SQL SAFE BACKUP

8.6.1

Table of Contents

1 Hands-free backup across your SQL Servers

- **Save time**. Back up faster than native SQL with dynamic compression
- **Reduce failures**. No disruption during network outages
- **Automate**. Full, differential, and transaction log backups
- **Enterprise management**. Scalable, centralized console and repository
- **Save space**. Intelligent compression across SQL versions
- **Instant restore**. Immediate access to backup data without waiting for a lengthy restore

2 SQL Safe Backup Release notes

IDERA SQL Safe Backup provides a high-performance backup and recovery solution for Microsoft SQL Server. SQL Safe Backup saves money by reducing database backup time by up to 50% over native backups and reducing backup disk space requirements by up to 95%. SQL Safe Backup also enables complete 'hands-free' automated backup of your entire SQL Server infrastructure and ensures compliance with your organization's backup and recovery policies. From implementations with tens of SQL servers to enterprises with hundreds of servers spread around the globe, SQL Safe Backup is the only SQL Server backup and recovery solution that scales to meet the challenge.

To get a quick glimpse into the newest features, fixed issues, and known issues in this release of SQL Safe Backup, review the following sections of the Release Notes:

- New features and fixed issues.
- Previous features and fixed issues.
- Known issues.
- See the list of recommended IDERA Solutions.

2.1 New features and fixed issues

SQL Safe Backup provides the following new features and fixed issues.

2.1.1 8.6.1 New Features

Backup and Restore operations

- SQL Safe Backup now enables compression on SQL Server databases which have Transparent Data Encryption (TDE) enabled.
- SQL Safe Backup now supports the preservation of Change Data Capture (CDC) settings across backup and restore cycles.

Support for Microsoft SQL Server 2019

- IDERA SQL Safe Backup 8.6.1 version now supports Microsoft SQL Server 2019 for all operations.

2.1.2 8.6.1 Fixed Issues

Backup

- SQLSAFE-14263 Users can now perform properly native backups (SQL Server format) to a Microsoft Azure Cloud storage location.

CLI and T-SQL Commands

- SQLSAFE-14275 SQLSAFE-14269 CLI and T-SQL commands are now working properly for all SQL Safe Backup operations.

Policies

- SQLSAFE-14247 Backup operation statuses are now displayed properly in the Policy Status tab.

Restore

- SQLSAFE-14267 Users can now perform properly restore operations using the point in time selection.

2.1.3 8.6 New Features

Azure / Amazon S3 Cloud

- SQL Safe Backup 8.6 version now supports Azure and Amazon S3 subfolders.

Cloud Environment

- Redesigned cloud backup and restore operations for Azure and Amazon S3 storage.

Support for Microsoft Windows 2019

- IDERA SQL Safe Backup 8.6 version now supports Microsoft Windows 2019.

2.1.4 8.6 Fixed Issues

Availability Group

- SQLSAFE-14063 SQL Safe Backup operations are now working properly in availability groups.

Backup

- SQLSAFE-14206 Users can now perform backup operations to an Azure location specifying the container name as "Page blob".

Backup / Restore

- SQLSAFE-11856 SQLSAFE-13585 SQLSAFE-13890 SQLSAFE-13924 SQLSAFE-14009 SQLSAFE-14161 Users can now backup and restore large databases to/from Azure and Amazon S3 locations.
- SQLSAFE-13539 SQLSAFE-14083 Users can now backup and restore to/from Azure and Amazon S3 cloud base storage using subfolders.
- SQLSAFE-14193 SQLSAFE-14194 Users can now perform AdHoc backup and AdHoc restore operations to Amazon S3 and Azure cloud storage using subfolders. They need to make sure to select the Force Restore (replace) option when performing restore operations.
- SQLSAFE-14087 SQLSAFE-14091 Backup and restore operations are now working properly for TSM and Amazon S3 locations.

Cluster Environment

- SQLSAFE-13853 Users can now perform Object Level Recovery and Virtual Database operations to an agent in a clustered environment.

FIPS Compliance

- SQLSAFE-13847 Users can now perform backup and restore operations when enabling FIPS compliance. They can save Azure Pub Account details in the Web and Desktop Console.

Policies

- SQLSAFE-13546 Users cannot run backup policies to a snapshot database.
- SQLSAFE-14192 Users can now perform backup policies using subfolders from azure locations.

Restore

- SQLSAFE-13952 The Restore wizard is now working properly when restoring backed up databases to Amazon S3 locations in a remote instance.
- SQLSAFE-14053 Instant Restore operations to large databases are now working properly when it is restored to a new database or to an existing database using the "Force Restore" option.

Upgrades

- SQLSAFE-14049 After upgrading from SQL Safe Backup 8.5.2 to 8.6, the registration of the Web Console is now successful.
- SQLSAFE-14123 Users can now upgrade the SQL Safe Agent Components from an old SQL Safe Backup product version to 8.6 version properly.

2.2 Previous features and fixed issues

2.2.1 8.5.2 New Features

Backup

- The SQL Safe Management Console has been improved to allow users to perform backups in native (.bak) format.

IDERA Dashboard

- SQL Safe Backup is now integrated with IDERA Dashboard 4.6.0.9.

SQL Safe Web Console

- The SQL Safe Web Console includes an additional tab named Advanced for editing properties of a registered instance.

2.2.2 8.5.2 Fixed Issues

Availability Groups

- SQLSAFE-13888 Users can now perform Full/Differential backups on the secondary replica and Full/Differential policies on the primary replica.

Backup

- SQLSAFE-12892 The verify bakcup option is now working properly when performing backups to TSM.
- SQLSAFE-13438 Users can now backup large databases to Amazon S3.
- SQLSAFE-13521 Scheduled backup jobs are now working properly in a failover cluster environment.
- SQLSAFE-13938 Users can now perform backups with a remote instance using a non-default port number.
- SQLSAFE-13908 Improved user experience with T-Log backup operations.

Cluster Environment

- SQLSAFE-13875 Users can now deploy the Backup Agent onto a Cluster Environment from the SQL Safe Management Console.

Desktop Console

- SQLSAFE-13766 The Last Operation Status section is now updating properly the list of backups succeeded and failed.

Grooming

- SQLSAFE-12842 The repository grooming is now set to 365 days.
- SQLSAFE-13854 The grooming bitesize option is now set to 1000.

Installation

- SQLSAFE-13984 The recovery model and page verify settings are now set properly when installing SQL Safe Backup.

License

- SQLSAFE-13652 The license key is now working properly when adding instances in the Web Console.

Management Service

- SQLSAFE-12850 Log entries are now working properly.

Policies

- SQLSAFE-9560 Restore policies are now working properly when changing the target database name of a restore performed from one server to another.
- SQLSAFE-13406 The Copy Policy feature is now scheduling policies properly.
- SQLSAFE-14060 When performing a backup policy, the location type is now set properly.

Restore

- SQLSAFE-13903 Users can now restore backup files that are moved manually to Amazon S3. Once the file is moved to Amazon S3, rename the file, and perform the restore.
- SQLSAFE-13938 Users can now perform restores with a remote instance using a non-default port number.

T-Log Backup

- SQLSAFE-13908 Improved user experience with T-Log backup operations.

Upgrade

- SQLSAFE-13976 Users can now upgrade manually SQL Safe Backup from 8.4.2 to 8.5.2 using the agent.

2.2.3 8.5.1 New Features

Amazon

- SQL Safe Backup now supports the Amazon AWS GovCloud S3 in the SQL Safe Management Console and SQL Safe Web Console.

Azure

- SQL Safe Backup now supports the Azure Government Blob Storage in the SQL Safe Management Console, SQL Safe Web Console, SQL Safe CLI, and SQL Safe XSP.

Tivoli Storage Manager (TSM)

- SQL Safe Backup now allows users to create striped files when backing up to TSM through the SQL Safe Management Console.

2.2.4 8.5.1 Fixed Issues

Access Files

- SQLSAFE-13737 The SQL Safe Web Console will now prompt users for credentials to access the file system when the initial attempt fails.

Backup

- SQLSAFE-13346 SQL Safe Backup policies automatically performs a full backup operation when a differential backup operation fails due to a lack of a full backup.
- SQLSAFE-12893 The storage options of the Backup again with different options to Tivoli Storage Manager (TSM) are working properly now in the SQL Safe Management Console.
- SQLSAFE-12894 The High Level and Low Level text boxes of the Backup with different options to Tivoli Storage Manager (TSM) are recognizing the data properly in the SQL Safe Web Console.
- SQLSAFE-9555 SQL Safe Backup displays now properly the progress of a backup operations that are being performed without compression.

- SQLSAFE-13892 Backup and restore operations performed in native file format can have timeout values modified through a registry key update.
- SQLSAFE-13225 Percentage values are no longer included in the Result Text of backup and restore operations.

Instant Restore

- SQLSAFE-9722 Resolved an issue that would prevent an InstantRestore operation from completing successfully.

Management Console Preferences

- SQLSAFE-13350 The preferences set on either the SQL Safe Web Console or SQL Safe Management Console are kept the same when performing operations.
- SQLSAFE-13891 Users can now set long passwords for the deployed Backup Agents service account in the Agent Deployment option.
- SQLSAFE-13842 The SQL Safe Management Console is no longer reporting a false failure when deploying the SQL Safe Backup Agent to a remote server.

Object Level Recovery

- SQLSAFE-13577 Users can now perform Object Level Recovery operations to a large backup file.

Operation History

- SQLSAFE-13137 The Database name of every operation performed is kept in the Operation History tab.
- SQLSAFE-12154 The SQL Safe Web Console now displays times in the local time zone.

Policies

- SQLSAFE-13848 Users can now edit SQL Safe Backup Policies which includes unreachable SQL Server instances and/or SQL Safe Backup Agents.

Repository

- SQLSAFE-11932 Improvements to the grooming operation.

Restore

- SQLSAFE-13489 Optimized the mc_GetRestorableDatabaseBackupsets stored procedure which is used to populate the list of backup sets in the SQL Safe Restore wizard.
- SQLSAFE-13839 SQL Safe Backup can now successfully restore large backup files that are stored in an Amazon S3 bucket.

SQL Safe Backup Service

- SQLSAFE-13804 Optimized the SQL Safe Backup Agent to prevent crashes in certain scenarios.

SQL Safe Management Console

- SQLSAFE-13893 The SQL Safe Management Console is now able to connect to the SQL Safe Management Service without any problems.

SQL Safe Scripts

- SQLSAFE-12867 Removed the 3200 character limit when performing an operation through SQL Safe Backup.

SQL Safe XSP

- SQLSAFE-12465 Users can now perform Object Level Recovery operations using the XSP commands.

2.2.5 8.5 New Features

Grooming

- The groom settings for SQL Safe Backup have been improved to allow users to granularly control the amount of operational history that is to be maintained by SQL Safe Backup. This improvement will provide users with more control on maintaining the size of the SQL Safe repository database.

IDERA Dashboard

- SQL Safe is now integrated with IDERA Dashboard 3.8.1.9.

New Installer

- The installer has been improved to allow the installation of all SQL Safe Backup components and the IDERA Dashboard in one single wizard.

SQL Safe Agents view

- The SQL Safe Agents tab now includes SQL Virtual Database properties.

Support

- SQL Safe now supports SQL Server 2017 CU4+ on a Windows operating system.

Virtual Database

- The Virtual Database page has been updated to allow users to easily manage the virtual databases that have been mounted. Users can quickly and easily identify which virtual databases are mounted on which SQL Server instance.

2.2.6 8.5 Fixed Issues

Administration

- SQLSAFE-13282 In the SQL Safe Web Console, users are now able to select the account type and specify the session timeout for the account they are adding in the Manage Users option.
- SQLSAFE-13280 In the SQL Safe Web Console, the user list in Manager Users is now viewable when the list spans several pages.
- SQLSAFE-13274 In the SQL Safe Web Console, users are now able to Add Groups in the Manage Users option.
- SQLSAFE-13228 In the SQL Safe Web Console, users can successfully configure the credentials to be used by the SQL Safe Management Service when connecting to the SQL Safe repository database.

Availability Group

- SQLSAFE-11993 SQL Safe Backup Policies which include databases in an availability group are now accurately reporting the Backups did not start as scheduled status.

CLI Commands

- SQLSAFE-13351 Users can now use the CLI commands for "restorelast" and “xp_ss_restorelast” to restore the latest backup without an associated backup policy.

Event Logs

- SQLSAFE-1620 The SQL Safe Backup Service, by default, is no longer writing entries into the Windows Application Event log when policies are configured to have jobs created by the SQL Safe Backup Agent.

IDERA Dashboard

- SQLSAFE-13483 In the SQL Safe Web Console, users are now able to perform backup operations successfully where the target location uses a UNC path.

Object Level Recovery

- SQLSAFE-13320 The Object Level Recovery functionality is working properly now when the repository is hosted on SQL Server 2014+ instance.

Operation History

- SQLSAFE-13254 In the SQL Safe Web Console, the instance column is now displaying the full instance name on the Operational History page.

Point in Time

- SQLSAFE-13560 The point in time functionality is working properly when mounting a virtual database.

Policies

- SQLSAFE-13366 Users can now add a database to an existing backup policy.
- SQLSAFE-13304 In the SQL Safe Web Console, the filename is now auto-generated when enabling the option to delete backup files in the backup policy wizard.
- SQLSAFE-13194 In the SQL Safe Web Console, the "Access Files As" option is now able to switch to default state after providing different credentials in Restore policy.
- SQLSAFE-13191 When creating or editing a policy in the SQL Safe Web Console, the credentials specified for accessing the file system is now properly applied.
- SQLSAFE-13159 In the SQL Safe Web Console, users can run now a policy manually.
- SQLSAFE-12673 In the SQL Safe Web Console, restore policies are working properly when configured to connect to the SQL instance using SQL Server Authentication.
- SQLSAFE-13513 Users can now supply a UNC path when creating or modifying a policy in the SQL Safe Web Console.

Restore

- SQLSAFE-13448 In the SQL Safe Web Console, users can now perform the verify operation using the restore wizard.
- SQLSAFE-13372 In the SQL Safe Web Console, the point in time selection is working properly now in the restore wizard.
- SQLSAFE-13173 In the SQL Safe Web Console, users are now able to select multiple files for restore through the File System and Target Server options.
- SQLSAFE-12860 The restore wizard now correctly selects the SQL Server instance based on the user selection on the Databases tab when the Repository option is selected.
- SQLSAFE-12049 The restore wizard now displays the list of instances in a sorted order on the Databases tab when the Repository option is selected.

SQL Safe installation

- SQLSAFE-12858 Users can now successfully install SQL Safe Backup while using quotes (") in their service account passwords.

Upgrade

- SQLSAFE-13243 Upgrading will use the existing installation directory rather than using the default path of C:\Program Files\Idera\SQLsafe.
- SQLSAFE-12574 The SQL Safe Agent Components can now be upgraded through the SQL Safe Management Console.

2.2.7 8.4.2 Fixed Issues

Dashboard

- In the Operation History tab, users can now use the Virtual Database filter.

E-mail

- Users can now receive e-mail notifications for Backup/Restore operations performed from the SQL Safe Web Console.
- Users can now update e-mail settings within the IDERA Dashboard.

Licensing

- Updated licensing files that will affect expiring subscription keys (license keys that have an expiration date). If applying a new subscription license key, you must upgrade prior to applying the new key. Perpetual license keys (non-expiring license keys) are not impacted.

Object Level Recovery

- Object Level Recovery Restore Operations are no longer generating the "An item with the same key has already been added" error.

Policies

- When creating/editing a policy through the IDERA Dashboard, users can now set an account for file system access without any problem.

SQL Safe Installation

- When installing SQL Safe Backup, the "Log on as a service" rights are now granted to the accounts specified for the SQL Safe services.
- The installation of SQL Safe Backup is no longer checking for an existing installation of .NET Framework 2.0 as SQL Safe requires an installation of .NET Framework 4.0 or higher.

SQL Safe Re-registration

- Users can now register SQL Safe Backup with the IDERA Dashboard (Dashboard > Administration > Manage Products > Register a Product).

Virtual Database

- Users can mount a virtual database from backup file(s) located on a UNC share, as long as the SQL Safe Backup Service and SQL Safe OLR Service has permissions to the UNC share.
- Mount Operations through the SQLvdb CLI Commands now recognizes the default location settings in the Graphical User Interface (GUI) and do not require the -Move parameter for the database file.

2.2.8 8.4 New Features

Native Backups

- SQL Safe Backup supports SQL Server native backup format for backups along with the ability to restore from native backups.

Virtual Database

- Virtual Database (VDB) is integrated to SQL Safe 8.4.

2.2.9 8.4 Fixed Issues

Add Instances

- When adding an instance to SQL Safe Backup, users can now validate the entered credentials by running Test Credentials button.

Backup Policy

- Users can select databases when adding all instances in the Backup Policy Wizard.

Backup Policy Schedule

- SQL Safe Backup policies configured with the "Monitor + Automatically create backup jobs using the SQLsafe Backup Agent" action, are now running all operations as scheduled.

Cloud Instances

- All Azure restore operations are recorded and displayed in the Operation History tab.

Encrypted Backup

- Users are no longer prevented from completing the SQL Safe Backup Wizard when selecting the "Backup with different options" setting on a backup operation that was previously completed which resulted in an encrypted backup file.

License

- The IDERA Dashboard now displays the SQL Safe Backup License Key information.

Operation History

- Selecting the “Backup with different options” option in the Operation History tab no longer generates exceptions.

Restore Database

- In the Restore Database wizard, the browse button, located in the Databases option under File System tab, is now enabled.
- In the SQL Safe Restore Wizard, selecting a file path using the Browse feature correctly populates the file path that was chosen.

SQL Safe Backup Service

- The installation or the upgrade of the remote SQL Safe Backup Agent service is no longer displaying errors.

SQL Safe Product Migration

- SQL Safe product migration on CWF is no longer an issue.

SafeToSQL Utility

- SafeToSQL Utility can convert .safe backups into .bak backups and it is no longer having .NET versions conflicts.

Virtual Database

- The servers that have SQL Safe Agents installed (Version 8.0.0.423) are no longer reporting “SQLvdb Filter Service version 8.0.0.423: License has expired”.

2.2.10 8.3 New Features

IDERA Dashboard

- SQL Safe is now integrated with IDERA Dashboard 3.0.

Microsoft Azure Blob Storage

- When performing backup and restore operations, users are now able to specify Microsoft Azure Blob Storage as the location for their backup and restore files.

New Installer

- The installer has been improved to allow the installation of all SQL Safe components and the Idera Dashboard in one single wizard.

Policies

- Copying policies is now available from the Policy tab. Users can now copy preexisting policies and modifying their settings before copying them.

Tivoli Storage Manager

- Users can now perform striped backups to tape devices using Tivoli Storage Manager.

2.2.11 8.3 Fixed Issues

- Users do not experience error messages anymore when performing the bulk "Backup Again" operation in the Operation History tab.
- Users can now sort the databases list of the backup wizard according to Database Name, Last Backup, or Space Used.
- The Restore Wizard now updates the point-in-time date accurately after clicking the Apply option.
- "Restore again" and "Restore with different options" are now available for Instant Restore operations in the Operation History tab.
- Links from "Databases with failed backups" and "Databases with failed restores" in the "My Environment" section of the Home tab now direct to the correct information in the Operation History tab.
- Users can now access a tool tip that provides the complete name of SQL Server instances and databases displayed in the bar graphs of the Home tab.
- SQL Safe now supports TSM (Tivoli Storage Manager) 7.1.1.1.
- Users can now use wildcards such as %, _, [ ], [^] to find their information more quickly through the filters.
- Users can now perform the "Restore again" operation as a bulk operation in the Operation History tab without experiencing error messages.
- When upgrading to a new version, users can now select previous SQL Safe display names or create a new one.

2.2.12 8.2 New Features

SQL Safe 8.2 includes the following features in its web console:

Policies

- Administrators and users can now create backup, restore, and log shipping policies with the same capabilities as in the desktop console. The options for launching the Policy Wizards are available from the Home, Policies, Instances, and Databases tabs.
- SQL Safe now allows you to access and edit any policy from the Policies tab.

Cloud Storage

- SQL Safe now includes Amazon S3 Cloud as a storage option for backup and restore operations.

Enhanced Options in the Alerts section

- Alerts related to connection failures allow administrators and users to:
    - Start/Restart SQL Safe Backup Agents.
    - Install/Upgrade the SQL Safe Backup Agents.

- Administrators and users are now able to create backup policies or add databases to existing policies in alerts notifying that databases have not been backed up.

Instances tab

- Administrators and users can now Start/Restart SQL Safe Backup Agents and/or Install/Upgrade the SQL Safe Backup Agents from the instances tab.

SQL Safe Agents tab

- Administrators and normal users are now able to edit the properties of the servers hosting SQL Safe Backup Agents.
- Administrators and users can now find the following options:
    - Install/Upgrade the SQL Safe Backup Agents
    - Enable/Disable SQL Safe Instant Restore

General Preferences

The General Preferences section of the Administration tab now includes the following options:

- The Agent Deployment section allows Administrators to specify the service account to be used for deploying Backup Agents.
- The Policy Data section lets Administrators specify a different path for the policy data files location.
- The Cloud Settings section allows Administrators to set their preferred configuration options for Amazon S3 cloud storage.

Saving Filtered Views

- Users can now save their preferred filters in the Policies, Operation History, Databases, Instances, and SQL Safe Agents tabs.

> As of this release, SQL Safe now requires .NET4.0 or higher and as a result it no longer supports Windows 2000.

2.2.13 8.2 Fixed Issues

- SQL Safe currently shows backup operations progress even when the compression option is not selected.
- If a policy with a differential backup has a database with no previous full backup, SQL Safe takes first a full backup before executing the policy.
- SQL Safe no longer restores logins when the option ***Include database logins in backup file*** is selected for backup operations.
- SQL Safe accurately sends warning notifications for the ***Verify step*** of Backup + Verify Policies.
- SQL Safe has now improved the logic for calculating policy schedules and sending email notifications.
- SQL Safe currently considers the creation date of a new database before sending notifications for missed operations schedules.

2.2.14 8.0 New Features

Web Console Application

Besides the desktop application, a new Web Application Console has been included that allows users to monitor their registered SQL Server instances and their respective SQL Safe operations. The web console provides most of the same capabilities as the desktop application. Actions such as adding new SQL Server instances, backing up, restoring, or performing other operations depend on the role assigned to the respective user.

The main features of this Web Console Application are:

- Home tab - with a general status of the environment through alerts, summary information of SQL Safe operations, managed instances, and other important data.
- Policies tab - with information about all SQL Safe policies existing in the users environment.
- Operations History tab - with of all SQL Safe operations performed in the users environment and their current status.
- Instances tab - with a general view of all SQL Server instances registered in the users environment.
- Databases tab - with general information of the databases that belong to the SQL Server instances registered in the users environment.
- SQL Safe Agents tab - provides information of all computers that host registered SQL Server instances and the details of related Safe Agents.

Integration with the IDERA Dashboard

Version 2.0 allows you to register as many instances of your SQL Safe with your chosen IDERA Dashboard. The **IDER A Dashboard** provides a platform of services that allow an integrated user experience across multiple IDERA products.

User level permissions

- SQL Safe Web Application now allows to assign three different roles to users: Administrator, User, and Guest. Each role has its own specific capabilities and/or restrictions.
- Support for EMC Data.
- SQL Safe now backs up and restores to/from EMC Data Domain.

2.2.15 8.0 Fixed issues

- The collection service is no longer set to run as Local System by default and users no longer need to ensure this account has sysadmin rights in the SQL Server instance hosting the repository. They can specify the service account during installation.

2.2.16 7.4 New features

Support for SQL Server 2014

- IDERA SQL Safe 7.4 now supports SQL Server 2014.

Support for SQL Server Express

- IDERA SQL Safe now supports SQL Server Express in all its editions and versions.

Availability to use the SQL Safe Backup Agent to schedule policies

- Users can now choose between using the SQL Safe Backup Agent for scheduling backup, restore, and log shipping policies or the SQL Server Agent as another option for scheduling these policies. Previous versions only allowed to use the SQL Server Agent to schedule policy jobs but now the user can choose the SQL Safe Backup Agent as a scheduler for these jobs too.

Centralized license management in the Management Service

- License Management has now been centralized in the SQL Safe the Management Service which is in charge of keeping track of those SQL Server instances that are licensed for backup operations. The user specifies which instances they want to license through the License Management view in the Management Console and the Management Service will contact the respective Backup Agents for licensing.
- The new License Management view allows users to add multi-instance license keys with no expiration date. On this view users can see which instances are licensed and which ones are only registered but not licensed yet. Users can manage licenses on this view and select those instances that they need to be licensed.

New Upgrade Installer

- Users can now access upgrade production installers from our Customer Support Portal. These installers are different from the trial installer, which now generates a trial license for unlimited instances with a 14-day expiration key on a fresh install.

Support on Always On Availability Groups

- SQL Safe now supports SQL Server Availability Groups and allows you to perform backup and recovery strategies on your primary and secondary replicas.

2.2.17 7.4 Fixed issues

- You no longer need to restart the InstantRestore service when adding a new drive to a server.

2.2.18 7.2.1 New features

Improved backup performance

- SQL Safe now offers faster backup times when backing up SQL Server instances in the following types of environments:
    - When the instance is hosted on server computer that is experiencing heavy resource loads.
    - When the instance is running on a virtual machine (VM).

2.2.19 7.2.1 Fixed issues

- SQL Safe now ensures backup jobs do not fail when the SQL Safe Backup Agent cannot read the registry on the target SQL Server instance.
- SQL Safe now provides more stability and better performance when using the Management Console.
- SQL Safe now displays the correct file locations for mirrored and striped backup sets in the following windows:
    - View Policy Settings
    - Summary tab of Backup Policy wizard
- SQL Safe now handles NULL values that may be returned from the operating system when an InstantRestore is in progress, allowing the operation to continue.

2.2.20 7.2 New features

New ability to specify mirror backup locations for Log Shipping policies

- You can now store mirrored copies of transaction log backup files in multiple locations, and then select which location should be used as the backup source when the log is shipped. This setting can be configured for each secondary server in the Log Shipping policy.

2.2.21 7.2 Fixed issues

- The SQL Safe Management Console now displays the correct database name when you attempt to cancel an operation and the operation status grid has been sorted by a column other than Start Date.
- When a server is reinitialized for a Log Shipping Policy, the Management Console now correctly refreshes and no longer returns the error “System.InvalidOperationException: Collection was modified; enumeration operation may not execute."
- SQL Safe now retries failed restore operations associated with Log Shipping policies.
- The SQL Safe Backup Agent now incurs minimal performance impacts when running backup operations on servers that have heavy loads.

2.2.22 7.1 New features

Enhanced Log Shipping Policy features

- Log shipping policy enhancements provide a way to specify an alternate network path from where you want the secondary database to pull the file to restore. For additional information about the new log shipping policy features, see Configure Secondary Options window of the Log Shipping Policy wizard.

Enhanced Restore wizard features

- Restore wizard enhancements provide an easier way to restore from a mirrored database when the primary database location is unavailable. For more information about the Restore wizard enhancements, see Backup Sets tab of the Restore Wizard.

Enhanced Cluster support

- SQL safe now supports failovers for Instant Restore during hydration. For additional information about the using SQL Safe in a clustered environment, see Using SQL Safe on a Microsoft Windows cluster environment.

SQL Server 2012 experimental support

- SQL Safe 7.1 is SQL Server 2012 RC0 compatible. This version of SQL Safe is not certified against newer builds of SQL Server and should not be used with these builds in a production environment. IDERA provides experimental support while you use your installation in a testing environment to ensure the features you rely on most are working as or better than expected.

2.2.23 7.1 Fixed issues

- Users who have a case-sensitive SQL Server user name no longer finds SQL Safe failing to create a job. This issue was the result of SQL Safe adding the characters in an all lowercase format.
- SQL Safe now prompts for a user name and password if a user runs the SQL SafeCmd DELETE command when they do not have access to the remote file system.
- This release fixes an issue that caused some users attempting to upgrade to SQL Safe 7.0 to receive a message stating that a previous version already exists.
- An issue preventing the backup service from starting after some users upgraded to SQL Safe 7.0 is resolved.
- Restore policies no longer become out of sync with the database for users who have multiple data files.
- An issue occurring during an upgrade that prevented backups from starting is resolved. In the job history, this issue logged an error stating, "The process could not be created for step 1 of job *X* (reason: The system cannot find the file specified). The step failed."
- A data format issue that caused some users to see a number of their policies as "Not Loaded" is fixed. All policies should successfully appear after any SQL Safe version upgrade.
- SQL Safe now prompts the user to sync policies after changing SQL authentication details for a SQL Server.

2.2.24 7.0.2 Fixed issues

- SQL Safe applications and services no longer experience a long delay when starting if Windows cannot verify the Authenticode signature on the associated applications and services.
- SQL Safe Agent deployment no longer fails due to an issue that occurs when accessing the registry on the remote machine during installation.
- The SQL Safe Management Console now properly handles creating and re-initializing log shipping databases that include several data files.
- The SQL Safe Management Console now properly handles creating restore policies for databases that include several data files.

2.2.25 7.0.1 Fixed issues

- The SQL Safe Today page now accurately displays the status for each item on the policy list and includes the status for operations that occasionally did not appear because the UTC offset was set to hours instead of minutes.

2.2.26 7.0 New features

Access your database quickly during a restore

- SQL Safe gives you the option to bring your database back online quickly when performing a restore. The InstantRestore feature lets you work on restoring a database while allowing users to perform read and write operations to the database during this process. InstantRestore is available only for restoring full databases and does not support a restore of individual files or file groups.

Automatically run a Full backup prior to a Diff/Log backup

- SQL Safe simplifies the initial setup process by automatically detecting and performing a Full backup prior to a Differential or Transaction Log backup.

2.2.27 7.0 Fixed issues

- The **Retry reading backup files after network errors** check box on the Backup Sets tab of the SQL Safe Database Restore wizard is renamed **Enable network resiliency**. The functionality remains the same while the name of the field was changed to improve usability.
- The **Verify (checks integrity, no data restored)** option moved from the Recovery State tab to the Target tab in the SQL Safe Database Restore wizard.
- SQL Safe now properly displays the selection in the **Select backup sets manually** box when the user switches from one database to another using the Restore wizard. This issue affected users attempting to restore multiple databases.
- New rolling logs improve troubleshooting by avoiding issues that result when a single log file continues to store information and grows without a limit. This file can cause performance issues and be hard to search for clues to find the issue you are trying to resolve. This feature is recommended for use only when instructed by IDERA support.
- SQL Safe Reporting no longer displays an error message when a user attempts to run the Last Backup report.
- Dependent SQL Safe operations in a series are now associated so that when one of the operations fails, all of the following operations are canceled.
- The Restore wizard now properly handles the LSN chain when there is an intermediate Full backup.
- Performance updates improve the speed of the SQL Safe installation.
- Enhancements to SQL Safe memory usage decrease the chance of memory leaks or fragmentation.
- Users can now backup a database using only one thread as specified in the Thread Count field on the SQL Safe Backup Policy wizard Options tab.
- SQL Safe alerting now properly handles log shipping restore schedule start times when set to a non-default value.
- SQL Safe network retry updates fixed an issue that resulted when the SQLAgent job hung during a backup and the network retry function is disabled.
- SQL Safe no longer causes an extreme load on the tempdb while backing up a database.
- SQL Safe now properly handles FQDN names when connecting to the Backup Service.
- SQL Safe no longer truncates text within the **Result Text** field.
- Users can now sort the list of databases in the SQL Safe Backup wizard by clicking the appropriate column name.
- SafeToSQL users who submit an encryption password that fails verification now receive the correct error message.

- The Backup Policy wizard no longer re-runs the file access check after a user edits the policy unless the change was made to the Location tab.
- SQL Safe now allows encryption passwords of more than 24 characters. This update allows users to implement pass phrases as a more effective method of security.
- Users no longer experience an issue causing the default SQL Server instance file path to change when re-running a failed or skipped backup.
- SQL Safe now features **Cancel** buttons in a number of areas available when the user runs a task. You can cancel a task when performing log shipping re-initialization, deleting a policy, enabling or disabling a policy, running a job, re-synching a policy, or updating a license.
- SQL Safe now prompts the user immediately after a user account credential check fails.
- Improved Command-line Interface (CLI) documentation regarding SQL Safe and TSM server is located in the SQL Safe Help topic, Back up to the TSM Server.
- Accessing sample Command-line Interface (CLI) and Extended Stored Procedure (XSP) script sample access is now documented in the Product components and architecture
- Users who submit a script that contains unnecessary backslash characters in the file path no longer receive an error message stating, "Value cannot be null." SQL Safe now omits the unnecessary backslash characters and continues the operation.
- Users can now create and run log shipping or restore policies on a database that contains multiple files on different drives.
- The SQL Safe Management Console now contains the proper certificate so the user no longer receives a request for credentials each time they launch the Console. This issue affected Windows 2008 users replying on user account control functionality.
- The default **Connection Settings** detail on the SQL Safe Backup wizard Locations tab no longer retains any changes made during previous use.
- Users can now quickly find an instance in the SQL Safe Database Restore wizard Databases tab by typing the name directly into the instance field and selecting the appropriate instance when it appears.
- Users can now successfully change the IP address on the server hosting the SQL Safe Management Service without causing IP address resolution issues with the Backup Agent.
- Users attempting to restore an older, inactive backup file stored on a TSM server no longer receive an error message.
- SQL Safe XSP now correctly handles wide-character Unicode file names.
- Updated file access permissions fixed a performance issue caused when SQL Safe ran the Backup Policy File Access Check on every database within a SQL Server instance.
- SQL Safe performance is improved when a user attempts to create or run a policy or load the policy status pages.
- Users with large SQL Safe repositories no longer encounter a timeout when accessing the Backup Sets page in the Management Console during a restore.
- If a backup policy specifies both a FULL as well as DIFF or LOG backups to be performed, the FULL backup is automatically run for new databases that have no previously-performed FULL backup existing at the time the FULL, DILL, or LOG backup is scheduled, whichever occurs first.
- The SQL Safe Management Console no longer prevents users from deleting some policies. These policies failed when loading from the Repository before the user attempted to delete the policy.
- PDF files of the SQL Safe Help and SQL Safe Release Notes now include hot links to access related information within the document.

