



**SQL Protection
Evaluator's Guide**



©2024 Software Pursuits, Inc.

Table of Contents

Introduction	2
System Requirements	2
Contact Information	3
Our Scenario.....	3
Understanding SQL Protection’s Architecture	4
A Basic UNC Path Backup	4
SQL Backups with the Communications Agent	5
Understanding the Types of Backups	6
Determining Acceptable Data Loss	7
A Simple Acceptable Data Loss Example	7
Understanding the Purpose of the Standby Server.....	7
Installation.....	8
Understanding the SureSync Desktop	9
If you have SureSync deployed	9
If you’re new to the SureSync Data Protection & Availability Suite	9
Launch the SureSync Desktop for the First Time.....	9
Creating a SureSync Database.....	9
Configuring the Local Communications Agent.....	11
Importing a License	13
If you have SureSync deployed	13
If you’re new to the SureSync Data Protection & Availability Suite	13
Installing the SureSync Scheduler.....	13
Configuring the Communications Agent Machines	15
Creating a SQL Protection Job.....	18
Name Your Job	18
Define the Source Database	19
Define the Authentication Type	20
Define the Database & Source Staging Area.....	20
Define the Destination Staging Area	22
Define the Authentication Type for the Destination Database	25
Define the Database Name	25
Set Restore Options	26
Set Restore Files	27
Define Scheduling Frequency	27
Configuration Complete	29
Configuring E-mail Alerts	29
Configuring a SMTP Profile.....	29
Configuring an Alert Profile	31
Configuring the SQL Protection Schedule to Send Alerts.....	32

Introduction

SureSync SQL Protection is an easy-to-use and affordable software solution for protecting critical Microsoft SQL Server databases. Information is the lifeblood of modern business. That information is often stored in Microsoft SQL Server databases. Protecting these databases is an essential component of any company's data protection and availability plans. With SureSync SQL Protection, you can quickly and easily protect these databases.

SureSync SQL Protection shares a common interface with SureSync making it easy to manage both file replication/synchronization and SQL protection jobs from one solution. This reduces the information technology overhead involved in maintaining multiple solutions for different data management needs.

System Requirements

SureSync SQL Protection's basic operating system and hardware requirements are:

- **Supported Operating Systems:** Windows Server 2022; Windows Server 2019; Windows Server 2016; Windows Server 2012 R2; Windows Server 2012; Windows Server 2008 R2 with SP1; Windows 11; Windows 10; Windows 8.1; Windows 8; Windows 7 SP1
- **Processor:** Minimum: Dual-core CPU of at least 2.5Ghz; Recommended: Quad-core CPU or greater of at least 2.5Ghz
- **RAM (total for system):** 4GB of free memory (recommended minimum)
- **Hard Disk:** 100MB for application files; 20MB+ for database

For File Locking, ReFS volumes are not supported on Windows 2008 R2, Windows 2012 or Windows 7.

SureSync can synchronize data to operating systems such as Windows 2008, and non-Windows machines such as Mac and Linux via UNC path but the software itself must be installed on one of the supported operating systems.

Virtualization

SureSync can be run on Windows operating systems hosted in virtualization software such as VMWare or Hyper-V without issue. Each virtual machine involved in the synchronization / replication requires appropriate licensing.

SureSync Database Requirements

SureSync requires a SQL database to store configuration information. The following versions are supported:

- SQL Express 2012, 2014, 2016, 2017, 2019 and 2022
- SQL Server 2012, 2014, 2016, 2017, 2019 and 2022

While SureSync will operate with older versions of SQL Server, it is strongly recommended to use the newest possible release to take advantage of performance and reliability enhancements in those versions of SQL Server.

SureSync requires some Microsoft components to be installed on the system. The SureSync installer will inspect your system for these components and offer to upgrade or install them as needed.

- Microsoft .NET Framework 4.8
- Microsoft Visual C++ Runtime 14.0 Update 3

Please note that using the SureSync installer to install these prerequisites could result in a reboot being necessary before the setup can continue. If a reboot is necessary, the installer will prompt you. In environments where a reboot is disruptive, we recommend installing the required components manually during your normal maintenance schedules and then proceeding to install SureSync.

Contact Information

If you need further information about SureSync or need clarification on anything within this guide, please contact our support group and they will be happy to assist you with your evaluation.

Software Pursuits, Inc.
140 Chestnut Ln
San Mateo, CA 94403

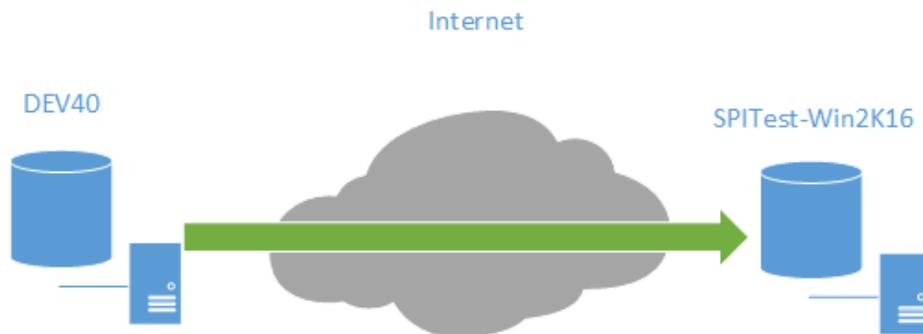
Phone: +1-650-372-0900
Fax: +1-650-372-2912

Sales e-mail: sales@softwarepursuits.com
Support e-mail: support@softwarepursuits.com

Technical support is available between 7:00AM and 4:00PM PST Monday through Friday.

Our Scenario

This guide will walk you through the configuration of SureSync SQL Protection to generate SQL backups from a source server named DEV40. These files will be transferred using the TCP/IP based Communications Agent over the Internet to a machine named SPITest-Win2K16. The backup files sent to SPITest-Win2K16 will be automatically restored into a standby SQL server on that same machine.



In this scenario, a full database backup will be performed once a day at midnight. A differential backup will be performed once an hour. In addition, SQL Protection will thin the staging folders to store only the 4 most recent full/differential backup sets.

SureSync SQL Protection can also be used to generate and copy SQL backup files to UNC paths allowing storage on devices that cannot run the Communications Agent.

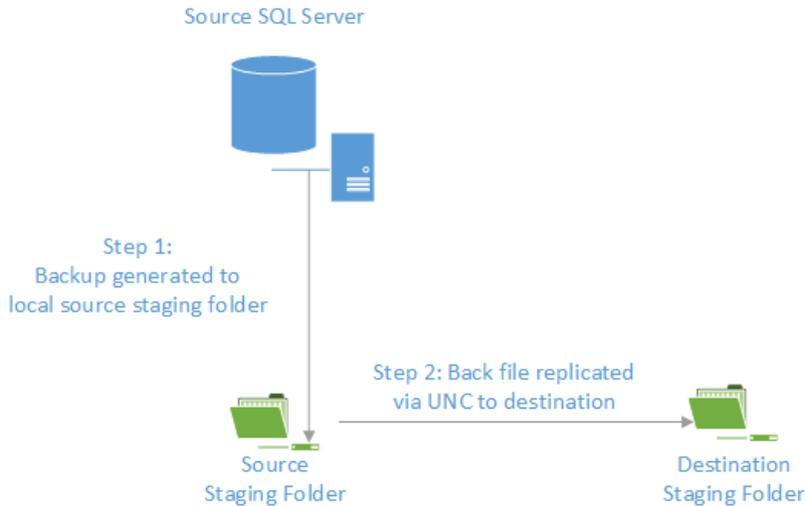
Understanding SQL Protection's Architecture

SQL Protection provides a comprehensive set of functionality to protect your SQL databases. This functionality includes:

- **Communications Agent:** The Communications Agent is a TCP/IP based agent that allows you to copy your SQL backup files securely to remote locations even over the Internet. Storing SQL backups at remote locations is an excellent way to protect mission critical SQL databases from natural disasters and other threats.
- **Encrypted data transmission:** The Communications Agent offers a number of FIPS certified encryption algorithms to ensure data security when transferred over public connections like the Internet.
- **Automated Restore to Destination SQL Server:** SureSync SQL Protection allows you to automatically restore the SQL backup files to a destination SQL server on a scheduled basis. On the destination SQL server, the database can be in a "restoring" state where the database is not accessible to users or in a "standby / read-only" state where users can execute read-only queries against the database.. This state allows you to continue restoring additional backup files and prevents users from accessing the database. With either type of restore, the administrator can switch the database to ready if necessary.
- **Automatic thinning of old backups:** Manage storage usage by automatically thinning old backup files. SQL Protection can be configured to keep x number of full backups. When a new full backup is generated, the oldest full and all associated differential or log backups can be automatically deleted.
- **Quick and easy restores:** An easy to use restore wizard is provided to allow your organization to quickly recover from any SQL database issues. You can also use the built-in SQL Management Studio tools to restore backups generated by SQL Protection giving you flexibility.
- **UNC path support:** SQL Protection can process SQL backups to any machine accessible via UNC path allowing you to store backups on devices that do not support running the Communications Agent. However, doing so eliminates the ability to use features of the Communications Agent including encryption. You also cannot automatically restore to a destination standby SQL server.

A Basic UNC Path Backup

SureSync SQL Protection is a flexible software solution allowing you to design a backup procedure that meets your specific needs. In the most basic form, SureSync SQL Protection will allow you to generate SQL backups on a schedule and replicate those files to another machine accessible via UNC path.



The basic backup process consists of two steps:

- In Step 1, the backup file is generated. SureSync SQL Protection uses standard Microsoft APIs to generate these backups ensuring that your backup files are consistent with Microsoft standards and are supported. This backup file is generated into a folder named the Source Staging Folder. Generally, this folder is on the same machine as the source SQL Server software. The Source Staging Folder can also be a share on the same local network as the SQL server that can be accessed via UNC path. The Source Staging Folder is simply a staging area for the backup files.
- In Step 2, the backup file generated in the Source Staging Folder will be replicated to the Destination Staging Folder. The Destination Staging Folder is the final destination for the backup files. This folder is on a second machine that is accessible via UNC path.

With this solution, you have two usable copies of the SQL backup files generated automatically. A copy is available locally in the source staging folder and in the remote destination staging folder. This provides some additional redundancy of the backup files.

