you already copied a `root.pem` file to the client for another IBM product, append the trusted root certificate authority information from your authority to the existing `root.pem` file. By default, all IBM client products look in this location for trusted self-signed certificate files. If you would like to use another location, create an `SSL_CERT_DIR` environment variable and set the value of the variable to the location.

   - Windows 7 and higher: `C:\ProgramData\IBM\SPSS\certificates`

- Mac: `/Library/Application Support/IBM/SPSS/certificates`
- UNIX and Linux: `/opt/IBM/SPSS/certificates`

Configuring client certificates

When SPSS Statistics Server is configured to use an SSL connection, and you are using a self-signed certificate, you must copy and configure the trusted certificate authority to all client workstations.

An example would be when IBM SPSS Collaboration and Deployment Services submits a job to SPSS Statistics Server (which is SSL enabled). In this situation, IBM SPSS Collaboration and Deployment Services is the client. The trusted certificate authority (`root.pem` on the SPSS Statistics Server) should be copied to and configured on all IBM SPSS Collaboration and Deployment Services machines.

Configuring certificate files for IBM SPSS Collaboration and Deployment Services

IBM SPSS Collaboration and Deployment Services support can be deployed on a Web Application Server (for example, IBM WebSphere and RedHat JBoss EAP).

The first step in configuring certificate files for IBM SPSS Collaboration and Deployment Services support is to retrieve the SPSS Statistics Server `root.pem` file from your administrator.

The SSL certificate configuration process depends on which Web Application Server is employed.

IBM WebSphere and RedHat JBoss EAP

The following instruction apply to both IBM WebSphere and RedHat JBoss EAP.

RedHat JBoss EAP note: When the SPSS Statistics Server uses IBM GSKit SSL, you must use the IBM JDK when configuring IBM SPSS Collaboration and Deployment Services on JBoss EAP.

1. Distribute the `root.pem` file to the IBM SPSS Collaboration and Deployment Services Server machine. If you have multiple trusted certificate authorities, copy them into a single `root.pem` file (trusted certificate authorities are text files, so you can copy and paste the certificates) Copy `root.pem` to the following location on the IBM SPSS Collaboration and Deployment Services Server.

   If you have already copied a `root.pem` file to the client for another IBM product, append the trusted root certificate authority information from your authority to the existing `root.pem` file. Create an `SSL_CERT_DIR` environment variable and set the variable's value to the desired server location that contains the `root.pem` file.
2. Verify that IBM SPSS Collaboration and Deployment Services Server user adds the `SSL_CERT_DIR` environment variable.

   Note: The IBM SPSS Collaboration and Deployment Services Server must be restarted after adding the environment variable.

Installing unlimited strength encryption

The Java Runtime Environment shipped with the product has US export-strength encryption enabled. For enhanced security of your data, upgrading to unlimited-strength encryption is recommended.

1. Extract the unlimited jurisdiction policy files that are packaged in the compressed file. The compressed file contains a `US_export_policy.jar` file and a `local_policy.jar` file.
2. Replace the existing copies of *US_export_policy.jar* and *local_policy.jar* files with the two files that you downloaded and extracted.

Instructing users to enable SSL

When users connect to the server through a client product, they need to enable SSL in the dialog box for connecting to the server. Be sure to tell your users to select the appropriate check box.

Enabling SSL using OpenSSL

Securing client/server and server-server communications with OpenSSL

The main steps in securing client/server and server-server communications with SSL are:

1. Install OpenSSL on the server computer.
2. Obtain and install the SSL certificate and keys.
3. Enable and configure SSL in the server administration application (IBM SPSS Deployment Manager).

   Note: IBM SPSS Statistics Server supports the TLSv1 protocol. SSLv3 has proven to have a security vulnerability and should not be used.
4. If using encryption certificates with a strength greater than 2048 bits, install unlimited strength encryption on the client computers.
5. If using a self-signed certificate, copy the certificate on the client computer.
6. Instruct users to enable SSL when connecting to the server.

Note: Occasionally a server product acts as a client. An example is IBM SPSS Statistics Server connecting to the IBM SPSS Collaboration and Deployment Services Repository. In this case, IBM SPSS Statistics Server is the *client*.

Install OpenSSL

If OpenSSL is not already available on the server, you must install it.