2.3 Known issues

IDERA strives to ensure our products provide quality solutions for your SQL Server needs. The following known issues are described in this section. ***If you need further assistance with any issue***, please contact Support.

2.3.1 Known Issues for 8.6.1

Management Service

- The Management Service may leave open SQL Server connections in an 'sleeping' state, until the SQL Server or SQL Safe Management is restarted. In extreme cases, the number of open connections may cause the number of concurrent connections to reach the maximum allowed by SQL Server. Please contact Support fo r further assistance.

Policies

- Email notifications are not sent for Restore policy failures. Please contact Support for further assistance.
- When creating a restore policy from a cloud backup, specifying the Temporary Download Location path may produce an error if the SQL Server is a named instance. Please contact Support for further assistance.

Restore

- After a successful restore using the 'Change Path' option to change the data and log file paths, the 'Restore Again' feature does not show the altered paths.
- When performing restore operations from Microsoft Azure Cloud with T-SQL and CLI commands, the WITHMOVE option is broken.
  The error presented is:
  Value cannot be null.
  Parameter name: sqlUsername

2.3.2 Known Issues for 8.6

Backup

- When saving native format SQL Server backups and storing to Azure cloud storage, customers need to confirm that their Azure accounts are of the type "General Purpose". This is not a SQL Safe Backup limitation but a Microsoft mandate, please refer to the Microsoft Support document.
- On SQL 2012 instances, databases may be skipped from jobs without displaying its skipped status or an error in the Policy Status view. Please contact Support for further assistance.

Cluster Environment

- When a Virtual Database is mounted to a cluster and a failover occurs, the Virtual Database entry will disappear from the Virtual Database list in the Web Console. The Virtual Database will still be present and functional on the instance but will no longer be seen in the SQL Safe Virtual Database list.

Policies

- Statuses for deleted databases are not removed from the Last Operation Status section of the Policy Status view. Please contact Support for further assistance.
- When performing a restore policy from Amazon S3 location for a large database with the "Download File from Cloud" option unchecked, the SQL Safe Backup product will download the file from the cloud and choose the SQL Server data folder default destination.

Restore

- Restore operations to TSM Striped Files may result in an error message reporting that the second striped file could not be found.

Upgrades

- When upgrading from Microsoft Visual C++ 2015 to Microsoft Visual C++ 2017 version 14.14 there is a bug in the Microsoft upgrade that incorrectly removes the registry key that the SQL Safe installation is checking. This bug is detailed in the following Developer Community. The bug was fixed in later versions of Microsoft Visual C++, so the workaround is to manually install the latest Microsoft Visual C++ version from the latest supported Visual C++ downloads.

2.3.3 Known Issues for 8.5.2

IDERA Dashboard

- Setting email notification configurations via the web UI may show an error "Failed to decrypt password" when attempting an ad-hoc backup from the web UI.

  > Email notification configurations set through the desktop client are not affected by this bug.

Installation

- SQL Safe Backup can be installed without the IDERA Dashboard on an Operating System 2003.
  Running the combined installer will display the following message: *"The current Operating System is not supported by this product"*.
- The remote installation of SQL Safe Backup agent fails in scenarios where the Microsoft Visual C++ 2017 Redistributable or greater is previously installed in the machine.
  To install SQL Safe Backup agent properly, log onto a machine directly and install the Agent\Management components.

Policies

- Statuses for deleted databases are not removed from the Last Operation Status section of the Policy Status view. Please contact Support for further assistance.

Upgrades

- After upgrading from SQL Safe Backup 8.5.1 to 8.5.2, the Web Console may need to be re-registered. Go to the product registration in the Web Console, delete the SQL Safe Backup 8.5.1 product, and register the SQL Safe Backup 8.5.2 product.
- Email notifications for SQL Safe Backup operations are not generated after upgrading to SQL Safe Backup 8.5.2. Please contact Support for further assistance.
- The upgrade to SQL Safe Backup 8.5.2 may fail if your SQL Safe Backup installation uses a repository with a non-default name. Please contact Support for assistance in completing the upgrade.
- The upgrade to SQL Safe Backup 8.5.2 may fail for customers who have previously backed up the system database TempDB using SQL Safe Backup. Please contact Support for assistance in completing the upgrade.

2.3.4 Known Issues for 8.5.1

Installation

- The installation of SQL Safe Backup fails in scenarios where the Microsoft Visual C++ Redistributable requires Operating System updates.
  The following patches are required depending on your Operating System:
  - For Windows 8.1, install KB2999226 which has a pre-requisite of KB2919355.
  - For Windows Server 2012 R2, install KB2919355 which has a pre-requisite of KB2919442.

Policies

- Statuses for deleted databases are not removed from the Last Operation Status section of the Policy Status view. Please contact Support for further assistance.

Upgrades

- After upgrading from SQL Safe Backup 8.5 to 8.5.1, the Web Console needs to be re-registered. Go to the product registration in the Web Console, delete the SQL Safe Backup 8.5 product, and register the SQL Safe Backup 8.5.1 product.

2.3.5 Known Issues for 8.5

Backup

- SQL Safe Backup Policies only supports availability groups where the Backup preferences are configured for the Primary Replica. If configured with any other option, Full Backup operations will result in an error when running against any Secondary Replica.

Installation

- The installation of SQL Safe Backup fails in scenarios where an existing SQL Safe Backup database already exists.
  To avoid a failure, make sure that the specified name of the SQL Safe repository database matches the existing database.

Policies

- The SQL Safe Management Console can become unresponsive when editing a policy that contains a server where the SQL Safe Backup Agent is unreachable.
- Statuses for deleted databases are not removed from the Last Operation Status section of the Policy Status view. Please contact Support for further assistance.

SQL Safe Backup Agent

- When deploying the SQL Safe Backup Agent components from the SQL Safe Management Console, an error message appears indicating an installation failure. Refreshing the SQL Server instance will detect if the SQL Safe Backup version or the deployment was successful. Deployments of the SQL Safe Backup Agent components from the SQL Safe Web Console does not present this issue.

2.3.6 Known Issues for 8.4.2

Installation

- The SQL Safe Filter Service and SQL Safe OLR Service can be installed on Windows version 10 and later once you disabled the Secure boot (in the VM properties for a VM or the BIOS settings for a physical machine).

Object Level Recovery

- The Object Level Recovery feature is dependent on an existing installation of SQL Server 2008 R2 Management Objects from the Microsoft SQL Server 2008 Feature Pack.

Policies

- Statuses for deleted databases are not removed from the Last Operation Status section of the Policy Status view. Please contact Support for further assistance.

2.3.7 Known Issues for 8.4.1

Native Backup

- When submitting an ad-hoc native backup operation against SQL Safe Agents prior to 8.4 version, the operation will not perform any work. Native backup operations are only supported for SQL Safe Backup Agents 8.4 version and later.

2.3.8 Known Issues for 8.4

Cloud Support

- When backups fail while using Windows Azure, partial files will remain unless the user deletes them manually.

IDERA Dashboard

- SQL Safe Backup does not currently support IDERA Dashboard's tag management.
- Instances removed from web and desktop console are still displaying under Dashboard Managed Instances.
- IDERA Dashboard does not support Microsoft Windows 2003.

Installation

- When installing or upgrading SQL Safe Backup, the installer is not able to parse SQL Server instance names with a space within the entered name. For example, use "SERVER,1433" instead of "SERVER, 1433".

Licensing

- The licenses users add should be valid for the Centralized Licensing model. When Centralized Licensing is enabled, all licensing information is managed using the SQL Safe Management Service and each of the individual SQL Safe Backup Agent licenses are removed.

Object Level Recovery

- The SQL Safe Management Service and SQL Safe Backup Agent must be upgraded to 8.4 version in order to run Object Level Recovery successfully.
- Object Level Recovery operations can fail when the SQLsafe repository database is hosted on a SQL Server 2014 or SQL Server 2016 instance with the following error: "Could not load file or assembly 'Microsoft.SqlServer.Smo, Version=10.0.0.0, Culture=neutral, PublicKeyToken=89845dcd8080cc91' or one of its dependencies. The located assembly's manifest definition does not match the assembly reference. (Exception from HRESULT: 0x80131040)". The recommended workaround is to use the InstantRestore or Virtual Database feature to recover the database in an expedited fashion.

Policies

- Users should take into account that time fields for the Policies, Operation History, and Databases tabs in the SQL Safe Web Console are currently expressed in the GMT time zone.
- In the web console Policies tab, options to start jobs and disable policy are not available.
- SQL Server Agent service account authentication for the Network path provided as Primary location in Log Shipping Policy wizard fails with an error even though the Service account has full permissions to the shared path.

TSM

- Low-level file selection is not working properly with TSM backup/restore.

Virtual Database

- SQL Virtual Database does not support compressed native backup from SQL Server 2014 and SQL Server 2016 instances. Attempting to mount these kinds of backups results in the following error: "The header for the backup archive cannot be read. The file may not be a valid backup archive".
- When using an appended SQL Safe Backup file, mounting a virtual database automatically selects the first backup set within the backup file.

Other Issues

- After registration, SQL Safe Web Console will list all available instances in the repository. Users have to use the options to Bulk Edit credentials to change credentials for those instances they are not able to monitor.
- The SQL Safe Web Console is not currently sending e-mail notifications for Restore Operations.
- "Agent deployment service account" changes made in the desktop console are not replicating in the web console.

2.3.9 Known Issues for 8.3

Cloud Support

- When backups fail while using Windows Azure, partial files will remain unless the user deletes it manually.

IDERA Dashboard

- Safe does not currently support IDERA Dashboard's tag management.
- Product registration at the IDERA Dashboard is currently not supported in this release.
- Instances removed from web and desktop console are still displaying under Dashboard Managed Instances.

Installation

- When installing or upgrading SQL Safe, the installer is not able to parse SQL Server instance names with a space within the entered name. For example, use "SERVER,1433" instead of "SERVER, 1433".

Licensing

- The licenses users add should be valid for the Centralized Licensing model. When Centralized Licensing is enabled, all licensing information is managed using the SQL Safe Management Service and each of the individual SQL Safe Backup Agent licenses are removed.

Policies

- Users should take into account that time fields for the Policies, Operation History, and Databases tabs in the SQL Safe Web Console are currently expressed in the GMT time zone.
- In the web console Policies tab, options to start jobs and disable policy are not available.
- SQL Server Agent service account authentication for the Network path provided as Primary location in Log Shipping Policy wizard fails with an error even though the Service account has full permissions to the shared path.

TSM

- Low-level file selection is not working properly with TSM backup/restore.

VDB

- Users who plan to install Safe 8.0 on the same system as VDB should obtain the VDB 2.1 buddy drop which addresses compatibility issues between the two products. Please contact IDERA Support for more information.
- After installing SQL Safe, existing virtual databases may not be accessible. IDERA recommends to first upgrade with the VDB 2.1 buddy drop, unmount virtual databases, install SQL Safe, and then recreate the respective VDBs. Contact IDERA Support for more information.
- When uninstalling SQL Safe, users may need to reboot their computers to remove the SQLvdb Filter Service.

Other Issues

- After registration, SQL Safe Web Console will list all available instances in your repository. Users have to use the options to Bulk Edit credentials to change credentials for those instances they are not able to monitor.
- The SQL Safe Web Console is not currently sending e-mail notifications for Restore Operations.
- Agent deployment service account changes made in the desktop console are not replicating in the web console.

2.3.10 Known Issues for 8.2

IDERA Dashboard

- Instances removed from web and desktop console are still displaying under Dashboard Managed Instances.

Policies

- SQL Server Agent service account authentication for the Network path provided as Primary location in Log Shipping Policy wizard fails with an error even though the Service account has full permissions to the shared path.
- Users may find the option for wildcards is not available when defining databases to backup in the respective policy wizard.
- In the web console Policies tab, options to start jobs and disable policy are not available.

TSM

- Low level file selection is not working properly with TSM backup/restore.

VDB

- Users who plan to install Safe 8.0 on the same system as VDB should obtain the VDB 2.1 buddy drop which addresses compatibility issues between the two products. Please contact IDERA Support for more information.
- After installing SQL Safe, existing virtual databases may not be accessible. IDERA recommends to first upgrade with the VDB 2.1 buddy drop, unmount virtual databases, install SQL Safe, and then recreate the respective VDBs. Contact IDERA Support for more information.

Other Issues

- After registration, SQL Safe Web Console will list all available instances in your repository. Users have to use the options to Bulk Edit credentials to change credentials for those instances they are not able to monitor
- Users should take into account that time fields for the Policies, Operation History, and Databases tabs in the SQL Safe Web Console are currently expressed in the GMT time zone.
- When uninstalling SQL Safe, users may need to reboot their computers to remove the SQLvdb Filter Service.
- The SQL Safe Web Console is not currently sending e-mail notifications for Restore Operations
- Agent deployment service account changes made in the desktop console are not replicating in the web console.

2.3.11 Known issues for 8.0

- To upgrade from build 8.0.0.423 to Hotfix build (8.0.0.503) users need to follow these steps:
  - Open the installer to prompt the upgrade.
  - Type the credentials for the newly created SQLsafe REST service.
  - When upgrading the instance of SQL Safe, the installer displays the value to be added to the location property.

    - Access the web application. When an "exception" message is displayed, select the option "Redirect me to Dashboard."
    - Go to the Administration tab of the IDERA Dashboard, select Manage Products, and edit the respective SQL Safe instance. In the Location field, add the value specified during installation.
    - Restart the IDERADashboardCoreService (Automatically the IDERADashboardWebAppService service is restarted.)
- After registration, SQL Safe Web Console will list all available instances in your repository. Users have to use the options to Bulk Edit credentials to change credentials for those instances they are not able to monitor.
- Users should take into account that time fields for the Policies, Operation History, and Databases tabs in the SQL Safe Web Console are currently expressed in the GMT time zone.
- The Progress bar of the Operation History tab in the SQL Safe Web Console does not update its progress during the operation execution.
- When uninstalling SQL Safe, users may need to reboot their computers to remove the SQLvdb Filter Service.
- The SQL Safe Web Console is not currently sending e-mail notifications for Restore Operations.
- When uninstalling SQL Safe, users should manually remove the product from their IDERA Dashboard before rebooting their system.
- Users who plan to install Safe 8.0 on the same system as VDB should obtain the VDB 2.1 buddy drop which addresses compatibility issues between the two products. Please contact IDERA Support for more information.
- After installing SQL Safe, existing virtual databases may not be accessible. IDERA recommends to first upgrade with the VDB 2.1 buddy drop, unmount virtual databases, install SQL Safe, and then recreate the respective VDBs. Contact IDERA support for more information.

2.3.12 Known issues for 7.4

- When a backup operation in SQL Safe is performed at the same time as the native SQL Server, the successful backup job on SQL Safe may not always show the correct timestamp in the file name of the repository
- When upgrading from an older version, the user may experience Log Shipping policies with an "out of date" message in the console. Clicking the "out of date" link will fix this issue.
- Instead of being assigned the default location set in Preferences, users may find that location paths of stripped files are the same as that of the mirror paths when changing from single to stripped location type.
- Users may find that pressing the "Enter" key in the Backup, Restore and Log Shipping wizards may lead them to the consecutive pages instead of inserting new lines in fields as it is done in the Backup Policy Wizard.
- When setting up a log shipping wizard with a cluster instance, users may find that secondary database file location does not display the same path as the one configured in the respective wizard but it displays the location from the primary database.
- Users may get a Last operation status of "Backups did not start as scheduled" in policies that are configured to run full and differential backups at specific times and where backups are done with no compression and no encryption.
- When running Instant Restore, users may experience problems if they have the same drive mounted as a drive letter and as a folder and they are using both paths for the Instant Restore procedure: the backup file accessed via the drive letter and the data files accessed via the folder path.
- Users may experience timeouts with the Instant Restore processes over a SQL Server 2012 SP1 with cumulative updates.
- Users may find that when the InstantRestore process is running in a clustered SQL Server and a failover occurs during the Hydration process, the Management Console displays the InstantRestore and Hydration processes as halted. The operation will not complete until the cluster is failed back to the original node where the operation was started.

- Users that select the SQL Safe Backup Agent to create policies on servers where the timezone has been changed may need to restart the SQL Safe Backup Agent service to update the timezone and ensure policies run on the correct schedules.

2.3.13 Previous known issues

SQL Safe Repository no longer supports SQL Server 2000

SQL Safe Repository no longer supports SQL Server 2000. Supported versions include:

- SQL Server 2008 R2
- SQL Server 2008 Standard and Enterprise Editions
- SQL Server 2005 Standard and Enterprise Editions SP1 or later

SQL Safe no longer supports Itanium

SQL Safe 7.0 and later does not support the Itanium processor architecture. For more information, see the software requirements.

Pentium II processors are not supported

You should not install SQL Safe on a computer running a Pentium II processor. For more information, see the hardware requirements.

User must select the SQL Server hosting the Repository when using the Maintenance wizard

Users of the SQL Safe Maintenance wizard to modify, repair, or remove this version of SQL Safe must click **Browse** to select the current SQL Server hosting the Repository in the SQL Safe Repository window of the wizard. The wizard does not let you continue until an entry appears in the **SQL Server hosting the Repository** field.

Backup file names that use the %timestamp% macro may change when upgrading to SQL Safe 6.5 or later

When some users upgrade to SQL Safe 6.5 or later, the backup file names using the `%timestamp%` macro may change. This issue affects users who have SQL Safe groom their backup files at backup time, using either the `-delete` command line option or the **Remove files older than** option in the Backup Policy wizard. Previous versions expand `%timestamp%` to the UTC time of the backup.

Beginning with SQL Safe 6.5, `%timestamp%` expands to the local time of the backup. As a result, SQL Safe may write new backups to files already created by an earlier version of SQL Safe immediately after upgrading. By default, SQL Safe appends to backup files and this issue does not occur as the new backup appends to the existing file. This situation resolves itself after the time difference between UTC and local time passes. For example, this issue is resolved after five hours in the Central Standard Time zone (US).

Note that if you specify to overwrite, SQL Safe overwrites the existing files instead of appending the new information. If you upgrade from a release earlier than SQL Safe 6.4, appends fail and display an error message.

Setup program removes previous version when upgrade fails

If the upgrade fails while you are upgrading from a previous version of SQL Safe, the setup program removes the previous version from the SQL Server computer on which you attempted the upgrade.

XSP installation fails on clustered SQL Server instances

When you use the Agent Only install to manually deploy the SQL Safe Backup Agent to a clustered SQL Server instance, the corresponding SQL Safe XSP installation will fail. After the Backup Agent install completes, you can manually install the SQL Safe XSP.

For more information, see the Using the SQL Safe XSP Technical Solution located in the Documentation folder (by default, `C:\Program Files\IDERA\SQL Safe\Documentation`).

Remote Backup Agent install fails when SQL Server is not installed

In order to install the SQL Safe Backup Agent remotely, the computer from which you install SQL Safe must have a version of SQL Server already installed. For more information, see the software requirements.

Table Restore wizard is no longer available in SQL Safe version 6.0 or later

To restore objects and data from your backup files, use the new IDERA SQL virtual database tool. For more information, see Recover objects using Virtual Database.

FIPS-compliant encryption no longer requires additional software when installing SQL Safe version 6.0 or later

In a FIPS-compliant environment, SQL Safe uses only FIPS-compliant algorithms to encrypt your backup files. These encryption methods do not require any additional software. For more information, see Ensure FIPS compliance.

Upgrade any Backup Agents that perform TSM backups

Due to the extensive TSM enhancements included in SQL Safe 6.4 and later, older Backup Agents are not compatible with 6.4. To ensure you can continue backing up your SQL Server data to TSM, upgrade any Backup Agent that is used to perform TSM backups in your environment.

64-bit users need additional steps to install reports

Users with 64-bit installations must follow different steps to install reports. For more information, see IDERA solution 3891, "Where do I find the SQL Safe reports," in the knowledge base on Support (www.IDERA.com/support).

SQL Safe 4.0 users who upgrade to SQL Safe 7.1 or newer cannot use existing backup policies as part of new restore policies

SQL Safe 4.0 users who upgrade to SQL Safe 7.1 or newer receive error messages if they attempt to create and then run a restore policy that includes a backup policy created on the earlier version of SQL Safe.

SQL Safe Management Service logging multiple grooming events per day

Some users may notice the SQL Safe Management Service logging multiple grooming events in the Windows Application log each day. SQL Safe should be logging only one such event per day.

Attempting to restore a database from the list of backups on the SQL Server details page fails

A failure results when you attempt to restore a database file by right-clicking a file backup in the Backup/Restore Operation Status list and select **Restore Database**. To avoid this issue when restoring a file backup, click **Restore > Database Files** from the menu and complete the available restore wizard. You can also access the wizard from the Servers tree by right-clicking the appropriate SQL Server instance and selecting **Restore Database(s) Files**.

InstantRestore performance is affected by whether the SE_MANAGE_VOLUME_NAME privilege is on your SQL Server

Enabling the SE_MANAGE_VOLUME_NAME privilege for your SQL Server account improves general SQL Server file I/O performance as well as SQL Safe InstantRestore. If this privilege is not enabled for the SQL Server Service, InstantRestore performance could be negatively impacted, just as with SQL Server itself. The degree of impact varies depending on environmental conditions. For more information about SQL Server Instant File Initialization, see the Microsoft Knowledge Base article located at Database Instant File Initialization.

InstantRestore appears to stall when restoring databases that contain read-only file groups

SQL Safe 7.0 Beta hydration appears to stall at 99% complete when restoring databases that contain read-only file groups. SQL Server triggers InstantRestore hydration when it performs read/write I/O on the database files. Because SQL Server does not perform read/write I/O on the read-only files, hydration does not begin. Eventually, hydration begins when SQL Server performs read I/O on the files. You can delete the database if you experience this issue.

Adding a new drive requires you to restart the InstantRestore Service

When you add a new drive to a server, you must restart the SQL Safe Filter Service to make sure that the SQL Safe Filter driver is attached to the new drive. When the SQL Safe Filter Service starts, it attaches the SQL Safe Filter driver to all the fixed drives on the server. If you add a new drive after the service starts, the driver is not attached and any files created on this drive during InstantRestore do not function correctly. To avoid this issue, simply restart the SQL Safe Filter Service after adding any new drive.

Not all files are removed when you delete a database restored using InstantRestore

Some files may remain after you attempt to delete a database previously restored using the InstantRestore feature. In most cases, you can manually delete these mdf, ndf, ldf, and vbm files. If the files are locked, restart either the SQL Safe Filter Service or the SQL Server Instance and then delete the files manually.

Offline SQL Safe Web Help may display a blank page

Some users experience a blank page when pressing F1 and using the offline SQL Safe Help. If this issue occurs, access the online version of SQL Safe 7.1 Help at http://www.IDERA.com/help/SQL Safe/7-1/web/default.htm.

SQL Safe Backup Agent may stop unexpectedly

The SQL Safe Backup Agent may stop unexpectedly and SQL Safe displays an error similar to, ".NET Runtime version 2.0…-Fatal Execution Engine Error." Microsoft recommends that users make sure that their environments include the following patches:

- Windows 2003: MS11-044: Description of the security update for the .NET Framework 3.5 Service Pack 1 and .NET Framework 2.0 Service Pack 2 on Windows XP Service Pack 3 and on Windows Server 2003 Service Pack 2: June 14, 2011
- Windows 2008 R2/Windows 7: MS11-044: Description of the security update for the .NET Framework 3.5.1 on Windows 7 Service Pack 1 and on Windows Server 2008 R2 Service Pack 1: June 14, 2011

Importing backup archive sets may result in an error

SQL Safe may experience an issue when you attempt to import backup archive sets into your Repository.

Logins data archived only on Full backups

SQL Safe archives Logins data only when you perform a Full backup. SQL Safe does not archive this data when you perform a Differential or Log backup. You can restore Logins data only when you use a single backup set. When you specify multiple backup sets such as Full, Differential, and Log, you cannot restore Logins data.

Policy views may be blank after upgrading to version 6.6

The new granular alert notifications available in version 6.6 provide more detailed feedback about policy compliance and status. Because policy jobs created with SQL Safe 6.4 or earlier do not support this feature, the Management Console policy views will not display compliance status related to previous backup or restore operations. Instead, the policy views will track the policy status from the time you upgraded. To see the status of previous backup and restore operations, use the backup/restore operation status pane on the instance and database status views.

No Restore Policy support for backup files stored on TSM Servers

The SQL Safe 6.6 Restore Policy does not support restoring a database from a backup file stored on a TSM Server.

Metadata for SQL virtual database is not generated

SQL Safe is unable to generate SQL virtual database metadata for backups that use the following options:

- SQL Server 2008 databases that use FILESTREAM to manage unstructured data
- Read-write file groups
- File backups

Errors occurring when saving changes may delete policies

If an error occurs while saving changes to an existing policy, the policy may be deleted.

InstantRestore appears to stall when restoring databases that contain read-only file groups

SQL Safe 7.0 Beta hydration appears to stall at 99% complete when restoring databases that contain read-only file groups. SQL Server triggers InstantRestore hydration when it performs read/write I/O on the database files. Because SQL Server does not perform read/write I/O on the read-only files, hydration does not begin. Eventually, hydration begins when SQL Server performs read I/O on the files. You can delete the database if you experience this issue.

Adding a new drive requires you to restart the InstantRestore Service

When you add a new drive to a server, you must restart the SQL Safe Filter Service to make sure that the SQL Safe Filter driver is attached to the new drive. When the SQL Safe Filter Service starts, it attaches the SQL Safe Filter driver to all the fixed drives on the server. If you add a new drive after the service starts, the driver is not attached and any files created on this drive during InstantRestore do not function correctly. To avoid this issue, simply restart the SQL Safe Filter Service after adding any new drive.

Not all files are removed when you delete a database restored using InstantRestore

Some files may remain after you attempt to delete a database previously restored using the InstantRestore feature. In most cases, you can manually delete these mdf, ndf, ldf, and vbm files. If the files are locked, restart either the SQL Safe Filter Service or the SQL Server Instance and then delete the files manually.

InstantRestore Hydration statistics are incorrect if the IR Server restarts during Hydration

During the Hydration phase of the InstantRestore feature, if the IR filter service is restarted, the statistics incorrectly show the hydration process reset to zero. This is not accurate as hydration correctly picks up where it left off in the process.

2.4 Recommended IDERA Solutions

IDERA strives to ensure our products provide quality solutions for your database needs. The following IDERA Solutions have been recently added to the knowledge base at our Support portal.

Number	Title
203	SQL Safe Backup fails with the error message "Could not initialize Virtual Device Set" if the Backup Agent's service account is not a member of the `sys_admin` server role.
727	The SQL Safe Console doesn't show status for backup and restore operations.
1109	SQL Safe backup/restore operation or agent deployment returns error "The CPU type and CPU family of <server> could not be determined."
1384	How to manually install the SQL Safe extended stored procedures.
1394	How to install the SQL Safe Backup Agent on a clustered SQL Server.

3 Welcome to SQL Safe Backup

SQL Safe Backup saves money by reducing database backup time by up to 50% over native backups, reducing backup disk space requirements by up to 95%, and providing automated, 'hands-free', multi-server backup management and monitoring.

Need help using SQL Safe Backup? See the following sections:

- Create a backup policy.
- Create a log shipping policy.
- Restore databases.

3.1 What is SQL Safe Backup?

The IDERA SQL Safe Backup (SQL Safe) provides a high-performance backup and recovery solution for Microsoft SQL Server. SQL Safe Backup saves money by reducing database backup times by up to 50% over native backups and reducing backup disk space requirements by up to 95%. SQL Safe Backup also enables complete "hands-free" automated backups of your entire SQL Server infrastructure and ensures compliance with your organization's backup and recovery policies. From implementations with tens of SQL Servers to enterprises with hundreds of instances spread around the globe, SQL Safe Backup is the only SQL Server backup and recovery solution that scales to meet the challenge.

3.2 How does SQL Safe Backup help me?

In many organizations today, SQL Server databases are the repositories for large volumes of business-critical data. As database size grows, the time required to back up your data using native tools can easily exceed your maintenance windows, plus a huge amount of storage space is needed for the files. Restore operations also become time-consuming. DBAs need a powerful backup and recovery solution that greatly reduces backup and recovery time, minimizes storage requirements, and provides enterprise management capabilities to conduct backups across a large number of servers simultaneously. SQL Safe Backup has been specifically designed to meet these requirements, resulting in increased application and business availability for your critical SQL Server infrastructure.

As a state-of-the-art backup and recovery solution, SQL Safe Backup provides:

- Maximum backup file compression.
- Minimum backup times.
- Reduced failures due to network glitches.
- Accelerated average restore time.
- Ensured compliance with corporate backup policies.

3.3 Find Answers

This documentation set includes a comprehensive online Help system as well as additional resources that support you as you install and use the product. You can also search IDERA Solutions, available at the IDERA Customer Service Portal.

3.3.1 Document conventions

IDERA documentation uses consistent conventions to help you identify items throughout the printed online library.

Convention	Specifying
Bold	Window items
Italics	Book and CD titles Variable names New terms
Fixed Font	File and directory names Commands and code examples Text typed by you
Straight brackets, as in [value]	Optional command parameters
Curly braces, as in {value}	Required command parameters
Logical OR, as in value 1 | value 2	Exclusively command parameters where only one of the options can be specified

3.3.2 How to use the Help

The IDERA wiki includes a comprehensive online Help system as well as additional resources that support you as you install and use IDERA products. You can also search multiple IDERA support solutions, available at www.idera.com/support/.

Additionally, IDERA helps you by providing:

- 24/7 technical support for critical issues.
- Availability to report cases and access a web-based customer portal for update status.
- Access to our Knowledge Center where you can find FAQs, How To's, Best Practices, and Webcasts.

This wiki includes the following Web browser minimum requirements:

- Internet Explorer 8.0
- Mozilla Firefox 4
- Google Chrome 6

You can access the IDERA SQL Safe Help system through the **Help** icon on the top right section of your window or by pressing F1 on the section where you need more information.

You can print a help topic from the wiki using the Print function in your browser.

3.3.3 Definition of terms

The following terms are used in the product and throughout the documentation.

Application Feature

SQL Safe Backup performs tasks and displays information depending on the Application Feature you have selected. You can change the application feature by clicking a button in the navigation pane on the Management Console. SQL Safe Today, an additional feature, can be reached by clicking the globe icon on the menu bar, or through the View menu.

Backup Agent

The Backup Agent is a service that runs on each of the SQL Server instances hosting databases you want to backup and restore. Before you can deploy a Backup Agent to a SQL Server instance, you must Register the SQL Server instance with SQL Safe Backup.

Operation

An Operation is a work item that can be scheduled to be performed by the Backup Agent. Backups and restores are executed as operations.

Policy

A policy consists of a list of databases, a set of operations to be performed on those databases, and a set of schedules according to which the operations will be performed. Policies allow you to define a maintenance plan across multiple SQL Server instances, which can reside on one or more physical servers. You can then use the Management Console to monitor the status of policies and their associated database backup operations.

Server Groups

Server Groups are collections of similarly tasked SQL Server instances, whose performance and policy status is more easily monitored together. You are not required to place SQL Server instances into groups but, in an enterprise with hundreds of servers, compliance review can be greatly simplified.

4 Getting Started

Learn the basics of IDERA SQL Safe Backup installation, the product requirements, the supported installation scenarios, and the installation or upgrade instructions. Later on, learn about how to register your instances, subscribe to alerts, manage your users, manage licenses, and other key IDERA SQL Safe Backup features.

SQL Safe Backup integrates seamlessly with the IDERA Dashboard, a common technology framework designed to support multiple IDERA products.

The following topics are included in this section:

- Installation and deployment.
- SQL Safe Backup Upgrades.
- How the InstantRestore Service works.

4.1 Installation and deployment

In this section, review all the installation process for SQL Safe Backup. Install and deploy SQL Safe Backup in any environment that meets the minimum requirements described in the following sections:

- Learn about the product components and architecture.
- Review the hardware, software, permission, and port requirements.
- Check the supported installation scenarios.

4.1.1 Product components and architecture

SQL Safe Backup provides a robust, easy-to-use SQL Server database backup and restore solution. Behind a simple user interface, SQL Safe Backup offers an architecture that is both flexible and extremely powerful. SQL Safe Backup fits your environment, no matter how simple or complex.

The SQL Safe Backup architecture easily runs in your SQL Server environment with minimal configuration. All SQL Safe Backup components run outside and separate from SQL Server processes. SQL Safe Backup does not add to or modify any of your native SQL Server files or services. After you install these components, you can implement features such as Reports.

> ⚠ You must use the same Windows account for the Backup Agent and InstantRestore Service. During installation, you are asked to enter credentials for only one account and the other is created with the same information. If you manually change your account information, make sure you change it in the other service as well to avoid any issues.

Product components

Management Console

The Management Console is a centralized, intuitive user interface that allows you to easily and quickly back up and restore data on specific SQL Server instances.

The Management Console also provides a T-SQL generator, allowing you to create backup and restore T-SQL scripts. You can execute these scripts through scheduled SQL Server jobs or combine several scripts into a single SQL Server scheduled batch job.

Repository Database

The SQL Safe Repository (Repository) is a central database that tracks all SQL Safe backup and restore operations and the corresponding backup archive file paths for your enterprise.

Management Service

The Management Service receives events from the Backup Agent, and then relays the status of all current and completed operations to the SQL Safe Repository.

Backup Agent

The Backup Agent performs backup and restore operations. The agent is a service that runs on the target SQL Server computer.

InstantRestore Service

The InstantRestore Service is used by the Backup Agent to query and change any InstantRestore properties not managed by the Agent. For more information about InstantRestore properties, see InstantRestore.

