

SG-Agent

sg-agent.zip is Microsoft Windows™ executable recommended for routine, automatic uploads. SG-Agent can be run as a program to streamline integration, an SFTP client, or both.

- Supported platforms: Windows™
- Connection requirements: Use your PPK-style private key (**id_rsa.ppk**) to connect.

When run as a program to streamline integration, SG-Agent:

- Connects to your ODBC compliant third-party information system database server. ODBC (Open Database Connectivity) is part of the international SQL Standard; all major database systems are compliant. Example ODBC compliant technologies include: Microsoft SQL Server™, Oracle Database™, FileMaker Pro™, MySQL™, and SQL Anywhere™. Unless you explicitly know otherwise, assume your information system database is ODBC compliant.
- Runs a series of SQL statements to extract the desired information from your database.
- Generates CSV files using the information extracted from your database. SG-Agent takes care of the gritty details like quoting values, escaping characters as necessary, and properly handling non-printable characters.

When used as an SFTP client, SG-Agent connects to a SameGoal SFTP server and uploads files using a directory structure recognized by SameGoal.

Steps *To use SG-Agent:*

1. Download **sg-agent.zip**
2. On a host from which an ODBC connection can be established with the database server, extract the program files to a directory of your choice. If you have no preference, extract these files to **C:\sg-agent**.
3. Place your OpenSSH-style public key (**id_rsa.pub**) and OpenSSH-style private key (**id_rsa**) in the certs directory (eg **C:\sg-agent\certs**) to enable public/private key authentication. The keys are those generated when you configured your SFTP account.

4. Modify **config.txt** (eg **C:\sg-agent\config.txt**). See instructions below.
5. Schedule a task to run **C:\sg-agent\sg-agent.exe** nightly.

config.txt:

SG-Agent is configured using a single text file located at **C:\sg-agent\config.txt**. The format of config.txt is:

- One line per command.
- Valid commands include:
 - `odbc_connection_string`
 - `sftp_username`
 - `sftp_host_public_key_md5`
 - `sftp_public_key_file`
 - `sftp_private_key_file`
 - `tables`
 - `sql`
 - `files`
- Each line must start with a valid command followed by a colon and a space:
<command>: <argument 1> ... <argument N>

Arguments for each command:

- *odbc_connection_string* - The connection string used to connect to your database server. This should include information such as your database server username and password as well as the hostname of the database server. Please consult your database server documentation or connection strings examples for help in finding the correct database server connection string. This command may be specified zero or one times in the config.txt file. Example valid connection strings:
 - **odbc_connection_string: <connection_string> # syntax**
 - **odbc_connection_string: Driver=SQL Anywhere 10;Uid=<sql_anywhere_username>;Pwd=<sql_anywhere_password>;Eng=psi;Dbn=gatekeep;Links=tcPIP(Host=<sql_anywhere_host_name>);CharSet=UTF8; # Gatekeeper (SQL Anywhere)**
 - **odbc_connection_string: Server=<host_name>;Database=<database_name>;Uid=<mssql_server_username>;Pwd=<mssql_server_password>; # SQL Server**
- *sftp_username* - Your district domain. This command must be specified exactly once.
 - **sftp_username: <domain> # syntax**

- **sftp_username: examplecityschools.org # example**
- *sftp_host_public_key_md5* - md5 hash of SameGoal SFTP server's public key. This field is for security purposes and should never be changed. This command must be specified exactly once.
 - **sftp_host_public_key_md5: a12d727365de9d232663438bc9070876 # never change this**
- *sftp_public_key_file* - The relative path to your public key file. This value defaults to **certs\id_rsa.pub**. Using this command is only necessary when you have a non-standard named public key file, or if a single install of sg-agent is uploading files for several different districts (eg at an ITC). This command is commonly used with the -config flag. This command can be specified zero or one times.
 - **sftp_public_key_file: <public_key_file> # syntax**
 - **sftp_public_key_file: certs\examplecityschools.org-id_rsa.pub # example**
- *sftp_private_key_file* - The relative path to your private key file. This value defaults to **certs\id_rsa**. Using this command is only necessary when you have a non-standard named private key file, or if a single install of sg-agent is uploading files for several different districts (eg at an ITC). This command is commonly used with the -config flag. This command can be specified zero or one times.
 - **sftp_private_key_file: <private_key_file> # syntax**
 - **sftp_private_key_file: certs\examplecityschools.org-id_rsa # example**
- *tables* - A space separated list of tables to upload. Information will be extracted from each table by running a query of the form: **SELECT * FROM <table_name>**. Then SG-Agent will create a CSV file using the database column names in the CSV header line and add one line in the CSV file for each row in the database. The filename for the generated CSV file matches the name of the extracted table. This command may be specified zero or one times. If this command is specified one time, the *odbc_connection_string* command is also required. For example:
 - **tables: <table 1> ... <table N> # syntax**
 - **tables: Students Guardians # uploads two files with information from the Students and Guardians tables (students.csv and guardians.csv)**
- *sql* - Create a new CSV file and populates it with the result set from the specified SQL statement. This command can be specified zero or more times (one time per SQL statement you wish to run). If this command is specified one or more times, the *odbc_connection_string* command is also required. NOTE: This command is rarely used (if possible, use "tables" instead).

- **sql: <csv_file_name> <sql_query> # syntax**
- **sql: "SELECT * FROM Students WHERE Enrolled = 1;" students.csv # example**
- *files* - Upload the specified file(s) for import by SameGoal. Use this option if you wish to create CSV file(s) with an external program/script but upload them using SG-Agent. This command can be specified zero or one times. For example:
 - **files: <fully_qualified_csv_file 1> ... <fully_qualified_csv_file N> # syntax**
 - **files: C:\students.csv C:\guardians.csv # uploads students.csv, guardians.csv**

Command Line Usage:

For most users (those uploading data for a single district), sg-agent.exe can be run directly without any flags. Simply add a Windows™ Scheduled Task to run the sg-agent.exe program once per day. Advanced users: If you wish to check the return status of sg-agent or configure uploads for multiple districts, SG-Agent returns 0 in the event of success and a non-zero value in the event of failure. In Windows, to echo the return code of the preceding command, use **ECHO.%ERRORLEVEL%**. Example:

- **C:\>cmd**
- **C:\>exit 0**
- **C:\>echo.%ERRORLEVEL%**
- **0**
- **C:\>cmd**
- **C:\>exit 1**
- **C:\>echo.%ERRORLEVEL%**
- **1**

To check for a non-zero return code use **IF %ERRORLEVEL% NEQ 0** Example:

- **C:\sg-agent>sg-agent.exe**
- **...**
- **C:\sg-agent>IF %ERRORLEVEL% NEQ 0 ECHO failure**

SG-Agent uses Google-style command line flags to configure execution. To see a list of all supported flags, run **sg-agent.exe** with the **-help** flag. Example:

- **C:\sg-agent>sg-agent.exe -help**

SG-Agent can be run with multiple different config files using the **-config** flag. Example:

- **C:\sg-agent>sg-agent.exe -config=examplecitychoools.org_config.txt**