1. Download OpenSSL from http://www.openssl.org/. Be sure to use the version of OpenSSL appropriate for the server version:

Table 2. OpenSSL versions

Server Product	Compatible OpenSSL Version
IBM SPSS Statistics 28	1.1.1f or later
IBM SPSS Statistics 27.0.1	1.1.1f or later **Note:** Support for the insecure SSLv3 protocol has been deprecated.
IBM SPSS Statistics 26-27	1.0.2 or later
IBM SPSS Statistics 24-25	1.0.1f or later
IBM SPSS Statistics 20-23	1.0.0
IBM SPSS Statistics 17-19 (*not* Linux® on System z®)	0.9.8 and its subversions (0.9.8a, 0.9.8b, and so on)
IBM SPSS Statistics 19 (Linux® on System z®)	1.0.0

2. Follow the instructions for installing and configuring the software. It is recommended to build OpenSSL yourself, with the following guidelines:

   Windows. OpenSSL should be built with DLLs (which are multithreaded by default).

   UNIX. OpenSSL should support multiple threads (which is not always by default) and shared libraries.
3. Make sure the OpenSSL modules are included on the system path.

   Note: If there is more than one version of the OpenSSL modules on the server computer, then copy the OpenSSL modules for the IBM SPSS Statistics Server to the directory where IBM SPSS Statistics Server is installed.

Obtaining and installing SSL certificate and keys

1. Obtain an SSL certificate and key file. There are two ways you can do this:
   - Purchase them from a public certificate authority (such as Comodo, Symantec, or GoDaddy). The public certificate authority signs the certificate to verify the server that uses it. This is the recommended method.
   - Generate the key and certificate files with an internal self-signed certificate authority. OpenSSL provides a certificate management tool for this purpose, or you can search the Internet for instructions on creating a self-signed SSL certificate.
2. Copy the certificate and key file or files to a local directory or directories on the server. The public and private keys can be stored in separate directories. They can also be stored in a single file. Ensure that the private key is not in a location that might be encountered during casual browsing of the file system.
3. Copy trusted certificate authority named *root.pem* to the following location on the server computer. If you would like to use another location, create an `SSL_CERT_DIR` environment variable and set the value of the variable to the location.

   Windows 7 and higher: `C:\ProgramData\IBM\SPSS\certificates`

   Mac: `/Library/Application Support/IBM/SPSS/certificates`

   UNIX and Linux: `/opt/IBM/SPSS/certificates`

Enable and configure SSL in IBM SPSS Deployment Manager

1. Start the server administration application (IBM SPSS Statistics Administration Console, which is installed as part of IBM SPSS Deployment Manager) and connect to the server.
2. On the configuration page, set **Secure Sockets Layer** to `Yes`.
3. In **SSL Public Key File**, specify the full path to the public key file.
4. In **SSL Private Key File**, specify the full path to the private key file.

   Note: If the public and private keys are stored in one file, specify the same file in **SSL Public Key File** and **SSL Private Key File**.
5. From the menus choose:

   File > **Save**
6. Restart the server service or daemon. When you restart, you will be prompted for the SSL password. On Windows, you can select **Remember this password** to store the password securely. This option eliminates the need to enter the password every time the server is started.

Installing unlimited strength encryption

The Java Runtime Environment shipped with the product has US export-strength encryption enabled. For enhanced security of your data, upgrading to unlimited-strength encryption is recommended.

1. Extract the unlimited jurisdiction policy files that are packaged in the compressed file. The compressed file contains a `US_export_policy.jar` file and a `local_policy.jar` file.
2. Replace the existing copies of *US_export_policy.jar* and *local_policy.jar* files with the two files that you downloaded and extracted.

Copying the certificate file to client computers

Note: Skip this step if you are using a certificate that is signed by a certificate authority.

If you are using a self-signed certificate, you need to copy the trusted certificate authority to the *client* computers. Be aware that a server computer may also act as a client. An example is IBM SPSS Statistics Server connecting to the IBM SPSS Collaboration and Deployment Services Repository. In this case, IBM

SPSS Statistics Server is the *client*, and therefore you need to copy the certificate for the IBM SPSS Collaboration and Deployment Services Repository server to the IBM SPSS Statistics Server.

1. Create a trusted certificate authority named *root.pem*. For example, if you were creating the trusted certificate authority with OpenSSL, use the `-out` switch to specify the output file as *root.pem*. If you have multiple trusted certificate authorities, copy them into a single *root.pem* file. Trusted certificate authorities are text files so you can copy and paste the certificate or certificates.
2. Copy *root.pem* to the following location on the client computers. If you already copied a *root.pem* file to the client for another IBM Corp. product, append the trusted root certificate authority information from your authority to the existing *root.pem* file. By default, all IBM Corp. client products look in this location for trusted self-signed certificate files. If you would like to use another location, create an `SSL_CERT_DIR` environment variable and set the value of the variable to the location.

   Windows 7 and higher: `C:\ProgramData\IBM\SPSS\certificates`

   Mac: `/Library/Application Support/IBM/SPSS/certificates`

   UNIX and Linux: `/opt/IBM/SPSS/certificates`

Instructing users to enable SSL

When users connect to the server through a client product, they need to enable SSL in the dialog box for connecting to the server. Be sure to tell your users to select the appropriate check box.