Command-line Interface and Extended Stored Procedures

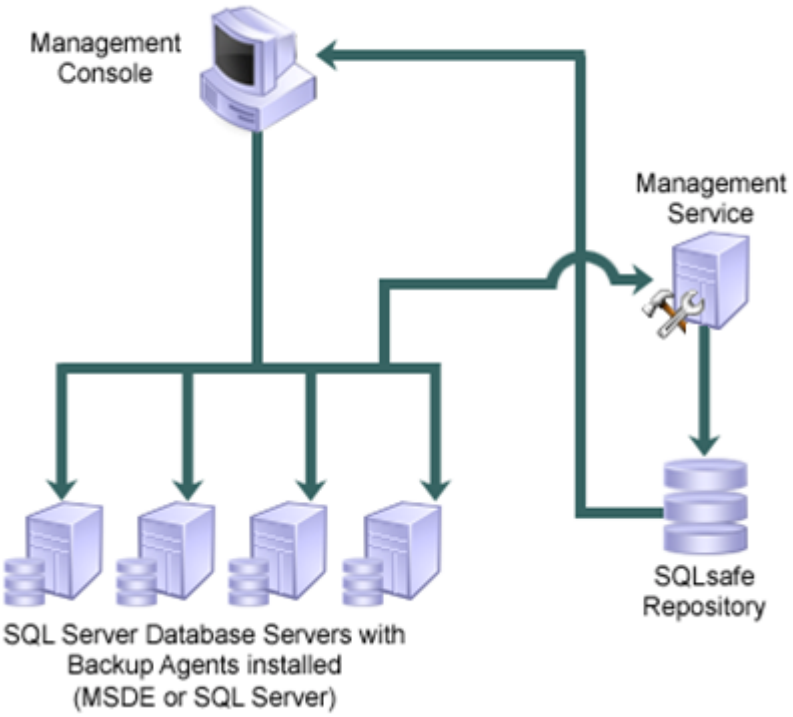
The SQL Safe command line interface (CLI) and extended stored procedures (XSPs) allow you to execute SQL Safe backup and restore procedures with batch files or through your preferred scripting language. You can also use the CLI or XSPs as an alternative to the Management Console.

For sample XSP scripts, see the Sample Scripts programs menu shortcut (**Start > All Programs > IDERA > SQL Safe > XSP > Sample Scripts**). The following scripts are available:

- `xp_ss_backup`
- `xp_ss_browse`
- `xp_ss_expire`
- `xp_ss_extract`
- `xp_ss_restore`
- `xp_ss_restorefilelistonly`
- `xp_ss_restoreheaderonly`
- `xp_ss_restorelast`
- `xp_ss_verify`

Product architecture

The following diagram illustrates the components of the SQL Safe Backup architecture.



4.1.2 Product requirements

Before installing the IDERA Dashboard and IDERA SQL Safe Backup review the system requirements. It is important that you learn about the hardware and software requirements, and the required accounts and permissions you need to have before installing these products.

The following topics are included in this section:

- Hardware requirements.
- Software requirements.
- Permission requirements.
- Port requirements.
- TSM requirements.

Hardware requirements

SQL Safe Backup requires the following hardware.

Hardware Type	Requirement	Recommendation
CPU	1 GHz	2 GHz
Memory	512 MB	1 GB
Hard Drive Space	80 MB (installation files only)	1 GB (temporary disk space for backup and restore operations as they write data to and from files)
Monitor Resolution	1024 by 768 pixels	1280x1024 pixels

Software requirements

The SQL Safe Backup components have the following general software requirements, as well as specific requirements outlined in the following sections. ***If a service pack is not specified***, a service pack is not required for that version of the software.

SQL Safe Backup is signed by SHA-2 code signing hash algorithms.

The following Windows Operating System versions require a Windows update to support SHA-2 code signing:

- Windows Server 2008 SP2
- Windows Server 2008 R2 SP1

For more information about Windows versions and Microsoft updates, please visit Windows Support.

SQL Safe Backup supports the built-in clustering technology included with the following versions of Microsoft Windows operating systems:

- Microsoft Windows Server 2003 R2
- Microsoft Windows Server 2003 SP2
- Microsoft Windows Server 2008 SP2
- Microsoft Windows Server 2008 R2 SP1
- Microsoft Windows Server 2012
- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2016

Keep in mind, SQL Safe 8.5.2 can be installed without the IDERA Dashboard on an Operating System 2003.

Running the combined installer will display the following message: "The current Operating System is not supported by this product".

General Software Requirements

- Microsoft Data Access Components (MDAC) 2.8 or later.
- Microsoft .NET Framework version 4.0 or higher. ***If this software is not already installed on your computer***, you must install it prior to the installation of SQL Safe Backup. This software can be installed from the installation kit by clicking **Prerequisites** on the Install window of the setup program. For more information about the .NET Framework, see the .NET Framework Versions and Dependencies article on MSDN.
- The following lists the web browser minimum requirements:
  - Internet Explorer IE 10.x+
  - Microsoft Edge (new MS Browser in Windows 10)
  - Mozilla Firefox
  - Google Chrome
- Microsoft Visual C++ 2015 Update 3 Redistributable.

SQL Safe Management Components

(SQL Safe Management Console, SQL Safe Collection Service, SQL Safe Management Service, and SQL Safe Rest Service)

The SQL Safe Management components can run on both 32- and 64-bit computers. Each component requires one of the following operating systems:

- Microsoft Windows Server 2003 SP2 or later
- Microsoft Windows Server 2008 SP2 or later
- Microsoft Windows Server 2008 R2 SP1
- Microsoft Windows Server 2012
- Microsoft Windows Server 2012 R2
- Microsoft Windows 8
- Microsoft Windows 10
- Microsoft Windows 2016
- Microsoft Windows 2019

SQL Safe Backup Agent

(SQL Safe Backup Service, SQL Safe Filter Service, and SQL Safe Object Level Recovery Service)

The SQL Safe Backup Agent is supported to run on both 32- and 64-bit computers. The Backup Agent requires one of the following operating systems and one of the following Microsoft SQL Server versions:

- Microsoft Windows Server 2003 SP2 or later
- Microsoft Windows Server 2008 SP2 or later
- Microsoft Windows Server 2008 R2 SP1
- Microsoft Windows Server 2012
- Microsoft Windows Server 2012 R2
- Microsoft Windows 8
- Microsoft Windows 10
- Microsoft Windows 2016
- Microsoft Windows 2019
- Microsoft SQL Server 2005 SP1 or later - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2008 - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2008 R2 - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2012 - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2014 - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2016 - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2017 CU4+
- Microsoft SQL Server 2019 - All Editions (Express, Standard, Enterprise, etc.)

SQL Safe Repository

The computer hosting the SQL Safe Repository requires one of the following operating systems and one of the following Microsoft SQL Server versions:

- Microsoft SQL Server 2008 - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2008 R2 - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2012 - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2014 - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2016 - All Editions (Express, Standard, Enterprise, etc.)
- Microsoft SQL Server 2017 CU4+
- Microsoft SQL Server 2019 - All Editions (Express, Standard, Enterprise, etc.)

Keep in mind, hosting the SQL Safe repository on Microsoft SQL Server Express editions is supported; however, the repository database size is restricted to the limit imposed by those editions.

Supported EMC Environments

- Data Domain OS 5.5, 5.6

Permission requirements

SQL Safe Backup requires specific permissions and rights to successfully execute backup and restore operations. Generally, the rights of the Management Console user dictate the rights available to SQL Safe Backup.

If you are deploying SQL Safe Backup to a non-trusted domain, specify an account with sysadmin fixed role rights for the Management Service and Backup Agent Service accounts, and ensure that SQL Authentication is enabled on each SQL Server instance where the SQL Safe Backup components have been installed.

Recommended permissions for trial installations

Type	Requirement
Windows Permissions	Your Windows logon account has local Administrator permissions.
SQL Server Privileges	Your Windows logon account is a member of the sysadmin fixed server role on the SQL Server instance.

Required permissions for production installations

Account	Action	Permissions Required
Windows user account	• Allows you to install the Backup Agent on local or remote SQL Server instances. • Allows you to install SQL Safe Backup components. • Allows you to perform SQL Safe tasks, such as executing a backup or restore operation, using standard Windows authentication. • Allows you to create the SQL Safe Repository database. • Allows you to read and write backup files. • Allows you to access the SQL Safe Repository.	• Windows administrator permission on the Management Console computer and target database server. • Windows administrator permission on the target computer. • `db_owner` or `db_backupoperator` role on each user or system database and VIEW SERVER STATE permission on the registered SQL Server instance. • Create Database rights on the target SQL Server instance. • Windows account credentials with read and write permission on the volume of share you want to write or read backup files. • Read and write privileges on the SQL Safe Repository database, execute privileges for stored procedures.

Account	Action	Permissions Required
SQL Server login	• Allows you to perform SQL Safe tasks, such as executing a backup or restore operation, using standard SQL authentication. • Allows you to create the SQL Safe Repository database.	• `db_owner` or `db_backupoperator` role in each user or system database and VIEW SERVER STATE permission on the registered SQL Server instance. • Create Database Rights on the target SQL Server instance.

Management Service account	• Allows the SQL Safe Management Service to access the SQL Safe Repository database.	• `db_owner` role or the following SQL permissions on the SQL Safe Repository database:
		  • EXECUTE
		  • INSERT
		  • SELECT
		  • UPDATE
		  • DELETE

Backup Service account	• Allows the Backup Agent to access the SQL Server instances in your environment.	• `sysadmin` privileges on each SQL Server instance.
MSSQLSERVER service	• Allows SQL Safe XSP to read and write backup files. • Allows SQL Safe to read and write backup files in native format. • Allows SQL Safe to create temporary files for OLR operations.	• Read and write permission on the volume or share you want to write or read backup files. • Read and write permission on the directory specified as the Temporary Location when performing an OLR operation.
TSM Server	• Allows you to configure TSM Server and client nodes for communication.	• Administrator privileges within TSM Server.

Port requirements

IDERA SQL Safe Ports

The SQL Safe Backup services use specific ports to communicate to each other as well as other SQL Safe Backup components. Before installing SQL Safe Backup, ensure the following ports are available:

Service	Port for trusted domains	Port for non-trusted domains
SQL Safe Backup Service	5164	5165
SQL Safe Collection Service	9299	9299
SQL Safe Management Service	5162	5163
SQL Safe Rest Service	9998	9998

SQL Safe Backup automatically detects whether its components have been installed in trusted or non-trusted domains.

IDERA Dashboard Ports

The IDERA Dashboard services use specific ports for communication. Before installing the IDERA Dashboard, ensure the following ports are available:

Service	Port
IDERA Dashboard Core Service	9292
IDERA Dashboard Web Application Service (HTTP)	9290
IDERA Dashboard Web Application Service (HTTPS)	9291
IDERA Dashboard Web Application Service (Monitor)	9094

TSM requirements

SQL Safe Backup supports the following versions of the TSM Client application:

- TSM Client 7.1.x.x
- TSM Client 6.4.x.x
- TSM Client 6.3.x.x
- TSM Client 6.2.x.x
- TSM Client 6.1.x.x
- TSM Client 5.6.x.x
- TSM Client 5.5.x.x
- TSM Client 5.4.x.x

By default, SQL Safe Backup supports any version of the TSM Server to which the supported TSM Client versions can connect. For more information about TSM Server requirements, see your IBM TSM documentation.

4.1.3 SQL Safe Backup Installation

This section provides you step-by-step information for installing the IDERA SQL Safe Backup and IDERA Dashboard.

The following topics are included in this section:

- Supported installation scenarios.
- Installing SQL Safe on IDERA Dashboard.
- Using SQL Safe on a Microsoft Windows Cluster environment.

Supported installation scenarios

You can install and deploy SQL Safe Backup to meet your unique backup, recovery, and SQL Server environment needs.

Typical environment

The typical SQL Safe Backup implementation scenario includes the following installations:

- Management Console on your workstation.
- Repository and Management Service on a SQL Server instance.
- Backup Agents on each computer hosting databases you want to back up and recover.

Clustered environment

You can install and configure SQL Safe to backup and recover virtual SQL Servers. A virtual SQL Server is a SQL Server running on a Microsoft failover cluster managed by Microsoft Cluster Services.

This configuration can be limited to deploying the Backup Agent to your virtual instances, or can include a full SQL Safe Backup deployment.

A Backup Agent deployment to a virtual instance includes the following installations:

- Management Console on your workstation.
- Repository and Management Service on a SQL Server instance (not located in the cluster).
- Backup Agents on each cluster node hosting the virtual SQL Server you want to manage.

For more information, see installing backup/restore components in a Clustered Environment.

A full SQL Safe Backup deployment on a cluster includes the following installations:

- Management Service and Backup Agent on each node of the Windows cluster.
- Repository on any virtual SQL Server instance.
- Management Console on your workstation (can also be installed on the cluster nodes).

For more information, see installing SQL Safe Backup on a Microsoft Windows Clustered Environment and installing backup/restore components in a Clustered Environment.

Non-trusted environment

You can install and configure SQL Safe to backup and recover SQL Server databases running in non-trusted domains or workgroups.

This configuration includes the following installations:

- Management Console on your workstation in a trusted or non-trusted domain.
- Repository and Management Service on a SQL Server instance in a trusted or non-trusted domain.
- Backup Agents on each SQL Server instance you want to manage (server can belong to a trusted or non-trusted domain or workgroup).

When deploying SQL Safe Backup to a non-trusted domain, specify an account with sysadmin fixed role rights for the Management Service and Backup Agent Service accounts, and ensure that SQL Authentication is enabled on each SQL Server instance where a SQL Safe Backup component has been installed.

Installing SQL Safe on IDERA Dashboard

You can install the IDERA Dashboard and SQL Safe Backup 8.5.1 on any computer that meets or exceeds the product requirements.

Virtual Database is integrated to SQL Safe Backup 8.5.1. SQL Safe Backup installs VDB at all times. There is no option to install either product separately.

Keep in mind, SQL Safe 8.5.2 can be installed without the IDERA Dashboard on an Operating System 2003.

Running the combined installer will display the following message: "The current Operating System is not supported by this product".

Review the following sections before your installation:

- Product Components and Architecture.
- Hardware Requirements.
- Permission Requirements.
- Software Requirements.
- TSM Requirements.

The installation of SQL Safe Backup fails in scenarios where the Microsoft Visual C++ Redistributable requires Operating System updates.
The following patches are required depending on your Operating System:

- For Windows 8.1, install KB2999226 which has a pre-requisite of KB2919355.
- For Windows Server 2012 R2, install KB2919355 which has a pre-requisite of KB2919442.

Installing SQL Safe Backup

1. Log on with an administrator account to the computer on which you want to install **SQL Safe Backup**.
2. Run the **SQL Safe Backup** installer.
3. Before you begin, the SQL Safe Backup wizard displays information about **what is needed** to complete the installation successfully.
4. Click **Next** and select a setup type to get started.
    a. The Full setup installs the **SQL Safe Backup** and the **IDERA Dashboard**.
    b. The Custom setup allows you to specify the components that you would like to install.
5. Click **Next**.

Full Setup

If you select to install the full setup, the SQL Safe Backup and the IDERA Dashboard components are installed.

The **SQL Safe Backup** components:

- SQL Safe Management Console
- SQL Safe Repository Database
- SQL Safe Backup Service
- SQL Safe Filter Service
- SQL Safe Object Level Recovery Service
- SQL Safe Collection Service

- SQL Safe Rest Service

The **IDERA Dashboard** components:

- IDERA Dashboard Repository Database
- Dashboard Core Service
- IDERA Dashboard Web Application Service

1. Review the license agreement. To accept this license agreement, select the **I accept the Terms and Conditions** checkbox. Click **Next**.
2. Specify the **Destination Folder** where you want to install the **SQL Safe Backup** and **IDERA Dashboard**. Specify different path for each one. Click **Next.**
3. Specify the **SQL Server Instance, Database Name, and Authentication** you want to use for the SQL Safe Repository and IDERA Dashboard Repository.
   By default, the **Windows Authentication** is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the login name and password for this account, and click save. Before you continue, you can **Test Connections** to make sure the SQL Server instance is connected. Then, click **Next**.
4. Specify the **Service Account** under which the SQL Safe Backup and IDERA Dashboard services will run under.
5. Specify the account to be used by the **SQL Safe Backup Management Service** to connect to the SQL Safe Backup Repository Database.
   By default, the **Windows account** specified on the previous step is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the respective SQL Sever login credentials, and click **Next**.
6. Once the **SQL Safe Backup** is ready to install, review the settings you selected and click **Install** to install the program.
   If you want to edit your settings, select **Back** and make your changes.
7. When the SQL Safe Backup installation completes, you can either:
   - Go to the **Start Menu**, select **IDERA** - **SQL Safe Management Console**.
   - Open the **IDERA Dashboard** through the URL https://localhost:9291 from a web browser.
   - Configure the **Firewall**.
   - Launch the **SQL Safe Management Console**.
   - Click **Finish** to exit the wizard.

Custom Setup

If you select to install the custom setup, you can specify the components that you would like to install.

- SQL Safe Backup
- IDERA Dashboard
- SQL Safe Agent Components
- SQL Safe Management Console and SQL Safe Agent Components
- IDERA Dashboard and SQL Safe Backup
- IDERA Dashboard and SQL Safe Agent Components
- IDERA Dashboard, SQL Safe Management Console, and SQL Safe Agent Components

Installing only the IDERA Dashboard

If you only want to install the **IDERA Dashboard**, then follow these steps:

1. In the installation wizard, select **IDERA Dashboard**.
2. Review the license agreement. To accept this license agreement, select the **I accept the Terms and Conditions** checkbox. Click **Next**.
3. Specify the **Destination Folder** where you want to install the **IDERA Dashboard**. Click **Next.**

4. Determine the **SQL Server Instance, Database Name, and Authentication** you want to use for the IDERA Dashboard Repository.
   By default, the setup program uses your **Windows credentials** to create this Database. If you want to use **Microsoft SQL Server Authentication**, select this option, then specify the login name and password for this account. Click **Save**.
   Before you continue, you can **Test Connections** to make sure the SQL Server instance is connected. Then, click **Next**
5. Specify the **Service Accounts** under which the IDERA Dashboard services will run under.
6. Once **IDERA Dashboard** is ready to install, review the settings you selected and click **Install** to install the program.
   If you want to edit your settings, select **Back** and make your changes.
7. When the installation completes, you can either:
   - Open the **IDERA Dashboard** through the link https://localhost:9291.
   - Configure the **Firewall**.
   - Click **Finish**.

Installing only the SQL Safe Backup Components

If you already have the **IDERA Dashboard** installed elsewhere and you only want to install the **SQL Safe** Management Components, then follow these steps:

1. In the installation wizard, select **SQL Safe Backup**.
2. Review the license agreement. To accept this license agreement, select the **I accept the Terms and Conditions** checkbox. Click **Next**.
3. Determine if you are going to register the **SQL Safe Backup** with an existing **IDERA Dashboard**.
   By default, the option Yes is selected. SQL Safe Backup automatically detects if you have a local IDERA Dashboard installed in your computer. If your IDERA Dashboard is located on a remote server, then specify the HostName and Port number of this server. Type administrator credentials to access this Dashboard and click **Next**.
   If you select No, then proceed to the next step.
4. Specify the **Destination Folder** where you want to install the SQL Safe Backup application. Click **Next.**
5. Specify the **SQL Server Instance, Database Name, and Authentication** you want to use for the SQL Safe Backup Repository.
   By default, the **Windows Authentication** is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the login name and password for this account, and click save.
   Before you continue, you can **Test Connections** to make sure the SQL Server instance is connected. Then, click **Next**.
6. Specify the **Service Account** under which the SQL Safe Backup and IDERA Dashboard will run under.
7. Specify the account to be used by the **SQL Safe Backup Management Service** to connect to the SQL Safe Backup Repository Database.
   By default, the Windows account specified on the previous step is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the respective SQL Sever login credentials, and click **Next**.
8. Once the **SQL Safe Backup** is ready to install, review the settings you selected and click **Install** to install the program.
   If you want to edit your settings, select **Back** and make your changes.
9. When the installation completes, you can either:
   - Go to the **Start Menu**: select **IDERA** - **SQL Safe Management Console**.
   - Configure the **Firewall**.
   - Launch the **SQL Safe Management Console**.
   - Click **Finish**.

Installing only the SQL Safe Agent Components

If you only want to install the **SQL Safe Backup Agent** on your SQL Server instance, then follow these steps:

1. In the installation wizard, select **SQL Safe Agent Components**.
2. Review the license agreement. To accept this license agreement, select the **I accept the Terms and Conditions** checkbox. Click **Next**.
3. Specify the **Destination Folder** where you want to install the SQL Safe Backup Agent application. Click **Next.**
4. Specify the **Service Account** under which the SQL Safe Backup and IDERA Dashboard will run under.
5. Specify the name of the **server** that hosts the SQL Safe Management Service.
6. Once the **SQL Safe Backup** is ready to install, review the settings you selected and click **Install** to install the program.
   If you want to edit your settings, select **Back** and make your changes.
7. When the installation completes, you can either:
   - Configure the **Firewall**.
   - Click **Finish**.

Installing only the SQL Safe Management Console and SQL Safe Agent Components

If you only want to install the **SQL Safe Management Console** and **SQL Safe Agent Components** on your SQL Server Instances, then follow these steps:

1. In the installation wizard, select the **SQL Safe Management Console** and **SQL Safe Agent Components**.
2. Review the license agreement. To accept this license agreement, select the **I accept the Terms and Conditions** checkbox. Click **Next**.
3. Specify the **Destination Folder** where you want to install the SQL Safe Backup application. Click **Next.**
4. Specify the name of the **server** that hosts the SQL Safe Management Service.
5. Once the **SQL Safe Management Console** and **SQL Safe Agent Components** are ready to install, review the settings you selected and click **Install** to install the program.
   If you want to edit your settings, select **Back** and make your changes.
6. When the installation completes, you can either:
   - Go to the **Start Menu**: select **IDERA** - **SQL Safe Management Console**.
   - Configure the **Firewall**.
   - Launch the **SQL Safe Management Console**.
   - Click **Finish**.

Installing the IDERA Dashboard and SQL Safe Backup Components

If you want to install the IDERA Dashboard and the SQL Safe Backup components, then follow the Full setup steps.

Installing the IDERA Dashboard and SQL Safe Agent Components

If you want to install the **IDERA Dashboard** and the **SQL Safe Agent Components** on your SQL Server Instances, then follow these steps:

1. In the installation wizard, select **IDERA Dashboard** and **SQL Safe Agent Components**.
2. Review the license agreement. To accept this license agreement, select the **I accept the Terms and Conditions** checkbox. Click **Next**.
3. Specify the **Destination Folder** where you want to install the SQL Safe Backup and IDERA Dashboard. Specify different path for each one. Click **Next.**
4. Specify the **SQL Server Instance, Database Name, and Authentication** you want to use for the IDERA Dashboard Repository.
   By default, the **Windows Authentication** is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the login name and password for this account, and click save.

Before you continue, you can **Test Connections** to make sure the SQL Server instance is connected. Then, click **Next**.
5. Specify the **Service Account** under which the SQL Safe Backup and IDERA Dashboard will run under.
6. Specify the name of the **server** that hosts the SQL Safe Management Service.
7. Once the **IDERA Dashboard** and **SQL Safe Agent Components** are ready to install, review the settings you selected and click **Install** to install the program.
   If you want to edit your settings, select **Back** and make your changes.
8. When the installation completes, you can either:
   - Open the **IDERA Dashboard** through the link https://localhost:9291.
   - Configure the **Firewall**.
   - Click **Finish**.

Installing the IDERA Dashboard, SQL Safe Management Console, and SQL Safe Agent Components

If you want to install the **IDERA Dashboard**, the **SQL Safe Management Consosle**, and the **SQL Safe Agent Components** on your SQL Server Instances, then follow these steps:

1. In the installation wizard, select **IDERA Dashboard**, **SQL Safe Management Console**, and **SQL Safe Agent Components**.
2. Review the license agreement. To accept this license agreement, select the **I accept the Terms and Conditions** checkbox. Click **Next**.
3. Specify the **Destination Folder** where you want to install the SQL Safe Backup and IDERA Dashboard. Specify different path for each one. Click **Next.**
4. Specify the **SQL Server Instance, Database Name, and Authentication** you want to use for the IDERA Dashboard Repository.
   By default, the **Windows Authentication** is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the login name and password for this account, and click save. Before you continue, you can **Test Connections** to make sure the SQL Server instance is connected. Then, click **Next**.
5. Specify the **Service Account** under which the SQL Safe Backup and IDERA Dashboard will run under.
6. Specify the name of the **server** that hosts the SQL Safe Management Service.
7. Once the **IDERA Dashboard**, the **SQL Safe Management Consosle**, and the **SQL Safe Agent Components** are ready to install, review the settings you selected and click **Install** to install the program.
   If you want to edit your settings, select **Back** and make your changes.
8. When the installation completes, you can either:
   - Go to the **Start Menu**: select **IDERA** - **SQL Safe Management Console**.
   - Open the **IDERA Dashboard** through the URL https://localhost:9291 from a web browser.
   - Configure the **Firewall**.
   - Launch the **SQL Safe Management Console**.
   - Click **Finish**.

Accessing SQL Safe Backup with IDERA Dashboard

To access the **IDERA Dashboard**, type the following URL into your browser: https://localhost:9291 or the port you specified.

> The URL used to access the IDERA Dashboard is constructed from the name of the machine on which the IDERA Dashboard was installed and the port specified for the IDERA Dashboard Web Application Service. For example, if the IDERA Dashboard was installed on a machine named *MYSERVER* and the Idera Dashboard Web Application Service used the default port of *9291*, then the URL to access the IDERA Dashboard would be the following: http://MYSERVER:9291

Logging into the Web Application

To log into **SQL Safe Backup** web application you can use one of the following alternatives.

- Type an account with permissions to access SQL Safe Backup (the account you provided at installation for example).
- Select **Log on using Windows credentials** to use Windows integrated authentication and bypass the login process. (You may need to configure your browser for this option, go to Configuring your browser for Windows Authentication to find out more information).

Installing SQL Safe Backup on a Microsoft Windows Clustered Environment

SQL Safe Backup is comprised of multiple application components. Each component provides specific functionality in different areas of the product. These different areas of functionality have specific concerns when deploying in a clustered environment, as redundancy and high availability are an issue.

Follow these steps to install SQL Safe Backup on a Microsoft Windows Clustered Environment:

1. Before to install SQL Safe Backup in a clustered environment, review the clusters configuration considerations.
2. Install SQL Safe on all the cluster nodes you want to host your SQL Safe Management Service, whether active or inactive.
3. Configure the SQL Safe services as a Generic Service Resource.
4. The IDERA Dashboard does not provide support for clustered environments. For installation and configuration instructions, review deploy the IDERA Dashboard in a clustered environment.
5. Registering the SQL Safe Backup with the IDERA Dashboard.

Clustering configuration considerations

The following table includes the applications and services that SQL Safe Backup is comprised of, with a summary of the requirements of each to be fully fault tolerant using the Microsoft Windows Clustering technology.

SQL Safe Backup components can work in either *active-passive* or *active-active* cluster configurations.

Component	Clustering Configuration Considerations
SQL Safe Management Console	None.
SQL Safe Command Line	None.
SQL Safe Management Service, SQL Safe Collection Service, and SQL Safe Rest Service.	This components are responsible for managing operational and policy status and alerting for your entire SQL Safe Backup deployment. If you require these SQL Safe Backup functions to be fault tolerant, configure this components as a *Generic Service Resource* on a clustered server. You need to deploy this components only once onto a single cluster, not on each server you are backing up. For more information, visit Registering SQL Safe Backup services as a Clustered Resource.
SQL Safe Repository Database	This component is a SQL Server database used to stored operational history and policy configuration information. The SQL Safe Repository Database is used by the Management Console to display operational history and by the Management Service to process and send alerts. For this component to be fault-tolerant, simply host the database on a clustered SQL Server.
SQL Safe Backup Service	This component is responsible for executing backup and restore operations on nodes hosting any clustered SQL Servers. You need to install this service only on each node of the cluster. No failover or cluster resource configuration is necessary. You can deploy the agent from the Management Console or the product installer.
SQL Safe Filter Service	This component is responsible for performing instant restore operations on nodes hosting any clustered SQL Servers. If you require that SQL Safe Instant Restore functionality be fault tolerant, configure this component via the SQL Safe Command Line. You should not configure this service as a clustered resource. Instead, a failover mechanism is activated via a *Generic Script Resource*, and is automatically configured for you using a SQL Safe command. For more information about installing the SQL Safe Filter Service on a Windows cluster, see Installing backup/restore components in a clustered environment.

Deploying SQL Safe Backup in a clustered environment

The following instructions guide you through the installation of SQL Safe Management components in a clustered environment.

> SQL Safe Backup must be installed on each node within the cluster.

Perform the following steps:

1. Log on with an Administrator Account to the clustered environment where you want to install **SQL Safe Backup**.
2. Run the **SQL Safe Backup** installer.
3. On the first screen, the SQL Safe wizard displays information about **what is needed** to complete the installation successfully, click **Next**.
4. Select the **Custom** setup type, then click **Next**.
5. Select the **SQL Safe Backup** option.
6. Review the license agreement. To accept this license agreement, select the **I accept the terms and conditions** checkbox. Click **Next**.
7. Select the **No** option to skip the registration with **IDERA Dashboard** as it will be performed later. Click **Next** to proceed.
8. Specify the **Destination Folder** where you want to install the SQL Safe Backup application. Click **Next.**
9. Specify the **clustered SQL Server instance, Database Name, and Authentication** you want to use for the SQL Safe Backup Repository.
   By default, the **Windows Authentication** is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the login name and password for this account, and click **Save**.
   Before you continue, you can **Test Connections** to make sure the SQL Server instance is connected. Then, click **Next**.
10. Specify the **Service Account** under which the SQL Safe Backup will run under.
11. Specify the account to be used by the **SQL Safe Backup Management Service** to connect to the SQL Safe Backup Repository Database.
    By default, the **Windows account** specified on the previous step is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the respective SQL Sever login credentials, and click **Next**.
12. Once the **SQL Safe Backup** is ready to install, review the settings you selected and click **Install** to install the program.
    If you want to edit your settings, select **Back** and make your changes.
13. When the installation completes, **uncheck** the ***Launch the SQL Safe Management Console*** option, then click **Finish**.
14. Open **Windows Services** (right-click the *Start menu*, select *Run*, type *services.msc*, then hit the *Enter* key).
15. Right-click on the **SQL Safe Collection Service** and select *Properties*.
    Change the **Startup Type** to Manual, click the Stop button, then click **OK** to save the changes.
16. Right-click on the **SQL Safe Management Service** and select *Properties*.
    Change the **Startup Type** to Manual, click the Stop button, then click **OK** to save the changes.
17. Right-click on the **SQL Safe Rest Service** and select *Properties*.
    Change the **Startup Type** to Manual, click the Stop button, then click **OK** to save the changes.

Once you complete the SQL Safe Backup installation on each cluster node, configure the SQL Safe services as a **Generic Service Resource**. For more information, visit registering SQL Safe Backup services as a Clustered Resource.

Registering SQL Safe Backup services as a Clustered Resource

Registering the SQL Safe Backup services with Microsoft Failover Cluster Manager allows the Microsoft Cluster Service to manage the services in failover situations.

Review the following topics to register SQL Safe Backup as a clustered resource:

- Configuring in a Windows Server 2003 clustered environment.
- Configuring in a Windows Server 2008 / Windows Server 2008 R2 clustered environment.
- Configuring in a Windows Server 2012 and later clustered environment.

Windows Server 2003 clustered environment

Follow the steps to configure the SQL Safe Backup services as a Generic Resource on a Windows Server 2003 clustered environment:

1. Open the **Windows Cluster Administrator** application named `cluadmin.exe`.
2. **Create or choose a resource group** containing defined IP Address and Network Name resources.
   Note that the **IP Address** and **Network Name** resources in this group are used to access your **SQL Safe Management Service**.
   If you choose to use an *existing group*, note that fail over for each component in this group occurs together.
   If you choose to create a *new group* and resources, make the owners of the new resources the same nodes on which you installed the Management Service.
   Once you configure or choose a group for the SQL Safe Management Service, you can add the **Generic Service Resource** using the following steps:
3. Right-click the group, and then select **New** > **Resource**.
   On the **New Resource** window, type a name for the **Management Service resource**, such as **SQL Safe Management Service**.
   Select the **Generic Service** resource type, and then click **Next.**
   On the **Possible Owners** window, click **Next**.
   On the **Dependencies** window, add the IP address and Network Name resources by selecting the appropriate address and name, and then click **Add >** to move them to the dependencies area. Click **Next.**
   On the **Generic Service Parameters** window, type **SQL Safe Management Service** as the service name.
   Check the **Use Network Name for Computer Name** check box, and then click **Next** .
   On the **Registry Replication** window, click **Add**.
   In the **root registry** field, type the following text: *Software\Idera\SQLsafe\Management Service.* Click **OK** and **Finish**.
4. Right-click the group, and then select **New** > **Resource**.
   On the **New Resource** window, type a name for the **Management Service resource**, such as **SQL Safe Collection Service**.
   Select the **Generic Service** resource type, and then click **Next.**
   On the **Possible Owners** window, click **Next**.
   On the **Dependencies** window, add the IP address and Network Name resources by selecting the appropriate address and name, and then click **Add >** to move them to the dependencies area. Click **Next.**
   On the **Generic Service Parameters** window, type **SQL Safe Collection Service** as the service name.
   Check the **Use Network Name for Computer Name** check box, and then click **Next** and **Finish**.
5. Right-click the group, and then select **New** > **Resource**.
   On the **New Resource** window, type a name for the **Management Service resource**, such as **SQL Safe Rest Service**.
   Select the **Generic Service** resource type, and then click **Next.**
   On the **Possible Owners** window, click **Next**.
   On the **Dependencies** window, add the IP address and Network Name resources by selecting the appropriate address and name, and then click **Add >** to move them to the dependencies area. Click **Next.**
   On the **Generic Service Parameters** window, type **SQL Safe Rest Service** as the service name.
   Check the **Use Network Name for Computer Name** check box, and then click **Next** .
   On the **Registry Replication** window, click **Add**.
   In the **root registry** field, type the following text: *Software\Idera\SQLSafeRestService.* Click **OK** and **Finish**.
6. Bring each of the newly created roles for the **SQL Safe Backup services** online. Right-click on the new **Generic Service Resource**, and then select **Bring Online**.