SQL Backups with the Communications Agent

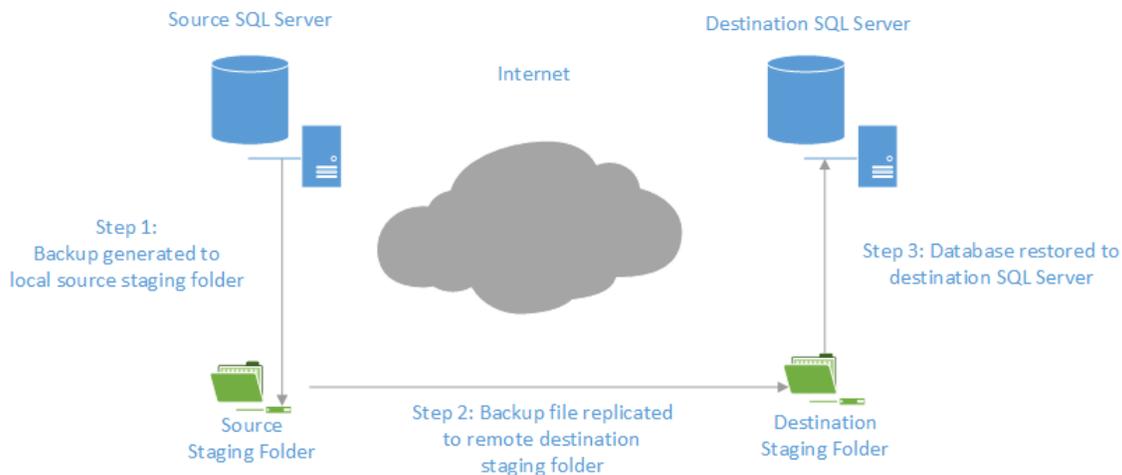
When implementing the Communications Agent in SureSync SQL Protection, the basic architecture is unchanged. The Schedule produces a backup from the SQL server and stores it in a local Source Staging Folder. From there, the backup file(s) are replicated to a Destination Staging Folder.

The Communications Agent adds the following:

- The Destination Staging Folder does not have to be accessible via UNC path. The Communications Agent is installed on the destination Windows based machine. The Communications Agent is a TCP/IP enabled Agent. With Communications Agents installed, SureSync SQL Protection can replicate the backup files to a Destination Staging Folder accessible by IP address or DNS name even over the Internet.
- The Communications Agent encrypts data transmitted with FIPS certified encryption algorithms. This enables the safe transmission of the SQL backup files to remote Destination Staging Folders even over the Internet.
- SureSync does not need to be installed on each SQL source server. With a Communications Agent installed on the SQL server, you can perform backup operations from a SureSync installation on a different machine.

- SQL Protection provides the option of automatically restoring the backup file onto a destination SQL server in the remote location. The database will be in the “restoring” state on the destination server. This allows you to continue restoring additional backup files and prevents users from accessing the database. As an administrator, you can switch the database out of the “restoring” state when necessary to allow users to access the database.

A visual representation of a fully implemented SureSync SQL Protection environment with the Communications Agent deployed and an automatic restore to a standby server would look like:



SQL Protection provides for the automatic thinning of backups stored in the staging folders. This feature helps control storage space consumed by backups. In-depth detail about SureSync SQL Protection can be found in the SureSync Data Protection & Availability Suite help file distributed with the product. Pressing F1 on any tab of the user interface will provide detailed context sensitive help about the options available on that screen.

Understanding the Types of Backups

If you are unfamiliar with SQL database backups, the first concept that must be understood is the difference between database backup types. Three different types of available backups are available: Full, Differential, and Log.

- **Full Backup:** A complete backup of the SQL database. This type of backup contains all of the data in the defined database. A full backup provides the foundation for Differential and Log backups. Without a Full database backup, the other database backup types are useless. Your SureSync SQL Protection Schedule must create a Full backup.
- **Differential Backup:** A Differential backup includes all of the data that has changed since the last Full backup. You must have access to the Full backup when performing a restore with Differential backups. For example, assume your full backup runs on Sunday nights at 11:00PM. For the remaining days of the week, you run a Differential backup at 11:00PM. If the database failed on Wednesday, you would need the Differential backup from Tuesday night and the Full backup from Sunday to perform a restore.
- **Log Backup:** This type of backup can only be used when the SQL database being backed up is in full recovery or bulk-logged recovery models. SQL databases using the simple recovery model cannot use a Log backup. Like Differential backups, a Log backup is dependent upon a Full backup of the database. A log backup contains the part of the

transaction log that was active when the backup was created and includes all records not backed up in the previous log backup. If you maintain an uninterrupted set of log backups and are running the full recovery model then you can use the logs to restore to a point in time using SQL Management Studio.

Determining Acceptable Data Loss

In any disaster recovery scenario, your company must determine the acceptable amount of data loss. For example, could your company recover a day worth of lost data? An hour? If you're running something simple like a database of prices for products then the information lost could be entered into the database again after recovery. Generally, the more sensitive the data in the database the smaller the amount of data loss that is considered acceptable. When dealing with databases that have a very low level of acceptable data loss, you have to perform much more frequent backups. The ideal backup configuration is going to be the one that provides your acceptable measure of data loss.

Constraints exist on how frequently you can backup your SQL servers. These constraints are no different than for file backups. One, you have a limited amount of bandwidth available to move data. If you're copying a TB worth of data, that will take a certain amount of time based on the speed of the connections involved. Two, the machines and storage systems involved have a maximum capacity. For example, storing 6 months worth of backups for a 1TB database where a full backup is generated once a week will consume 24TB of storage for just the full backups. Any differential and/or log backups add to the storage requirement. The machines involved must have adequate storage available to meet your backup goals.

A Simple Acceptable Data Loss Example

A company is protecting a SQL database that includes information used by the company's human resources department. The company decides that 1 hour of data loss is acceptable. This is a commonly accepted measure and can be easily obtained with Scheduled backups in most situations.

The company creates a SureSync SQL Protection Schedule that runs a Full backup once a day. In addition, Differential backups are configured to be run once an hour. By taking the latest full and differential backup the company will be able to restore to the acceptable level.

Understanding the Purpose of the Standby Server

When running SureSync SQL Protection, you have the option of automatically restoring the backup files to a standby SQL server. The files restored to the standby SQL server are the backup files copied into the destination staging folder by the SQL Protection Job.

When restoring the backup to the Standby Server, three modes of restore are available:

- **Restore in Ready State:** SQL refers to this method as "Restore with Recovery." When restoring in this mode, no additional restores can be performed except for full backup restores. The database is left ready for use
- **Allow Additional Restores:** SQL refers to this method as "Restore with NoRecovery." The database is left in a "Restoring" state and is not accessible by clients. This mode allows additional backups to be restored.
- **Restore with Standby:** The database will be left in a "Standby / Read-Only" mode and allows additional restores. This mode allows users to perform read-only queries against the database on the destination but increases the overhead of restores on that machine.

A Standby Server can provide the following benefits:

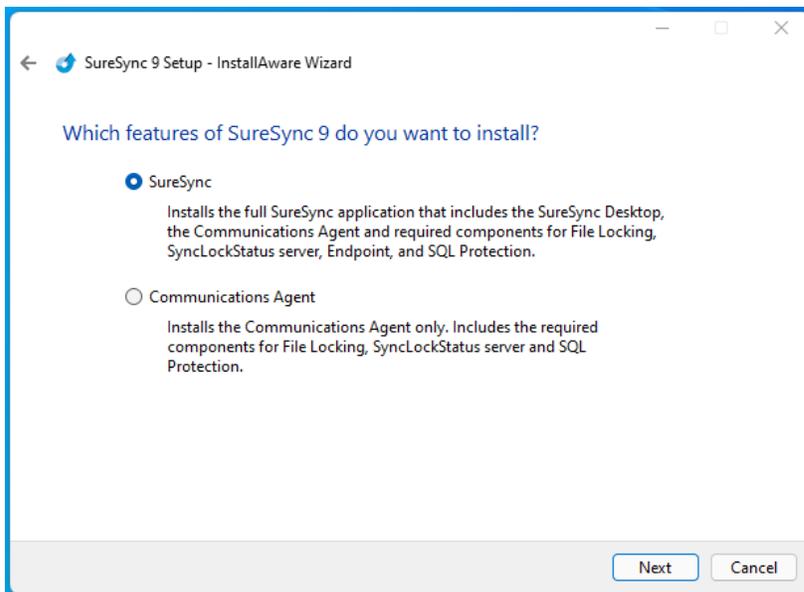
- **Automatic validation of backups:** Simply having a backup file doesn't mean that it will restore properly. The entire restore process completing to the standby SQL server proves that the backup files are valid and function correctly.
- **Increase speed of recovery:** A Standby Server can increase the speed of recovery in the event of a natural disaster or server failure. SQL Protection keeps the database in another the remote office current as of the last backup. Users could be redirected to this machine in the event of a major failure.
- **Offload processing of reports or queries:** Using "Restore with Standby," an administrator could have users generate reports on the Standby Server and save that load on the production machine. However, care must be taken that the Standby Server has a current copy of the database so report information is accurate.

It is critical to use the Standby Server correctly. You never want to allow write access to the Standby Server unless you're permanently switching to the machine. Write changes to the Standby Server database copy will result in two different copies of the same database with no way to resolve the difference. you would have to perform backups from the destination machine and restore those backups on the source before allowing anyone to make additional changes to the original source machine.

The standby server provides organizations with an affordable means of quickly moving to a new SQL server that contains copies of the databases current as of the last backup. This process is quicker in the event of a major failure than having to build a new server, install SQL server and have to restore copies of all of the databases. Instead, the server is already there with the databases and it's ready to go.

Installation

To begin your SureSync SQL Protection trial, you must install the appropriate components on the correct machines. For SQL Protection, the full SureSync application must be installed on one machine and the Communications Agent installed on any other machine that will be used as either a source SQL server or a remote staging / remote restore location.



In our example, SureSync will be installed on the DEV40 database server. To install the SureSync components, launch the SureSync9Setup.exe installer on the machine in question and follow the prompts. Selecting “SureSync” will install the SureSync Desktop and all required components.

With SureSync SQL Protection, the Communications Agent must also be installed on the destination side. For this example scenario, the destination machine is SPITest-Win2K16. The same SureSync9Setup.exe should be run on that machine and the “Communications Agent” option selected.