Setting a Locale

The server software and the client that connects to it must run in the same character set, encoding, and locale. The server software gets its locale from the client. By default, this is the client's *system* locale. However, the client can override the default to process data files in other locales. By overriding the default, the user instructs the server software to run in a specified locale without changing the client's system locale.

Syntax

The user overrides the default by using the `SET LOCALE` syntax command:

```
SET LOCALE="localeid"
```

`localeid` is a string that identifies the locale in which the server software will run. `SET LOCALE` writes a registry entry on the client machine. This entry persists so that the next time IBM SPSS Statistics is started on the client machine, IBM SPSS Statistics will run in that locale.

The naming convention for the locale ID can differ among platforms and vendors. Therefore, there is an XML file installed with the server that maps client locales to server locales. This file, *loclmap.xml*, is located in the server installation directory on Windows and the */bin* subdirectory on UNIX.

loclmap.xml

The root element in *loclmap.xml* is the following. The root element also identifies the schema location.

```
<locale-map xmlns="http://xml.spss.com/spss/mls"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xml.spss.com/spss/mls
http://xml.spss.com/spss/mls/locale-map-1.0.xsd">
```

The root element contains `<client-locale>` elements with a `name` attribute identifying the client locale. The `<client-locale>` elements contain one or more `<server-locale>` elements. Each `<server-locale>` element has a `name` attribute identifying a server locale that corresponds to the client locale. The server software translates the client locale ID into one that can be used on the server machine. It checks each server locale in order, until it finds one that is valid on the server machine.

None of the default server locales in *loclmap.xml* are Windows locales. Windows system locales are generally not needed because the server software first tries to use the same locale as the client's system locale. A Windows server should have the locale that matches the client locale. However, you can add Windows server locales to *loclmap.xml* if you need to substitute a different but similar Windows locale.

You can modify *loclmap.xml* as needed. Just be aware that your XML elements must validate against the schema.

Example

Following is an example of the contents of *loclmap.xml*:

```
<client-locale name="French">
    <server-locale name="fr_FR.cp1252"></server-locale>
    <server-locale name="fr_FR.IBM-1252@euro"></server-locale>
    <server-locale name="fr_FR.IBM-1252"></server-locale>
    <server-locale name="fr_FR.8859-15"></server-locale>
    <server-locale name="fr_FR.ISO8859-15"></server-locale>
    <server-locale name="fr_FR.iso885915@euro"></server-locale>
    <server-locale name="fr_FR@euro"></server-locale>
    <server-locale name="fr_FR"></server-locale>
    <server-locale name="fr"></server-locale>
    <server-locale name="fr_FR.iso88591"></server-locale>
    <server-locale name="fr_FR.ISO8859-1"></server-locale>
    <server-locale name="fr_FR.windows-1252"></server-locale>
    <server-locale name="fr_FR.utf8"></server-locale>
    <server-locale name="fr_FR.UTF-8"></server-locale>
    <server-locale name="French_France.1252"></server-locale>
</client-locale>
```

In this case, if the user issues SET LOCALE="French", the server software checks `fr_FR.cp1252` first. Consider the case of an AIX server. The `fr_FR.cp1252` locale does not work on AIX, so the server software continues checking until it reaches `fr_FR.windows-1252`, which does work on AIX.

Using a Server Locale

If the user issues SET LOCALE using a server locale ID not recognized on the client machine, the client machine uses *loclmap.xml* to find client locale ID associated with a server locale ID. It writes this locale ID to the registry. For example, if the user issues SET LOCALE="fr_FR.windows-1252", French is written to the registry. To see which entry in *loclmap.xml* applies to the client, you can run the SHOW LOCALE command in local mode.

Potential Issues

Be aware that using the SET LOCALE command can cause functional problems in some cases:

- The current variable names might not be legal in the new code page.
- Case-insensitive name matches might fail. The failure might occur because strings are converted to upper-case characters in case-insensitive name matches (for example, when comparing variable names). If the locale is incorrect, this conversion would change the character (for example, in the Central European code page, 1250).
- Some bytes could be interpreted incorrectly as lead bytes, and a problem might occur because of an unexpected trail byte.
- SET LOCALE does not change the client's system locale. Therefore, if the IBM SPSS Statistics locale associated with SET LOCALE is different from the client's system locale, there will be display problems in various places. In this situation, a user is also unable to use an Input Method Editor (IME) to enter national characters.
- The IBM SPSS Statistics locale, OLANG setting, and the encoding used for the data must be compatible. Otherwise, output might be unusable and unreadable.