Windows Server 2008 / Windows Server 2008 R2 clustered environment

There are two ways to register the SQL Safe Backup services as a Generic Resource on a Windows Server 2008/ Windows Server 2008 R2 clustered environment.

- Creating a new role for the SQL Safe services.
- Adding the SQL Safe services to an existing role.

Creating a new role

Use the following steps if you **do not** have an existing application or service that you want to use for the SQL Safe Management Service:

1. Open the **Windows Failover Cluster Management** application named **cluadmin.msc**.
2. Right-click on **Services and Applications**, and then select **Configure a Service** or **Application**.
   On the Select Service or Application window, click **Generic Service**, and then click **Next**.
   On the Select Service window, click **SQL Safe Management Service**, and then click **Next**.
   On the Client Access Point window, type or select the IP address and network name that you want to use to access the SQL Safe Management Service, and then click **Next**.
   On the Select Storage window, click **Next**.
   On the Replicate Registry Settings window, click **Add**.
   In the root registry field, type the following text: *Software\Idera\SQL Safe\Management Service.* Click **OK** and **Next**.
   On the Confirmation window, click **Next**.
   On the Summary window, click **Finish**.
3. Right-click on **Services and Applications**, and then select **Configure a Service** or **Application**.
   On the Select Service or Application window, click **Generic Service**, and then click **Next**.
   On the Select Service window, click **SQL Safe Collection Service**, and then click **Next**.
   On the Client Access Point window, type or select the IP address and network name that you want to use to access the SQL Safe Management Service, and then click **Next**.
   On the Select Storage window, click **Next**.
   On the Replicate Registry Settings window, click **Next**.
   On the Confirmation window, click **Next**.
   On the Summary window, click **Finish**.
4. Right-click on **Services and Applications**, and then select **Configure a Service** or **Application**.
   On the Select Service or Application window, click **Generic Service**, and then click **Next**.
   On the Select Service window, click **SQL Safe Rest Service**, and then click **Next**.
   On the Client Access Point window, type or select the IP address and network name that you want to use to access the SQL Safe Management Service, and then click **Next**.
   On the Select Storage window, click **Next**.
   On the Replicate Registry Settings window, click **Add**.
   In the root registry field, type the following text: *Software\Idera\SQLSafeRestService.* Click **OK** and **Next**.
   On the Confirmation window, click **Next**.
   On the Summary window, click **Finish**.
5. Once you register the **SQL Safe services as a Generic Resource**, right-click each of the new Generic Service Resource, and then select **Bring this resource online**. Your SQL Safe Management Service is now configured for cluster failover.

Adding to an existing role

Use the following steps if you **do** have an existing application or service, already configured with the IP address and network name that you want to also use for the SQL Safe Management Service.

1. Open the **Windows Failover Cluster Management** application named **cluadmin.msc**.
2. Right-click the application, and then select **Add a resource > 4 - Generic Service**.
   On the Select Service window, wait while the services populate the fields with a list of services installed on this computer. Click on **SQL Safe Management Service**, and then click **Next**.
   On the Confirmation window, click **Next**. The wizard configures the service as displayed on the Configure Generic Service window, and then displays the Summary window.
   On the Summary window, click **Finish**.
3. Right-click your **new Generic Service** resource, and then select Properties. SQL Safe now displays the SQL Safe Management Service Properties window which contains many tabs.
   On the General tab, check the **Use Network Name for Computer Name** check box.
   On the Dependencies tab, add the **Network Name** and **IP Address** resources belonging to this application.
   On the Registry Replication tab, click **Add**.
   In the root registry field, type the following text: *Software\IDERA\SQL Safe\Management Service.* Click **OK**.
   On the SQL Safe Management Service Properties window, click **OK**.
4. Right-click the application, and then select **Add a resource > 4 - Generic Service**.
   On the Select Service window, wait while the services populate the fields with a list of services installed on this computer. Click on **SQL Safe Collection Service**, and then click **Next**.
   On the Confirmation window, click **Next**. The wizard configures the service as displayed on the Configure Generic Service window, and then displays the Summary window.
   On the Summary window, click **Finish**.
5. Right-click your **new Generic Service** resource, and then select Properties. SQL Safe now displays the SQL Safe Management Service Properties window which contains many tabs.
   On the General tab, check the **Use Network Name for Computer Name** check box.
   On the Dependencies tab, add the **Network Name** and **IP Address** resources belonging to this application.
   On the Registry Replication tab, click **OK**.
   On the SQL Safe Management Service Properties window, click **OK**.
6. Right-click the application, and then select **Add a resource > 4 - Generic Service**.
   On the Select Service window, wait while the services populate the fields with a list of services installed on this computer. Click on **SQL Safe Rest Service**, and then click **Next**.
   On the Confirmation window, click **Next**. The wizard configures the service as displayed on the Configure Generic Service window, and then displays the Summary window.
   On the Summary window, click **Finish**.
7. Right-click your **new Generic Service** resource, and then select Properties. SQL Safe now displays the SQL Safe Management Service Properties window which contains many tabs.
   On the General tab, check the **Use Network Name for Computer Name** check box.
   On the Dependencies tab, add the **Network Name** and **IP Address** resources belonging to this application.
   On the Registry Replication tab, click **Add**.
   In the root registry field, type the following text: *Software\Idera\SQLSafeRestService.* Click **OK**.
   On the SQL Safe Management Service Properties window, click **OK**.
8. Once you register the SQL Safe Backup services as a Generic Resource, right-click each of the new Generic Service Resource, and then select **Bring this resource online**. Your SQL Safe Management Service is now configured for cluster failover.

Windows Server 2012 and later clustered environment

Registering the SQL Safe Backup services with Microsoft Failover Cluster Manager allows the Microsoft Cluster Service to manage the services in failover situations. The following configuration ensures the high availability of the services during a failover.

There are two ways to register the SQL Safe Backup services as a clustered resource:

- Creating a new role for the SQL Safe Backup services.
- Adding the SQL Safe Backup services to an existing role.

Creating a new role for the SQL Safe Backup services

Creating a new role for the SQL Safe Backup services ensures the high availability of the services during a failover. This set of instructions helps you to configure SQL Safe Backup services in its own cluster. This option requires an available disk on the cluster to store the temporary files used by the SQL Safe Backup services.

1. Log onto the active cluster node using an Administrator Account.
2. Launch the **Microsoft Failover Cluster Manager**.
3. Right-click on **Roles** and select **Create Empty Role**.
4. Right-click on the newly created role and select **Properties**.
5. **Rename the role** to a unique name (e.g. SQLSafeServices), then click **OK**.
6. Right-click on the role and select **Add Resource** > **Client Access Point**.
   On the **Client Access Point** screen, enter a unique **Network Name** (e.g. SQLSafeCluster) and an available IP Address that has been reserved for the clustered resource, then click **Next**.
   On the **Confirmation** screen, click **Next**.
   On the **Summary** screen, click **Finish**.
7. Right-click on the role and select **Add Resource** > **Generic Service**.
   On the **Select Service** screen, locate and select the **SQL Safe Collection Service**. Click **Next** to proceed.
   On the **Confirmation** screen, click **Next**.
   On the **Summary** screen, click **Finish**.
8. Right-click on the role and select **Add Resource** > **Generic Service**.
   On the **Select Service** screen, locate and select the **SQL Safe Management Service**. Click **Next** to proceed.
   On the **Confirmation** screen, click **Next**.
   On the **Summary** screen, click **Finish**.
9. Right-click on the role and select **Add Resource** > **Generic Service**.
   On the **Select Service** screen, locate and select the **SQL Safe Rest Service**. Click **Next** to proceed.
   On the **Confirmation** screen, click **Next**.
   On the **Summary** screen, click **Finish**.
10. Right-click on the newly added Role for the **SQL Safe Collection Service** and select **Properties**.
    On the **Properties** window, go to the **Dependencies** tab.
    Add the following dependencies:
    a. Name of the clustered role.
    b. IP address.

    Click **Apply**.
    Go to the **General** tab.
    Enable the **Use Network Name for computer name** checkbox, then click **OK**.
11. Right-click on the newly added Role for the **SQL Safe Management Service** and select **Properties**.
    On the **Properties** window, go to the **Dependencies** tab.
    Add the following dependencies:
    a. Name of the clustered role.
    b. IP address.

    Click **Apply**.
    Go to the **General** tab.
    Enable the **Use Network Name for computer name** checkbox, then click **Apply**.
    Go to the **Registry Replication** tab. Click the **Add** button and enter *SOFTWARE\Idera\SQLsafe\Management Service* into the Root Registry Key field, then click **OK**.

    > There is a bug in Windows 2012 where the Registry Replication tab is not available. If the tab is unavailable, use the Add-ClusterCheckpoint PowerShell cmdlet to add the necessary setting. EXAMPLE: Add-ClusterCheckpoint -ResourceName "SQLsafe Management Service" -RegistryCheckpoint "SOFTWARE\Idera\SQLSafe\Management Service" -Cluster "[ClusterName]"

12. Right-click on the newly added Role for the **SQL Safe Rest Service** and select **Properties**.
    On the Properties window, go to the **Dependencies** tab.
    Add the following dependencies:
    a. Name of the clustered role.
    b. IP address.

    Click **Apply**.
    Go to the **General** tab.
    Enable the **Use Network Name for computer name** checkbox, then click **Apply**.
    Go to the **Registry Replication**tab. Click the **Add**button and enter*SOFTWARE\Idera\SQLSafeRestService*into the Root Registry Key field, then click **OK**.

    > There is a bug in Windows 2012 where the Registry Replication tab is not available. If the tab is unavailable, use the Add-ClusterCheckpoint PowerShell cmdlet to add the necessary setting. EXAMPLE: Add-ClusterCheckpoint -ResourceName "SQLsafeRestService" -RegistryCheckpoint "SOFTWARE\Idera\SQLSafeRestService" -Cluster "[ClusterName]"

13. Right-click on the Role and select **Start Role**.
14. Launch the **SQL Safe Management Console** (Start > All Programs > IDERA > SQL Safe Management Console).
15. Go to **Tools** > **Repository and Management Service Settings**.
16. Next to the **Computer** field, click the **Change** button.
17. On the window that appears, enter the **cluster name** that SQL Safe Management Service is associated to, then click **OK**.
18. Click **OK** on the **Repository and Management Service Settings** window.

Adding the SQL Safe Backup services to an existing role

Adding the SQL Safe Backup services to an existing role helps with the high availability of the services in failover situations. The following set of instructions helps you to add the SQL Safe Backup services to an existing role:

1. Log onto the active cluster node using an Administrator Account.
2. Launch the **Microsoft Failover Cluster Manager**.
3. Right-click on the role created for the clustered instance and select **Add Resource** > **Generic Service**.
   On the **Select Service** screen, locate and select the **SQL Safe Collection Service**. Click **Next** to proceed.
   On the **Confirmation** screen, click **Next**.
   On the **Summary** screen, click **Finish** .
4. Right-click on the role created for the clustered instance and select **Add Resource** > **Generic Service**.
   On the **Select Service** screen, locate and select the **SQL Safe Management Service**. Click **Next** to proceed.
   On the **Confirmation** screen, click **Next**.
   On the **Summary** screen, click **Finish**.
5. Right-click on the role created for the clustered instance and select **Add Resource** > **Generic Service**.
   On the **Select Service** screen, locate and select the **SQL Safe Rest Service**. Click **Next** to proceed.
   On the **Confirmation** screen, click **Next**.
   On the **Summary** screen, click **Finish**.
6. Right-click on the newly added Role for the **SQL Safe Collection Service** and select **Properties**.
   On the **Properties** window, go to the **Dependencies** tab.
   Add the following dependencies:
   a. Name of the clustered role.
   b. IP address.

   Click **Apply**.
   Go to the **General** tab.
   Enable the **Use Network Name for computer name** checkbox, then click **OK** .
7. Right-click on the newly added Role for the **SQL Safe Management Service** and select **Properties**.
   On the **Properties** window, go to the **Dependencies** tab.
   Add the following dependencies:
   a. Name of the clustered role.
   b. IP address.

   Click **Apply**.
   Go to the **General** tab.
   Enable the **Use Network Name for computer name** checkbox, then click **Apply**.
   Go to the **Registry Replication** tab. Click the **Add** button and enter *SOFTWARE\Idera\SQLsafe\Management Service* into the Root Registry Key field, then click **OK**.

   > There is a bug in Windows 2012 where the Registry Replication tab is not available. If the tab is unavailable, use the Add-ClusterCheckpoint PowerShell cmdlet to add the necessary setting.
   > EXAMPLE: Add-ClusterCheckpoint -ResourceName "SQLsafe Management Service" -RegistryCheckpoint "SOFTWARE\Idera\SQLSafe\Management Service" -Cluster "[ClusterName]"
8. Right-click on the newly added Role for the **SQL Safe Rest Service** and select **Properties**.
   On the **Properties** window, go to the **Dependencies** tab.
   Add the following dependencies:
   a. Name of the clustered role.
   b. IP address.

   Click **Apply**.
   Go to the **General** tab.
   Enable the **Use Network Name for computer name** checkbox, then click **Apply**.

Go to the **Registry Replication** tab. Click the **Add** button and enter *SOFTWARE\Idera\SQLSafeRestService* into the Root Registry Key field, then click **OK**.

> There is a bug in Windows 2012 where the Registry Replication tab is not available. If the tab is unavailable, use the Add-ClusterCheckpoint PowerShell cmdlet to add the necessary setting. EXAMPLE: Add-ClusterCheckpoint -ResourceName "SQLsafeRestService" -RegistryCheckpoint "SOFTWARE\Idera\SQLSafeRestService" -Cluster "[ClusterName]"

9. Bring each of the **newly created roles** for the **SQL Safe services** online (in any order).
10. Launch the **SQL Safe Management Console** (Start > All Programs > IDERA > SQL Safe Management Console).
11. Go to **Tools** > **Repository and Management Service Settings**.
12. Next to the **Computer** field, click the **Change** button.
13. On the window that appears, enter the **cluster name** that SQL Safe Management Service is associated to, then click **OK**.
14. Click **OK** on the **Repository and Management Service** Settings window.

Deploying the IDERA Dashboard in a clustered environment

The **IDERA Dashboard** does not currently support clustered environments. Given this limitation, IDERA Dashboard must be installed on a stand-alone server.

1. Log onto the stand-alone server using an Administrator Account.
2. Run the **SQL Safe Backup** installer.
3. On the first screen, the **SQL Safe Backup** wizard displays information about what is needed to complete the installation successfully, click **Next**.
4. Select the **Custom** setup type, then click **Next**.
5. Select the **IDERA Dashboard** option.
6. Review the license agreement. To accept this license agreement, select the **I accept the terms and conditions** checkbox. Click **Next**.
7. Specify the **Destination Folder** where you want to install the **IDERA Dashboard**. Click **Next.**
8. Determine the SQL Server Instance, Database Name, and Authentication you want to use for the **IDERA Dashboard Repository.**
   By default, the setup program uses your **Windows credentials** to create this Database.
   If you want to use **Microsoft SQL Server Authentication**, select this option, then specify the login name and password for this account. Click **Save**.
   Before you continue, you can **Test Connections** to make sure the SQL Server instance is connected. Then, click **Next**
9. Specify the **Service Accounts** under which the **IDERA Dashboard** services will run under.
10. Once **IDERA Dashboard** is ready to install, review the settings you selected and click **Install** to install the program.
    If you want to edit your settings, select **Back** and make your changes.
11. When the installation completes, Click **Finish**.

Registering the SQL Safe Backup with the IDERA Dashboard

The **IDERA Dashboard** does not currently support clustered environments. Given this limitation, IDERA Dashboard must be installed on a stand-alone server. For more information, visit Deploying the IDERA Dashboard in a clustered environment.

The following instructions guide you to register SQL Safe Backup on the IDERA Dashboard, allowing you to access the product using a web browser:

1. Log into **IDERA Dashboard** using a Dashboard Administrator Account.

   > ⚠ The *URL* used to access the **IDERA Dashboard** is constructed from the name of the machine on which the IDERA Dashboard was installed and the port specified for the IDERA Dashboard Web Application Service. For example, if the IDERA Dashboard was installed on a machine named MYSERVER and the Idera Dashboard Web Application Service used the default port of 9290, then the URL to access the IDERA Dashboard would be the following: http://MYSERVER:9290

2. Go to **Administration** > **Manage Products**.
3. Click on the **Register a Product** link.
4. On the **Add a Product to IDERA Dashboard** window, fill in the required fields:
   a. **Product**: Select or enter *SQLSafeRestService*.
   b. **Display Name**: Enter a unique name to allow you to easily identify the installation of **SQL Safe Backup**.
   c. **Host**: Enter the cluster name where the SQL Safe Rest Service is located.
   d. **Port**: Enter the port number used by the SQL Safe Rest Service. The default port is 9998.
   e. **User Name and Password**: Enter the credentials of a Dashboard administrator account.
5. Click **Register**.
6. Click **Yes** to confirm the registration of the SQL Safe Backup product.

Installing backup/restore components in a clustered environment

For a fault-tolerant backup and restore infrastructure, you must install the SQL Safe Backup Service and SQL Safe Filter Service on the computers hosting your SQL Servers. Although you are using a clustered environment, **you should not configure these services as clustered resources**. Doing so would compromise their functionality in *active-active* clustering configurations. In the case of InstantRestore, you can perform the optional step of executing a one-time command to enable failover support for SQL Safe hydration operations. See the following topic to enable fault tolerance for IR hydration per SQL Server instance.

Install the SQL Safe backup and restore components via the SQL Safe Management Console

1. **Install** the **SQL Safe Management Console** onto a computer using the **FULL** Installer. You can install on any computer, as long as it can access the servers to which you want to deploy the SQL Safe Management Console.
2. **Launch** the **SQL Safe Management Console**.
3. **Register** the **clustered SQL Server instances** on which you want to install the SQL Safe Backup Agent Components by right-clicking SQL Server Instances, and then selecting Register SQL Server.
4. Right-click on the newly registered SQL Server instance in the Servers tree, and then select **Properties**.
5. On the **Advanced** tab, update the **Network Name** field to include the current active node of the cluster. Click **OK** to save the changes. If a connection error message appears, click **Yes** to proceed.
6. Go to **SQL Safe Backup Agent** tab in the navigation pane.
7. **Validate** that the **node** name of the clustered SQL Server is displayed.
8. Right-click on the node name and select **Install SQLsafe Backup Agent**.
9. Click **OK** on the **Install SQLsafe Backup Agent** wizard window.
10. Once the installation has completed, right-click on the node name and select **Refresh**.
11. Once the management console refreshes, it should detect and list the remaining nodes of the cluster.
12. Right-click the first of the remaining nodes, and then select **Install SQLsafe Backup Agent**. Repeat this step for each of the remaining nodes to install the SQL Safe Agent components on the remaining nodes.
13. Verify that the **Management Server** column matches the correct management service. This setting is automatically set during your installation.
14. Go to the **Servers** tab in the navigation pane.
15. Right-click on the clustered SQL Server instance name and select **Properties**. On the **Advanced** tab, update the **Network Name** field to include the current active node of the cluster (and instanced name if it is not the default instance). Click **OK** to save the changes.
16. On the **Advanced** tab, update the **Network Name** field to the clustered SQL Server instance name. Click **OK** to save the changes.
17. ***If you want to use InstantRestore functionality***, right-click the first node, and then choose **Enable SQL Safe Instant Restore**. Repeat this step for each of the remaining nodes. Your SQL Safe Backup Services are now installed.

Enable fault tolerance for InstantRestore Hydration per SQL Server instance (Optional)

This process is necessary only if you want fault tolerance for databases still going through the hydration process of InstantRestore. Before you begin, make sure that your SQL Safe Backup service components are already installed.

1. Remotely **log on to a node** of the cluster.
2. Open a new **Command Window**.
3. For each clustered SQL Server instance hosted on that cluster server, **run the** following **command**:
   SQL SafeCmd Cluster FilterService `sqlserver_name`
   Where `sqlserver_name` is the full SQL Server Instance name of the clustered SQL Server instance.

This command creates a **Generic Script Resource** that controls failover for Hydration operations for that specific SQL Server. Note that this command is **run only once per clustered SQL Server**, not per node.

Installing SQL Safe Backup in an AlwaysOn Environment

AlwaysOn Availability Groups are part of an integrated solution, introduced in SQL Server 2012 with the goal of achieving the highest level of data availability and disaster recovery for organizations.

Availability Groups grant DBAs the ability to automatically or manually failover a group of databases as a single unit with support for up to four secondary replicas. For additional information on availability groups, see the Microsoft document, Overview of AlwaysOn Availability Groups (SQL Server).

The SQL Safe Backup supports SQL Server Availability Groups and allows you to perform backup and recovery strategies on your primary and secondary replicas.

To install the SQL Safe Backup Management Components while targeting an availability group for the databases, follow these steps:

1. Install the SQL Safe Backup with repository databases hosted in an availability group. For more information, see deploy SQL Safe Backup in an AlwaysOn Environment.
2. Once you install the SQL Safe Backup, configure your databases for High Availability.
3. Launch the SQL Safe Management Console from the Start menu.

> Keep in mind, an agent needs to be installed on all nodes of the availability group that will be considered for backups. To install the SQL Safe Backup Agent on each node of the availability group, please follow the instructions. For more information, see install and configure the SQL Safe Backup Agent.

To perform Backup and Restore operations on your primary and secondary replicas, see availability groups.

> In an AlwaysOn Environment, the SQL Safe Backup will require a license key for each replica to ensure continued functionality of the product after a failover event.
>
> The license key for SQL Safe Backup is tied to the name of the SQL Server instance hosting the repository database.

Deploying SQL Safe Backup in an AlwaysOn Environment

The following instructions guide you through the installation of SQL Safe Backup while hosting the repository databases in an AlwaysOn Environment.

> Keep in mind that an agent needs to be installed on all nodes in the availability group that will be considered for backups. To install the SQL Safe Backup Agent on each node of the availability group, please follow the instructions. For more information, see install and configure the SQL Safe Backup Agent.

Perform the following steps:

1. Log on with an administrator account to the AlwaysOn environment where you want to install **SQL Safe Backup** (primary node of the availability group).
2. Run the **SQL Safe Backup** installer.
3. Before you begin, the SQL Safe Backup wizard displays information about **what is needed** to complete the installation successfully. Click **Next**.
4. Select the **Full** setup type to get started. Click **Next**.
5. Review the license agreement. To accept this license agreement, select the **I accept the Terms and Conditions** checkbox. Click **Next**.
6. Specify the **Destination Folder** where you want to install the SQL Safe Backup and IDERA Dashboard. Specify different path for each one. Click **Next.**
7. Specify the **SQL Server Instance** (listener name of the availability group), **Database Name, and Authentication** you want to use for the **SQL Safe Backup Repository**.
   By default, the **Windows Authentication** is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the login name and password for this account, and click save.
   Make sure you enable the **Same Instance and Authentication as above** checkbox for the **IDERA Dashboard Repository**.
   Before you continue, you can **Test Connections** to make sure the SQL Server instance is connected. Then, click **Next**.
8. Specify the **Service Account** under which the SQL Safe Backup and IDERA Dashboard services will run under.
9. Specify the account to be used by the **SQL Safe Backup Management Service** to connect to the SQL Safe Backup Repository Database.
   By default, the **Windows account** specified on the previous step is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the respective SQL Sever login credentials, and click **Next**.
10. Once the **SQL Safe Backup** is ready to install, review the settings you selected and click **Install** to install the program.
    If you want to edit your settings, select **Back** and make your changes.
11. When the **SQL Safe Backup** installation completes, you can either:
    - Go to the **Start Menu**, select **IDERA** - **SQL Safe Management Console**.
    - Open the **IDERA Dashboard** through the URL https://localhost:9291 from a web browser.
    - Configure the **Firewall**.
    - Launch the **SQL Safe Management Console**.

    By default, the **Launch the SQL Safe Management Console** checkbox is enable. To complete the installation, disable it, then click **Finish**.

Once you complete the SQL Safe Backup installation, configure your databases for High Availability.

Configuring the Databases for High Availability

The following steps guide you to make the IDERA Dashboard Repository and SQL Safe Backup Repository databases highly available using the Windows Failover Cluster Service for Windows Server 2008 and above.

1. Open the **SQL Server Management Studio (SSMS)** and connect to the **listener** name.
2. Set the **recovery model** of the **IDERA Dashboard Repository and SQL Safe Repository databases** to **Full**.
   Go to Databases, right click on IDERA Dashboard Repository, and select Properties. Go to options and set the Recovery Model to Full. Click OK.
   Go to Databases, right click on SQL Safe Repository, and select Properties. Go to options and set the Recovery Model to Full. Click OK.
3. **Perform a Full backup** of the **IDERA Dashboard Repository and SQL Safe Repository**.
   Go to Databases, right click on IDERA Dashboard Repository, select Tasks, and Backup. The Backup Database - IDERA Dashboard Repository window appears, select the Backup Type as Full, and click OK.
   Go to Databases, right click on SQL Safe Repository, select Tasks, and Backup. The Backup Database - SQL Safe Repository window appears, select the Backup Type as Full, and click OK.
4. **Sign in to the listener machine**, open the **SQL Server Management Studio (SSMS),** and connect to the **listener** name.
5. **Create an availability group**.
   Go to AlwaysOn High Availability, right click on Availability Groups, and select New Availability Group.
   The New Availability Group wizard opens. On the General tab assign the Availability group name and in the Availability Replicas Add the information for the secondary replica. Make sure the Readable Secondary option is set as Yes for both replicas. On the Backup Preferences tab select the location where your backup should occur. Click OK.
6. Right click on the Availability Group you just created and select **Add Database**.
   The Add Database to Availability Group wizard opens displaying information about what is needed to add one or more availability databases to an existing availability group. Click **Next.**
   Select the IDERA Dashboard Repository and SQL Safe Repository **databases for the availability group**. Click Next.
   Select your **data synchronization** preference, then click Next.
   Connect to all the existing **secondary replicas**, then click Next.
   Ensure the **availability group validation** tests are successful. Click Next.
   Once the **Databases** are ready to add to the availability group, review the Summary and click **Finish**. If you want to edit your settings, select **Previous** and make your changes.
   When the Databases are included to the availability group, the results section displays it. Click **Close** to exit the wizard.
7. In **SQL Server Management Studio (SSMS)**, connect to the **secondary replica(s)** and ensure that the databases have been added successfully.

Once you complete it, you have successfully installed SQL Safe Backup with the repository databases hosted in an availability group.

You may begin using SQL Safe Backup by launching the SQL Safe Management Console from the Start menu.

In an AlwaysOn Environment, SQL Safe Backup will require a license key for each replica to ensure continued functionality of the product after a failover event.

The license key for SQL Safe Backup is tied to the name of the SQL Server instance hosting the repository database.

4.1.4 SQL Safe Backup Upgrades

Upgrading SQL Safe Backup allows you to take advantage of the new features available in this version.

- The backup file names using the `%timestamp%` macro may change and cause issues with your backup files. This issue only affects some users. For more information, see the known issues section of the Release Notes.

- SQL Safe Backup includes a file system filter driver to support the InstantRestore feature. The driver, named SQLsafeFilterDriver, allows SQL Server to access database data while SQL Safe is executing InstantRestore. The driver is only used during this action and is no longer in use once the database is completely hydrated.

Upgrade checklist

To successfully upgrade your Microsoft SQL Server environment to this build, complete the procedures outlined in the following table. These procedures support upgrades from SQL Safe Backup versions 5.0 or later.

☑	**Follow these steps...**
☐	Ensure the computers on which you want to upgrade SQL Safe Backup meet or exceed the product requirements for this version of SQL Safe Backup. For example, ensure .NET Framework 4.0 or later is running on the target computer.
☐	Ensure your Windows logon account has local administrator permissions on the computers you intend to upgrade.
☐	Review the Product components and architecture.
☐	Review the Supported installation scenarios.
☐	Close all open applications on the computers you intend to upgrade.
☐	Upgrade your SQL Safe Backup installation.

Available upgrade paths

Because each component can be installed separately on different computers, the type of upgrade you will need to perform will depend on your environment. The following table describes the conditions under which you would follow a typical or staged upgrade path.

Environment Description	Recommended Path	Why
Simple environment, where the Management Console, the Repository, and the Management Service all reside on the same computer.	Typical	A simple environment can be upgraded through the setup program.
Distributed environment, where each SQL Safe Backup component resides on a different computer.	Staged	A distributed environment requires a staged upgrade to maintain backup continuity while each component is upgraded.
Multiple Management Consoles deployed to different computers.	Staged	A SQL Safe Backup installation with multiple Management Consoles requires a staged upgrade in order to maintain connection with all Backup Agents while each component is upgraded.
Backup Agents from different SQL Safe Backup versions.	Staged	An environment that manages Backup Agents from different SQL Safe Backup versions requires a staged upgrade in order to maintain connection with all Backup Agents while each component is upgraded.
Change control policies that require multi-phased upgrades.	Staged	An environment with stringent change control policies requires a multi-phased upgrade in order to test each updated component thoroughly before moving on to the next step.

For more information, see available upgrade paths.

New encryption options in 6.0 and later

SQL Safe Backup 6.0 and later provides new, more secure encryption algorithms. To use these new algorithms, upgrade your Backup Agents to the latest version.

Previous Encryption Options	SQL Safe 6.x Encryption Options
AES	AES-128
DES	AES-256
Triple-DES	
RC2	

These new encryption options replace the options previously available in SQL Safe Backup 5.0 or earlier. You can select the new encryption options when you manually perform a backup, or create and edit existing Backup Policies.

If you had set encryption options when creating your Backup Policies, the encryption method specified in the corresponding SQL Server job will be automatically updated to AES-128 when you upgrade the associated Backup Agent. You can later change this setting by editing the policy.

SQL Safe Backup 6.0 and later does support previously encrypted archives; you can continue to restore any encrypted backup file created with a previous version of SQL Safe.

More upgrade paths

- To upgrade the SQL Safe Management Components, review the Installing SQL Safe Backup and IDERA Dashboard.
- To upgrade the Backup Agents, visit Upgrade deployed Backup Agents.
- To upgrade SQL Safe Backup in non-trusted domains, see Upgrade SQL Safe Backup in non-trusted domains.
- To upgrade SQL Safe Backup when your SQL Server is running SQL Safe Lite or SQL Safe Freeware edition, see Upgrade the Lite or Freeware Edition or Upgrade the SQL Safe Freeware Edition.
- To upgrade SQL Safe Backup in a clustered environment, see Upgrade backup/restore components in a clustered environment.

Upgrading instructions

To upgrade SQL Safe Backup to the newer version, follow these steps:

1. Log on with an administrator account to the computer where the server hosting the SQL Safe Backup is located.
2. Open the **SQL Safe Backup installer** and click **Run**.
3. Before you begin, the **SQL Safe Backup** wizard displays information about what is needed to complete the installation successfully. Click **Next**.
4. By default, the installer automatically detects the features you have already installed. Those features are selected to upgrade.
5. Review the license agreement. To accept this license agreement, select the **I accept the Terms and Conditions** checkbox. Click **Next**.
6. Determine if you are going to **register the SQL Safe Backup** with an existing IDERA Dashboard.
   By default, the option Yes is selected. SQL Safe Backup automatically detects if you have a local IDERA Dashboard installed in your computer. If your IDERA Dashboard is located on a remote server, then specify the HostName and Port number of this server. Type administrator credentials to access this Dashboard and click **Next**.
   If you select No, then proceed to the next step.
   A message appears letting you know that an older version of IDERA Dashboard is found and proceeding further will upgrade the existing one. Click **OK** to continue.
7. By default, the installer uses the **Destination Folder** you specified when you installed the product. Click **Next.**
   A message appears letting you know that an existing SQL Safe Backup product is registered on this Dashboard and the current instance will be upgraded. Click **OK** to continue.
8. The **SQL Server Instance and Database Name** remains the same as your previous installation of the product. You can change the **Authentication** for the **SQL Safe Backup Repository**. By default, the **Windows Authentication** is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the login name and password for this account, and click Save.
   Before you continue, you can **Test Connections** to make sure the SQL Server instance is connected. Then, click **Next**.
9. Specify the **Service Account** under which the **SQL Safe Backup** and **IDERA Dashboard** services will run under. Click **Next**.
10. Specify the account to be used by the **SQL Safe Backup Management Service** to connect to the SQL Safe Backup Repository Database. By default, the **Windows account** specified on the previous step is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the respective SQL Sever login credentials, and click **Next**.
11. Once the **SQL Safe Backup** is ready to install, review the settings you selected and click **Install** to install the program.
    If you want to edit your settings, select **Back** and make your changes.
12. When the SQL Safe installation completes, you can either:

    - Go to the **Start Menu**, Select IDERA > SQL Safe Management Console.
    - Open the **IDERA Dashboard** through the URL https://localhost:9291 from a web browser.
    - Configure the **Firewall**.
    - Launch the **SQL Safe Management Console**.
    - Click **Finish** to exit the wizard.

Available upgrade paths

Each component can be installed separately on different computers, the type of upgrade you will need to perform will depend on your environment.