Understanding the SureSync Desktop

The SureSync Desktop is a user interface component shared between all members of the SureSync Data Protection & Availability Suite. The SureSync Desktop is used to configure and manage SureSync SQL Protection Jobs, Schedules and Restores.

If you have SureSync deployed

SureSync SQL Protection shares the SureSync Desktop and your SureSync database. If you intend to run SureSync SQL Protection Jobs and Schedules from the same main SureSync machine used to run your SureSync tasks then you can skip ahead in this guide to the “Importing a License” section.

If you're new to the SureSync Data Protection & Availability Suite

If you are new to the SureSync Data Protection & Availability Suite, you will be installing components for the first time and will need to configure a SureSync database.

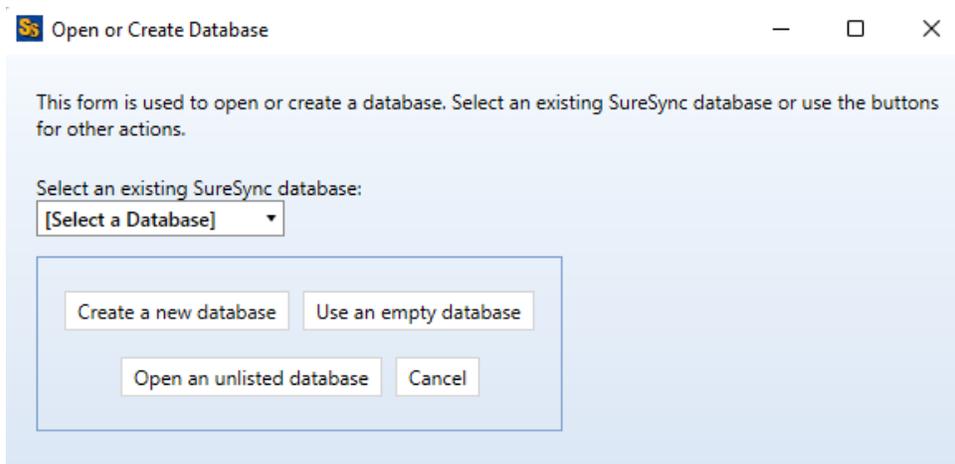
Launch the SureSync Desktop for the First Time

Now that the required components have been installed, we can continue with the configuration. To launch SureSync for the first time, go to the Start menu, select the SureSync 9 folder and click on the SureSync 9 Desktop icon. This will launch the SureSync Desktop where you will perform your entire synchronization job configuration.

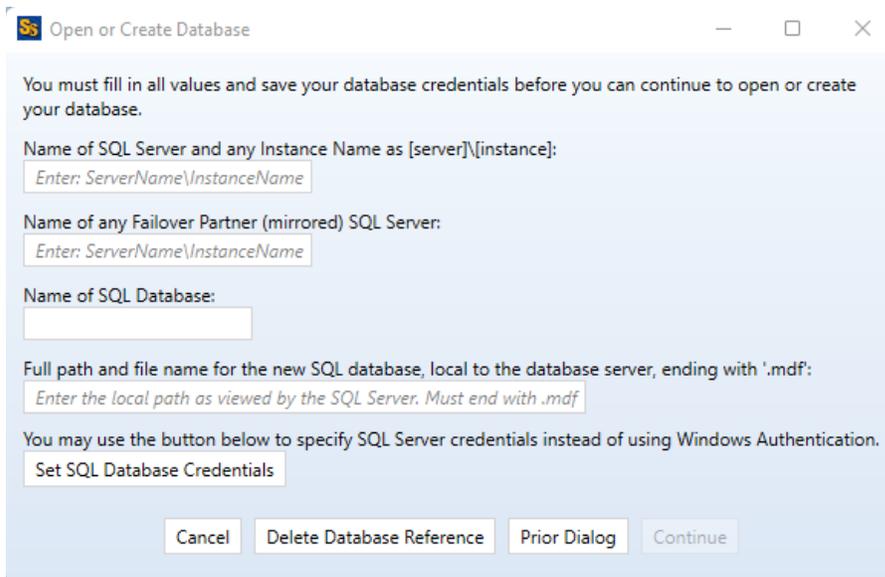
On the first launch of the application, SureSync will present you with a series of questions to aid in completing the initial configuration.

Creating a SureSync Database

The first dialog displayed is used to create or open an existing SureSync database. The database is used to store your synchronization/replication configuration and related information.



Click on the 'Create a new database' button to continue.



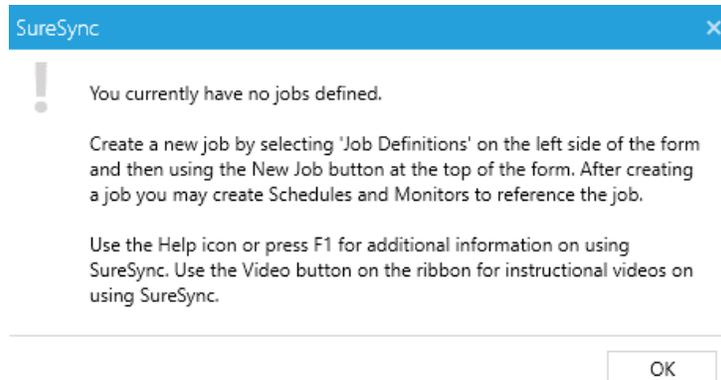
Enter the following information to create the SureSync database:

- **Name of the SQL Server and any Instance Name:** Enter the name of the SQL server and an instance name if used. For example, dev40\sqlexpress.
- **Name of any Failover Partner (mirrored) SQL Server:** This is an optional field that allows you to use a SQL Failover Partner. This functionality has been deprecated from SQL but is currently still available. Should be defined as machinename\instancename.
- **Name of SQL Database:** Enter the name of the SQL database that should be created.
- **Full path and file name of the new SQL database:** Enter the path on the SQL server where the database files should be created. This folder must already exist. For example, G:\Databases\SureSync.mdf.

If you want to use Windows authentication with the SQL database, you can click 'Continue' to create the database.

If you want to use SQL authentication with the SQL database, you will click on the 'Set SQL Database Credentials' button to provide the credential. After that, you can click 'Continue' to create the database.

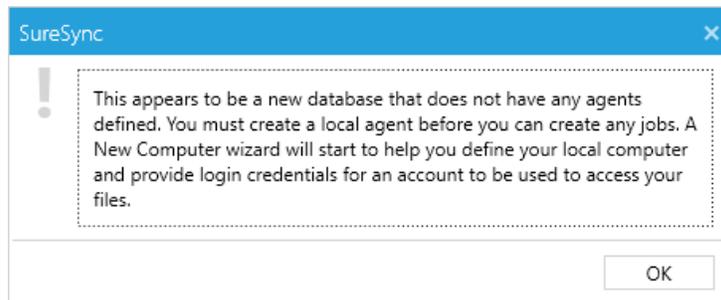
Next, the following dialog will be displayed:



Click 'OK' to continue. The help file will be displayed that can either be reviewed now or closed.

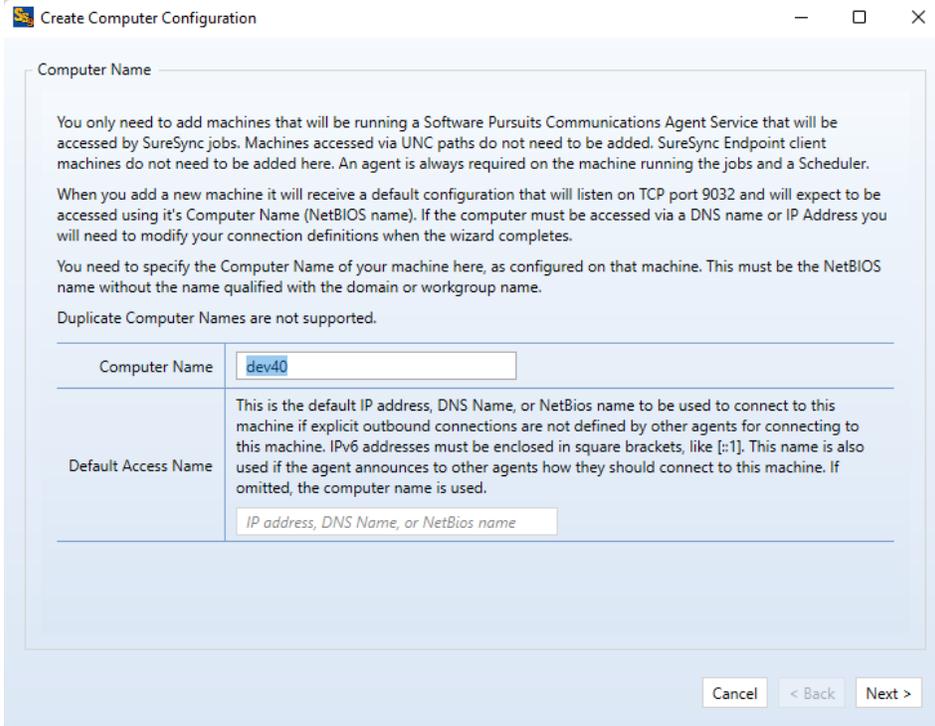
Configuring the Local Communications Agent

The next step in the initial configuration involves defining a credential for the local Communications Agent to run under.

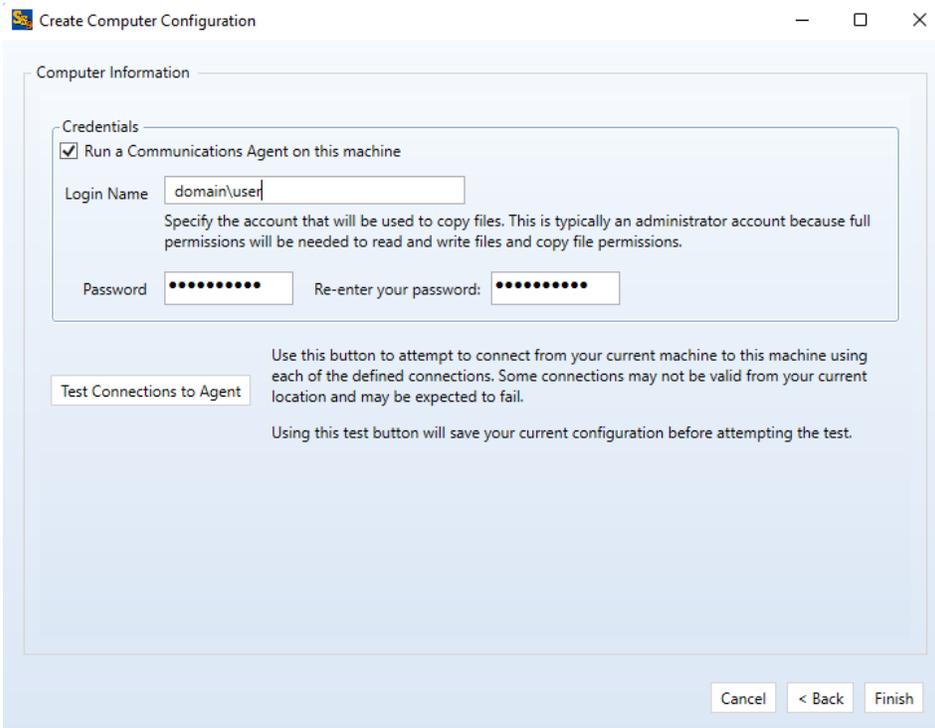


Click "OK" and the Communications Agent Configuration panel will be launched.