Connecting to the Server Software

The end user connects to the server software by logging in from the client application. To login an end user needs the following information from you:

- **Computer name or IP address.** When ends users connect to the server software, they log in from the client application. To do that, they need to correctly specify the name of the computer running the server software. The server computer can be identified by an alphanumeric name (for example, `myserver`) or an IP address assigned to the server computer (for example, `202.123.456.78`)—whichever you prefer. If you configure the server and client desktop computers to use Secure Sockets Layer (SSL), the end user must use a fully-qualified domain name (e.g., `myserver.mycompany.com`).
- **Port number.** End users need to correctly specify the port on which the server software is listening for connections. The port number is the default for the server, or whatever you specified when you configured the server software.
- **Domain name (Windows only).** End users may also need to specify a domain name. A domain name is required only when the server computer is in a different domain than the end-user desktop computers.
- **User ID and password.** End users are required to log in to the server computer. To do this, the users need a valid account, with appropriate permissions, for the computer on which the server software is running.
- **Secure Socket Layer (SSL).** If you use SSL to encrypt the communications that occur when end users to connect to the server software, tell the users to enable SSL when they set up the server connection. The clients do not need to know which SSL protocol is being used by the server. The client software will try both and use the one that works.

Accessing Data and Files

When end users connect to the analytic sever software, their view of data sources and files is from the perspective of the server computer, not their desktop computers.

- **ODBC data sources.** If your end users need access to ODBC data sources defined on the server computer, distribute the names, descriptions, and login information for those data sources. See Chapter 3, “Data Access,” on page 9 for a discussion of database access from the server software.
- **File access.** Distribute the names and locations of the files on the server computer that you want end users to access. See the topic “Referencing Data” on page 10 for more information.

Saving Data and Files

When end users save files while they are connected to the server software, the default location for the save is the directory from which the file was opened. In many cases, this is the local desktop computer; however for data files, it will often be a write-protected location on the server computer. Tell users where to save data files. Typically, the location is the user’s home directory somewhere on your network.

UNIX note: Tell end users to use the full file specification and forward slashes when saving files (for example, */public/myhome/myserverdata/data.sav*). Avoid using the backslash character in the UNIX directory and filenames used with the server software.

Chapter 6. Analyzing and Improving Performance

If you need to improve the performance of the server software, refer to this chapter for various strategies, ranging from configuration changes to hardware upgrades. Before making these changes, obtain performance information so you know which areas are problematic.

We also provide a white paper that includes additional information about improving performance. Go to http://www.ibm.com/developerworks/spssdevcentral and look for the link to "Books and Articles."

Obtaining Performance Information

To check performance, compare usage in the following areas when the server is not being used to when it is being heavily used.

- Disk usage
- CPU usage
- Memory usage
- Network usage

Logging

The administration application (IBM SPSS Statistics Administration Console, which is installed as part of IBM SPSS Deployment Manager) allows you to configure the server software to log performance information. Using the **Performance Log Interval** node, you can specify how often the server software writes performance information to the log. See the topic Logging in the *Deployment Manager User's Guide* (included in the help for IBM SPSS Collaboration and Deployment Services) for more information. You can also get performance information directly from the operating system.

Getting Performance Information on Windows

On Windows, you can obtain performance information by using the Performance Monitor.

Table 3. Windows Performance Information

Area	**Windows Performance Monitor Object**	**Useful Counter**
Disk Usage	PhysicalDisk	% Idle Time
CPU Usage	Processor	% Processor Time
Memory Usage	Memory	Committed Bytes
Network Usage	Network Interface	Bytes Total/sec for each physical interface instance (not the MS TSP loopback interface)

Getting Performance Information on UNIX

On UNIX, there are various commands for obtaining performance information, depending on the vendor.

Table 4. UNIX Performance Information

Area	**Vendor**	**Command**	**Note**
Disk Usage	Linux	iostat -x	Check the **%util** column.

Table 4. UNIX Performance Information (continued)			
Area	**Vendor**	**Command**	**Note**
Disk Usage	AIX	iostat -d	Check the **% tm_actl** column.
CPU Usage	Linux	top	Use the P interactive command to sort by CPU usage.
CPU Usage	AIX	ps aux	Pipe to the `sort` command to sort by the **%CPU** column.
CPU Usage	All	uptime	Check the load average.
Memory Usage	Linux	top	Use the `M` interactive command to sort by CPU usage.
Memory Usage	AIX	ps aux	Pipe to the `sort` command to sort by the **RSS** column.
Network Usage	All	netsat	Use the `-i` and `-s` switches for information.

Next Step

After gathering this information, you should be able to identify the area or areas that are problematic. The following sections describe possible solutions and recommendations for each area.

Improving Disk Usage

Consider the following for improving disk usage.