The following table describes the conditions under which you would follow a typical or staged upgrade path:

Environment Description	Recommended Path	Reason
Simple environment, where the Management Console, the Repository, and the Management Service all reside on the same computer.	Typical	A simple environment can be upgraded through the setup program.
Distributed environment, where each SQL Safe Backup component resides on a different computer.	Staged	A distributed environment requires a staged upgrade to maintain backup continuity while each component is upgraded.
Multiple Management Consoles deployed to different computers.	Staged	A SQL Safe Backup installation with multiple Management Consoles requires a staged upgrade in order to maintain a connection with all Backup Agents while each component is upgraded.
Backup Agents from different SQL Safe Backup versions.	Staged	An environment that manages Backup Agents from different SQL Safe Backup versions requires a staged upgrade in order to maintain a connection with all Backup Agents while each component is upgraded.
Change control policies that require multi-phased upgrades.	Staged	An environment with stringent change control policies requires a multi-phased upgrade in order to test each updated component thoroughly before moving on to the next step.

Perform a staged upgrade

Use a staged process to upgrade your SQL Safe Backup installation if you have a distributed environment, have deployed multiple Backup Agents that cannot be upgraded during the same time period, or you need to adhere to change control policies.

What is a distributed environment?

A distributed environment consists of the SQL Safe management components running on different physical computers. The management components are:

- Repository database
- Management Service
- Management Console

The time scale over which you choose to perform a staged upgrade will depend on the size of your SQL Server environment and your corporate change control policies. For example, you may choose to perform one of the following steps per day or per week; however, the order in which you perform them should remain the same.

How to perform a staged upgrade

Use this process if your environment meets the following conditions:

- The SQL Safe Backup components are installed on different computers
- All Backup Agents are version 5.0 or earlier

To perform a staged upgrade:

1. Upgrade the Management Service and Repository database to the newest version of SQL Safe Backup by using the setup program to perform a Custom install.
2. Upgrade your Management Console installations to the newest version of SQL Safe Backup. Although you can upgrade the Management Console installations over time, keep in mind that 5.0 and earlier Management Consoles should not be used with the latest version of the Backup Agents.
3. ***If you use policies to automate your backup and restore operations***, start the SQL Safe Management Console to synchronize the jobs associated with each policy. This synchronization should happen automatically.
4. Upgrade deployed Backup Agents according to your change control policies. As you upgrade your Backup Agents, ensure you use a matching version of the Management Console to manage the corresponding instances and Backup Policies.

Perform a typical upgrade

Use the Typical install to upgrade your SQL Safe Backup installation if you have centralized the SQL Safe Backup deployment or are upgrading from a trial installation. This process assumes you can upgrade all deployed Backup Agents during the same time period.

What is a typical environment?

In a typical environment, the SQL Safe management components will be installed on the same physical computer. These components include:

- Repository database
- Management Service
- Management Console

Existing backups executed through maintenance plans, SQL Server jobs, or the CLI will continue to run successfully using older Backup Agents. Once you have completed the installation of the management components, you can upgrade your Backup Agents.

How to perform a typical upgrade

A typical upgrade can be easily completed during off-hours.

To perform a typical upgrade:

1. Perform a Typical install to upgrade the management components to the newest version of SQL Safe Backup. When prompted, verify the name of SQL Safe Repository database.
2. ***If you use policies to automate your backup and restore operations***, start the SQL Safe Management Console to synchronize the jobs associated with each policy. This synchronization should happen automatically.
3. Upgrade all previously deployed Backup Agents.

Upgrade deployed Backup Agents

Consider upgrading previously deployed Backup Agents off-hours during a single time period or in stages according to your corporate change policies. You can remotely upgrade previously deployed agents through the SQL Safe Agents view in the Management Console.

The Backup Agent is automatically installed on the local computer when you install or upgrade any of the other SQL Safe Backup components.

Backup Agent compatibility with Tivoli Storage Manager (TSM)

SQL Safe Backup 6.4 and later includes extensive enhancements to how SQL Safe Backup handles backing up to and restoring from TSM Server. ***If you store backup files on a TSM Client node***, ensure you upgrade the Backup Agents that perform backups and restores using those files.

Backup Agent compatibility with Backup Policies

The latest version of SQL Safe Backup requires version 8.2 or higher Backup Agents to successfully run new Backup Policies and edit existing Backup Policies (created with SQL Safe 5.0 or earlier).

If you want to continue using 5.0 or earlier Backup Agents for existing Backup Policies, install the current version of SQL Safe Backup alongside your current SQL Safe Backup deployment. Use the latest installation of SQL Safe Backup to create and manage new Backup Policies and use SQL Safe Backup 5.0 to maintain existing policies (created with SQL Safe Backup 5.0 or earlier).

Backup Agent compatibility with Log Shipping Policies

Because SQL Safe 5.0 Backup Agents cannot restore backups written by more recent versions of the Backup Agents, environments that include Log Shipping policies require a specific upgrade path. First upgrade the Backup Agents running on the SQL Server computers hosting the secondary databases and then upgrade the Backup Agents running on the SQL Server computers hosting the primary databases. Otherwise, when SQL Safe Backup attempts to ship a new transaction log backup, the restore operation will fail and the Log Shipping policy status will show that the associated jobs are out of date until all the Backup Agents have been upgraded.

How to upgrade a Backup Agent using the Management Console

This procedure guides you through the process of upgrading previously deployed Backup Agents using the Management Console. You can also manually upgrade a Backup Agent using the setup program.

To upgrade deployed Backup Agents:

1. Verify that your environment includes the **newest version of the SQL Safe Management Console**.
2. Start the new SQL Safe Management Console, and **navigate** to the **SQL Safe Agents view**. To access the SQL Safe Agents view, click the SQL Safe Agents tab at the bottom of the **Servers** tree.
3. For each Backup Agent you want to upgrade, select the SQL Server computer that is hosting the agent, and then click **Upgrade SQL Safe Backup Agent** on the right-click context menu. You can include the following options to upgrade your agent:
   - Install SQL Safe Agent Extended Stored Procedures.
   - Include Backup Service Install Log.

Upgrade SQL Safe in non-trusted domains

SQL Safe Backup 6.4 and later supports seamless integration and communications between trusted and non-trusted domains. ***If you want to take advantage of this feature***, use the following instructions to upgrade your SQL Safe Backup 6.3 or earlier environment.

To upgrade SQL Safe Backup 6.4 or later to the current release version, follow the standard Typical or Staged upgrade instructions.

How to upgrade a single Repository deployment

This deployment consists of:

- One SQL Safe Repository hosted in the trusted domain.
- A Management Service in each non-trusted domain.
- A single Management Service that manages SQL Server instances in your trusted domain.
- Management Consoles in trusted and non-trusted domains.
- Backup Agents in trusted and non-trusted domains.

To upgrade this deployment:

1. Uninstall the Management Service from each non-trusted domain.
2. Ensure your Backup Service and Management Service accounts have the appropriate permissions.
3. Ensure the required ports are open in all trusted and non-trusted domains.
4. Decide how many SQL Safe Management Consoles you need to keep in each domain, and uninstall those consoles you no longer need. Maintaining a Management Console in a non-trusted domain is no longer required.
5. Run the setup program to upgrade:
    - The SQL Safe Repository and Management Service in the trusted domain.
    - The Management Consoles in the trusted domain.
    - The Management Consoles in the non-trusted domains. Note that this step is optional, depending on whether you decided to maintain these consoles.
6. ***If you use policies to automate your backup and restore operations***, start the SQL Safe Management Console to synchronize the jobs associated with each policy. This synchronization should happen automatically.
7. Upgrade all deployed Backup Agents in your trusted and non-trusted domains.

How to upgrade an island deployment

This deployment consists of:

- A complete SQL Safe Backup installation in each non-trusted domain.
- A single centralized installation that manages SQL Server instances in your trusted domain.

To upgrade this deployment:

1. Decide how many SQL Safe Management Consoles you need to keep in each domain. Maintaining a Management Console in a non-trusted domain is no longer required.
2. Identify which domain you want to host your new centralized SQL Safe Backup deployment.
3. Ensure your Backup Service and Management Service accounts have the appropriate permissions.
4. Ensure the required ports are open in all trusted and non-trusted domains.
5. Uninstall the SQL Safe Backup management components from the domains you do not intend to use to host SQL Safe Backup. Remember to preserve the Management Consoles you identified in Step 1.

6. Upgrade the SQL Safe Backup management components that were previously deployed to the selected domain.
7. Run the setup program to upgrade all remaining Management Consoles.
8. ***If you use policies to automate your backup and restore operations***, start the SQL Safe Backup Management Console to synchronize the jobs associated with each policy. This synchronization should happen automatically.
9. In each Management Console, verify that:
    - It can connect to the upgraded SQL Safe Repository.
    - It is using the correct Management Service.
10. Upgrade all deployed Backup Agents in your trusted and non-trusted domains.

Upgrade the Lite or Freeware Edition

The Management Console detects when a registered SQL Server instance is running SQL Safe Lite or SQL Safe Freeware Edition. You can upgrade the Backup Agent to the current version of SQL Safe enterprise edition using one of the following methods:

- Deploy the current version of the agent. For more information, see Install the SQL Safe Backup Agent.
- Select the instance in the Servers tree, and then click **Enable trial license** on the Instance View. This upgrades the license to an enterprise edition trial license.
- Click **Upgrade** on the SQL Safe Agent Properties window. For more information, see Modify the Backup Agent properties.

When you upgrade the Backup Agent, SQL Safe Backup deploys the current version of the Backup Agent with a trial license enabled. The trial license allows you full access to the SQL Safe enterprise features for all SQL Server instances hosted on that computer. The trial license is a limited-time, limited-instance license that you will need to upgrade with a production license key.

Upgrade the SQL Safe Freeware Edition

SQL Safe Freeware Edition is no longer available. However, you can easily upgrade your SQL Safe Freeware Edition deployment to either SQL Safe or SQL Safe Lite.

Upgrade SQL Safe Freeware Edition to SQL Safe Lite

You can upgrade SQL Safe Freeware Edition to SQL Safe Lite 6.x by running the SQL Safe setup program on the target SQL Server computer and selecting **Agent Only Install**.

The first time you perform a backup using SQL Safe Lite, the tool will generate a trial license. To install and activate your production license, click **IDERA > SQL Safe Lite > Activate License** on the **Start** menu.

Upgrade SQL Safe Freeware Edition to SQL Safe

To upgrade from SQL Safe Freeware Edition to SQL Safe Backup 6.x, first perform a full installation of SQL Safe Backup, and then Upgrade the Lite or Freeware Edition through the Management Console.

If you install SQL Safe Backup on the same computer as your existing SQL Safe Freeware Edition, the corresponding Backup Agent will be automatically upgraded as part of this installation.

After installation, apply your new SQL Safe Backup 6.x production license by clicking **Manage License** on the **Tools** menu in the Management Console.

Upgrade backup/restore components in a clustered environment

To upgrade the SQL Safe backup and restore components via the SQL Safe Management Console, follow the steps:

1. **Launch** the **SQL Safe Management Console**.
2. Click the **SQL Safe Agents** tab. SQL Safe Backup displays the active and inactive node names of the clustered SQL Server onto which you installed the SQL Safe Backup Agent.
3. Right-click the first node, and then select **Upgrade SQLsafe Backup Agent**. Repeat this step for each of the remaining nodes to upgrade the SQL Safe Agent components on the remaining nodes.
4. Verify that the **Management Server** column matches the correct management service. This setting is automatically set during your upgrade.
5. ***If you want to use InstantRestore functionality***, right-click the first node, and then choose **Enable SQL Safe Instant Restore**. Repeat this step for each of the remaining nodes. Your SQL Safe Backup Services are now upgraded.

Upgrade the SQL Safe Backup in an AlwaysOn Environment

To upgrade the SQL Safe Backup in an AlwaysOn Environment for SQL server 2012 and later, follow these steps:

> The databases hosting on the availability group needs to be removed before performing an upgrade.

1. Prepare the Repository Databases from the availability group.
2. Upgrade the SQL Safe Backup and IDERA Dashboard.
3. Configure your Databases for High Availability.
4. Launch the SQL Safe Management Console from the Start menu.

> In an AlwaysOn Environment, the SQL Safe Backup will require a license key for each replica to ensure continued functionality of the product after a failover event.
> The license key for SQL Safe Backup is tied to the name of the SQL Server instance hosting the repository database.

Preparing the Repository Databases

The following steps will guide you to prepare the repository databases from the ability group.

> Keep in mind, the databases hosting on the availability group needs to be removed before performing an upgrade.

1. Open the **SQL Server Management Studio (SSMS)** and connect to the **SQL Server Instance** (listener) hosting the IDERA Dashboard Repository and SQL Safe Repository.
2. Go to **AlwaysOn High Availability** and expand the following folders: Availability Groups, the availability group you created to configure your databases high available, and Available Databases.
   Right click on **IDERA Dashboard Repository**, and click on **Remove Database from Availability Group**. The Remove Database from Availability Group wizard appears to confirm the removal of the database form the availability group, click **OK**.
   Right click on **SQL Safe Repository**, and click on **Remove Database from Availability Group**. The Remove Database from Availability Group wizard appears to confirm the removal of the database form the availability group, click **OK**.

Once you remove the databases from the availability group, upgrade the SQL Safe Backup and IDERA Dashboard.

Upgrading the SQL Safe Backup and IDERA Dashboard

The following instructions guide you through the upgrade of the SQL Safe Backup and IDERA Dashboard in an AlwaysOn Environment.

1. Log on with an administrator account to the computer where the server hosting the SQL Safe Backup and IDERA Dashboard is located.
2. Open the **SQL Safe Backup installer** and click **Run**.
3. Before you begin, the **SQL Safe Backup** wizard displays information about what is needed to complete the installation successfully. Click **Next**.
4. By default, the installer automatically detects the features you have already installed. Those features are selected to **upgrade**.
5. Review the license agreement. To accept this license agreement, select the **I accept the Terms and Conditions** checkbox. Click **Next**.
6. Determine if you are going to **register the SQL Safe Backup** with an existing IDERA Dashboard.
   By default, the option Yes is selected. SQL Safe Backup automatically detects if you have a local IDERA Dashboard installed in your computer. If your IDERA Dashboard is located on a remote server, then specify the HostName and Port number of this server. Type administrator credentials to access this Dashboard and click **Next**.
   If you select No, then proceed to the next step.
   A message appears letting you know that an older version of IDERA Dashboard is found and proceeding further will upgrade the existing one. Click **OK** to continue.
7. By default, the installer uses the **Destination Folder** you specified when you installed the product. Click **Next.**
   A message appears letting you know that an existing SQL Safe Backup product is registered on this Dashboard and the current instance will be upgraded. Click **OK** to continue.
8. The **SQL Server Instance and Database Name** remains the same as your previous installation of the product. You can change the Authentication for the SQL Safe Backup Repository. By default, the **Windows Authentication** is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the login name and password for this account, and click Save.
   Before you continue, you can **Test Connections** to make sure the SQL Server instance is connected. Then, click **Next**.
9. Specify the **Service Account** under which the **SQL Safe Backup** and **IDERA Dashboard** services will run under. Click **Next**.
10. Specify the account to be used by the **SQL Safe Backup Management Service** to connect to the SQL Safe Backup Repository Database. By default, the **Windows account** specified on the previous step is selected. If you want to use the **Microsoft SQL Server Authentication**, select this option and specify the respective SQL Sever login credentials, and click **Next**.
11. Once the **SQL Safe Backup** is ready to install, review the settings you selected and click **Install** to install the program.
    If you want to edit your settings, select **Back** and make your changes.
12. When the installation completes, you can either:
    - Go to the **Start Menu**, Select IDERA > SQL Safe Management Console.
    - Open the **IDERA Dashboard** through the URL https://localhost:9291 from a web browser.
    - Configure the **Firewall**.
    - Launch the **SQL Safe Management Console**.

    By default, the **Launch the SQL Safe Management Console** checkbox is enable. To complete the installation, disable it, then click **Finish**.

Once you complete upgrading the SQL Safe Backup and IDERA Dashboard, configure the Databases for High Availability.

Configuring Databases for High Availability

The following steps guide you to configure the IDERA Dashboard Repository and SQL Safe Repository databases highly available using the Windows Failover Cluster Service for Windows Server 2008 and above.

1. Open the **SQL Server Management Studio (SSMS)** and connect to the **secondary nodes** of the availability group.
2. **Delete** the copies of the **IDERA Dashboard Repository and SQL Safe Repository databases** from the **secondary nodes**.
   Go to Databases, right click on IDERA Dashboard Repository, and select Delete.
   Go to Databases, right click on SQL Safe Repository, and select Delete.
3. In **SQL Server Management Studio (SSMS)** connect to the **active node** of the availability group.
4. Set the **recovery model** of the **IDERA Dashboard Repository and SQL Safe Repository databases** to **Full**.
   Go to Databases, right click on IDERA Dashboard Repository, and select Properties. Go to options and set the Recovery Model to Full. Click OK.
   Go to Databases, right click on SQL Safe Repository, and select Properties. Go to options and set the Recovery Model to Full. Click OK.
5. **Perform a Full backup** of the **IDERA Dashboard Repository and SQL Safe Repository**.
   Go to Databases, right click on IDERA Dashboard Repository, select Tasks, and Backup. The Backup Database - IDERA Dashboard Repository window appears, select the Backup Type as Full, and click OK.
   Go to Databases, right click on SQL Safe Repository, select Tasks, and Backup. The Backup Database - SQL Safe Repository window appears, select the Backup Type as Full, and click OK.
6. **Sign in to the listener machine**, open the **SQL Server Management Studio (SSMS),** and connect to the **listener** name.
7. **Add Databases** to the Availability Group.
   Go to AlwaysOn High Availability, expand Availability Groups, right click on the availability group, and select **A dd Database**. The Add Database to Availability Group wizard opens displaying information about what is needed to add one or more availability databases to an existing availability group. Click Next.
   Select the IDERA Dashboard Repository and SQL Safe Repository **databases for the availability group**. Click **Next**.
   Select your **data synchronization** preference , then click Next.
   Connect to all the existing **secondary replicas**, then click Next.
   Ensure the **availability group validation** tests are successful. Click **Next.**
   Once the **Database** is ready to add to the availability group, review the Summary and click **Finish**. If you want to edit your settings, select **Previous** and make your changes.
   When the Database is included to the availability group, the results section displays it. Click **Close** to exit the wizard.
8. In **SQL Server Management Studio (SSMS)**, connect to the **secondary replica(s)** and ensure that the databases have been added successfully.

Once you complete it, you have successfully upgraded SQL Safe Backup and IDERA Dashboard with the repository databases hosted in an availability group.

You may begin using SQL safe Backup by launching the SQL Safe Management Console from the Start menu.

In an AlwaysOn Environment, SQL Safe Backup will require a license key for each replica to ensure continued functionality of the product after a failover event.

The license key for SQL Safe Backup is tied to the name of the SQL Server instance hosting the repository database.

4.1.5 Deploy the SQL Safe XSP

There are two ways to deploy the SQL Safe XSP: remotely through the Management Console or locally through the command-line interface (CLI).

If you are upgrading a Backup Agent, you will be prompted to perform an upgrade of the XSP for all instances on the target SQL Server.

If you do not want to deploy the XSP to all instances on a given SQL Server, you can deploy the XSP to a single instance through the Management Console or the CLI.

XSP deployment through the Management Console

You can upgrade the XSP when you deploy or upgrade the Backup Agent from the Management Console. At that time, you will be prompted for permission to install or perform an upgrade of the XSP for all instances on the target SQL Server. To install or upgrade the XSP, click **Yes**, and complete the authentication information as necessary. This action will install the new XSP on all instances on the SQL Server.

You can also deploy the XSP to a single instance.

To deploy the SQL Safe XSP to a single instance:

1. In the **Servers** tree, select the instance to which you want to deploy the XSP.
2. On the right-click context menu, click **Install SQL Safe Extended Stored Procedures**.
3. Click **OK**.

XSP deployment using the SQL Safe CLI

If you did not install the XSP during the Backup Agent upgrade, or you want to deploy the XSP to select instances on a given SQL Server, you can install the XSP manually.

To deploy the SQL Safe XSP with the standalone installer:

1. Log on with an administrator account to the SQL Server computer on which you want to install the SQL Safe XSP. Ensure your logon account also belongs to the System Administrators role on each SQL Server instance.
2. Ensure you have the most current version of SQL Safe Backup.
3. Open the Command Prompt, and navigate to the directory where the SQL Safe CLI is installed. By default, the CLI is installed in `C:\Program Files\IDERA\SQL Safe`.
4. Type `SQLsafeCmd InstallXsp –InstanceName` *`MyInstance`* `–Server` *`MyServerComputer`*, specifying the name of the instance and the SQL Server computer. For more information about available InstallXSP options, such as specifying authentication credentials, see the CLI Help. To view the CLI Help, type `SQLsafecmd help installxsp`.

4.1.6 How the InstantRestore Service works

The SQL Safe Filter service is responsible for performing InstantRestore operations. The service manages the following properties, which are stored in the registry:

- Max number of concurrent hydrations
- Number of driver threads
- Large raw buffer IO size
- Hydration chunk sizes
- Decompressed block cache size
- Driver active IO list size

> ⚠ You must use the same Windows account for the SQL Safe Backup Service and SQL Safe Filter Service. During installation, you are asked to enter credentials for only one account and the other is created with the same information. If you manually change your account information, make sure you change it in the other service as well to avoid any issues.

How do I enable or disable the InstantRestore feature?

SQL Safe Backup includes the SQL Safe Filter Service during an installation or upgrade. The SQL Safe Filter Service is responsible for performing InstantRestore (IR) operations. Users can enable or disable the IR component in the SQL Safe Management Console (right-click on the registered instance and select "Enable SQLsafe InstantRestore" or "Disable SQLsafe InstantRestore"). If an InstantRestore operation is in progress when a user attempts to disable the IR component, SQL Safe Backup returns an error.

5 Navigate the Web Console Dashboard

IDERA SQL Safe Backup provides a central point of control for managing SQL Server backups enterprise-wide and includes a powerful exception reporting failed backups. SQL Safe Backup enables complete, hands-free, automated management of your entire SQL Server backup infrastructure, and ensures compliance with your backups and recovery policies.

SQL Safe Backup can be managed through the web console Dashboard. To get started with this web console follow the Welcome Wizard steps.

5.1 Navigate the Welcome Wizard

The first time you access the SQL Safe Web console, a Welcome Wizard opens where you can configure the main settings of your environment. Follow these steps to navigate through this Wizard:

- On the **Welcome** tab, click **Next** to go to the next section
- On the **Add New Users** tab, you can add users. SQL Safe Backup let you select between three types of user roles: Administrator, User, and Guest. Each user role can perform different actions and can access to different sections in the interface. To specify new users, click **Manage Users**. On the **Manage Users** dialog window, use the option **Add User / Group** to specify a new user. To find more information about user roles and what options you have when registering a new user, click Manage users. After adding your users, click **Close** and **Next** to go to the next section.
- On the **Add Instances** tab, you can register those SQL Server instances you want to monitor with SQL Safe Backup. Click **Add Instances** to register your new instances. For more information about registering new instances, go to Add instances. After you register your instances, click **Next** to go to the next section.
- The **Finish** tab allows you to learn about the new options available in SQL Safe Backup. Click **Finish** to exit the Welcome Wizard.

You should use a SQL Safe Backup license that is valid for Centralized Licensing in order to be able to register instances.

5.2 Adding SQL Server instances

In order to perform backup and restore operations on SQL Server instances, you have to register them with SQL Safe Backup. To register SQL Server instances, you can find the option **Add Instance** available on the following tabs:

- **Home** tab
- **Policies** tab
- **Operation History** tab
- **Instances** tab
- **Databases** tab

You should use a SQL Safe Backup license that is valid for Centralized Licensing in order to be able to register instances.

When you click **Add Instance**, the wizard for registering new instances opens and you can find the following sections:

5.2.1 INSTANCE

Use this section to specify the instance or instances you want to register with SQL Safe Backup for monitoring. To add new instances you can use any of the following options:

- Type the name of the instance inside the SQL Server instance box. Use a semicolon to separate multiple instances.
- Use the option **BROWSE** to get a list of instances discovered by the CWF service. Select those instances you want to register.

Click **NEXT** to go to the following section.

5.2.2 CREDENTIALS

In this section you can specify the accounts SQL Safe Backup uses to gather information about your SQL Server instances. You have to define two type of credentials in this section:

SQL Connection Credentials

SQL Safe Backup uses these credentials to connect and perform queries against your new registered SQL Server instances and collect configuration, availability, performance, and capacity data. You can specify a Windows user or a SQL Server login Account Type.

WMI Connection Credentials

SQL Safe Backup uses these credentials to connect to the instance's host computer and gather configuration and performance data.

Use the option **TEST CREDENTIALS** to make sure the credentials specified can gather data for instances and host computers.

Click **NEXT** to go the following section.

5.2.3 FINISH

Use this section to review all your settings. You can go to previous sections to make any needed changes. Click **FINISH** to finish registering your SQL Server instances.

SQL Safe Backup collects initial configuration and availability data from the instances and their databases. When the initial collection is complete, you can see their information and their respective health recommendations.

5.3 What information is available on the Home tab?

On the **Home** tab you can find the following information:

- Alerts from your environment
- Largest databases and longest backups in your environment
- Managed Instances
- Summary information about your environment

Go to each link to view more detailed information about each **Home** tab section.

5.3.1 What alerts are available on the Home tab?

On this section of the SQL Safe Backup **Home** tab, you can find a summary of your environment's alerts. SQL Safe Backup groups these alerts by level of severity: OK (green), Informational (gray), Warning (yellow), Critical (red).

What actions can you perform on Alert Rules?

In each Alert rule, you can find the following options:

- **Show Details** - to view the complete list of instances or databases affected by the alert rule. Use the option **Hide Details** to display only the summarized form of the alert rule.
- **Dismiss** - to ignore the alert rule with their associated alerts.
- **Refresh** - to get the latest status of the alert rule.

What actions can you perform individually on each instance alert?

When you choose to **Show Details** of a specific alert rule, you can see all instances or databases associated with that specific alert rule. On each alert you can:

- **Dismiss** - to ignore the alert for that specific instance or database.
- **Refresh** - to get the latest status of that specific alert.

> When you **Dismiss** an alert, it will not be displayed on the **Home** tab temporarily. SQL Safe will raise the alert when the alert rule condition triggers it again.

Additional options

According to the type of alert, you can find additional options, relevant to the alert, that can help you solve the situation:

- **SQL Safe Backup Agent connection failures** - you can select to Start/Restart the SQL Safe Backup Agent
- **SQL Safe Backup Agent should be upgraded** - you can select to Upgrade SQL Safe Backup Agent. When you select this option, SQL Safe Backup opens a new window with the following options: **Install SQL Safe Agent Extended Stored Procedures** and **Include Backup Service Install Log**. Select your options and click **OK** to install them.
- **Databases never been backed up** - you have the following options:
  - **Add to Backup Policy** - Use this option to select an existing policy. The wizard for the selected policy opens with the respective database added to the policy. Make any necessary changes or updates and save your policy.
  - **Create Backup Policy** - Use this option to open the Create Policy Wizard. The respective Database will be added to the policy.
- For alerts showing **Errors or Failed operations** - you can perform the following options:
  - **Backup again**.
  - **Backup with different options**.
  - **Restore again** - Click this option to execute the operation again with the previously defined options.
  - **Restore with different options** - Click this option to open the Restore wizard and view the settings defined for the respective operation. You can review and change the restore options.

Take into account that actions available for SQL Safe Backup connection failures and Databases never been backed up are only available for Administrators.

What other actions can you perform on the Alerts section?

SQL Safe Backup allows you to perform the following actions on the Alerts section of the **Home** tab:

- **Export** - you can export your information in any of the following formats: PDF, XLS, or XML. Additionally, you can choose from one of the following options:
  - **Summary of recommendation categories** - select this option to export all alert rules in their summarized version.
  - **Details for all recommendations** - select this option to export all alert rules information including the details of the instances or databases that are associated with them.
  - **Details for selected categories** - you can use this option to select those specific alert rule details you want to export.
- **Hide alerts** - use this option if you do not want to see the displayed alerts.

What SQL Safe Backup alerts can you find?

To find out more about what alerts SQL Safe Backup notifies you about, go to Available Alerts and view the complete list of alerts, their description, severity level, and links to the respective knowledge base articles.

5.3.2 Largest databases and longest backups in your environment

Below the **Alerts** section of the **Home** tab, you can view the following two bar graphs:

- **Top Databases by Size (MB)** - this graphic displays the top largest databases in your environment in MB.
- **Longest running backups by Database (minutes)** - this graphic displays the top longest running backups by database in your environment in minutes.

5.3.3 Viewing SQL Server Instances on the Home tab

On the **Managed Instances** section of the **Home** tab you can see a summary view of all your registered SQL Server instances on your environment. For each instance, SQL Safe Backup displays the following information:

- **Status** - you can see an up status when the Instance is connected and the SQL Safe Backup Agent is running. You get a down status when either the Instance connection failed or the SQL Safe Backup Agent is not running.
- **Instance Name** - the name of your SQL Server instance.
- **SQL Server version** - the version of your SQL Server instance.
- **Status text** - in this column you can view the detailed description of your Status. For example, if it shows a down status, SQL Safe Backup displays whether is due to an Instance connection failure, the SQL Safe Backup Agent could not be contacted, or some other error occurred.
- **# of Databases** - displays the number of databases per SQL Sever instance. Click this number and SQL Safe Backup displays the **Databases** tab filtered by the selected instance.
- **# of Policies** - displays the number of policies that include at least one database of the respective SQL Sever instance. Click this option and SQL Safe Backup takes you to the **Policies** tab where you can see the list of all the policies related to your SQL Server instance.
- **# of Operations with failure** - displays the number of operations for this instance where the last status is a failure. Click this option and SQL Safe Backup takes you to the **Operation History** tab where you can see all those operations with an **Error** status and filtered by your selected instance.

What other actions can you perform on the Managed Instances section?

- **Add instance** - click this option to register new instances. Go to Adding SQL Server instances to find more information about registering new instances.
- **Create policy** - click this option to access the Backup, Restore, and Log Shipping Policy wizards.
- **Backup** - this option opens the Backup Wizard. Go to Using the Backup Wizard to know more about the steps for setting up backup operations.
- **Restore** - this option lets you choose among restoring: Database(s) or Object Level Recovery. Select the respective option and SQL Safe Backup opens the Restore Wizard.
- **Export** - you can export the information available about your SQL Server instances in any of the following formats: PDF, XLS, or XML.
- **Additional options** - Click the gear icon  to access this wiki or check for updates.

> ⚠ Take into account that none of these options are available for Read-only users.

5.3.4 What Summary information can you see on the Home tab?

On the right side of your **Home** tab, you can view a summary section with the most important information from your environment. This information is divided in the following sections:

My Environment

In this section you can view the number of:

- **Managed instances** - click this option and SQL Safe Backup takes you to the **Instances** tab.
- **Not-contacted instances** - click this option and SQL Safe Backup takes you to the **Instances** tab where you can see the instances in your environment filtered by **Error** status.
- **Not-contacted Backup Agents** - click this option and SQL Safe Backup takes you to the **SQL Safe Backup Agents** tab where you can see those instances whose SQL Safe Backup Agent have an **Error** status.
- **Databases** - click this option and SQL Safe Backup takes you to the **Databases** tab.
- **Databases with failed backup** - click this option and SQL Safe Backup takes you to the **Databases** tab filtered by **Error** status.
- **Databases with failed restore** - click this option and SQL Safe Backup takes you to the **Databases** tab filtered by **Error** status.
- **Backup Policies** - click this option and SQL Safe Backup takes you to the **Policies** tab where you can see policies filtered by **Backup** Policy Type.
- **Restore policies** - click this option and SQL Safe Backup takes you to the **Policies** tab where you can see policies filtered by **Restore** Policy Type.
- **Log Shipping policies** - click this option and SQL Safe Backup takes you to the **Policies** tab where you can see policies filtered by **Log Shipping** Policy Type.

Status Details

In this section you can view a summary of the general status of your environment operations. The number of policies with OK and not OK status, and the number of successful and failed operations.

Disk Space Savings

In this section you can view the amount of disk space saved by SQL Safe Backup. You can view this information for Today, this Month, this Year, the total savings, and the ROI (Return on Investment).

SQL Safe Backup provides a built-in calculator to help you calculate your monetary return on your SQL Safe Backup investment. The calculator attempts to measure the time and monetary savings you gain through using the SQL Safe Backup compression scheme. The Return On Investment (ROI) calculator bases your ROI on the total cost of ownership of your storage devices multiplied by the amount of disk space savings you achieve using SQL Safe Backup. SQL Safe Backup defaults to the commonly used estimate of $200 per GB of storage. You can change this estimate to reflect your particular hardware configuration by accessing the Basic Configurations of the **General Preferences** window on the **Administration** tab.

5.3.5 Available Alerts

What SQL Safe Backup alerts can you find?

You can find the following alerts:

Alert	Description	Severity
SQL Server Instance Connection Failure	displays those instances that cannot be contacted	Critical
SQL Safe Backup Agent Connection Failure	displays those SQL Servers in your environment where the SQL Safe Backup Agent could not be contacted.	Critical
SQL Safe Backup Agent upgrade needed	displays the instances where your SQL Safe Backup Agent needs to be upgraded.	Informational
Databases never backed up	displays that have never been backed up by SQL Safe.	Warning
Backups failed	shows operations where the last backup for a specific database and backup type failed.	Critical
Backups did not start as scheduled	shows operations where the last backup for a specific database and backup type did not start as scheduled.	Critical
Backups were skipped	shows operations where the last backup for a specific database and backup type were skipped.	Informational
Backups were cancelled by user	shows operations where the last backup for a specific database and backup type were cancelled by the user.	Warning
Backups completed with warnings	shows operations where the last backup for a specific database and backup type were completed with warnings.	Warning
Backups succeeded	shows operations where the last backup for a specific database and backup type were successful.	OK
Restores failed	shows databases where the last restore operation for a specific database failed.	Critical
Restores did not start as scheduled	shows databases where the last restore operation for a specific database did not start as scheduled.	Critical
Restores were cancelled by user	shows databases where the last restore operation for a specific database were cancelled by the user.	Warning
Restores completed with warnings	shows databases where the last restore operation for a specific database completed with warnings.	Warning

Alert	Description	Severity
Restores succeeded	shows databases where the last restore operation for a specific database completed successful.	OK
Log Shipping data too old	show databases where the last log shipping backup or restore for a specific database is too old.	Warning

5.4 Using the Backup Wizard

SQL Safe Backup helps you see the status of your environment; when databases were last backed up or when backup operations finished with warnings. Almost all tabs provide information about your recent backup operations and backup policies.