The local Communications Agent must be given a credential for SureSync to function properly. We recommend this account be either a local administrator on the machine or a domain administrator to ensure rights to all the files and folders being synchronized. The local Communications Agent must be configured even if you're not licensed for the Communications Agent Add-on. The Add-on functionality is enabled by a license but the Communications Agent on the SureSync machine is responsible for all the basic I/O as well.



When the “Create Computer Configuration” wizard loads, the name of the computer SureSync is installed on is automatically filled in. For this example, the machine name is dev40. Click “Next” to continue.



The “Computer Information” panel of the wizard is where you define the user account and password that the Communications Agent should use to access the files on the machine.

To ease initial configuration, the “Run a Communications Agent on this machine” option will be checked. In addition, the account you are logged into the machine as will be prefilled in the “Login Name” field.

If you want the Communications Agent to run under a different user account, you can change it here. The account must be in domain\user or machinename\user format. You can also enter .\username to define a local account.

Enter the password for the defined account twice in the password fields.

When a Communications Agent configuration is saved, a default connection for TCP port 9031 is created automatically. In most environments, only the default connection is used.

To test the connection, click the “Test Connections to Agent” button.

Click “Finish” to complete Communications Agent configuration for the local agent. You will be taken to the SureSync Desktop

Importing a License

To enable functionality, you must import a trial license or a purchased license.

If you have SureSync deployed

If you are a current SureSync user and want to trial SQL Protection, you should request trial licensing be added to your existing license. This can be done by e-mailing our sales team at sales@softwarepursuits.com. You can also call the sales team at 1-800-367-4823.

Once an updated license file has been provided to you, click the “Licenses” button () in the ribbon bar of the SureSync Desktop and click the “Import License...” button to import the updated file.

If you're new to the SureSync Data Protection & Availability Suite

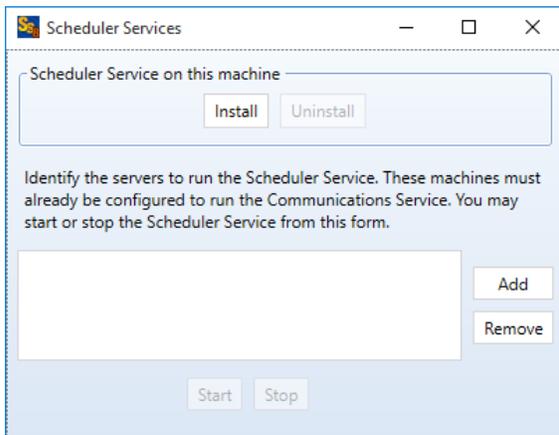
If you are new to the SureSync Data Protection & Availability Suite, you should request a trial license from <https://www.softwarepursuits.com/suresync>. A license file will be generated and sent to you via e-mail. You can also call our sales team at 1-800-367-4823.

Once an updated license file has been provided to you, click the “Licenses” button () in the ribbon bar of the SureSync Desktop and click the “Import License...” button to import the updated file.

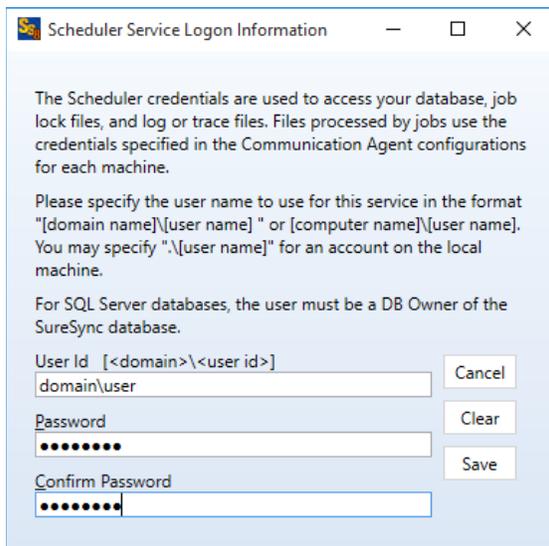
Installing the SureSync Scheduler

The SureSync Scheduler is a Windows service that runs on the main SureSync machine. This service is responsible for launching Schedules at the correct times and for running Real-Time Monitors. You must have a Scheduler running for your Schedule to execute at the configured time.

To install the SureSync Scheduler service, click on the Home button in the upper left-hand corner of the SureSync Desktop and click on “Scheduler Services.”



To install the Scheduler, click the “Install” button, which will launch a window like the one below.



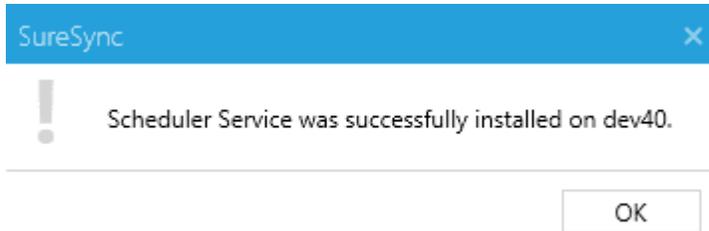
The user account the Scheduler service runs under must be a member of the local administrators group on the SureSync machine.

The account must also be a DBOwner on the SureSync database. Depending on the account you're using and how your SQL Server is configured, you may need to do this manually if the account is not automatically assigned DBOwner on newly created databases.

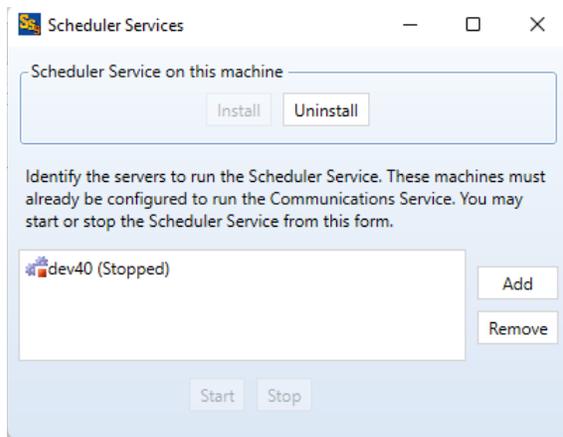
To do so, launch SQL Server Management Studio. Connect to the instance where the new SureSync 9 database is located. Expand Security and then Logins. Locate the account that the Scheduler will be running under. Right click and select Properties. Click on User Mappings. Make sure there is a checkmark next to the SureSync 9 database and that db_owner is checked for membership.

Enter the username formatted as *machinename\username* or *domainname\username*. This account must be an administrator on the machine. Click “Save” to install the service.

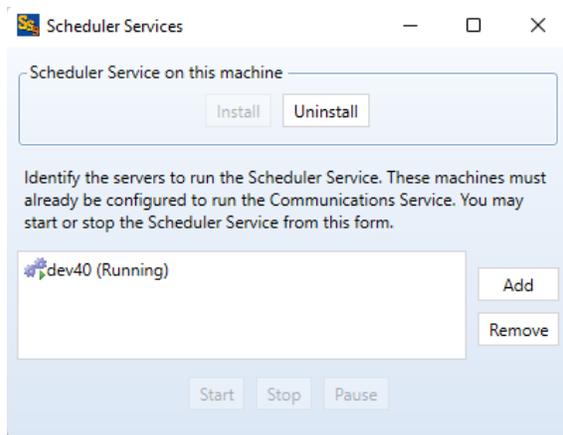
You will receive a confirmation dialog box that the Scheduler service was installed successfully as shown below.



The Scheduler will appear in the dialog with a status of (Stopped).



Once the service has been installed, click on the machine name and click “Start.” The Scheduler will now be listed as Machine Name (Running).



Configuring the Communications Agent Machines

The next step in configuring the environment is to define the remote Communications Agent machines.



To configure a Communications Agent, launch SureSync and click on the “Computer” (Computer) button in the ribbon bar.

Computer Name

You only need to add machines that will be running a Software Pursuits Communications Agent Service that will be accessed by SureSync jobs. Machines accessed via UNC paths do not need to be added. SureSync Endpoint client machines do not need to be added here. An agent is always required on the machine running the jobs and a Scheduler.

When you add a new machine it will receive a default configuration that will listen on TCP port 9032 and will expect to be accessed using its Computer Name (NetBIOS name). If the computer must be accessed via a DNS name or IP Address you will need to modify your connection definitions when the wizard completes.

You need to specify the Computer Name of your machine here, as configured on that machine. This must be the NetBIOS name without the name qualified with the domain or workgroup name.

Duplicate Computer Names are not supported.