Space. Allow enough disk space. Each user typically needs temporary disk space equal to twice the size of the data (SAV) file in use (the space needed ranges from 1 to 2.5 times). A user sorting a file might need temporary space more than three times the size of the file. For example, if six concurrent users are accessing a file and two are sorting at once, they might need as much as 17 times the size of the file. In practice, they will not be at peak usage simultaneously, so 12 times the size of the file would be sufficient.

Hardware. Use SCSI disks for fastest perfomance. Do not use IDE.

System configuration. Keep temporary files on a separate spindle. You can also define multiple temporary files locations using the administration application. Make sure each location is on a separate spindle. If you use RAID, use RAID0 for the temporary file spindle. The scratch files speed gained from RAID0 is preferred over the redundancy gained from RAID1. If your CPU is not a problem and the server computer runs Windows, you can also compress the data directory or data files on disk. Do not allocate more virtual memory.

IBM SPSS Statistics configuration. If memory is not a problem but disk usage is, increase the workspace in IBM SPSS Statistics for faster performance. Try setting it by dividing the amount of RAM on the server computer by the expected number of concurrent users. For example, if the server computer has 1 GB of RAM, set the workspace to 0.25 GB RAM.

Temporary files directory. Modify the user profile or groups settings so that the temporary files directories for each user are located on different physical drives.

Cache compression. If your users are consistently working with large data files (especially if the size of the files is greater than half of the server's RAM), try enabling cache compression in the administration application.

Improving CPU Usage

Consider the following for improving CPU usage:

Number. Add more processors. If you want to approach the speed that a user would experience when running IBM SPSS Statistics locally, try to have one processor for every two concurrent users. Also use processors that are as fast as, or faster than, the processor on the desktop computer. For example, if you expect an average of four concurrent users, configure the server computer with two fast processors.

Hardware. Use fast processors. Adding a few really fast processors is better than adding many slow ones. If CPU usage is still a problem with fast processors, consider adding more server computers to your system.

SAV file locations and access. If certain files are used often by many concurrent users, consider moving the files across multiple servers to balance the user load. For example, if *TestScores.sav* and *GPA.sav* are both heavily used, put them on separate servers. Control access to the files with the operating system permissions (per group or per user) instead of controlling access through server accounts.

CPU priority. If certain users need a higher CPU priority than other users (for example, users who run quick jobs versus those who run long jobs), modify the user profile or group settings.

Cache compression. Cache compression has some CPU overhead for the compression and decompression of scratch files. If your users are not working with large data files, you may want to consider turning it off.

Improving Memory Usage

Consider the following for improving memory usage:

Amount. Add as much RAM as possible. Try to have 128 MB of RAM for each concurrent user. So, if there are four concurrent users, configure the server with 512 MB of RAM.

IBM SPSS Statistics configuration. Decrease the workspace in IBM SPSS Statistics.

Improving Network Usage

Consider the following for improving network usage:

System configuration. Schedule network-intensive operations for times when the server software is not in use (for example, run system backups overnight). If you identify a problem with network traffic on a computer on which the server is running, IBM Corp. will work with you to diagnose the problem further.

Using IBM SPSS Statistics Efficiently

In addition to focusing on specific problematic areas, you can also improve performance by adhering to the following guidelines for using IBM SPSS Statistics efficiently.

Data management. If you have large data files that require regular updating and are shared by users, consider doing the updates once and then releasing the files to the users for analysis. For example, if you regularly add monthly data to a file, sort it, and perform transformations, designate one person to run the job on the file. The other users can get the data they need without having to repeat the merge, sort, and transformations.

Interactive vs. batch. If you have regular, time-consuming operations that you do with IBM SPSS Statistics, consider running them from the IBM SPSS Statistics Batch Facility rather than from a client connected to the server. Use the client for building the reports, and run them from the IBM SPSS Statistics Batch Facility after the reports are ready.

Appendix A. Troubleshooting

Server Software

Port number conflict. If there is a port number conflict, the server software may fail to start. Correct the problem by using the administration application (IBM SPSS Statistics Administration Console, which is installed as part of IBM SPSS Deployment Manager) to change the port number. See the topic Connections in the *Deployment Manager User's Guide* (included in the help for IBM SPSS Collaboration and Deployment Services) for more information. Be sure to distribute the new port number to the end users.

Erratic behavior. The server software may behave erratically if its configuration file (for example, *statisticsd.conf)* is corrupted or missing. To correct the problem, restore the configuration file from your backup copy. Copy it to the location specified in the administration application or the configuration file environment variable and restart the server software. For information about restarting, see "Starting and Stopping the Server Software" on page 18 in .