SQL Safe Backup Web Console allows you to access the Backup Wizard from the following sections:

- **Home** tab.
- **Instances** tab.
- **Databases** tab.

The Backup Wizard contains the following sections:

- Databases
- General
- Locations
- Options
- Notifications
- Finish

Go to each section to review the respective steps and details.

Take into account that the Backup option is not available for **Guest** role users.

5.4.1 Selecting databases for backup

The **Databases** section of the Backup Wizard allows you to specify the SQL Server instance that hosts the databases and the specific databases you want to back up.

What can you do on the Databases tab?

You can select the SQL Server instance that hosts your target databases. Use the drop down arrow to access those instances in your environment.

After selecting your instance, the database list is populated. From the database list, select the databases you want to backup. You can select from the list of **System Databases** or **User Databases**. SQL Safe allows you to see when the Last Backup was performed on each database and how much space was used.

If you want to ...	**Select this option ...**
Back up all databases on the selected SQL Server instance	All Databases
Back up only User databases on the selected SQL Server instance	All User Databases
Back up only System databases on the selected SQL Server instance	All System Databases (master, model, msdb, distribution)
Back up only the databases you specify on the selected SQL Server instance	Specific Databases, and then choose the appropriate databases

Why is the target instance not listed?

The instance list only includes SQL Server instances that have been registered with SQL Safe. If the instance is not in the drop-down list, make sure the instance is registered by using the Add Instance Wizard.

Once you select the databases for your backup, click **NEXT** to choose the backup type.

5.4.2 Choosing the backup type

The **General** section of the Backup Wizard allows you to specify the backup type, name, and description of the backup you are creating.

What types of backups can you choose?

SQL Safe supports the standard SQL Server database backup types:

- Full.
- Differential.
- Transaction Log.
- File.

What should you do for your initial backup?

If you are backing up the database for the first time, select **Full**. A full backup will provide a comprehensive data set and SQL Safe requires it to perform differential backups or transaction log backups later on. For more information about backup types, go to understand backup types.

What to choose in the Format drop-down option?

By default, the SQL Safe format is displayed in this option. Backups created using the SQL Safe file format can only be restored by SQL Safe. The SQL Server format is the native SQL Server backup file format. Backups created using the SQL Server format can be restored using SQL Safe and native SQL Server tools.

When should you specify a description?

After you specify the name for you backup operation, you should provide a description to identify important details about this operation so you can easily recognize which backup sets should be restored later. The backup description will appear in the status view of past and current backups and will allow you to identify problems when they occur.

What is a copy-only backup?

A copy-only backup is a copy of the database, not a true backup, and cannot be used as a part of a restore strategy or restore chain. It is a backup that does not affect the log sequence numbers (LSN) of the database.

How do you verify the integrity of your backup?

You can choose the option to **Verify Backup**. When this option is selected, SQL Safe performs a data integrity check after the backup has been created. SQL Safe only verifies the integrity of the data files in the backup set created by this backup.

Verifying the backup helps identify potential issues that could occur when restoring data files.

Once you choose your backup type, click **NEXT** to select the location of your backup files.

5.4.3 Selecting the location of your backup files

The **Locations** section of the Backup Wizard allows you to specify the backup location you want to use to store the backup set.

For a TSM backup, you can change the TSM connections settings to override the values set in the client options file if you need to write the backup files to a TSM Server other than the TSM Server already specified in the dsm.opt file.

Where can you store your backup set?

SQL Safe supports the following location types:

- Single File.
- Striped Files.
- Tape (Tivoli Storage Manager).
- Data Domain.
- Amazon S3 Cloud.
- Microsoft Azure Cloud.
- Tape (Tivoli Storage Manager) Striped Files.

When you perform a backup under the SQL Server format (native backup), the following location types are available:

- Single File.
- Striped Files.
- Data Domain.
- Microsoft Azure Cloud.

What actions do you have for each location type?

Location Type	Actions
Single File	• Choose if you want to Overwrite or Append the existing archive file, the name of the Primary Archive file, and the Mirror Archives Filenames.
Striped Files	• Choose if you want to Overwrite or Append the existing archive files. • Specify as many filenames as the number of striped files you want.
Tape (Tivoli Storage Manager)	• Choose if you want to Overwrite or Append the existing archive file. • Enter the path for the Primary Archive file (High Level, Low Level, and Management Class). • Specify the TSM Client Settings (Client Options File and Connection Settings). For these settings, you have to specify the name of the node you want to use, the password required to access the node, and the TCP/IP Server address and port.

Location Type	Actions
Data Domain	• Choose if you want to Overwrite or Append the existing archive file, the name of the Primary Archive file, and the Mirror Archives Filenames.
Amazon S3 Cloud	• Choose if you want to Overwrite or Append the existing archive file. • Specify your Amazon S3 cloud storage options to be used for your backup. These options include: File Name, Access Key, Secret Key, Region, Bucket Name, and File Size. For more information, go to Amazon Settings.
Microsoft Azure Cloud	• Choose if you want to Overwrite or Append the existing archive file. • Specify the information needed to access your Microsoft Azure cloud such as: File Name, Container Name, Azure Storage Account Name, Azure Access Key, and Sector Type. For more information, go to Azure Settings.
Tape (Tivoli Storage Manager) Striped Files	• Choose if you want to Overwrite or Append the existing archive file. • Determine the path to the High Level primary archive, its Management Class, and the information related to the TSM Client Settings (Client Options File, Node Name, Server Address, and Server Port). • Specify the number of striped archives and the details for each low level archive. ⚠ Take into account that if the number of stripes is greater than the available sessions on TSM server, the backup fails with a message "sessions are not available on TSM". There is no available way for the TSM client to find out available sessions on the TSM server.

What are striped files?

If you want to take advantage of distributing I/O overhead for a large database, you can select striped files and choose backup locations on different local disks.

How do you handle errors encountered while writing to the network during a backup?

Select **Enable Network Resiliency** and access the **Network Resiliency Settings** window where you can specify how often you want to retry when errors occur and after how much time the operation will fail. You can also configure how much time is allowed for the total retry time of the backup.

ⓘ This option is not available when backing up to tape using Tivoli Storage Manager or Amazon S3 Cloud.

What do you do if you do have an existing archive?

If the **Archive Exist**, you can select to **Append** it to an existing archived backup set or choose to **Overwrite** it.

What happens if you do not have an existing archive file?

If you do not have an existing archive file, SQL Safe Backup creates a new archive file to include the backup set.

> Keep in mind, the filename extension for all backups performed under the SQL Safe format are .safe and for all backups performed under the SQL server format are .bak.

How do you mirror your backups?

Select **Mirror Archives**, then specify where you want the mirror copies to be stored.

For each mirror, SQL Safe creates a copy of the backup archive set. You can specify up to two mirror archives for each backup operation. Keep in mind that creating mirrors can impact the performance of your backup operation.

If you want to stop the backup operation when the mirror location is unavailable, select **Abort backup if a mirror location reports a failure**.

> This option is only available when backing up to Single File and Data Domain.

Once you select the location of your backup files, click **NEXT** to configure options for your backup.

5.4.4 Configuring options for manual backup

The **Options** section of the Backup Wizard allows you to select additional options, such as compression and encryption to use for the current backup operation.

What types of compression algorithms are available?

- None.
- IntelliCompress, optimize for size (iSize).
- IntelliCompress, optimize for speed (iSpeed).
- Levels 1, 2, 3, 4.

A backup operation using Level 1 completes fastest but achieves the least amount of compression. Level 4 achieves maximum compression, but the backup operation may take longer.

For more information about backup compression, see how to choose compression and encryption.

When performing a backup under the SQL Server format, the compression options available changes to an option to use compression.

What types of encryption algorithms are available?

- None.
- AES (128-bit).
- AES (256-bit).

If your SQL Server environment requires FIPS compliance, use the AES encryption option. For more information, see ensure FIPS compliance.

When performing a backup under the SQL Server format, the encryption options are not available.

Does encryption require a password?

When you choose to encrypt an archive, you must assign a password. For security reasons, SQL Safe Backup does not store this password. Ensure you remember the password you select.

What are the advanced options?

The following options are available as Advanced Options:

Options	Description
Number of threads	Allows you to specify how many threads you want SQL Safe to use to distribute the backup operation for compression across multiple processors on the target SQL Server computer. Use this setting to optimize backup performance. Select **Auto** to have SQL Safe determine the optimal thread count for your environment.

Options	Description
Remove inactive transaction log entries	Removes all completed transactions from the transaction log after SQL Safe finishes the backup. This option is only available for Log backups.
Generate maps (for InstantRestore and SQL virtual database)	Generates maps containing metadata for each database included in your backup file. Depending on the number of transactions completed since your last backup, generating maps may impact the performance of the backup operation. Generating maps is optional but must exist in the backup file for InstantRestore to accept and restore that file. SQL virtual database can attach SQL Safe files without the metadata, but the data files improve SQL VDB performance during creation of the virtual database. For more information, see recover objects using SQL virtual database. This option is selected by default.
Include database logins in backup file	Copies SQL login information for the selected databases, including credentials and privileges, when the backup files are written. To help you ensure the security of your SQL Server database, SQL Safe encrypts the login information. This option is available for full backups only.

⚠ Advanced options are only available when performing backups using the SQL Safe file format.

What are the advanced options for SQL Server 2005 and later?

The following options are available as Advanced Options for SQL Server 2005 and later:

Options	Description
Generate checksums	Generates a checksum for the backup file.
Ignore checksum errors	Select this option to ignore any errors from the generated checksum. If checksum errors are encountered, this option indicates that SQL Safe should continue to back up this database.
Read-write filegroups	Specifies a partial backup, which includes the primary filegroup and any read-write secondary filegroups. Read-write filegroups are not supported by SQL virtual database. If this option is selected, the Generate metadata option will be disabled. Additionally, backups created with the read-write filegroups option cannot be used by SQL virtual database to create virtual databases.

Once you configure the options for your backup, click **NEXT** to configure notifications.

5.4.5 Configuring notifications for manual backup

The **Notifications** section of the Backup Wizard allows you to email status notifications to the appropriate database administrators about this backup. Email notifications let you, and your staff, remotely monitor the status of your backups.

You can choose any of the following status to monitor:

- Backup fails.
- Backup is skipped.
- Backup is cancelled by user.
- Backup competes with warnings.
- Backup succeeds.

Type the email address of each recipient. Use semicolons to separate multiple email addresses.

> You must configure your mail server settings before SQL Safe can send email notifications. Click **Configure E-mail** to check your settings. For more information, see configure e-mail settings for status notifications.

When is the email sent?

SQL Safe sends an email to the specified recipients only when the selected backup status occurs. For example, if you chose to monitor whether the backup fails, you will not be emailed if the backup is skipped. Because you are performing a manual backup, you will receive one status notification.

Once you configure notifications for your backup, click **NEXT** to review details.

5.4.6 Reviewing details for manual backup

When you reach the **Finish** section of the Backup Wizard, you can go to previous sections and make any necessary changes.

You can also click **Generate Script** to have SQL Safe generate T-SQL and Command Lines of your backup operation settings.

After you have reviewed the information, click **Finish** to submit the backup job immediately.

How do you verify the status of your backup?

If you chose to run the backup job immediately, and want to verify a successful run, you can view its status in the **Operation History** tab.

5.4.7 Generating scripts for backup and restore operations

You can generate **CLI** and **T-SQL** scripts for backup and restore operations through the Backup and Restore wizards. SQL Safe generates the CLI or T-SQL script using the settings you specified for the backup or restore operation.

CLI scripts can be run as a batch file from the command line prompt. Generated CLI scripts use supported options for the backup and restore actions.

T-SQL scripts can be run through Query Analyzer or as a scheduled SQL Server job. Generated T-SQL scripts leverage the SQL Safe XSP to execute backups and restores.

If you need a command line or T-SQL script for your backup or restore, SQL Safe provides the Generate Script button to let you generate CLI and T-SQL scripts for these operations. When you use a wizard to run a backup or restore, SQL Safe disables this button until sufficient criteria exists to generate a script. SQL Safe generates the CLI or T-SQL script using the settings you specified for the backup or restore operation.

How do you generate script?

You can generate script through the Backup Wizard or the Restore Wizard once your settings provide SQL Safe with enough information to create the script. Click Generate Script, and SQL Safe displays command line script by default. Click the T-SQL button and SQL Safe displays the script in T-SQL format.

To retain your script in either format, click the Save to a file or Copy script to clipboard icon. SQL Safe also allows you to use normal select, cut, copy, and paste functionality directly on the displayed script.

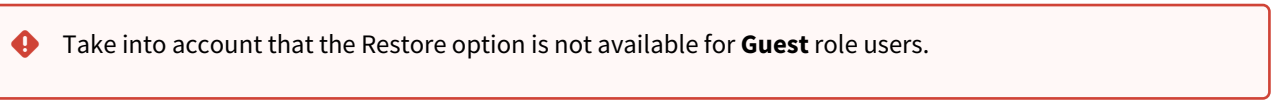
5.5 Using the Restore Wizard

The SQL Safe Backup web console allows you to access the **Restore Wizard** from the following sections:

- **Home** tab.
- **Instances** tab.
- **Databases** tab.

When you click the **Restore** option from any of the previous locations, SQL Safe Backup prompts you to choose a restore operation for:

- Database(s) or
- Object Level Recovery



5.5.1 Restoring Databases

When you select to **Restore Database(s)**, the Restore Database dialog box opens with the following sections:

- Target
- Databases
- Backup Sets
- Database Files
- Recovery State
- Restore Type
- Notifications
- Finish

Selecting the target instance for restore

The **Target** section of the Restore Wizard allows you to select the instance to where you will restore the database.

On this section, you can follow these steps:

- Select from the drop-down list the SQL Server instance where you want to restore your databases. If your instance is not listed, then register it by using the Add Instance Wizard.
- Choose which action you want to perform:
    - **Restore** - use this option to restore your databases. You can instruct SQL Safe to disconnect users from the databases before performing the restore by selecting the option **Disconnect users before the restore.**
    - **Verify** - select this option when you want to ensure your backup operations are successful without actually restoring your data. Consider using this restore verification option on all critical backups after executing the backup operation.
- You can specify the number of threads for decompressing data or you can choose the **Auto** option so that SQL Safe calculates the optimum number of threads for your operation.

Once you select the target instance for your restore, click **NEXT** to select databases.

Selecting the databases you want to restore

The **Databases** section of the Restore Wizard allows you to specify the databases you want to restore and the general location of the corresponding archive files. You can select backups from the following locations:

- **Repository** - use this option when the backup files reside in your repository. Choose the SQL Server where the database(s) to be restored were backed up, then select the databases you want to restore.
- **File System** - use this option when the archive file was written to the local File System. Type the path from the network share or local drive. Use the **Browse** option to load the backup info of an specified file in your File System. This path must be accessible by the Backup Agent installed on the Agent Computer.
- **Target Server** - use this option when a network share is available on a remote file system (Target Server). Type the respective path or click **Browse** to select the file from the file list.
- **Tivoli Storage Manager** - use this option when the backup was performed using TSM. Specify the TSM Client Option File, Node Name and Password, Server Address and Port, and Path. When restoring from a TSM Server, browse for the correct database archive. You can change the TSM connections settings to override the values set in the client options file. If you have to specify TSM striped files, click **Fetch Stripes** to access them. You can also include inactive files.
- **Amazon S3 Cloud** - select the location of your cloud storage where the respective backup files are located. You have to specify the Access Key, Secret Key, Bucket Name, and Region. For more information, go to Amazon Settings. Click **Load Bucket** and **Load Backups** to access your files in your Cloud Storage Account. Select the files and/or databases you want to restore.
- **Microsoft Azure Cloud** - use this option to access your Microsoft Azure Cloud where the backup files are stored. You have to specify your Container Name, Azure Storage Account Name, Azure Access Key, and Sector Type. For more information, go to Azure Settings. Click **Load files** and **Load db** to access your files in your Microsoft Azure Account. Then select the files and/or databases you want to restore.

Once you select databases for your restore, click **NEXT** to select backup sets.

Selecting the backup set for the restore

The **Backup Sets** section of the Restore Wizard allows you to choose which backup sets you want to use to restore the selected databases.

> The backup sets listed in the main window can be reorganized by Type, Backup Set Name, Format, Date/Time, and Encryption.

How do you select Backup Sets?

For each database you have selected to restore, the backup set listing is populated with the available backup sets from the Repository. You can easily identify your SQL Safe backups or native backups (SQL Server) in the format column. You can choose the specific backup sets you want to restore, or you can select a point in time to which you want to restore data. To restore more than one database, select a backup set for each database.

What are the encryption settings?

Use the option **Encryption Settings** for backup sets that were encrypted during its backup. Use this option to specify the encryption password.

How do you keep your restores running despite network errors?

Click **Configure Resiliency** to configure your **Network Resiliency Settings** and keep the restore operations running despite network errors. By default, SQL Safe will retry the restore operation every 10 seconds and then fail after 5 minutes (300 seconds) of continuous errors. Also, over the course of the operation, SQL Safe allows a total of 60 minutes in which to retry the restore before stopping the operation. You can configure these settings according to your requirements.

Once you select the Backup Sets for you restore, click **NEXT** to specify your database files.

Specifying database files

The **Database Files** section of the Restore Wizard allows you to change the names of the restored databases or edit the data file locations of the restored files.

What settings can you specify on the Database Files section?

For each database you have selected to restore, you can edit the following fields:

- **Restore As** - type a new name for the database you will restore.
- **Select restore options:**
  - **Force Restore (replace)** - choose this option to ensure the selected backup files are restored, even if that means overwriting an existing database.
  - **Restore Database logins** - choose this option to restore the SQL logins associated with the selected databases. This option is available when you are restoring a full backup that contains the database login information.
  - **Keep CDC** - choose this option to restore databases that uses Microsoft SQL Server Change Data Capture (CDC) feature.
  - **Ignore Checksum Errors** - choose this option to ignore any errors from the generated checksum. If checksum errors are encountered, SQL Safe should continue to restore the backup file.
  - **Preserve replication settings** - choose this option to retain the settings used when the selected databases were replicated.
- **Change Path** - type a new path for the target database.
- **Edit the filename of the restoring file** - Edit the names you want to change in the **Restore as Filename** field. When you select a database name from the drop-down list or edit the field, the **Restore As Filename** is automatically updated to reflect the new name, but you can edit this field by directly typing in it.

To restore a database over an existing database, select the **Force Restore** option to ensure SQL Safe writes the selected backup files over the existing database.

Once you specify your database files, click **NEXT** to select the recovery state.

Selecting recovery state

The **Recovery State** section of the Restore Wizard allows you to choose the recovery state that each database should be left in after the restore operation.

Which recovery states are supported?

SQL Safe supports the following recovery states:

- **Fully Accessible** - Leaves the database operational. No additional transaction logs can be restored.
- **Not Accessible (no recovery mode)** - Leaves the database non-operational but able to restore additional transaction logs.
- **Accessible but read-only (standby mode)** - Leaves the database in read-only state and able to restore additional transaction logs. You can specify an **Undo file** for this option.

Take into account that **Fully accessible** is the only recovery state supported by the **InstantRestore** feature. If you choose a partial recovery state, you cannot restore your database using InstantRestore. For additional information about performing an InstantRestore, see how InstantRestore works .

Once you select the recovery state, click **NEXT** to select the restore type.

Selecting restore type

The **Restore Type** section of the Restore Wizard allows you to choose whether you want to use the Normal SQL Safe Restore or the InstantRestore option when restoring your database.

⚠ Take into account when **InstantRestore** can or cannot be used:

- **InstantRestore** is not available for all restores as not all properties are supported; for example, you cannot restore a SQL Safe backup from a TSM Server.
- **InstantRestore** supports only complete database restores and does not support file or filegroup restores.
- **InstantRestore** can only work when you choose a Fully Accessible recovery state.
- **InstantRestore** does not support native format files.

What is the benefit of using InstantRestore?

In most cases, **InstantRestore** allows you to use the database almost immediately after starting the restore. If you have large databases that you need to access very quickly, this may be the best option for you, but take into account that there may be some performance issues if your users are making changes to the database while the restore is in progress.

InstantRestore will bring the database online quickly allowing you to access your data while SQL Safe continues to restore the database in the background.

What options do you have available in this section?

You can choose between:

- **Normal SQL Safe Restore** - SQL Safe will restore the database using the traditional restore engine and the database will become available when the restore completes.
- **SQL Safe InstantRestore** - the database will become available in a fraction of the time that a normal restore normally takes.

✓ If you have not enabled the **InstantRestore** feature yet, click **Enable InstantRestore**.

Once you select the restore type, click **NEXT** to configure notifications.

Configuring notifications for manual restore

The **Notifications** section of the Restore Wizard allows you to email a status notification to the appropriate database administrators about the restore operation. Email notifications let you and your staff remotely monitor the status of your restores.

Choose any of the following status you want to monitor:

- Restore fails
- Restore is cancelled by user
- Restore completes with warnings
- Restore succeeds

Type the email address of each recipient, separating them with semicolons.

> You must configure your mail server settings before SQL Safe can send email notifications. Click **Configure E-mail** to check your settings. For more information, see configure e-mail settings for status notifications.

When is the email sent?

SQL Safe sends an email to the specified recipients only when the selected restore status occurs.

Once you configure notifications, click **NEXT** to review details.

Reviewing details for manual restore

You are ready to start the restore operation, but if you want to change any of the restore settings, you can go to the previous sections and change the respective configurations.

You can click **Generate Script** to generate the respective Command Line and T-SQL scripts with all restore configurations. For more information about generating scripts, click here.

What do you do next?

After you have reviewed the respective configurations, click **Finish** to submit the restore job immediately. You can view its status in the Operation History tab.

5.5.2 Restoring Object Level Recovery

When you select to perform an **Object Level Recovery Restore**, the Restore Object Level Recovery dialog box opens with the following options:

- **Target** - in this section, you can select the SQL Server instance where you want to restore. Go to selecting the target instance for restore for more detailed information.
- **Databases** - specify the databases you want to restore and the general location of the corresponding archive files. Go to selecting the Databases for restore for more detailed information.

  > ⚠ Take into account that Tivoli Store Manager, Amazon S3 Cloud, and Microsoft Azure Cloud are not available for Object Level Recovery.

- **Backup Sets** - choose which backup sets you want to use for restore. Go to selecting the Backup set for restore for more detailed information.
- **Temporary Location** - select where to restore virtual databases. Type the path in **Temporary Location** and select your instance from the drop-down options.
- **Database Objects** - for each **Database Virtual Object**, specify in **Restore As** the name with which you want to restore your database virtual object and type your **File location**. You can also select the database objects to restore separately, such as tables, stored procedures, functions, views, schemas, etc.
- **Finish** - before clicking **Finish** to proceed with your restore operation, you can go to previous sections to review and make any necessary modifications.

> ⓘ SQL Safe Backup does not support Object Level Recovery options with compressed and encrypted native backups from SQL Server 2014 and higher.
>
> For more information about recovering objects using SQL Virtual Database, click here.

5.6 Viewing your Policies

On the **Policies** tab, you can see all the existing **SQL Safe** policies in your environment. For each policy, you can see the following information:

- **Status** - displays the status of the operations for each database The status are: wait, failed, missed, skipped, cancelled, completed with warnings, or succeeded.
- **Policy Type** - displays the type of policy: backup, restore, or log shipping.
- **Policy Name** - displays the name with which the policy was created.
- **Databases Covered** - displays the number of databases covered by the policy.
- **Instances Covered** - displays the number of instances covered by the policy.
- **Last Operation** - displays the last operation executed in the policy.
- **Last Operation with Failure** - displays the last failed operation in the policy.

Keep in mind:

- You can sort the information available on this tab by clicking the column header by which you want to order your instances.
- You can set the number of items you want to view per page by going to the lower section of the list grid, type a number in the box, and the page will be updated according to your requirements.

5.6.1 How do you filter your information?

In order to get more specific policy information from your environment, you can use the **Filtering** section on the left side of the **Policies** tab.

You can filter your information by:

- **Status** - select the status of the policies you want to view. The status are:
  - **Wait** - no operations have been performed for the policy.
  - **Failed** - the last policy operation for one or more databases failed.
  - **Missed** - one of the last scheduled policy operations for one or more databases did not start as scheduled.
  - **Skipped** - one of the last scheduled policy operations for one or more databases were skipped.
  - **Cancelled** - one of the last scheduled policy operations for one or more databases were cancelled by the user.
  - **Completed with warnings** - one of the last scheduled policy operations for one or more databases completed with warnings.
  - **Succeeded** - all of the last operations for each database and operation type completed successfully.
- **Policy Type** - select the type of policy: Backup, Restore, or Log Shipping.
- **Policy Name** - type the name with which the policy was created.
- **Databases Covered** - use the **From** and **To** options to set specific range of databases you want to filter by policy.
- **Instances Covered** - use the **From** and **To** options to set specific range of instances you want to filter by policy.
- **Last Operation** - select the date using the Calendar. On the **From** and **To** options, set specific date range to filter the last operation executed in the policy.
- **Last Operation with Failure** - select the date using the Calendar. On the **From** and **To** options, set specific date range to filter the last failed operation in the policy.
- **Instance** - type the name of the instance under which the policy is running under.
- **Database** - type the name of the database under which the policy is running under.

When using filters take into account:

- You can save your filtering options by selecting your filters, typing a name in the **By Custom Filter** field, and clicking **Add Filter**. To retrieve your saved filters, click the drop-down option in the **By Custom Filter** section and select your filter name.
- If you want to select filters first and apply the changes later, deselect the **Apply filter as it changes** option.
- To remove filters, use the specific **Remove Filter** option in each filter. For example, if you want to remove your Status filters, click **Remove Status Filter** under the same filter section.
- Under **APPLIED FILTERS** on the top section or your **Policies** tab, you can see the filters you have selected. Click the **X** icon next to the ones you want to remove.
- Use the option **Clear** on the top section of the **Filtering** section to remove all filters.

5.6.2 What other options are available from the Policies tab?

The **Policies** tab allows you to perform the following actions located on the upper section of this tab:

- **Add instance** - click this option to register new servers. Go to adding SQL Server instances to find more information about registering new instances.
- **Create policy** - use this option to access the wizard for creating backup, restore, or log shipping policies.
- **Edit policy** - select one of the available policies and click this option to edit its configuration settings.
- **Copy policy** - this option allows you to select an existing policy and create a copy. When you select this option, the Policy Wizard opens with the selected policy settings. You can change these settings according to your preferences. Take into account you have to specify a different name for the copy.
- **Actions** - use this option to enable the policy, disable the policy, or to Run a Backup.
- **Properties** - select this option to access the summary details of a selected policy. You can also use this option to change the configuration settings of a policy.
- **Remove/delete** - select a policy from the list of available policies and click this option to delete it. You can select to delete only the policy or also the jobs associated with the policy.
- **Export** - click this option to export the displayed information on the **Policies** tab. You can select exporting your information in PDF, XLS, or XML file.

5.6.3 Create Backup Policies

SQL Safe helps you create backup policies, restore policies, and log shipping policies to address different needs. Through backup policies, for example, you can define backup maintenance plans across multiple SQL Server instances in your enterprise.

What is a backup policy?

A backup policy consists of:

- A list of databases you want to back up.
- A set of backup operations to be performed on those databases.
- A set of schedules according on which the backups will be performed.

You can choose to run the associated backup jobs on a specific schedule, to run them on demand (execute the jobs manually from the Management Console), or to define a policy for monitoring purposes only. You can then monitor the status of each backup policy from the Management Console.

How do you incorporate backup strategies in your policies?

Implementing a policy requires that you have a clear understanding of your backup strategy. To determine a backup strategy to use, consider the following recovery model requirements.

Model	Full Backup?	Differential Backup?	Transaction Log Backup?	File or Filegroup Backup?
Simple Model	Required	Optional	N/A	N/A
Full Model	Required	Optional	Required	Optional
Bulk-Logged Model	Required	Optional	Required	Optional

What constitutes a good backup strategy?

Consider using all four backup types to maximize your recovery and minimize your data loss. A basic backup strategy fulfills the following needs:

- The creation of regularly scheduled database backups.
- The creation of frequent differential backups between full backups.
- The creation of transaction log backups more frequently than differential backups.

Database backup creation depends on server activity and data sensitivity. Ensure you implement a strategy that allows you to create policies that back up both user databases and system databases.

How do backup policies help you?

Backup policies allow you to plan and schedule your SQL Server backup maintenance. You can monitor operations success and failures from a single point of contact in the Management Console.

The process of updating your maintenance plans becomes quick and easier by allowing SQL Safe to schedule a set of backup operations across your SQL Server instances.

How do you access the Backup Policy Wizard?

The SQL Safe Backup Policy Wizard allows you to create backup maintenance plans across your enterprise. SQL Safe creates SQL Server jobs for the specified backups in your policy.

You can access the Backup Policy Wizard from the top options located on the Home, Policies, Instances, and Databases tab.

Once you open the backup wizard, you can define the following settings:

- Name the policy.
- Select the databases you want to back up.
- Select backup options.
- Specify where you want to store the backup files.
- Schedule when and how often you want the backup to occur.
- Get email notifications about the policy status.
- Review details.

Naming the policy

The **General** tab of the SQL Safe Backup Policy Wizard allows you to specify the basic properties of the backup policy.

Why should you specify a name or description?

SQL Safe requires that you enter a unique name for each policy. Both the name and description will appear in the status messages for your policies. Using a meaningful name and description allows you to easily identify problems when they occur. For example, consider specifying a description that will help you later choose the correct backup to restore during a disaster recovery situation.

⚠ When supplying a name and/or description, do not end with the backslash (\ ) character.

Which format should you choose to perform your backup?

Depending on the needs of the backup policy. The option to select between SQL Safe file format and native SQL Server file format is available.

✓ For native SQL Server file format select "SQL Server". For SQL Safe file format, which is compatible with InstantRestore, select "SQL Safe".

Which policy action should you choose?

Choose the action that best reflects how you want to use this policy. According to your requirements, you can select from one of the following options:

Policy Action	Description
Monitor and automatically create backup jobs using the SQL Server Agent	Creates the policy for monitoring database backups and automatically creates the backup jobs using the SQL Server Agent on your SQL Server instances. Creating jobs allows to enforce consistent backup settings across your environment.
Monitor and automatically create backup jobs using the SQL Safe Backup Agent	Creates the policy for monitoring database backups and automatically creates backup jobs using the SQL Safe Backup Agent which is in charge of executing and scheduling these policies.
Monitor only	Creates the policy to only monitor database backups and no jobs are created. By default, SQL Safe will monitor the status of any backup operation that meets the criteria of your policy.

Note that SQL Server Express does not support the SQL Server Agent. Use the second option if you add any instance with SQL Server Express in your policy. This option allows the SQL Safe Backup Agent (second option) to create your policy backup jobs.

If you choose to use the SQL Safe Backup Agent, policy data files will be stored by default at C:\Program Files\IDERA\SQL Safe\PolicyData. You can change these settings by going to the **Administration** tab, selecting the **General Preferences** option, and typing the preferred folder directory in the **Policy Data** tab.

Once you define some policy settings, click **NEXT** to select your databases.

Selecting Databases

Use the **Membership** tab of the SQL Safe Backup Policy Wizard to select which SQL Server instances and databases you want to monitor with this policy.

Follow these steps to select your instances and databases:

1. Click **Add/remove Instances** and choose the instances from where you want to backup databases. Click **OK**.
2. For each selected SQL Server instance, define which databases SQL Safe will take into account for this policy. You can select from the following options: **All Databases**, **All User Databases**, **All System Databases**, **Specific Databases**. When choosing one of the database options, the policy will automatically include the relevant databases as they are added on the server and the Edit Database Selection window opens.
    - In the Edit Database Selection window some options are available depending on the databases you chose in the previous step. You can use the WildCard option or the Manually option to exclude or include specific databases from your backup.

    When you use the WildCard options:
    - **%** or ***** - substitutes one or more characters
    - _ - substitutes a single character
    - \\ - type double backlash to avoid it acting as the escape character
    - [abc] - searches for abc
    - [**^**abc]- searches and matches for those characters that are not abc
    - [a-c] - searches and matches a, b, or c
3. Click **OK**.

Once you select your databases, click **NEXT** to configure your backup options.

Configuring options

The **Options** tab of the SQL Safe Backup Policy Wizard allows you to enter the backup types and options for each operation included the backup policy.

For each backup operation you include in the backup policy, you can select compression, encryption, verification options, and set additional advanced options.

What types of backup can you choose?

You can specify one, two, or the three types of backup: **Full**, **Differential**, **Log**.

Select which backup types you want for your policy and provide the respective settings. Take into account that the options for each backup type are hidden until the backup type is selected. For more information about backup types, view understand backup types.

What types of compression algorithms are available?

SQL Safe provides the following compression algorithms:

- None.
- IntelliCompress, optimize for size (iSize).
- IntelliCompress, optimize for speed (iSpeed).
- Levels 1, 2, 3, 4.

> Backup operations using Level 1 complete fastest but achieve the least amount of compression. Level 4 achieves maximum compression but the backup operation may take longer.

For more information about backup compression, see how to choose compression and encryption.

> When performing a backup under the SQL Server format, the compression options available changes to an option to use compression.