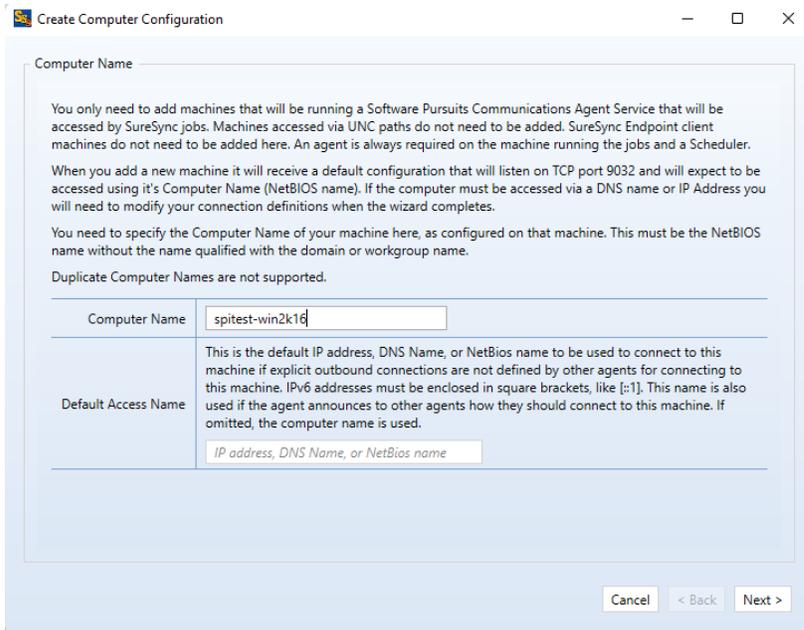
Computer Name	<input type="text" value="Please enter your new Computer Name"/>
Default Access Name	<input type="text" value="IP address, DNS Name, or NetBios name"/>

Cancel < Back Next >

The “Computer Name” panel of the Computer wizard is where you define the name of the machine running the Communications Agent that SureSync will be connecting to. This name must be the NetBIOS computer name of the computer.

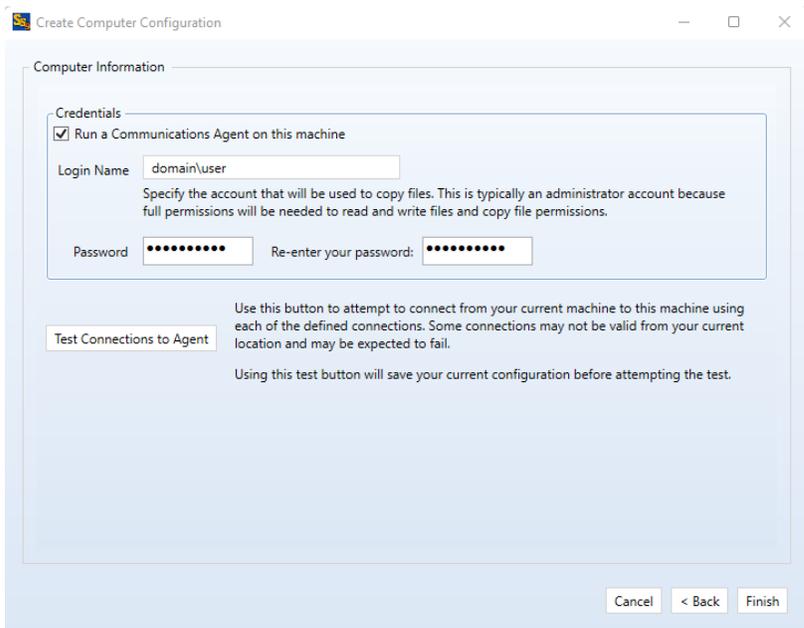
The computer name entered here must be the NetBIOS computer name of the computer in question. Using an IP address or a random name will result in the connection failing.

Enter the computer name into the field. For example, spitest-win2k16.



Click the “Next” button to continue.

The “Computer Information” panel of the wizard is where the credential used for accessing files on this Communications Agent machine is defined.



On this panel you will then want to:

1. Ensure that “Run a Communications Agent on this machine” is checked. It is by default.
2. Enter the account to be used in the “Login Name” field. This should be in domain\user or machinename\user format. .user can also be used to represent the machine name for local accounts.
3. Enter the password for the account in both password fields.

Click the “Test Connections to Agent” button to test the connection.

If the test does not complete successfully, there are some common causes to investigate:

1. Ensure the Software Pursuits SureSync 9 Communications Agent service is running on the machine.
2. Ensure the name you defined for the Computer name is the NetBIOS name of the computer.
3. Ensure that the clocks are accurate for the time zone the machine is in. .NET cryptography will reject requests when the clocks are greater than 5 minutes apart.
4. Ensure TCP port 9031 is forwarded to the correct machine in any firewall.

These steps should be done for each Communications Agent that will be used. In the example scenario, *SPITEST-WIN2K16* would be defined

Creating a SQL Protection Job

SQL Protection Jobs are run on a Scheduled basis. A SQL Protection Job consists of a Schedule and the Job definition. Both components created using the same wizard. The Schedule will automatically be given a name based on the name of the Job. To launch the wizard, click on the



“Job” button () in the ribbon bar.

Name Your Job

The first wizard panel allows you to configure the Job name, an optional description and define the number of full backups that the Job should keep. When a number of backups is defined in the “Set to the number of full backups you want to keep” option, SQL Protection pruning is enabled. This means that SQL Protection will keep the number of full backups and associated differential and log backups defined here. When a new full backup is generated, the oldest one and the related differential and log backups are automatically purged. This feature helps keep the storage requirements for your SQL backups under control.

For this Schedule, we will use the name “SQL Protection Demo” and define 4 as the number of full backups to maintain.

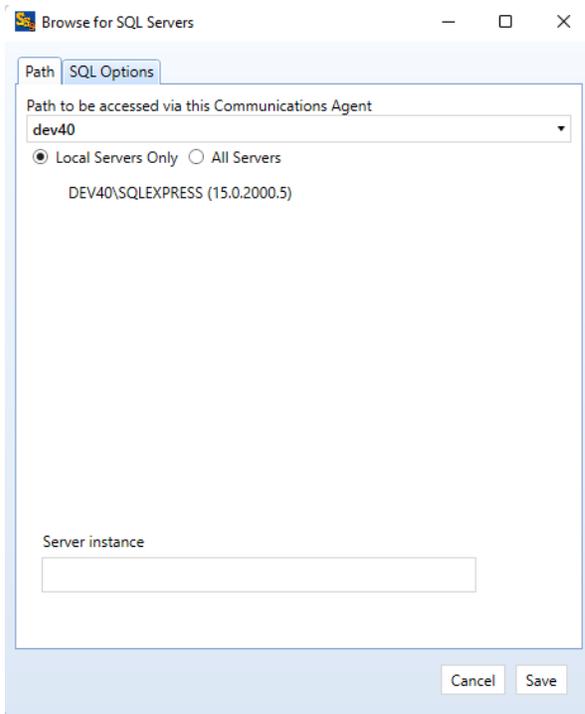
The screenshot shows a window titled "Create New SQL Job" with a "Job Name" section. It contains a text box with "SQL Protection Demo", a large empty text area for a description, and a spinner box set to "4" with the text "Set to the number of full backup files you want to keep. Set to zero to not thin the backup files". Navigation buttons "Cancel", "< Back", and "Next >" are at the bottom.

Define the Source Database

The next panel in the Wizard is where you provide details about the source SQL database that you are looking to protect with the Job.

The screenshot shows the "Define the Source" step of the "Create New SQL Job" wizard. It includes instructions on configuring backups and a license note for SureSync. The form fields are: "Server\Instance Name" with a "Browse" button, "Authentication" set to "Windows Authentication", "User Name" with a placeholder "[domain\user name]", "Password", "Database" with a dropdown menu "[Please select or enter a database]", and "Source Staging Area" with "[dev40]" and a "Browse" button. Navigation buttons "Cancel", "< Back", and "Next >" are at the bottom.

Click the “Browse” button next to “Server\Instance Name” to select the SQL Server that hosts the database to be backed up.



The browse will only return results if the SQL Browser service is enabled and the SQL instance in question is configured to respond to TCP/IP requests. In many cases, you will get no results. You can still protect databases on that instance by typing the server instance into the “Server Instance” field. For example dev40\SQLEXPRESS.

Select the SQL Server or type the instance into the “Server instance” field and click the “Save” button.

Define the Authentication Type

The next step is to define the correct authentication type for the SQL server using the “Authentication” drop-down menu. The available options are “Windows Authentication” or “SQL Server Authentication.” If using “SQL Server Authentication” you must also provide the SQL username and password in the available fields. The “Windows Authentication” option will use the username and password provided in Communications Agent Setup for the Agent in question.

For this example, we will use “Windows Authentication.”

Define the Database & Source Staging Area

The “Database” drop-down will display a list of all SQL databases on the configured SQL server. For this example, we will select the “SureSync9” database.

Define the Source

Use this panel to configure your job to create backups of your SQL Server database and copy the backups to a destination machine.

SQL Protect is licensed by each source machine configured. You may, however, backup your SureSync database without any additional license. The SureSync machine will not need a license unless that machine is used for other source databases.

Server Instance Name:

Authentication:

User Name:

Password:

Database:

Source Staging Area:

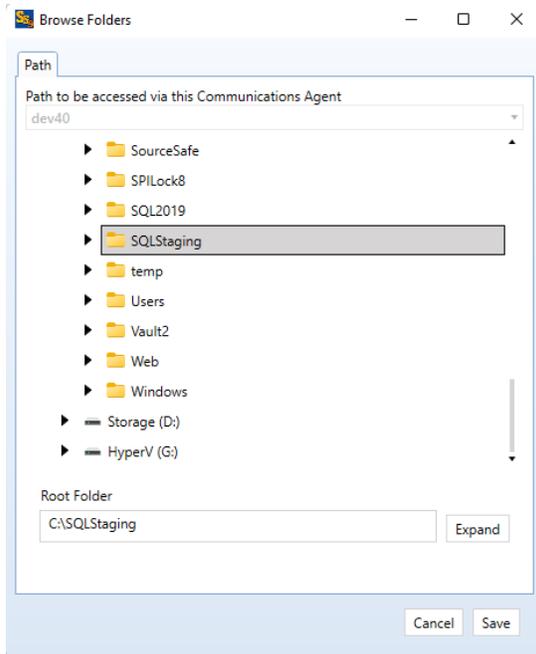
Cancel < Back Next >

Finally, you must define a source staging area. The source staging area is a folder on the source SQL machine where the SQL backup files will be generated before they are replicated to your remote destination machine.

The browse will only return results if the SQL Browser service is enabled and the SQL instance in question is configured to respond to TCP/IP requests. In many cases, you will get no results. You can still protect databases on that instance by typing the server instance into the “Server Instance” field. For example, dev40\SQLExpress.