Administration application doesn't work (UNIX only). If you use the administration application to control or configure the server software and it doesn't work (for example, you cannot stop the server), it may be because you did not start the server software with the startup script provided by IBM Corp.. Correct the problem by starting the server software with the *start_statistics_server* startup script. See the topic "To Stop the Service or Daemon" on page 18 for more information. If you get an error message when you attempt to control or configure the server software, it may be because you have connected with an account that does not have administrator permissions.

Can't change the location of temporary files (UNIX only). If you use the administration application to change the location of temporary files and the change is not effective, it may be because the new location doesn't have sufficient file permissions for the end users. Choose a location that has **read**, **write**, and **execute** access for all of the users who will connect to the server software.

Server won't start (UNIX only). If the server software will not start, it may be because you do not have the required operating system patches. To correct the problem, download and install the appropriate patch. The required patches are listed in the UNIX installation instructions for your server product.

Client Software

End user cannot connect to the server. The user may not have adequate permissions, or the firewall may be blocking the server software. For information about user permissions, see "Permissions" on page 28 . For information about configuring the firewall, see "Configuring connections through a firewall" on page 30 .

End user login fails with the "specified remote server computer was not found" message. The service or daemon may not be running. Confirm this by checking the status of the server software. To correct the problem, restart the service or daemon. See the topic "Starting and Stopping the Server Software" on page 18 for more information.

End user login fails with the "error connecting to package" message. The end user has specified the name or IP address of a server computer that isn't on the network. To correct the problem, ask the end user to enter a valid server name.

DataDirect ODBC data source fails with the "not licensed" message. The DataDirect data access technology is distributed with IBM Corp. products. It works only with newer IBM Corp. products—it doesn't work with earlier versions, nor does it work with non-IBM Corp. applications. If end users attempt to use DataDirect data sources with an older or unlicensed product, they will get a message containing the text **You are not licensed to use the DataDirect ODBC Driver**. To correct the problem with the IBM Corp. product, upgrade your users to a current version. To correct the problem with unlicensed products, upgrade your licensing with DataDirect or ask end users not to attempt to use the data sources that you have defined for IBM Corp. products with unlicensed applications.

End user can’t find a data file or ODBC data source. When end users are running in distributed analysis mode, they will have access only to data files and ODBC data sources on the computer that is running the server software. When end users are running in local analysis mode, they will have access only to data files and ODBC data sources on their desktop computers. To correct the problem, ask the end user to run the client application in the appropriate mode.

End user can’t run a statistical procedure (IBM SPSS Statistics Server only). When end users are connected to the server software, they have access only to the IBM SPSS Statistics options that were installed during the IBM SPSS Statistics Server installation. To correct the problem, ask the end user to run the procedure while in local analysis mode or install the requested procedure on the server computer.

Appendix B. The IBM SPSS Statistics Batch Facility

Note: The IBM SPSS Statistics Batch Facility is a batch processing utility included with **IBM SPSS Statistics Server**.

Typically the client for IBM SPSS Statistics Server is IBM SPSS Statistics running on a desktop computer. However, the IBM SPSS Statistics Batch Facility is an alternative way to use the power of IBM SPSS Statistics Server, and it runs on the server computer. The IBM SPSS Statistics Batch Facility is intended for **automated production** of statistical reports. Automated production provides the ability to run analyses without user intervention. Automated production is advantageous if users at your site regularly require a set of time-consuming analyses, such as weekly reports.

The IBM SPSS Statistics Batch Facility takes as its input a report request contained in a **command syntax** file. The IBM SPSS Statistics Batch Facility then automatically produces the statistical reports specified by the syntax.

What You Need to Know

Operating systems. The IBM SPSS Statistics Batch Facility is currently available with all IBM SPSS Statistics Servers, UNIX and Windows.

Installation. The IBM SPSS Statistics Batch Facility is automatically installed in the IBM SPSS Statistics Server installation directory on Windows and the */bin* subdirectory of the installation directory on UNIX.

Invoking. The IBM SPSS Statistics Batch Facility is run from the command line using the *statisticsb* executable file. It runs independently of IBM SPSS Statistics Server—IBM SPSS Statistics Server does not have to be started for it to run. It can also be run concurrently with IBM SPSS Statistics Server.

Modes of operation. Commands are submitted tothe IBM SPSS Statistics Batch Facility in either **batch** or **interactive mode**. In batch mode, the analyst or IT professional submits a command syntax file to the IBM SPSS Statistics Batch Facility for execution—the commands in the file are read and acted upon as a batch, and output is directed to a file. The IBM SPSS Statistics Batch Facility runs unattended and terminates after executing the last command. This is the typical way to use the IBM SPSS Statistics Batch Facility. In interactive mode, the analyst types commands one at a time at a command prompt. The commands are executed immediately, and output is displayed in the window. The IBM SPSS Statistics Batch Facility waits for the next command.