What types of encryption algorithms are available?

SQL Safe provides the following encryption algorithms:

- None.
- AES (128-bit).
- AES (256-bit).

If your SQL Server environment requires FIPS compliance, use the AES encryption option. For more information, see ensure FIPS compliance.

> When you choose to encrypt an archive, you must designate a password. For security reasons, SQL Safe does not store this password. Ensure you remember or take note of the password you select.

> When performing a backup under the SQL Server format, the encryption options are not available.

What additional options are available?

SQL Safe Backup provides additional options that can be applied to SQL Safe Backup Policies.

The following list describes options that are available depending on the backup format (SQL Safe or SQL Server) chosen on the General tab.

Options	SQL Safe	SQL Server	Description
Use compression	Not available	Available	• You designate the compression rate necessary to match your storage needs.
Encryption	Available	Not available	• Select the level of encryption you need to ensure data security within your environment.
Verify the integrity of the backup when complete	Available	Available	• Verifies the integrity of the backup set data files created by this backup. • Verifying the backup helps identify potential issues that could occur when restoring these data files.
Generate maps	Available	Not available	• Generates maps containing metadata for each database included in your backup file. This option is selected by default. • Depending on the number of transactions completed since your last backup, generating maps may impact the performance of the backup operation. • Generating maps is optional, but it should exist in the backup file for InstantRestore to accept and restore that file. • SQL virtual database can attach SQL Safe backup files without the metadata, but the data files improve SQL VDB performance during tcreation of the virtual database. • For more information, see recover objects using Virtual Database.

Options	SQL Safe	SQL Server	Description
Report T-Log operations that are skipped as SUCCESS	Available	Available	• Allows SQL Safe to report SKIPPED T-Log operations as SUCCESS. • Avoids backup policies from reporting a warning status when T-Log operations are skipped for databases that are in simple recovery.
Include database logins in backup file	Available	Not available	• Copies SQL login information for the selected databases, including credentials and privileges, when the backup files are written. • To help ensure the security of your SQL Server database, SQL Safe encrypts the login information. This option is available for full backups only.
Thread Count	Available	Not available	• Allows you to specify how many threads you want SQL Safe to use to distribute the backup operation across multiple processors on the target SQL Server computer. Use this setting to optimize backup performance. • When the resultant backup file is restored, SQL Safe uses the same thread setting to ensure consistent performance. • Select **Auto** to have SQL Safe determine the optimal thread count for your environment.
Transaction Log	Available	Available	• Removes all completed transactions and inactive entries from the transaction log after SQL Safe finishes the backup.
Checksum: Generate	Available	Available	• Generates a checksum for the backup file. • It must be enabled to allow the Checksum: Ignore Errors availability.

Options	SQL Safe	SQL Server	Description
Checksum: Ignore Errors	Available	Available	• If the Checksum: Generate option is enabled, then either format is available. • If checksum errors are encountered, this option determines that SQL Safe should continue to backup process.
Backup: Copy only	Available	Available	• Specifies a copy-only backup. This is a copy of the database and cannot be used as part of a restore strategy. • Allows to take a "snapshot" backup of your database without interfering the LSN (log sequence number) order of your backup strategy.
Backup: Read-write filegroups	Available	Available	• Specifies a partial backup, which includes the primary filegroup and any read-write secondary filegroups. • If this option is selected, the **Generate metadata** option (Generate maps for InstantRestore and SQL virtual database) will be disabled. • Backups created with the read-write filegroups option cannot be used by SQL virtual database to create virtual databases.

Once you configure options for your backup, click **NEXT** to select your backup location.

Selecting location

The **Locations** tab of the SQL Safe Backup Policy Wizard allows you to specify the backup location for each operation you include in the backup policy.

For each operation you have included in the backup policy, you can specify the location type, full path in which to store the backup file, an optional housecleaning schedule for existing disk archives, and the backup file extension.

What types of backup locations can you use?

SQL Safe supports the following location types:

- Back up to a single file on the local computer or a network share.
- Back up to tape using Tivoli Storage Manager.
- Back up to multiple striped files on the local computer or a network share.
- Back up to Data Domain.
- Back up to Amazon S3 Cloud.
- Back up to Microsoft Azure Cloud.
- Back up to tape using Tivoli Storage Manager Striped Files.

When you perform a backup under the SQL Server format (native backup), the following location types are available:

- Single File.
- Striped Files.
- Data Domain.
- Microsoft Azure Cloud.

What do you do if you do not have an existing archive?

If you do not specify an existing archive, SQL Safe creates a new backup set with the name you specify. The location entered for each backup type must be valid for all SQL Server instances. You can choose to **Append** or **Overwrite** if the archive already exists.

What accounts can you specify to access the backup files location?

Specify the account to access the filesystem when you select any of the following location types: Single File, Striped files, or Data Domain. Depending whether you selected the SQL Server Agent or the SQL Safe Backup Agent for your policy, you will be able to choose between SQL Server Agent service account/SQL Safe Backup Agent service account respectively or a Windows account. Click **Account** and select your preferred option.

The account specified must have read and write privileges on the directory selected for your backup file location.

How do you keep the backups running despite network errors?

Select **Enable network resiliency** and then click **Configure** to change the default settings. By default, SQL Safe will retry the backup operation every 10 seconds and fail after 5 minutes (300 seconds) of continuous errors. Over the course of the operation, SQL Safe allows a total of 60 minutes in which to retry the backup before stopping the operation.

This option is not available when backing up to tape using Tivoli Storage Manager or Amazon S3 Cloud.

Can you change the default file locations?

SQL Safe automatically populates the path using several available variables, depending on the location type. You can modify this path to suit your needs, taking advantage of all the available variables.

For a disk backup, browse for or type the directory in which to store the backup file.You can use the supplied macros in the way best suited to your storage needs. And if you want to limit the lifetime of your backup sets created by the policy, you can select the option that removes files older than the specified time.

For a TSM backup, browse for or enter the high level directory for the tape file. Then, browse for or enter the location of the TSM configuration file.

Keep in mind, the filename extension for all backups performed under the SQL Safe format are .safe and for all backups performed under the SQL server format are .bak.

What does removing old files do?

For backups written to a single file or mirrored files, you can choose to remove old files to prevent disk space limitations. When you select to remove files older than the specified time, backup files created with names of the same format will be deleted from that directory. You can configure SQL Safe to delete old backup files from the primary archive as well as from your mirror archives.

For backups written to a TSM Server, you can configure SQL Safe to mark these files as inactive after a specified age. This option is not available when using Amazon S3 Cloud.

How do you mirror the backups this policy creates?

When selecting Single File or Data Domain as location types, you have the option to create mirror archives. For each mirror archive, SQL Safe creates a copy of the backup archive set. Click **Mirror Archives** and specify where you want the mirrored files to be stored. Take into account that you can specify up to two mirrors for each backup operation.

If you want to stop the backup operation when mirror location is unavailable, select **Abort backup if a mirror location reports a failure**. You can also enable to remove files older than a specified time.

Keep in mind that creating mirrors can impact the performance of your backup operation.

What do you specify when backing up to a TSM Server?

When a TSM location is selected, you have to specify the following settings:

- The configuration file.
- High level file path.
- Low level file path.
- Management Class.

You can also configure SQL Safe to mark these files as inactive after a specified age.

Note that SQL Safe accepts up to 260 characters for the TSM file path name.

SQL virtual database is not available when backing up to a TSM Server.

How do you backup to multiple stripes using TSM Servers?

You can also back up to tape using Tivoli Storage Manager Striped Files. If you select this option, provide the following information:

- The configuration file.
- High level file path.
- Management class.
- The number of striped archives to use and the respective names for each low level file name.

You can also configure SQL Safe to mark these files as inactive after a specified age.

Take into account:

- If the number of stripes is greater than the available sessions on TSM server, the backup fails with a message "sessions are not available on TSM". There is no available way for the TSM client to find out available sessions on the TSM server.
- If a password is required for TSM, then policies created using TSM may not work correctly.

What do you specify when backing up to Amazon S3 Cloud?

When Amazon S3 Cloud backup location is selected, you have to specify the following fields:

- **Append/Overwrite** - select if you want to append the backup archive to an existing one or if you prefer to overwrite it.
- **Filename** - to be used as your primary backup archive.
- **Access Key** - specify the access key generated in your security credentials of your Amazon S3 web console. For more information, click here.
- **Secret Key** - specify the secret key generated in your security credentials of your Amazon S3 web console. For more information, click here.
- **Region** - select the region where your information will be stored. You can find more information about these regions here.
- **Bucket Name** - define the name of the Amazon S3 bucket where your backup will be stored.
- **File Size** - File size is used to determine the minimal parts of the backup file in bytes that will be sent to the bucket simultaneously. The minimal value for File Size is 1 000 000 bytes. Note that when backing up to a cloud location, the network quality may affect performance.

What do you specify when backing up to Microsoft Azure Storage?

When Microsoft Azure Storage is selected, make sure you specify the following fields:

- **Container name** - the name of the Azure container where your backup will be stored. Every blob in Microsoft Azure storage must reside in a container. The container forms part of the blob name. For more information, click here.
- **Azure Storage Account Name** - the account name of your storage account. Every object you store in Azure Storage has a unique URL address. The storage account name forms the subdomain of that address. You can find more information in the following link.
- **Azure Access Key** - the access key to your Azure Storage Account. For more information about Azure Keys, click here.
- **Sector Type** - define the Azure sector type:
  - *Public* - commercial cloud storage solution.

    - *Government* - cloud storage solution offered to US government customers and their partners.
- **Enable Network Resiliency** - enable or disable the network resiliency settings for your backup operations. You can click on Configure to define how to handle errors encountered while writing to the network during a backup.
- **Filename** - to be used as your primary backup archive.

Once you determine your backup location, click **NEXT** to configure a schedule for your backup operations.

Configuring schedule

The **Schedules** tab of the SQL safe Backup Policy Wizard allows you to schedule the frequency and duration of your backup operations. For each backup type, enter the appropriate information in the schedule fields according to your backup requirements.

You can specify when your operation will begin, how frequently backup jobs will be executed, and the respective duration of these operations. You can also choose to run the operation "On Demand," allowing you to manually execute the associated jobs with your preset options.

Each backup operation can have a different schedule. For instance, perhaps you decide you want to run full backups monthly, differential backups once a week, and transaction logs during business hours every day.

How do you know what frequency to set?

The schedule of your operations should be determined by how much data you can afford to lose in the event of a catastrophic failure. The schedule should be developed in agreement with your backup strategy. For example, for lab or development instances, you may want to schedule on-demand or have weekly backups. For critical production instances, you may want to schedule full backups every day with transaction log backups every hour.

How do you set the schedule?

You can set up a schedule by defining the following options:

Field	Description
Occurs	Unit of Frequency: • On Demand • Daily • Weekly • Monthly
Daily Frequency	Time of day: • Occurs once at `HH:MM:SS AM/PM` • Occurs every `N` Hours/Minutes starting at `HH:MM:SS AM/PM`, ending at `HH:MM:SS AM/PM`
Duration	Length of time: • Start date `mm/dd/yyyy` • End date `mm/dd/yyyy` • No end date

When an operation does not occur as scheduled, the backup policy will consider it "missed." SQL Safe can notify you about this missed operations if you configure this respectively.

Once you configure the schedule of your backup operations, click **NEXT** to configure notifications.

Configuring notifications

The **Notifications** tab of the SQL Safe Backup Policy Wizard allows you to choose from which backup status you want to receive alert notifications. Email notifications let you, and your staff, remotely monitor the status of the backups you have automated with this policy. The status of the backup operations determines the status of your policy. When your backups are successfully completed on scheduled, the policy is considered ok.

Choose the status you want to monitor, type the email address of each recipient, select the desired alert frequency for each operation, and then click **Next**.

> ⚠ You must configure your mail server settings before SQL Safe can send e-mail notifications. Click **Configure E-mail** to check your settings. For more information, see configure e-mail settings for alert notifications.

When is the email sent?

SQL Safe sends an email to the specified recipients when the selected operation status occurs. Because SQL Safe checks the status of your backup operations every minute, your alert notifications provide a real-time indication of the health of your service level agreements and disaster recovery plans for the SQL Server instances covered by this policy.

However, how often SQL Safe notifies about a specific status update depends on the notification frequency you select. For example, if you want to receive an email whenever a backup fails, even when the failures occur sequentially, choose to receive notifications every time the event occurs.

Once you configure notifications, click **NEXT** to review details.

Reviewing details

The **Summary** tab of the SQL Safe Backup Policy Wizard provides the summary of the specified values and options you have selected through the Backup Policy wizard tabs. If you want to change any of the configuration settings, go to the respective tab. After you review the information on the Summary tab, click **Finish** to create the policy and corresponding backup jobs.

If you want to create the policy but not the backup jobs, return to the General tab and select the **Monitor Only** action.

5.6.4 Create Restore Policies

The SQL Safe Restore Policy Wizard allows you to create restore maintenance plans across your enterprise. A restore policy is defined as a set of databases for which restore operations will be performed according to a defined schedule. By default, SQL Safe creates the SQL Server jobs for the specified restores.

You can create a restore policy for any database that belongs to a backup policy and has a full backup.

How do you access the Restore Policy wizard?

You can access the Restore Policy Wizard from the top options located on the Home, Policies, Instances, and Databases tabs.

To get started with the Restore Policy Wizard:

- Name the policy.
- Select the source database which contains the data you want to restore.
- Select the target database where the data will be restored.
- Get email notifications about the policy status.
- Review details.

Naming the restore policy

The **General** tab of the Restore Policy Wizard allows you to specify the basic properties of the restore policy. You are required to enter a unique name for each policy.

Both the name and description will appear in the status messages for your policies. Using a meaningful name and description will allow you to more easily identify problems when they occur. For example, consider specifying a description that will help you later choose the correct restore operation to monitor during a disaster recovery situation.

What options do you have for creating a restore policy?

When you create a restore policy, you can choose from between the two following options:

- Create Restore Jobs using the SQL Server Agent.
- Create Restore Jobs using the SQL Safe Backup Agent .

> ⚠ Note that SQL Server Express does not support the SQL Server Agent. Use the second option if you add any instance with SQL Server Express in your policy. This option allows the SQL Safe Backup Agent (second option) to create your policy restore job.

Once you define some policy settings, click **NEXT** to select your databases.

Selecting databases you want to restore

The **Source** tab of the Restore Policy Wizard allows you to specify the database you want to restore, the location of the corresponding backups, and which account SQL Safe should use to access these files. Follow these steps in this tab:

- Use the drop down arrows to specify the SQL Servers that host the databases you want to restore.
- Select the databases you want to restore.
- Choose the location of your backup. Click **Select** and SQL Safe displays the available backup locations from the databases backup policies.

> ⚠ If the database selected is not associated with a backup policy, SQL Safe prompts to create the respective policy. SQL Safe will automatically restore to the latest backup found in that location each time your restore policy runs.

How do you keep your restores running despite network errors?

Select **Enable Network Resiliency** and then click **Configure** to change the default settings. By default, SQL Safe will retry the restore operation every 10 seconds and then fail after 5 minutes (300 seconds) of continuous errors. Over the course of the operation, SQL Safe allows a total of 60 minutes in which to retry the restore before stopping the operation. You can change these settings according to your requirements.

What accounts can you specify to access the backup files?

Depending whether you selected to use the SQL Server Agent or the SQL Safe Backup Agent for your restore policy, on this section you have the option to select between SQL Server Agent service account/SQL Safe Backup Agent service account respectively or another account with the respective credentials.

> ⚠ The specified user account must have read and write privileges on the selected directory for the backup file location.

Once you specify the databases you want to restore, click **NEXT** to select the target database.

Selecting the target database

The **Target** tab of the Restore Policy Wizard allows you to specify the server, database name, and restore schedule for your new database.

You can perform the following actions:

- Select the SQL Server instance where your target database is.
- Select the database you want to update or create a new one.
- Specify the location of the data and log files of the restored database. SQL Safe displays each database file (by logical name) and the directory where each will be restored.
- Choose the appropriate recovery state for the database (Fully Accessible, Accessible but read-only, or Not Accessible).
- Schedule when the Agent should execute the restore job.
- Select other restore options.

How do you change the location of your database files?

If SQL Safe does not display the correct location where you want to restore a file, click **Select** in the Database File Locations section and change it to your preferred location.

The Database File Locations window allows you to manage the paths where SQL Safe restores new data and log files. Files names are created automatically, using the file type and destination database for easier identification.

Can you access your files during the restore operation?

To access your files while SQL Safe executes the restore operation, choose from the following options:

- **Fully Accessible** - Leaves the database operational. No additional transaction logs can be restored.
- **Accessible but read-only (standby mode)** - Leaves the database non-operational but able to restore additional transaction logs.
- **Not accessible (no recovery mode)** - Leaves the database in read-only state and able to restore additional transaction logs.

How do you set the restore schedule?

You can click **Schedule** on the Restore Job option and set the frequency and the duration of your restore policy job.

How do you restore the SQL logins for this database?

You can recover SQL logins associated with this database by selecting the **Restore database logins** option. You can use this option when the source backup files contain login information. To capture login information, configure your backup policy to include the database logins.

> When you choose a database file with ".bak" or native format, the option Restore database logins is disabled.

What do you do if you have users connected to the database?

You can instruct SQL Safe to disconnect users from the database before performing the restore. To do so, select the **Disconnect users** option from the restore options.

What other additional options do you have?

Additionally, you have the following options when performing your restore:

- **Ignore checksum errors** - Select this option to ignore any errors from the generated checksum. If checksum errors are encountered, this option indicates that SQL Safe should continue to back up this database.
- **Preserve replication settings** - Choose this option to retain the settings used when the selected databases were replicated.
- **Keep CDC** - choose this option to restore databases that uses Microsoft SQL Server Change Data Capture (CDC) feature.

Once you select the target database, click **NEXT** to configure notifications.

Configuring notifications for restore policy

The **Notifications** tab of the Restore Policy Wizard allows you to choose from which restore status you want to receive alert notifications. Email notifications let you, and your staff, remotely monitor the status of the restores you have automated with this policy. The status of the restore operations determines the status of your policy. When your restores are successfully completed on scheduled, the policy is considered okay.

Choose the status you want to monitor, type the email address of each recipient, select the desired alert frequency for each operation, and then click **Next**.

> ⚠ You must configure your mail server settings before SQL Safe can send e-mail notifications. Click **Configure E-mail** to check your settings. For more information, see configure e-mail settings for alert notifications

When is the email sent?

SQL Safe sends an email to the specified recipients when the selected operation status occurs. Because SQL Safe checks the status of your restore operations every minute, your alert notifications provide a real-time indication of the health of your service level agreements and disaster recovery plans for the SQL Server instances covered by this policy.

However, how often SQL Safe notifies about a specific status update depends on the notification frequency you select. For example, if you want to receive an email whenever a restore fails, even when the failures occur sequentially, choose to receive notifications every time the event occurs.

Once you configure notifications, click **NEXT** to review details.

Reviewing details for restore policy

The **Summary** tab of the Restore Policy Wizard provides the summary of specified values and options you have selected in the Restore Policy wizard. If you want to change any of the configuration settings, go to the respective tab.

After you have reviewed the information on the Summary tab, click **Finish** to create the policy and corresponding restore jobs.

5.6.5 Create Log Shipping Policies

Log shipping policies allow you to ship transaction logs between multiple SQL Server instances in your enterprise, on a scheduled basis. These instances can reside on one or more physical servers.

SQL Safe offers log shipping policies, backup policies, and restore policies to address different needs.

What is a log shipping policy?

A log shipping policy consists of primary and secondary databases you want to synchronize, a set of transaction log backup and restore operations to be performed on those databases, and a set of schedules according to which these operations will be performed. You can also choose to mirror the backup files, storing copies of the transaction logs in multiple secured locations. You can then monitor the policy status, all from a single point of contact in the Management Console.

How do log shipping policies help you?

Log shipping policies allow you to implement a disaster recovery strategy for your entire SQL Server environment. You can use log shipping policies to synchronize, or back up and restore one database to another. By synchronizing databases through this policy, you can save disk space, network bandwidth, and comply with security requirements. Each transaction log backup can be compressed and encrypted.

How do you access the Log Shipping wizard?

You can access the Log Shipping Policy Wizard from the following tabs: Home, Policies, Instances, and Databases. On any of these tabs, click **Create Policy** and then choose **Log Shipping Policy**.

To get started with the Log Shipping Policy wizard, follow these steps:

- Name the policy.
- Select the primary database that you want to back up.
- Specify where these transaction log files should be stored.
- Select the secondary database you want to synchronize with the primary.
- Select the notifications you want to receive for this policy.
- Review details.

Naming the log shipping policy

The **General** tab of the Log Shipping Policy Wizard allows you to specify the basic properties of the log shipping policy.

Why should you specify a name or description?

The wizard requires you to enter a unique name for each policy. Both the name and description will appear in the status messages for your policies. Using a meaningful name and description will allow you to more easily identify problems when they occur. For example, consider specifying a description that will help you later choose the correct backup to restore during a disaster recovery situation.

What options are available for creating a log shipping policy?

When you create a log shipping policy, you can choose between the two following actions:

- Create Backup and Restore Jobs using the SQL Server Agent.
- Create Backup and Restore Jobs using the SQL Safe Backup Agent.

> ⚠ Note that SQL Server Express does not support the SQL Server Agent. Use the second option if you add any instance with SQL Server Express in your policy. This option allows the SQL Safe Backup Agent (second option) to create your log shipping policy job.

How does SQL Safe determine that a log shipping policy is okay?

SQL Safe determines that the policy is okay by looking at the following statuses:

- Whether the transaction log backup on the primary database has completed on schedule.
- Whether the transaction log restore on the secondary database has completed without warnings or errors.
- Whether the data on the secondary database is stable.

How do you control when a log shipping policy is compliant?

You can control how SQL Safe determines a missed backup by changing these options:

- Select a time limit for the log backup to occur. This is the leeway time allowed for the log backup to occur. If the log backup occurs within this period from the scheduled time, the policy is still compliant.
- Select an age limit for the secondary's data. This is the tolerance level for how old the data in the secondary database can be.

Once you define some policy settings, click **NEXT** to select the primary database.

Selecting the primary database

Use the **Primary** tab of the Log Shipping Policy Wizard to select which SQL Server instance will be the primary source of the log files. This is the database you will be backing up by using log shipping.

What information is required on this tab?

On this tab, you have to specify the following fields:

- **SQL Server** - Select the SQL Server that contains the database to be backed up.
- **Database** - Select the database from which you will ship the backup logs.
- **Backup Job** - This schedule defines how often the backup job occurs. By default, SQL Safe schedules this job to occur every day, every 15 minutes between 12:00 AM and 11:59 PM, and to start on the current date. Click **S chedule** to change the frequency and start date.
- **Backup Options** - These options allow you to change the methods used for compression, encryption, and the number of threads used when performing a backup.

Log shipping cannot be performed on a database configured to use the simple recovery model. Your database should use the Full or the Bulk-logged recovery model. SQL Safe prompts you to change the recovery model if the simple recovery model is currently used at the database.

What types of compression algorithms are available?

You can select from the following compression algorithms:

- None.
- IntelliCompress, optimize for size (iSpeed).
- IntelliCompress, optimize for speed (iSize).
- Levels 1, 2, 3, 4.

Backup operations using Level 1 complete fastest but achieve the least amount of compression. Level 4 achieves maximum compression but the backup operation may take longer.

For more information about backup compression and encryption, see how to choose compression and encryption.

What types of encryption algorithms are available?

You can select from the following encryption algorithms:

- None.
- AES (128-bit).
- AES (256-bit).

If your SQL Server environment requires FIPS compliance, use the AES encryption option. For more information, see ensure FIPS compliance.

When performing a backup, ensure the backup does not truncate the transaction logs of the database. Truncating the log will cause this log shipping policy to fail.

Once you select the primary database, click **NEXT** to select the location.

Selecting location for log shipping

The **Location** tab of the Log Shipping Policy Wizard allows you to specify the location for the backups you are creating with this log shipping policy. Backups must be stored to a network path that all servers in the policy can write to.

What options can you set on this tab?

Access Filesystem As

This is the account SQL Safe uses to access the specified primary and mirror locations. Depending whether you selected the SQL Server Agent or the SQL Safe Backup Agent for your log policy, you can choose between SQL Server Agent service account/SQL Safe Backup Agent service account respectively or a Windows account. Click **Account** and select your preferred option.

> ⚠ Enter a user account that has access rights to the target locations. The user account used must have read and write permissions to the specified resource.

You can also choose how to handle errors encountered during a backup by selecting **Enable network resiliency**. By default, SQL Safe will retry the backup operation every 10 seconds and then fail after 5 minutes (300 seconds) of continuous errors. Over the course of the operation, SQL Safe allows a total of 60 minutes in which to retry the backup before stopping the operation. This option is not available when backing up to tape using Tivoli Storage Manager.

Primary Location

This is the first location where the backup files will be stored. By default, SQL Safe ships the backup files from this location to your secondary server. When you configure the secondary database settings, you can specify an alternate location.

Enter the network path of the location where you want the log backup archive to be kept. The destination folder must be configured as a network share. You can also specify how long you want to keep old backup files. By default, SQL Safe will delete files older than three (3) days.

> ⓘ SQL Safe detects if the Computer Browser service is not running on your computer. This service enables Windows to list other computers on the network. If this service is not running, Windows may not be able to list the computers on your network. SQL Safe allows you to start this service, but keep in mind that it may take several minutes for computers to become visible.

Mirror Archives

These are the locations where copies or "mirrors" of the backup files will be saved. For each mirror location, SQL Safe creates and stores a copy of the backup files. You can specify up to 2 mirrors for each log shipping operation. Keep in mind that creating mirrors can impact the performance of your log shipping operation.

You can also specify:

- How long you want to keep old backup files. By default, SQL Safe delete files older than three (3) days.
- Whether SQL Safe should cancel the backup when one of the specified mirror locations reports a failure like a connection timeout for example.

Once you determine the location, click **NEXT** to select secondary databases.

Selecting secondary databases

Use the **Secondary(s)** tab of the Log Shipping Policy wizard to select the SQL Server instances and databases where the log backups will be restored.

In this section, you can add, edit, or remove secondary databases. Each database can be restored with different options, schedules, recovery mode, etc.

How do you add secondary databases?

Click **Add Secondary Database** to add secondary databases. SQL Safe opens a new window where you can specify the SQL Server, the database, and the different restore options. You can determine the following fields:

- **SQL Server** - Select a registered SQL Server.
- **Database** - Create a new database or select the database that receives the transaction log restores. To create a new database, type directly the database name in the **Database** field. If you want to select an existing database, click **Select** to access the list of databases available on the selected instance.
- **Initialization** - Determine the initial state of the secondary database that receives the transaction log restores. By default, this field is set to Initialize database with a newly generated full backup. (copy only). Wh en you click **Change**, a window for Database Initialization options opens where you can choose:
    - **Do not initialize** - Database exists and has received the most recent full backup of a primary database.
    - **Initialize database with a newly generated full backup** - Selected by default. This is the only option available when SQL Safe detects the primary database was previously configured to use the simple recovery model. Since database backups using the simple recovery model lack log checkpoint information necessary for subsequent log restores, the database requires a new full backup to initialize the secondary database.
    - **Initialize database with these backups** - If you enable this option, you can specify the location of the backups and add encryption settings. In this section you can:
        - **Add** - type the UNC or local backup file path.
        - **Add from repository** - select this option, and click Load Backups.
    - **Database File Locations** - You can also click **Database File Locations** to choose where to store your database files. SQL Safe displays each database file (by logical name) and the directory where each will be stored. Filenames will be dynamically generated using the file type and destination database name.

Database State

Select the recovery mode in which the secondary database is left after each log restore, i.e. the status of the secondary database. You have two options:

- If you select **Not Accessible** (No recovery mode), then the secondary database shows the status as "Restoring". The database is unusable in this state.
- If you select **Accessible but read-only** (Standby mode), then the database is in a read-only state. In this option, you can also choose to disconnect users when performing the restore job.

Restore Job

This is how often the restore process will occur. By default, the restore job occurs every 15 minutes every day, between 12:00 AM and 11:59 PM, but by clicking **Schedule**, you can specify other settings for the daily frequency and duration of the job.

You can also choose to delay the restores by a number of minutes or hours. For example, setting this value to 15 minutes means the secondary database will always be, at least, 15 minutes out of sync.

Restore From

Specify the location from which the transaction log backup files will ship to the secondary database. You have two options:

- **Same location as backup** - To use the network path previously specified for the transaction log backup.
- **Different location** - To restore from a different location. If you select this option, type the appropriate network path or click **Select** to browse the location.

> ⚠ Take into account that to restore from a different location, the database must already be initialized.

Once you select secondary databases, click **NEXT** to configure notifications.

Configuring notifications for log shipping

The **Notifications** tab of the Log Shipping Policy wizard allows you to choose from which log shipping status you want to receive alert notifications. Email notifications let you, and your staff, remotely monitor the status of the backups and restores you have automated with this policy. The status of the log shipping operations determine the status of your policy. When your backups and restores are successfully completed on schedule, the policy is considered okay.

Choose the status you want to monitor, type the email address of each recipient, select the desired alert frequency for each operation, and then click **Next** .

> ⚠ You must configure your mail server settings before SQL Safe can send email notifications. Click **Configure E-mail** to check your settings. For more information, see configure e-mail settings for alert notifications.

When is the email sent?

SQL Safe sends an email to the specified recipients when the selected operation status occurs. Because SQL Safe checks the status of your backup and restore operations every minute, your alert notifications provide a real-time indication of the health of your log shipping policy and your primary and secondary servers.

However, how often SQL Safe notifies about a specific status update depends on the notification frequency you select. For example, if you want to receive an email whenever a restore fails, even when the failures occur sequentially, choose to receive notifications every time the event occurs.

Once you configure notifications, click **NEXT** to review details.

Reviewing details for log shipping

The **Summary** tab of the Log Shipping Policy wizard provides the summary of specified values and options you have selected in the Log Shipping Policy wizard. If you want to change any of the configuration settings, go to the respective tab.

After you have reviewed the information on the Summary tab, click **Finish** to create the policy and corresponding log shipping schedule. SQL safe opens a window with the list of tasks for your policy and verifies them.

5.7 Viewing your Operations History

The **Operations History** tab allows you to see all operations performed by SQL Safe in your environment. For each operation, you can see the following details:

- **Status** - this column lets you view the status of each respective operation. Status can be In Progress (percentage and bar indicating the progress of the operation), Queued, Successful, Successful with warnings, Error, Canceled, Skipped, or Deleted.
- **Details** - the details about the status of the respective operation.
- **Instance** - the name of the SQL Server instance where the operation was executed.
- **Database** - the name of the Database where SQL Safe executed the operation.
- **Operation** - the type of operation (Backup, Restore, Verify, InstantRestore, Hydrate).
- **Backup Type** - if it is a backup operation, then the type of configured backup.
- **Policy** - if the operation belongs to a policy, then the name of the policy.
- **Compressed**, **MB** - the size to which the data was compressed in MB.
- **Uncompressed**, **MB** - the uncompressed data in MB.
- **Ratio**, **%** - the ratio between compressed and uncompressed data in percentage.
- **Database size**, **MB** - the size in MB of the database where the operation was performed.
- **Compression** - the type of compression used for the backup operation.
- **Encryption** - the type of encryption selected for the backup operation.
- **Duration** - the total duration of the operation.
- **Start time** - the exact time when the operation started.
- **End time** - the exact time when operation finished.
- **Threads** - number of threads used by the operation.
- **Actions** - the gear icon under this column allows you to access options such as Set progress to, Backup again, or Backup with different options.

Keep in mind:

- You can sort the information available on this tab by clicking the column header by which you want to order your instances.
- You can set the number of items you want to view per page by going to the lower section of the list grid, type a number in the box, and the page will be updated according to your requirements.

5.7.1 How can you filter the information on the Operation History tab?

SQL Safe allows you to filter the information on the **Operations History** tab so that you can see quickly access your required information.

To filter your SQL Safe operations, go to the left section of the tab where you can find the following options for filtering:

- **Status** - specify the status of those operations you want to view. An operation can have any of the following status: In progress, Queued, Successful, Successful with warnings, Error, Canceled, Skipped, or Deleted.
- **Instance** - type those instances for which you want to view their SQL Safe operations.
- **Database** - specify those databases for which you want to view their operations.
- **Operation** - select those operation types (Backup, Restore, Verify, InstantRestore, Hydrate, or Object Level Restore) you want to view on the **Operation History** tab.
- **Backup Type** - enter the backup type for which you want to view its operations. You can choose from Full, Differential, Log, or File backup operations.
- **Policy** - specify if you want to include or exclude policy operations.

When using filters take into account:

- You can save your filtering options by selecting your filters, typing a name in the **By Custom Filter** field, and clicking **Add Filter**. To retrieve your saved filters, click the drop-down option in the **By Custom Filter** section and select your filter name.
- If you want to select filters first and apply the changes later, deselect the **Apply filter as it changes** option.
- To remove filters, use the specific **Remove Filter** option in each filter. For example, if you want to remove your Status filters, click **Remove Status Filter** under the same filter section.
- Under **APPLIED FILTERS** on the top section or your **Operation History** tab, you can see the filters you have selected. Click the **X** icon next to the ones you want to remove.
- Use the option **Clear** on the top section of the **Filtering** section to remove all filters.

5.7.2 What actions can you perform on operations?

Under the **Actions** column, you can find the following options when you click the gear icon:

- **Set progress to** - use this option to change the status of the operation. You can set it to Successful, Successful with warnings, Error, Canceled, Skipped, or Deleted.
- **Backup Again** - use this option if you want to repeat the backup operation. SQL Safe performs the backup with the established backup settings.
- **Backup with different options** - use this option if you want to review or change the backup settings. SQL Safe opens the Backup Wizard where you can review the settings used to perform the operation or edit according to your preferences.
- **Restore Again** - use this option if you want to repeat the restore operation. SQL Safe performs the restore with the established restore settings.
- **Restore with Different Options** - use this option if you want to review or change the restore settings. SQL Safe opens the Restore Wizard where you can review the settings used to perform the operation or edit according to your preferences.