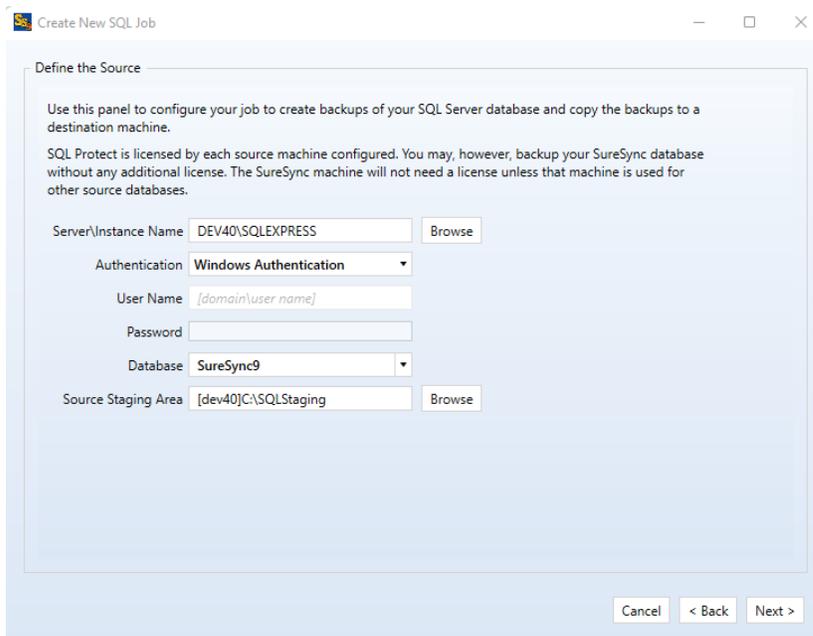
The Source Staging Area definition should be unique for each database being protected. One strategy is to create a root folder such as C:\SQLStaging and then create a subfolder for each database. For example, C:\SQLStaging\Database1. The Source Staging Area is then set to C:\SQLStaging\Database1 for the Database1 SQL Schedule.

To define the Source Staging Area, click the “Browse” button and select the correct folder. For this example, we will use a folder “C:\SQL Staging.”



Click the “Save” button to return to the main Job wizard.

The completed “Define the Source” panel looks like:



Click the “Next” button to continue the Job configuration.

Define the Destination Staging Area

The Destination Staging Area is a folder on a remote machine where the SQL Protection Job should copy the backup files generated on the source.

You cannot restore a SQL database backup to a SQL server running an older version of SQL than used on the source. For example, if your source SQL server is running SQL 2016, your destination server must be as well (or newer).

SQL Protection provides access to the Communications Agent. This powerful TCP/IP based agent allows you to communicate with any Windows machine accessible via DNS name or an IP address. The Communications Agent allows you to replicate SQL backups generated by SQL Protection to remote machines even over the Internet.

In addition, with SQL Protection you can chose to restore the backup files copied to the Destination Staging Area

When selecting a remote Communications Agent from the drop-down, the browse panel will display the drives and folders from that remote machine as if you were sitting down in front of the computer.

You do not have to restore to a standby server. If you do not provide the database details on the “Define the Destination” panel then the SQL backups will simply be copied to the Destination Staging Area for backup storage.

The screenshot shows a window titled "Create New SQL Job" with a "Define the Destination" tab. The window contains the following fields and controls:

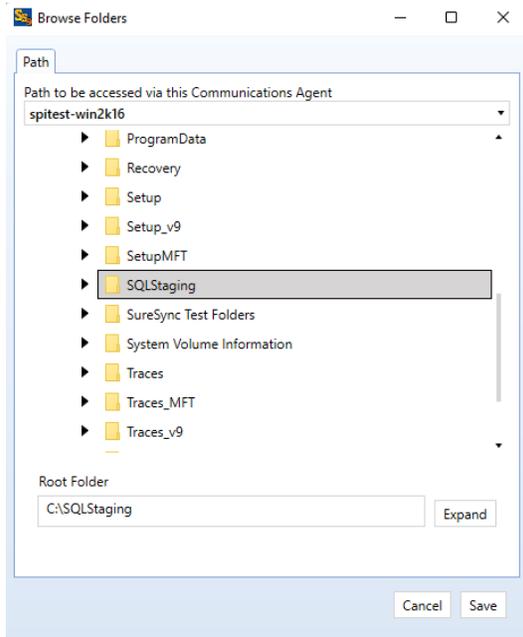
- Destination Staging Area:** A text box containing "[dev40]" and a "Browse" button.
- Use Compression:** A checkbox labeled "Check to have transmission compressed when coping source staging path to the destination." (Note the typo "coping" in the original image).
- Server\Instance Name:** A text box containing "[server name\instance name]" and a "Browse" button.
- Authentication:** A dropdown menu currently set to "Windows Authentication".
- User Name:** A text box containing "[domain\user name]".
- Password:** An empty text box.
- Database:** A dropdown menu with the text "[Please select or enter a database]".

At the bottom of the dialog are three buttons: "Cancel", "< Back", and "Next >".

To define the Destination Staging Area, click on the “Browse” button. In this scenario, the destination is the remote SPITest-Win2K16 machine. Select the “SPITest-Win2K16” machine from the “Path to be accessed via this Communications Agent” drop-down menu.

When selecting a remote Communications Agent from the drop-down, the browse panel will display the drives and folders from that remote machine as if you were sitting down in front of the computer.

The path that we will store the SQL backup files on SPITest-Win2K16 is “C:\SQLStaging” which we then select using the browse.

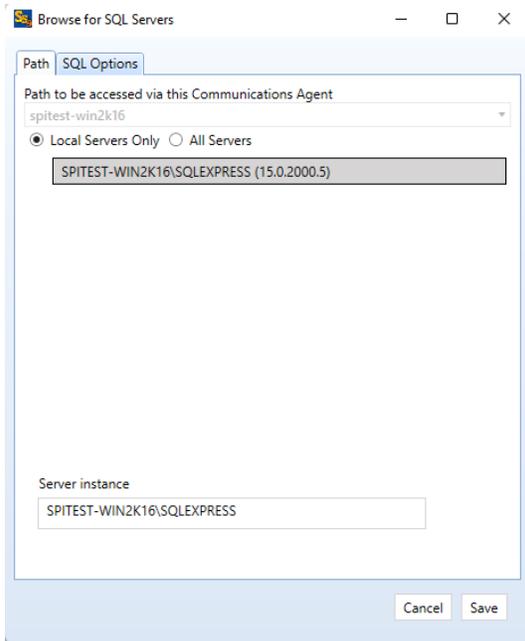


Click the “Save” button to return to the main Job wizard.

With SQL Protection, the option to restore the transferred backups to a standby server automatically is available. For this example, we will fill out the rest of the “Define the Destination” wizard panel.

Click the “Browse” button next to “Server\Instance Name” to select the SQL Server that will be used to host the standby database.

The browse will only return results if the SQL Browser service is enabled and the SQL instance in question is configured to respond to TCP/IP requests. In many cases, you will get no results. You can still protect databases on that instance by typing the server instance into the “Server Instance” field. For example dev40\SQLExpress.



Click the “Save” button to save your selection and return to the main wizard panel.

Define the Authentication Type for the Destination Database

The next step is to define the correct authentication type for the SQL server using the “Authentication” drop-down menu. The available options are “Windows Authentication” or “SQL Server Authentication.” If using “SQL Server Authentication” you must also provide the SQL username and password in the available fields. The “Windows Authentication” option will use the username and password provided in Communications Agent Setup for the Agent in question.

These options are defined on the “SQL Options” tab.

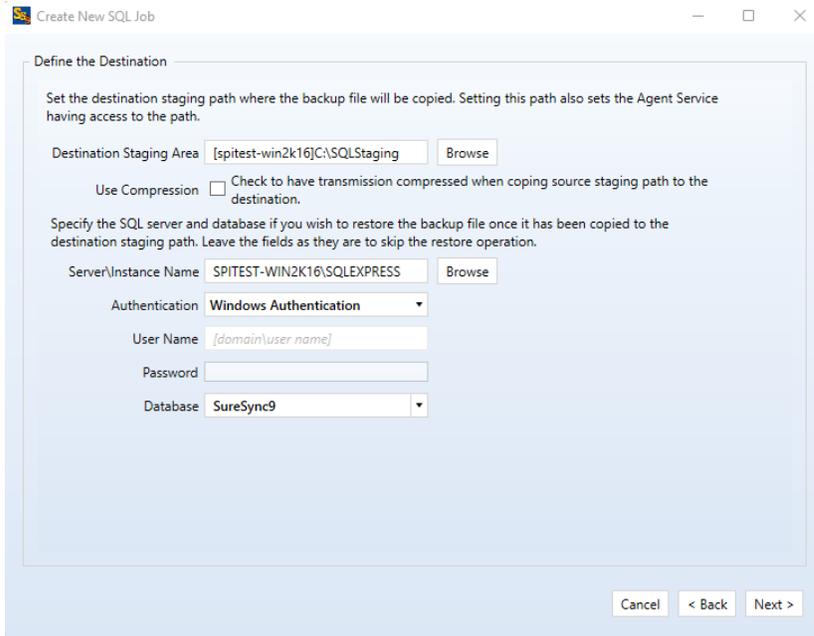
For this example, we will use “Windows Authentication.”

Define the Database Name

The “Database” drop-down on the “Define the Destination” panel allows you to select or enter the name of the database that SureSync SQL Protection should restore into. Often, this will be the same name as the source database. However, it can be a different name.

For this example, the database name on the destination will be “SureSync9.”

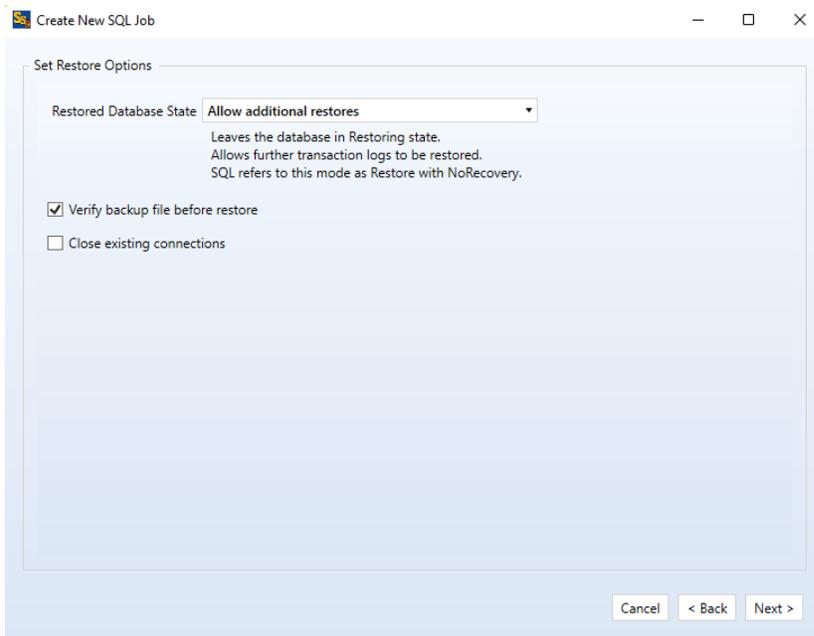
The completed destination database configuration will look like this:



Click the “Next” button to continue.