Documentation. The user's guide, written for the analysts and IT professionals at a site who will be using the IBM SPSS Statistics Batch Facility, is on the IBM SPSS Statistics Server DVD in */Documentation/<language>/Manuals*. The command syntax reference guide that analysts will need in order to create command syntax files for the IBM SPSS Statistics Batch Facility is on the IBM SPSS Statistics Server DVD in */Documentation/<language>/Manuals*. The IBM SPSS Statistics Batch Facility for UNIX is also distributed with a manual page, *statisticsb.1*, which is in the */bin* subdirectory of the IBM SPSS Statistics Server installation directory. If you are administering a UNIX system, copy it to the location where you keep your manual pages.

Additional documentation. The IBM SPSS Statistics Batch Facility user's guide contains sufficient information for an analyst who is experienced with the IBM SPSS Statistics command syntax language to build command syntax files for the IBM SPSS Statistics Batch Facility. If the analysts at your site are new to IBM SPSS Statistics, they may require additional documentation. If they do, direct them to our Web site at http://www.ibm.com/software/analytics/spss/, or ask them to contact your sales representative.

Appendix C. Windows Operating System Tasks

You can do most administrative tasks with the administration application ; however a few tasks may need to be done with the Windows operating system. Use the following operating system features to administer server software running on Windows:

- **File properties.** Used to set end-user access to the server software's installation directory, the temporary file location, and data files.
- **System properties.** Used to create environment variables.
- **User manager.** Used to create end-user accounts.
- **Services Control Panel.** Used to start, stop, and configure the service.
- **ODBC Administrator.** Used to configure data sources.

File Properties

Use File Properties to set permissions on files. For data files, how you do this depends on where the data are stored. When you store data on the same computer as the server software, you control access to the data directory by setting permissions on a directory on an NTFS drive.

On the server computer, logged on as an administrator:

1. Use the Windows Explorer to navigate to the data directory.
2. Click the directory, right-click, and click **Sharing** on the context menu.
3. Click the **Security** tab and configure the permissions.

   Note: The Security tab is available only on NTFS drives. If you're not sure what type of file system your hardware uses, follow these steps:
4. Use the Windows Explorer to navigate to the drive.
5. Click the drive, right-click, and click **Properties** on the context menu.
6. Click the **General** tab and look at the value for File System.

   When you store data on a computer on your network, you can control access to the data directory by creating a shared resource and setting the permissions appropriately.

   On the networked computer, logged on as an administrator:
7. Use the Windows Explorer to navigate to the data directory.
8. Click the directory, right-click, and click **Sharing** on the context menu.
9. Click the **Sharing** tab in the dialog box, click **Shared As**, enter a share name, and set the appropriate access.

System Properties

Use system properties to create environment variables.

On the server computer, logged on as an administrator:

1. On the Windows desktop, right-click on the icon for the computer. For example, right-click on **My Computer**.
2. Select **Properties** from the menu.
3. Click the **Advanced** tab and then click **Environment Variables**.
4. Click **New**.
5. Type the name of the new variable.
6. Type the value for the new variable.

User Manager

Use the User Manager to create end-user accounts.

On the server computer, logged on as an administrator:

1. From the Windows Start menu choose:

   Programs > **Administrative Tools**

   - Select **Computer Management** and then **Local Users and Groups**.
2. Create the user accounts.

Services Control Panel

Use the Windows Services Control Panel to:

- Stop and start the service.
- Change service startup parameters.
- Check the server status.

To access and use the Services Control Panel:

1. From the Windows Start menu choose:

   Settings > **Control Panel**
2. Select **Administrative Tools** and then **Services**.
3. Select the service. You can now check its status, start or stop it, and edit startup parameters.

Note: You can start, stop, and check the status of the server software with the administration application.

Task Manager

Use the Task Manager to see how many server-related processes are running.

1. Open the Windows Task Manager by pressing Ctrl-Alt-Delete and choosing **Task Manager**.
2. Click the **Processes** tab.
3. Click **Image Name** to sort the processes alphabetically.
4. Search for the filename of the server process (*statisticssrvr.exe*).
5. Search for the filename of the client process (*statisticsproc.exe*). There is one process for each end user currently connected to the server software.

Note: You can monitor server and client processes with the administration application.

ODBC Administrator

Use the ODBC Administrator to configure system and user data sources for use with the server software.

How the ODBC data source is created affects who can view and use it. Use *system* DSNs when you want to allow general access to the data source. Use *user* DSNs when you want to restrict access to sensitive information or when you want to tailor the DSN for a specific user.

To Configure a System DSN

System DSNs can be used by anyone logged on to the computer on which they are defined. System DSNs are easier for you to configure and administer because you do it just once for all users.

On the computer on which you want the data source to reside, logged on as an administrator:

1. From the Windows Start menu choose:

Settings > **Control Panel**

2. Select **Administrative Tools** and then **Data Sources**.
3. Click the **System DSN** tab.
4. Click **Add**.
5. Select a driver from the list. If you are configuring a data source that uses the IBM Corp. data access technology, the Connect ODBC driver names are labeled with the text IBM Corp. OEM.
6. Click **Finish**.
7. Enter the appropriate information into the **Driver Setup** dialog box.
8. Click **OK**.

To Configure a User DSN

User DSNs can be used only by the account of the user who created them. Configure user DSNs when you want to restrict access to sensitive information or when you want to tailor the DSN for a specific user.

Log in as the user and follow the steps for a system DSN, with this exception:

- Click the **User DSN** tab instead of the **System DSN** tab.

Appendix D. UNIX Operating System Tasks

You can do most administrative tasks with the administration application ; however a few tasks may need to be done with the UNIX operating system. Use the following operating system features to administer server software running on UNIX:

- **chmod**. Used to set end-user access to data files.
- **env**. Used to check values of environment variables.
- **scripts**. Used to start the server software and configure its environment.
- **ps** and **kill**. Used to check and stop server processes.
- **odbc.ini**. Used to configure ODBC data sources.

chmod

Use the `chmod` (or `chown`) command to change or assign the permissions mode for directories and data files. For example, to set the */usr/data* directory to read-only for everyone:

1. Log in as a super-user or as the owner of the directory.
2. At the UNIX prompt, type:

```
chmod a-w /usr/data
```

env

Use the `env` command to check the current values of environment variables. For example, to use `env` to check the current values of environment variables for the server software:

1. Log in as the account that started the daemon, typically *root*.
2. At the UNIX prompt, type:

   ```
   env
   ```
3. Check the settings for the variable(s) of interest.

Scripts

To change value of environment variables, edit the environment variable script that is called by the script that starts the server software. To edit the environment variable script:

1. Use a text editor to open the *statsenv.sh* script, which is included in the */bin* subdirectory of the IBM SPSS Statistics Server installation directory. For example, open */usr/local/myserverproduct/bin/statsenv.sh*.
2. If necessary, uncomment the line that defines the variable and then enter the new value for the variable.
3. Save the file.

statsenv.sh is called by the *start_statistics_server* script. Environment variables set and exported in *statsenv.sh* affect only the processes started with the *start_statistics_server* script.

ps and kill

Use the `ps` command to get information about what server processes are running and to report process status. For example:

1. At the UNIX prompt, type:

`ps -efl`.

2. Search for the filename of the daemon process (for example, *statisticsd*). This process has the **UID** of the user who started the server software daemon process (usually *root*).
3. Search for the filename of the client process, *statisticsproc.exe*. There is one process for each end user currently connected to the server software. The *UID* column displays the login ID of the end user who owns the client process.

   Use the `kill` command to kill a process. For example:
4. Log in as the user that started the daemon.
5. At the UNIX prompt, type:

`kill -9` *pid*

where *pid* it the process id of the process.

The server software daemon also automatically creates a file that contains its process ID. Instead of manually finding the PID with the `ps` command, you can use this file in conjunction with the `kill` command directly to kill the daemon process directly:

```
kill -9 `cat statisticsd.pid`
```

Note: If you want to use the administration application to monitor and kill processes, you must start the server software with the startup script provided by IBM Corp.. See the topic "Controlling Service Startup" on page 16 for more information.

odbc.ini

You may need to configure ODBC data sources on the server computer if:

- You are using the IBM Corp. Data Access Pack

and

- The server software needs to access databases

No ODBC administrator exists on UNIX. To configure an ODBC data source on UNIX, you edit a system information text file, *odbc.ini*. *Odbc.ini* is installed when you install the data access pack for UNIX. Installation instructions appear in *IBM Corp. Data Access Pack Installation Instructions for Unix.pdf* (the document is located in the */Documentation/<language>/InstallationDocuments* directory on the product DVD). Be sure to install the additional documentation so that you have access to the documents listed below.

Connect ODBC. For information about editing your *odbc.ini* file and setting important environment variables, see the "Configuring Drivers and Data Sources" section in the "Installation on UNIX" chapter of the *Connect ODBC Installation Instructions* for detailed instructions.

DataDirect's product documentation for Connect ODBC is included, by default, as part of the IBM SPSS Data Access Pack installation. The installer creates the entry IBM SPSS OEM Connect and ConnectXE for ODBC along with the entries for your other programs on the Start menu. The DataDirect product documentation is accessed from this menu item.

DataDirect's product documentation for Connect ODBC can be found under the directory where you extracted the files.

Note: The documentation can also be accessed from the DataDirect home page at http://www.datadirect.com .

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBMproducts. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Index

V

W