Set Restore Options

The next panel of the wizard allows you to set options related to the restored standby database.



You have the following restore options:

- **Restore in ready state:** Restores the database ready for use on the destination. Additional restores are not allowed.

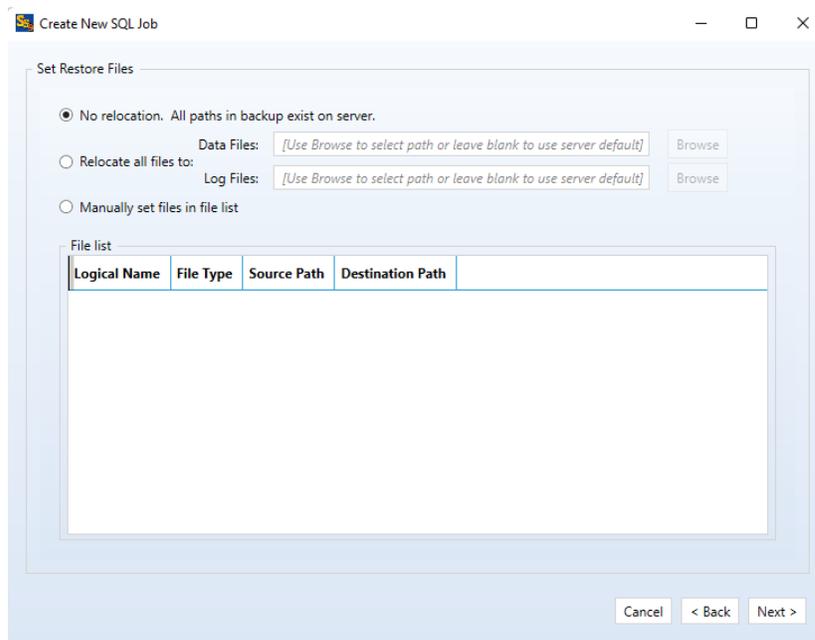
- Allow additional restores: Leaves the database in a Restoring state. Allows further transaction logs to be restored. SQL refers to this mode as “Restore with NoRecovery.”
- Restore with Standby: Leaves the restored database in read-only mode.

For this example, we will select “Allow additional restores” and check “Verify backup file before restore.”

Click the “Next” button to continue.

Set Restore Files

The paths where the data file (*.mdf) and log file (*.ldf) are located on the source server are stored within the SQL backup file. This next wizard panel sets options about the database paths for the destination server.



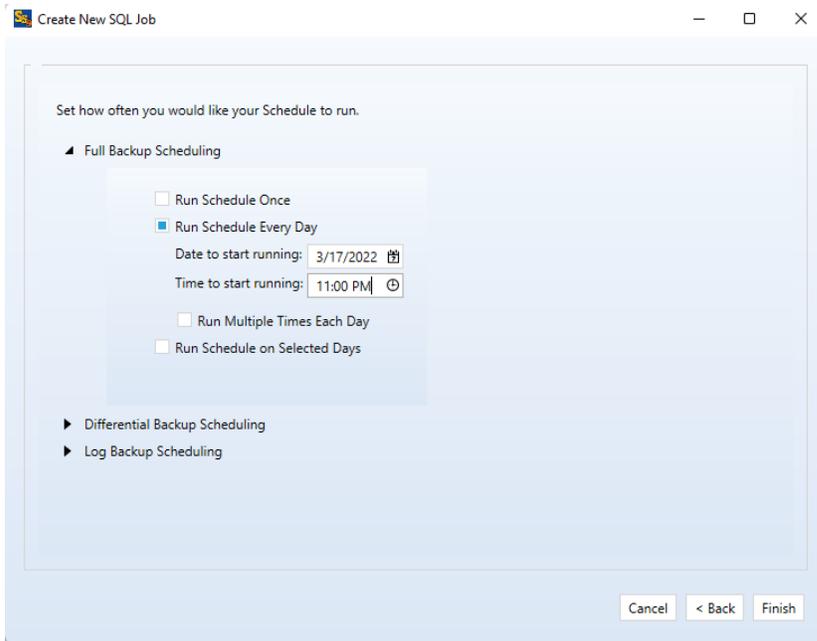
If the same folder structure exists on the destination to support storing the *.mdf and *.ldf files, the default “No Relocation” option can be selected. Otherwise, you must define a folder that exists on the destination server to serve as the storage location for the restored SQL data files.

Define Scheduling Frequency

SQL Protection provides powerful Scheduling of Full, Differential and Log backups. Each backup type can be configured with different repeat settings. For our scenario, a full backup will be performed once per day and a differential backup will be performed every hour.

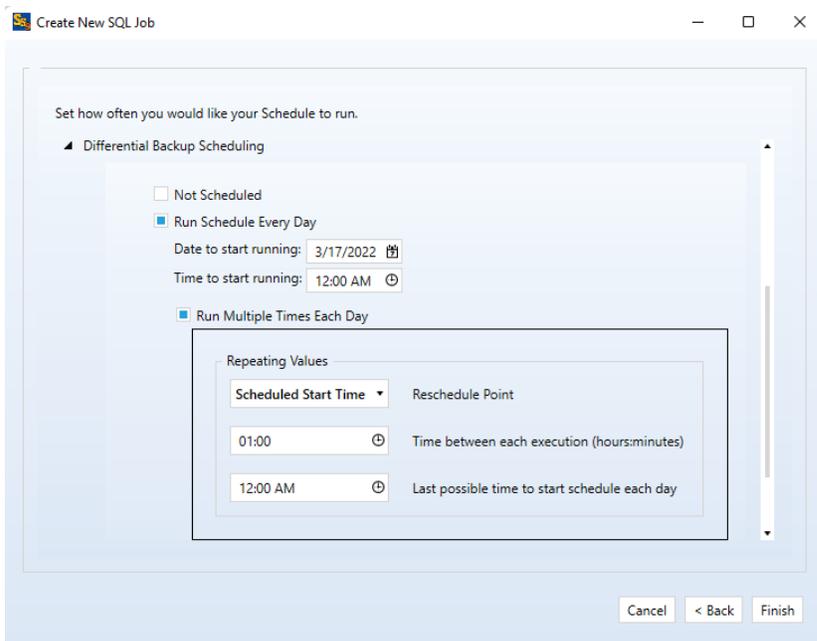
Please note that Log backups are only available when processing a database that is using the full or bulk-logged recovery model. Databases using the simple recovery model can only use Full and Differential backups.

First, we will configure the Full backup scheduling. To perform this task, click the triangle next to “Full Backup Scheduling” to expand the selection:



For the Full backup, we want this to run once per day at 11PM. To configure this, select “Run Schedule Every Day” and enter 11:00PM for the start time.

Second, we will configure the Differential backup scheduling. To perform this task, click the triangle next to “Differential Backup Scheduling” to expand the selection:



For the Differential backup, we want to run it every hour. To configure this, select “Run Schedule Every Day” and check the “Run Multiple Times Each Day” option. For the “Repeating Values” change the “Time between each execution (hours:minutes)” option to 01:00 to tell the Schedule to run the Differential every hour.

It is natural to have times for Full, Differential and Log backups that conflict. In our example scenario, the Full and Differential backups will want to run at the same time once per day at 11PM.

SureSync SQL Protection gives Full backups priority over all other backup types and Differential priority over Log. In this scenario, at 11PM the Full backup will “win” and be selected to run.

Click “Finish” and SureSync SQL Protection will build the SQL Protection Job and Schedule.

Configuration Complete

You have now successfully configured a SQL Protection Schedule and Job! For the Schedule to launch, you must take it off held status by unchecking “Hold Schedule” on the General tab. The SQL database will be backed up according to the scheduling options defined in the Schedule automatically if your Scheduler service is running.

Configuring E-mail Alerts

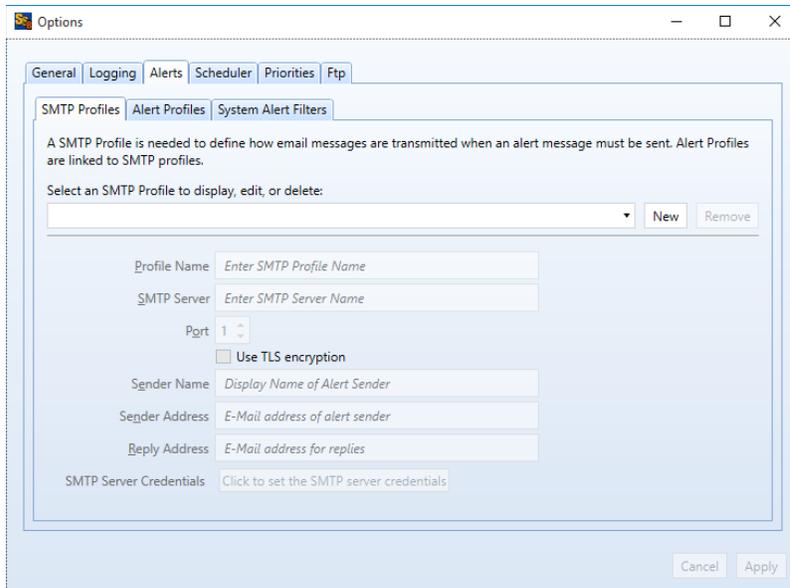
E-mail alerts are part of the Alert system within SureSync and can be a useful way of notifying the SureSync administrator of a problem that should be investigated.

Configuring a SMTP Profile



To start the configuration of e-mail alerts, you should click on the Options () button in the Ribbon bar. Click on the “Alerts” tab to access the appropriate configuration panel.

The first item configured is the SMTP Profile. The SMTP Profile provides SureSync with the necessary details about your SMTP e-mail server so the program can send e-mail messages to you.



To create a SMTP Profile, click on the “New” button on the SMTP Profiles tab.

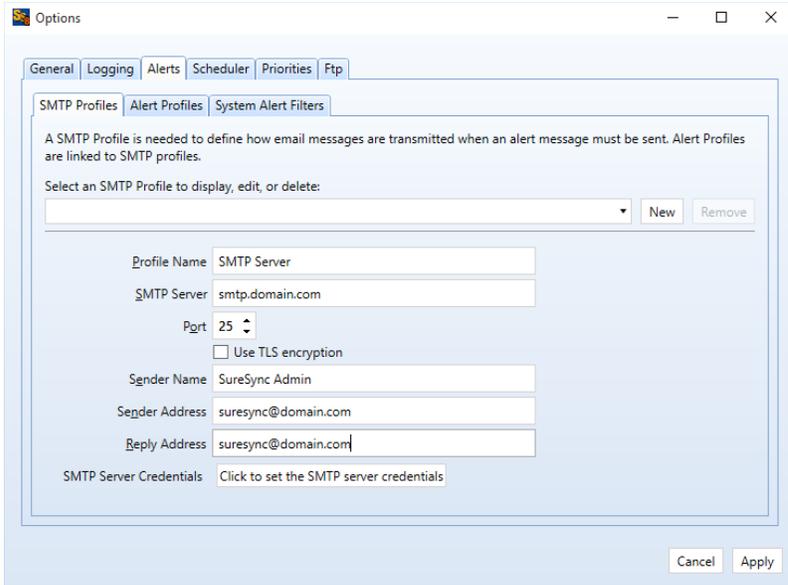
The following fields must be configured:

- **Profile Name:** The name of the profile that you are configuring. For example, you might use the name of the SMTP server.
- **SMTP Server:** The server address of the SMTP server. For example, smtp.mail.com.
- **Port:** The SMTP port for the SMTP server. The default SMTP port is 25.
- **Use TLS Encryption:** If your SMTP server requires encryption, check this box.
- **Sender Name:** The name that will appear in the From: field of messages sent by SureSync.
- **Sender Address:** The e-mail address that will appear on messages sent by SureSync.
- **Reply Address:** The reply address that will appear on messages sent by SureSync.

You can also define SMTP Server Authentication settings if your SMTP server requires a valid logon to send messages (most do). To define a credential, click the ‘Click to set the SMTP server credentials’ button. The following fields will be presented:

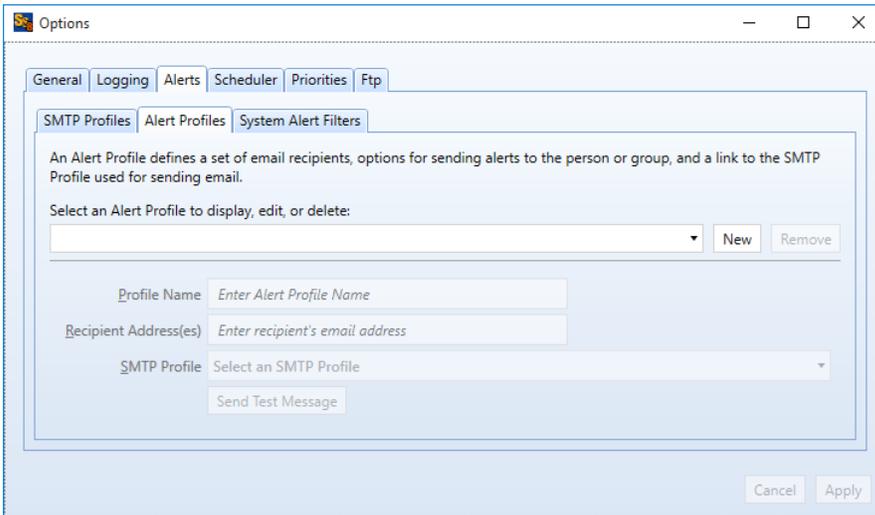
- **User Name:** The username that SureSync will log into this SMTP server with when sending an e-mail alert.
- **Password:** The password for the username defined in "User Name" should be entered here.
- **Repeat Password:** The password for the username defined in "User Name" should be confirmed here.

Click “Apply” to save the SMTP Profile.



Configuring an Alert Profile

The second part of configuring E-mail Alerts is to define an Alert Profile. The Alert Profile defines the e-mail address(s) that will be sent messages when errors occur. To configure an Alert Profile, click on the Alert Profiles tab and click the “New” button.



The following fields need to be configured:

- **Profile Name:** The name of the profile.
- **Recipient Address(es):** The e-mail address(es) that SureSync will send the alert messages to.
- **SMTP Profile:** The SMTP profile that will be used to send the message.

Multiple e-mail addresses can be added by separating the addresses with commas (,) or semi-colons (;). This allows you to easily send messages to a group of SureSync administrators.

Configuring the SQL Protection Schedule to Send Alerts

The final step for receiving e-mail alerts from the SQL Protection Schedule is to define an Alert Filter on the Schedule.

A global Alert Filter can be defined on the Alerts tab of Options. These global Alert Filter settings are inherited by all Jobs, Schedules and Real-Time Monitors defined in the database. Global Alert Filters can be overwritten for a specific Job, Schedule or Real-Time Monitor as appropriate.

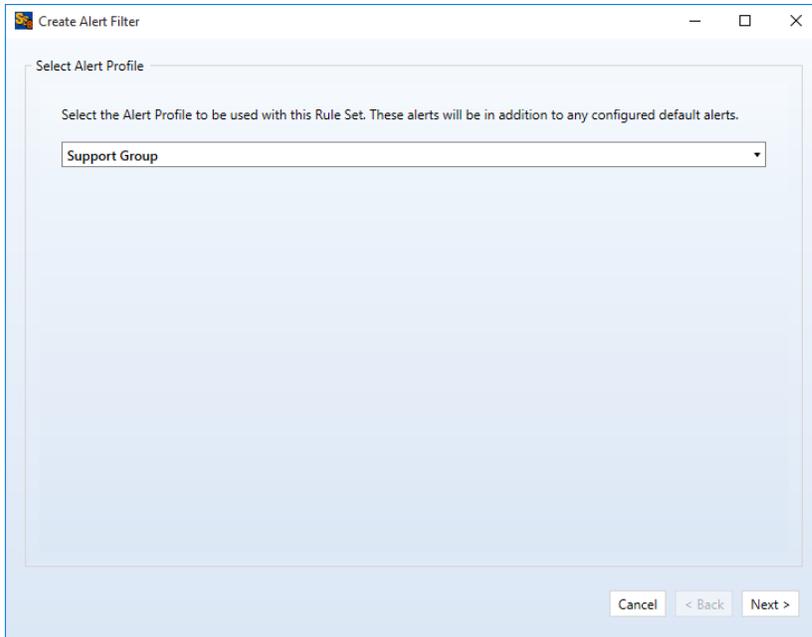
You can also define an Alert Filter on a specific Job, Schedule or Real-Time Monitor. This is what will be done in this guide.

Click on the “SQL Protection Demo Schedule” Schedule and click on the “Alert Filters” tab.

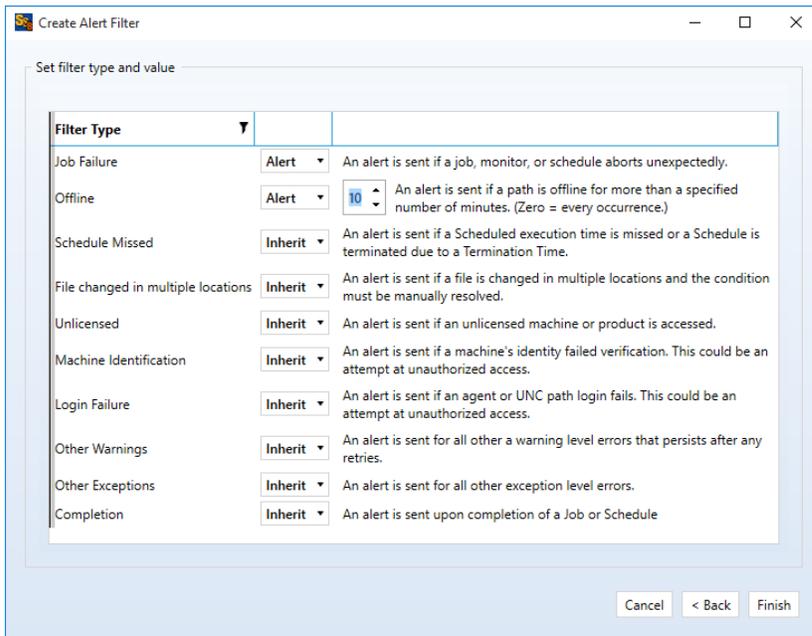


The screenshot shows a software interface for configuring a SQL Protection Demo Schedule. At the top, there is a header with the 'SQL' logo and a search box containing 'SQL Protection Demo Schedule'. Below the header is a ribbon with tabs for 'General', 'Scheduling', 'Scripts', 'Logging', 'Alert Filters', and 'Security'. The 'Alert Filters' tab is currently selected. The main content area contains two paragraphs of instructional text: 'Use the Options item in the desktop ribbon bar to configure and view the default Alert Filters used to control email messages sent on certain types of errors.' and 'This panel will allow you to create Alert Filters that can override individual system defined Alert Filters. Any system defined Alert Filter that is not listed and overridden here will remain in effect.' Below the text is a table with two columns: 'Alert Name' and 'Address'. The table is currently empty. At the bottom of the main content area, there are two buttons: 'New Alert Filter' and 'Delete Filters'. At the very bottom of the window, there are three buttons: 'View Errors', 'Cancel', and 'Apply'.

Click on the “New Alert Filter” button.



From the drop-down menu on the “Select Alert Profile” panel select the Alert Profile you want to configure. Click “Next” to continue.



On the next panel, you can define the types of alerts you would like to receive via e-mail. For a full description of all available options, press F1 to launch the context sensitive help.

Click “Finish” to create the Alert Filter. It will be displayed in the Schedule properties.

SQL Schedule 

General | **Scheduling** | Scripts | Logging | **Alert Filters** | Security

Use the Options item in the desktop ribbon bar to configure and view the default Alert Filters used to control email messages sent on certain types of errors.

This panel will allow you to create Alert Filters that can override individual system defined Alert Filters. Any system defined Alert Filter that is not listed and overridden here will remain in effect.

Alert Name	Address	
Support Group	support@domain.com	