Take into account that these options are not available for **Guest** role users.

5.7.3 What other options are available on the Operations History tab?

The **Operations History** tab allows you to perform the following actions located on the upper section of this view:

- **Add instance** - use this option to register new instances. Go to adding SQL Server instances to find more information about registering SQL Server instances in your environment.
- **Export** - use this option to export the information displayed on your **Operations History** tab. Select your preferred format for exporting your information: PDF, XLS, or XML.

5.8 View your Managed Instances

The **Instances** tab allows you to view all instances managed in your environment. For each instance, you can see the following information:

- **Status** - displays the status of the respective instance. The error icon means the connection to the instance failed. The up icon represents a successful connection with the instance.
- **Instance name** - the name of the respective SQL Server instance.
- **Status text**- specifies in detail the status of the SQL Server instance.
- **# of Databases** - displays the number of databases that belong to the respective instance. Click this option and SQL Safe takes you to the **Databases** tab filtered by the selected instance.
- **# of Policies** - shows the number of policies that cover at least one database that belongs to the respective instance. Click this option and SQL Safe takes you to the **Policies** view filtered by the selected instance.
- **# of Operations**- displays the number of operations in the instance. Click this option and SQL Safe takes you to the **Operation History** tab filtered by the respective instance.
- **SQL Server version**- displays the SQL Server version of the instance
- **Actions**- under this column you can find the gear icon with the following options for your instances: Remove/delete, Change Credentials, Perform Operation (AdHoc Backup, AdHoc Restore), Create Policy (Backup, Restore, Log Shipping), and Install SQL Safe Backup Agent (if relevant).

Keep in mind:

- You can sort the information available on this tab by clicking the column header by which you want to order your instances.
- You can set the number of items you want to view per page by going to the lower section of the list grid, type a number in the box, and the page will be updated according to your requirements.

5.8.1 How do you filter the information on your Instances tab?

SQL Safe allows you filter your information to access your required data easily. Go to the left section of the **Instance s** tab and filter according to:

- **Status**- you can select **Ok** or **Error**.
- **Instance name**- type the name of the instances you want to view.
- **Status text**- type an specific status.
- **# of Databases**- use the **From** and **To** options to specify a range of the number of databases for which you want to see your data.
- **# of Policies**- use the **From** and **To** options to specify a range of the number of policies for which you want to see your data.
- **# of Operations**- use the **From** and **To** options to specify a range of the number of operations for which you want to see your data.
- **SQL Server version** - type a SQL Server version for which you want to see its instances.
- **Policy Name** - type the name of the policies for which you want to view their respective instances.

When using filters take into account:

- You can save your filtering options by selecting your filters, typing a name in the **By Custom Filter** field, and clicking **Add Filter**. To retrieve your saved filters, click the drop-down option in the **By Custom Filter** section and select your filter name.
- If you want to select filters first and apply the changes later, deselect the **Apply filter as it changes** option.
- To remove filters, use the specific **Remove Filter** option in each filter. For example, if you want to remove your Status filters, click **Remove Status Filter** under the same filter section.
- Under **APPLIED FILTERS** on the top section or your **Managed Instances** tab, you can see the filters you have selected. Click the **X** icon next to the ones you want to remove.
- Use the option **Clear** on the top section of the **Filtering** section to remove all filters.

5.8.2 What other options are available on the Instances view?

On the upper section of your Instances list, you can find the following options:

- **Add instance**- You can register SQL Server instances to your monitored environment. Go to adding SQL Server instance for more information about registering new instances.
- **Create Policy**- use this option to create a backup, restore, or log shipping policy.
- **Backup** - use this option to open the Backup Wizard.
- **Restore**- use this option to open the Restore Wizard.
- **Export** - you can export the information displayed on your **Instances** view. Select your preferred format: PDF, XLS, or XML.

Take into account that these options are not available for **Guest** users.

5.8.3 What actions can you perform on instances?

You can select one or more instances from the **Instances** tab, click one of the respective  gear icon under the **Actions** column, and perform any of the following actions:

- **Remove/delete**- use this option if you no longer want to manage the selected instances.
- **Properties**- use this option to edit general and advanced properties of a registered Instance.
  The General tab, allows you set different credentials to connect to the SQL Server Instance. Go to add new SQL Server instances to find more information about the credentials needed to monitor an instance.
  The Advanced tab, allows you specify a Network Name to connect to the SQL Server Instance and/or the SQL Safe Agent Components.
- **Perform operations**- select this option to access any of the following operations:
  - **AdHoc Backup** - select this option to open the Backup Wizard with the selected instances and specify the respective settings for the backup.
  - **AdHoc Restore** - take into account that this option is only available when you select a single instance. SQL Safe opens the Restore wizard so that you can specify the respective settings for the restore operation.
- **Create Policy** - use this option to create a backup, restore, or log shipping policy.
- **Install SQL Safe Backup Agent** - This option lets you install the SQL Safe Backup Agent in the server hosting the selected instance.

Take into account that these options are not available for **Guest** users.

5.9 Databases view

SQL Safe allows you to view a list of all the databases that belong to your SQL Server instances. In this tab you can view the following information for each database:

- **Status** - it displays the status of the database, it can be OK, Error, or Warning.
- **Instance Name** - the name of the instance where the database resides.
- **Database Name** - the name of the respective database.
- **Database Type** - if it is a User or System database.
- **Last Full Backup** - the date and time when the last full database backup was executed.
- **Last Diff Backup** - the date and time when the last differential database backup was executed.
- **Last Log Backup** - the date and time when the last log database backup was executed.
- **# Of Policies** - the number of policies that cover the database.
- **Space used, MB** - the amount of space used by the database in MB.
- **Actions** - under this column you can find a  gear icon with the following options: **AdHoc Backup,AdHoc Restore,** and **Create Policy**.

Keep in mind:

- You can sort the information available on this tab by clicking the column header by which you want to order your instances.
- You can set the number of items you want to view per page by going to the lower section of the list grid, type a number in the box, and the page will be updated according to your requirements.

5.9.1 What information can you filter in the Databases view?

SQL Safe allows you to filter the information available on the **Databases** tab to quickly access your required data. To filter your information, go to the left section of the tab and filter your information according to:

- **Status** - select a database status. You can choose from Ok, Warning, or Error.
- **Instance Name** - type the names of the instances for which you want to see their databases.
- **Database Name** - type the names of the databases you want to view.
- **Database Type** - select if you want to view **User** databases or **System** databases.
- **Last Full Backup** - use the **From** and **To** options to set specific date range of last Full backups for which you want to view your databases.
- **Last Diff Backup** - use the **From** and **To** options to set specific date range of last Diff backups for which you want to view your databases.
- **Last Log Backup** - use the **From** and **To** options to set specific date range of last Log backups for which you want to view your databases.
- **# Of Policies** - use the **From** and **To** options to set a range of policies for which you want to view your databases.
- **Space used**, **MB** - use the **From** and **To** options to view the databases that belong to a specific space range in MB.
- **Policy Name** - type the name of the policies for which you want to view their databases.

When using filters take into account:

- You can save your filtering options by selecting your filters, typing a name in the **By Custom Filter** field, and clicking **Add Filter**. To retrieve your saved filters, click the drop-down option in the **By Custom Filter** section and select your filter name.
- If you want to select filters first and apply the changes later, deselect the **Apply filter as it changes** option.
- To remove filters, use the specific **Remove Filter** option in each filter. For example, if you want to remove your Status filters, click **Remove Status Filter** under the same filter section.
- Under **APPLIED FILTERS** on the top section or your **Databases** tab, you can see the filters you have selected. Click the **X** icon next to the ones you want to remove.
- Use the option **Clear** on the top section of the **Filtering** section to remove all filters.

5.9.2 What actions can you perform on Databases?

On the **Databases** tab, you can select one or several databases, and click one of the following options from the gear icon under the **Actions** column. You can also use the options from the upper section of this tab.

- **Adhoc Backup** - select this option to open the Backup Wizard.
- **Adhoc Restore** - select this option to open the Restore Wizard.
- **Create Policy** - use this option to create a Backup Policy, Restore Policy, or Log Shipping Policy. SQL Safe adds the selected databases to the respective policy.

These options are not available to **Guest** users.

5.9.3 What other options are available on the Databases tab?

You can also find the following options in the upper section of the Databases list:

- **Add instance** - use this option to register new instances and launch the Add Instance Wizard.
- **Create policy** - select this option to access the Backup, Restore, or Log Shipping policy wizards.
- **Backup** - select this option to open the backup wizard.
- **Restore** - select this option to open the restore wizard. You can select to restore databases or perform an object level recovery.
- **Export** - use this option to export the information displayed on the **Databases** tab, select your preferred format: PDF, XLS, or XML.

The **Add instance, Create policy**, **Backup**, and **Restore** options are not available to **Guest** users.

5.10 Managing SQL Safe Agents

The **SQL Safe Agents** tab allows you view all servers that host instances in your environment.

For each server you can see the following information:

- **Status** - it displays the status of the SQL Safe Backup Agent which could be:
  - **Ok** (green) - the SQL Safe Backup Agent is available and configured correctly.
  - **Warning** (yellow) - SQL Safe connected to the SQL Safe Backup Agent but the Backup Agent is not the most current version or is not configured for the same management service.
  - **Error** (orange) - SQL Safe cannot connect to the Backup Agent on the server.
- **Computer name** - the name of the computer where the SQL Safe Backup Agent is running.
- **Version** - the version of the SQL Safe Backup Agent.
- **Management Server** - the name of the management server that the SQL Safe Backup Agent is configured for.
- **Max Load** - the maximum number of concurrent operations that can be executed by the SQL Safe Backup Agent.
- **Priority** - displays the Windows thread priority at which the Backup Agent threads run.
- **Send Status** - displays the frequency with which the agent is configured to communicate with the Management Server.
- **SQL Timeout** - displays the SQL DMO timeout value, which determines how long the Backup Agent will wait for a response from SQL Server before timing out.
- **VDI Trans Limit**, **bytes** - displays the maximum size of a transfer block for the VDI operation.
- **VDI Buffers**, **bytes** - displays the number of buffers used for the VDI operation.
- **VDI Block Size**, **bytes** - displays the size of a VDI device block. All data transfers are integer multiples of this value.
- **VDI Timeout**, **seconds** - displays the timeout for configuring the VDI.
- **Actions** - allows to access the options for editing server properties, installing/upgrading SQL Safe Backup Agent, and Enabling/Disabling Instant Restore.

Keep in mind:

- You can sort the information available on this tab by clicking the column header by which you want to order your instances.
- You can set the number of items you want to view per page by going to the lower section of the list grid, type a number in the box, and the page will be updated according to your requirements.

5.10.1 How can you filter your information?

You can filter the information of the SQL Safe Agents tab to access your data quickly.

Filter your information by:

- **Status** - it displays the status of the SQL Safe Backup Agent which could be:
  -  **Ok** - the SQL Safe Backup Agent is available and configured correctly.
  -  **Warning** - SQL Safe connected to the SQL Safe Backup Agent but the Backup Agent is not the most current version or is not configured for the same management service.
  -  **Error** - SQL Safe cannot connect to the Backup Agent on the server.
- **Computer Name** - type the computer name.

- **Version** - type the Backup Agent version.
- **Management Server** - type the Management Server configured for the Backup Agent.
- **Max Load** - set a range using the **From** and **To** options.
- **Priority** - type the priority.
- **Send Status** - select from the **On** and **Off** options.
- **VDI Trans Limit, bytes** - set a range using the **From** and **To** options.
- **VDI Buffers, bytes** - set a range using the **From** and **To** options.
- **VDI Blocksize, bytes** - set a range using the **From** and **To** options.
- **VDI Timeout, seconds** - set a range using the **From** and **To** options.

When using filters take into account:

- You can save your filtering options by selecting your filters, typing a name in the **By Custom Filter** field, and clicking **Add Filter**. To retrieve your saved filters, click the drop-down option in the **By Custom Filter** section and select your filter name.
- If you want to select filters first and apply the changes later, deselect the **Apply filter as it changes** option.
- To remove filters, use the specific **Remove Filter** option in each filter. For example, if you want to remove your Status filters, click **Remove Status Filter** under the same filter section.
- Under **APPLIED FILTERS** on the top section or your **SQL Safe Agents** tab, you can see the filters you have selected. Click the **X** icon next to the ones you want to remove.
- Use the option **Clear** on the top section of the **Filtering** section to remove all filters.

5.10.2 What other options are available on the SQL Safe Agents tab?

You can also find the following options in the upper section of the SQL Safe Agents tab:

- **Export** - use this option to export the information displayed on the SQL Safe Agents tab. You can select your preferred format: PDF, XLS, or XLM.
- **Refresh**  - use this option to get the latest status of your SQL Safe Agents.

5.10.3 What options can you edit in the properties window?

Click the  gear icon at the end of the row and select properties. On the Agent Properties window, you can edit the following settings:

- On the **General** tab, you can edit the following settings:
  - **Status**
    - **Management Server** - the name of the server hosting the SQL Safe Management Service that the Agent is configured to communicate with.
    - **Send Status every X seconds** - select this option to have the SQL Safe Backup Agent communicate with the SQL Safe Management Service. Also, define the frequency of the communicate.
  - **Performance**
    - **Max Load** - the maximum number of concurrent operations that the backup agent can perform.
    - **Priority** - use this option to define the thread priority at which backup agent threads run.
  - **Troubleshooting**
    - **Enable Debug Mode** - select this option to enable debug logging on the Agent.

        - The **Advanced** button provides access to define the following options: More detailed messages, Backup Service Engine, Backup Service I/O, Map Generation, Filter Service Engine, and Filter Service Driver. Additionally, you can set **Log Files** options and define if you want to roll logs.

- On the **SQL Server** tab, you can edit by:
    - **SQL Server Connectivity**
        - **Timeout** - set the timeout in seconds.
    - **Virtual Device Interface (VDI) Defaults**
        - **Timeout** - set the timeout in seconds, which determines how long the Backup Agent will wait for a response from SQL Server before timing out.
        - **Buffers** - define the number of buffers used for the VDI operation.
        - **Transfer Limit** - set the maximum size of a transfer block for the VDI operation.
        - **Block Size** - set the size of a VDI device block. All data transfers are integer multiples of this value.
- On the **SQL Virtual Database** tab, you can edit by:
    - **Default Database File Location** - use this option to specify which folder SQL VDB uses to stores data files when creating a virtual database. Click **Browse** to change the Database File location.
    - **Cleanup Unused Files** - use this option to remove the temporary files used for a VDB that are left behind after the database is deleted. Click **Clean Up** to remove these files and free up space.

When you finish configuring your Agent Properties, click **Save** or **Cancel** the configuration.

5.11 Working with Virtual Database

Virtual Database (VDB) is a powerful solution that lets you attach SQL Server backup files and query them like real databases. Virtual database allows you to gain instant access to critical data in a backup file without spending the time and storage previously required for restore. In minutes, you can create a virtual database and then use any native SQL Server or third-party tools to query and extract the data you need.

Any operation that you can perform on a physical database can be performed on a virtual database. Likewise, applications that rely on getting information from this database can continue using the virtual version. You can also access this virtual database using Microsoft SQL Server tools, such as Management Studio, and other third-party applications.

You can modify the data and objects in the virtual database. However, because the virtual database is based on archived data, your changes will not persist when you detach the virtual database. To preserve your changes, back up the modified virtual database and then create a new virtual database using those backup files.

Virtual Database offers the following features:

5.11.1 Virtual recovery

Provides instant, feature-rich access to all data from within SQL Safe backup and native SQL Server backup files.

5.11.2 Point-in-time selection

Provides point-in-time selection and recovery, allowing granular control over the state of the data displayed in the virtual database.

5.11.3 Native SQL Server and third-party application access

Use existing SQL Server tools such as SQL Server Management Studio and third-party applications to interact with the new virtual database as though it were an actual physical database.

5.11.4 Intuitive Console

Allows virtual databases to be quickly and easily created, edited, or removed.

5.11.5 No impact to production servers

Installs to a single non-critical server and attaches all virtual databases to a single SQL Server instance.

5.11.6 Viewing your Virtual Databases

On the **Virtual Database** tab, you can see all the existing Virtual Databases in your environment. Create virtual databases to access, compare, recover, and report on data from previous backups of databases.

For each Virtual Database, you can see the following information:

- **Status** - displays the status (Online, Restoring, Recovering, Recovery Pending, Suspect, Emergency, Offline, Server Offline, Creating Content Map, Creating, Created Failed, License Expired, Detached, Unknown) of the Virtual Database.
- **Instance** - displays the name of the SQL Server instance under which the Virtual Database is mounted.
- **Database** - displays the name of the Virtual Database.
- **Point in Time** - displays the date and time when the Virtual Database was restored.
- **Backup Files** - displays the backup file where the Virtual Database is mounted.
- **Description** - details the description specified when mounting the Virtual Database.
- **Actions** - click the  gear icon to:
  - **Edit** - edit the Virtual Database options.
  - **Edit/Add description** - enter and/or edit the description of the Virtual Database.
  - **Remove** - remove the Virtual Database.

Keep in mind:

- You can sort the information available on this tab by clicking the column header by which you want to order your instances.
- You can set the number of items you want to view per page by going to the lower section of the list grid, type a number in the box, and the page will be updated according to your requirements.

How can you filter your information?

In order to get more specific Virtual Database information from your environment, you can use the Filtering section on the left side of the **Virtual Database** tab. You can filter your information by:

- **Status** - select a status (Online, Restoring, Recovering, Recovery Pending, Suspect, Emergency, Offline, Server Offline, Creating Content Map, Creating, Created Failed, License Expired, Detached, Unknown) to filter your Virtual Database.
- **Instance Name** - type the name of the SQL Server instance under which the Virtual Database is mounted. You can use the WildCards (&, ^, [], %, and _).
- **Databases Name** - type the name of the Virtual Database. You can use the WildCards (&, ^, [], %, and _).

When using filters take into account:

- You can save your filtering options by selecting your filters, typing a name in the **By Custom Filter** field, and clicking **Add Filter**. To retrieve your saved filters, click the drop-down option in the **By Custom Filter** section and select your filter name.
- If you want to select filters first and apply the changes later, deselect the **Apply filter as it changes** option.
- To remove filters, use the specific **Remove Filter** option in each filter. For example, if you want to remove your Status filters, click **Remove Status Filter** under the same filter section.
- Under **APPLIED FILTERS** on the top section or your **Virtual Databases** tab, you can see the filters you have selected. Click the **X** icon next to the ones you want to remove.
- Use the option **Clear** on the top section of the **Filtering** section to remove all filters.

What other options are available on the Virtual Database tab?

On the top section of the **Virtual Database** tab, you can find the following options:

- **Atthach Full backup** - use this option to create a Virtual Database from a single full backup file. Visit the Attach Full Backup section to find more information.
- **Attach Multiple Backups** - use this option to create a Virtual Database based on data from specific backup files. Visit the Attach Multiple Backups section to find more information.
- **Remove** - use this option to remove a Virtual Database. For more options visit the Remove a Virtual Database section.
- **Refresh**  - use this option to refresh your Managed Virtual Databases section.

Attach Full Backup

You can use the **Attach Full Backup** option to easily create a virtual database from a single full backup file. Follow these steps:

1. Click the **Attach Full Backup** option located on the top section of the Virtual Databases tab.
2. In the Attach Full Backup dialog box, click **Select** to choose one of your listed and registered instances.
3. Type the filename of the backup or click **Browse** to search the filename of the backup you want to use.
4. Type a unique name for your new VDB in the Virtual Database Name text box. This name will be displayed in the SQL Safe Console and in other database management tools.
5. Click **Create**.

> Click the **Help** button located in the upper right corner of the dialog box to get more information.

The progress bar window opens indicating the status of your VDB creation. The results are revealed in the VDB List grid of the **Virtual Database** tab.

Attach Multiple Backups

To create a VDB based on data from specific backup files, click the **Attach Multiple Backups** option located on the top section of the **Virtual Database** tab. The Attach Multiple Backups Wizard opens and different options are displayed. Follow these steps:

1. **Backup Files:**
   - Click **Select** to choose a Host Instance from the displayed list.
   - Click **Browse**, select the backup files (.bak & .safe) from the file explorer dialog box or enter manually the files you prefer in the text box and click **Add** to display them in the Selected Backups section below.
   - The **Encryption Settings** option is available when the chosen backup sets were encrypted during their backup.
   - Click **Remove** to delete all selected backup files from the display. A dialog box opens to confirm the backup removal.
   - Click **Next** to continue.
2. **Point in Time:**
   In this step, the user sets a specific restore point in time. The following options can be performed:
   - Use the point in time slider to select the backup sets you need. This ensures you are not restoring data time-stamped with dates later than the point in time you specified.
   - You can enter manually the specific point in time to restore. There are two ways to pick date and time to restore file(s): By Date or by LSN.
   - Click Next to continue.

   > When choosing Date use MM/DD/YYYY hh:mm:ss format. And for LSN use '#:#;#' format.

   > LSN (Log Sequence Numbers) is used internally during a RESTORE sequence to track the point in time to which data has been restored. When a backup is restored, the data is restored to the LSN corresponding to the point in time at which the backup was taken.
3. **Create As:**
   - The Host Instance textbox displays the instance name that hosts the VDB, by default.
   - In the Virtual Database Name textbox, type a unique name for your VDB. Click **Database Files...** to customize the location where to store the files for your new VDB and click **OK**.
   - Enter a Description.
   - Click **Next** to continue.
4. **Summary:**
   - Confirm the display information in the Summary step.
   - Click **Prev** to go back to previous steps and edit the information, select **Cancel** to cancel the process and close the wizard, or click **Finish** to start the process.
   - A progress bar displays showing the status of the process. When the process completes, the status changes to Complete.

The results are revealed in the VDB List grid of the **Virtual Database** tab.

5.11.7 Remove a Virtual Database

There are different options available to **remove a virtual database** which are described below:

Operation History

1. In the SQL Safe Web Console, go to the Operation History tab.
2. Locate the virtual database you want to remove.
3. Click the  located at the end of the row.
4. Select **Remove VDB** from the displayed options.
5. The message window *Remove Virtual Database* displays requesting confirmation of the removal of the virtual database.
6. Click **Yes** to remove; otherwise, click **No**.

Once you complete the removal, your Virtual Database will be displayed as deleted on the Operation History List grid of the **Operation History** tab and it will disappear from the VDB List grid of the **Virtual Database** tab.

Virtual Database

1. In the SQL Safe Web Console, go to the Virtual Database tab.
2. Locate and select the virtual database(s) you want to remove.
3. Click the  located at the end of the row.
4. Select **Remove** from the displayed options.
5. The message window *Remove Virtual Database* displays requesting confirmation of the removal of the virtual database.
6. Click **Remove** to remove; otherwise, click **Cancel**.

Once you complete the removal, your Virtual Database will be displayed as deleted on the Operation History List grid of the **Operation History** tab and it will disappear from the VDB List grid of the **Virtual Database** tab.

For detailed descriptions and available virtual database options, see the CLI Help (`SQLvdbCmd help <action>`).

5.11.8 CLI Commands

Virtual Database includes a command line interface (CLI). When you use the *sqlvdbcmd* command, the following actions can be performed:

Actions	Description
Create	Create a new virtual database
Remove	Delete a virtual database
Cleanup	Cleanup unused virtual database temporary files
EncryptWindowsPassword	Encrypt plain-text password for Windows logins
EncryptSqlPassword	Encrypt plain-text password for SQL Server logins
EncryptRestorePassword	Encrypt plain-text password for encrypted restores
Map	Generate maps <for InstantRestore or SQL virtual database>
Help	Display more detailed help

For detailed descriptions and available virtual database options, see the CLI Help (`SQLvdbCmd help <action>`).

SQL VDB CLI actions and options are not case-sensitive.

5.12 Options in the Administration tab

In the **Administration** tab, you can find the following options:

- Manage Users
- Configure Notification Settings
- General Preferences
- Manage Licenses

5.12.1 Manage Users

In this section you can **manage your users**, groups, add new ones, remove them, give them role permissions, and other options.

SQL Safe allows you to have three roles of users with the following permissions:

- **Administrator** - can view all tabs and dialogs in the SQL Safe Web console and perform all available operations
- **User** - can access all SQL Safe views except the **Administration** tab. They can perform all operations on the tabs they can access.
- **Guest** - can view all SQL Safe tabs except the **Administration** tab. They cannot perform operations and can only access read-only dialog windows.

The Manage Users section of the **Administration** tab is only accessible to Administrators. On this view you can perform the following actions:

- **Add new users or groups** - click the option **Add User/Group** on the top section of the **Manage Users** dialog window.
- **Edit existing users or groups settings** - select the pencil icon  next to the respective user to edit its settings.
- **Remove users or groups** - click the **X** icon next to the selected user to delete it.

What settings can you define for each user or group?

When you add a new user or group, you can define the following account details:

- **Account Name** - specify the name of the new user or group you want to add. Use the following format: domain\account name.
- **Account Type** - define the new account as a User or Group.
- **Account Enabled** - select this option to enable the selected account.
- **Session Timeout** - specify the amount of time in minutes the browser will wait before the session times out.
- **Product** - use the drop down option to select which IDERA product the user will be able to access.
- **Role** - define which role (Administrator, User, or Guest) the new user or group should have.
- **Email Address** - type the email address where you want the new user or group to receive the welcome email and other notifications. Take into account that you must configure your SMTP Server settings first in order to send emails.

5.12.2 SMTP settings for notifications

You can enable SQL Safe to send email notifications about the current status of your backup and restore operations.

Access these settings by clicking **Configure Notifications Settings** on the **Administration** tab. This option is only available to Administrators.

What email settings can you change?

If you enable **E-mail Notifications**, you can configure how the email will appear in your Inbox.

Sender Name

Enter the name that will appear as the sender of the email.

Reply to Address

Enter the email address that will appear as the sender and where replies to the message will be sent.

Priority

Select low, normal, or high priority for the email alerts.

What mail server information is required?

You must specify the mail server information so that SQL Safe can send email notifications.

Use Gmail as an SMTP Server

Select this option if you want to use Gmail as your preferred SMTP email server. SQL Safe automatically populates the respective SMTP Server Address (smtp.gmail.com) and Port (587). When you select this option, you need to provide the respective username and password.

> Take into account that if you choose to use Gmail as your SMTP email server, you could require to turn on **need to activate less security apps access**. In other words, click the respective link and select **Turn on** access. For more information, go to allowing less secure apps to access your account.

SMTP Server Address

If you do not want to use Gmail as your email server, you can also specify the SMTP address of your respective email server.

SMTP Server Port

Specify the port for your email server. You can also enable SSL encrypted connection.

SMTP Authentication

If your SMTP server requires authentication, you must type a valid **User Name** and **Password** that SQL Safe can use to access the email server.

Test your settings

To be sure that your settings are correct, click **Send Test Email** on the bottom section of the window, then check the test email sent to your email server.

5.12.3 General Preferences

This section allows you to configure your general preferences for your SQL Safe Web Console, Management Service and Repository Settings, Backup Preferences, Agent Deployment, Policy Data location, Amazon Settings, and Azure Settings.

To access this section, go to the **Administration** tab, and click**General Preferences.**

This option is only available to Administrators.

On the **Configure General Preferences**, you can find these tabs :

- Basic
- Repository and Management Service
- Backup
- Agent Deployment
- Policy Data
- Amazon Settings
- Azure Settings

Basic Configurations

To configure the basic SQL Safe configurations, go to the **Basic** tab of the **Configure General Preferences** option from the **Administration** tab.

In this section, you can set the following options:

- **Automatically refresh status** - check this option to have the status automatically refreshed on the screen. If you select this option, you can also define the **Number of seconds between each refresh.**
- **Return on Investment (ROI)** - In this option, you can define the TCO/GB (total cost of ownership per gigabyte) that SQL Safe will use to calculate the ROI. By default, this value is set to 200$/GB.
- **Troubleshooting** - you can define the following options:
    - **Enable Debug Logging** - where you can click **Advanced** to get additional logging options.
    - **Do not use ICMP Ping when contacting hosts.**

Repository and Management Service Settings

To configure the Repository and Management Service settings, go to the **Repo and Management service** tab of the **Configure General Preferences** option from the **Administration** tab.

In this section, you can define the following settings:

- **Location of Management Service** - specify the name of the computer that hosts the management service. Click **Change** to define a different computer than the one previously specified.
- **Repository database for Management Service** - use this option to configure the settings of your repository database such as:
    - **SQL Server** - type the SQL server hosting the repository database.
    - **Database** - type the repository database name.
    - **Authentication** - you can select to use a **Windows** or **SQL Server** account to connect to the repository database. If you select **SQL Server** account, you need to provide the respective username and password. Additionally, you can use the option **Test connection** to verify the provided account can successfully connect to the repository database.
- **Repository grooming** - use this option to specify the number of days SQL Safe will keep operational history before grooming it.

Backup Preferences

To configure the backup settings you want to use by default for all your backup jobs, go to the **Backup Preferences** tab of the **Configure General Preferences** option from the **Administration** tab.

In this tab you can set the following preferences:

Where to access backup archives on file system?

Use this section to configure:

- **File System location** - define the path where backup archives are located. By default, C:\Backup\ is used as the File System location. You can type a different path.
- **Account** - specify the account the Backup Agent will use to access the filesystem.
- **Retry writing backup files after network errors** - select this option if you want to retry writing backup files after network errors. Click **Configure** to specify retry settings such as the interval between each retry or how long SQL Safe will wait until it fails the operation.

Where to access backup archives on Tivoli Storage Manager?

In this section you can configure the High level, Low Level, and the path where to find the Configuration File for TSM. To change any of these settings, type directly into the box field.

How to store backup archives?

In this section you can configure or select the following options:

- **If archive exists** – use this option to define what to do if the archive already exists. You can **Append** to an existing archived backup set or choose to **Overwrite** it.
- **Compression** - use this option to specify your compression level (iSpeed, iSize, Level1, Level2, Level3 or Level4). Go to understand compression levels to know what each level of compression entices.
- **Encryption** - select what type of encryption you want to use. Go to understand encryption levels to know what each level of encryption means.
- **Generate metadata for quick access by SQL Virtual Database.**
- **Include database logins in backup file.**

What format to use to generate default file names?

SQL Safe allows you to define what format you want to use for generating backup file names. You have two box fields in this section:

- **Use the following template** - In this box field, click the place where you want to add syntax for your backup file name, then click **Insert**. You can select from Backup Type, Database Name, Instance Name, Stripe Ordinal, Stripe Total, Timestamp, or UTC Timestamp and the format will be inserted in the location selected. Use the option **Reset** to get back to a previous configuration.
- **Preview** - SQL Safe displays the format settings specified in the previous field.

Tune Performance

In this section you can specify the number of threads SQL Safe will use when performing a single archive backup. You can set it to Auto, 1, 2, 28, etc.

Agent Deployment

To configure the service account SQL Safe uses to deploy Backup Agents, go to the **Agent Deployment** tab of the **Configure General Preferences** option from the **Administration** tab.

In this section, you can define whether you want to use the Local System built-in account or another account for the backup agent service account. If you select another account, you have to provide the respective username and password.

Additionally, in this section, you can configure SQL Safe to **Automatically upgrade Backup Agent Extended Stored Procedures**.

For any changes you perform on this tab, click **Save** to apply them.

Policy Data

To configure where you want to save your policy data files of your policies, go to the **Policy Data** tab of the **Configure General Preferences** option from the **Administration** tab.

By default each backup agent stores policy data in its own installation path: <InstallPath>\PolicyData. You can change this path by typing your preferred location or by clicking **Browse** to find it.

After determining your policy data location changes, click **Save** to apply this new location.

Amazon Settings

To specify the Amazon cloud settings to be used through the different backups and restores, go to the **Cloud Settings** tab of the **Configure General Preferences** option from the **Administration** tab.

In this section, you can specify following default settings:

- **Access Key** - specify the access key generated in your security credentials of your Amazon S3 web console. For more information, click here.
- **Secret Key** - specify the secret key generated in your security credentials of your Amazon S3 web console. For more information, click here.
- **Region** - select the region where your information will be stored. You can find more information about these regions here.
- **Bucket Name** - define the name of the Amazon S3 bucket where your backup will be stored.
- **File Size** - determine the minimal parts in bytes of the backup file sent to the bucket simultaneously. The minimal value is 1000000 bytes (1,000,000 bytes).

When Amazon cloud settings are set in the preferences, they can be used in backup and restore operations.

Azure Settings

Blob storage is a type of Azure storage service. Blob Storage stores file data. A blob can be any type of text or binary data, such as a document, media file, or application installer. You can use Blob storage to store content such as backups of files, computers, databases, and devices. Blob storage is also referred to as Object storage.

Use the **Azure Settings** tab of the **Configure General Preferences** option from the **Administration** tab to specify the storage settings to be used through the different backups and restores.

In these settings you can specify the following fields:

- **Container Name** - the name of the Azure container where the new blob will be created and the backup stored. Every Azure blob must reside in a container. The container forms part of the blob name. If no container with the input name exists, a new one will be created. For more information, click here.
- **Azure Storage Account Name** - the account name of your storage account. Every object that you store in Azure Storage has a unique URL address. The storage account name forms the subdomain of that address. You can find more information in the following link.
- **Azure Access Key** - you can use any of the access keys provided to your Azure Storage Account. For more information about Azure Keys, click here.
- **Sector Type** - define the Azure sector type:
  - *Public* - commercial cloud storage solution.
  - *Government* - cloud storage solution offered to US government customers and their partners.

⚠ By default, SQL Safe Backup splits the backup into several files, each with size of 70MB.

You can configure the file size by updating the Windows Registry on the machine hosting the SQL Safe Backup Agent using the following steps:

- Add a new registry REG_DWORD key to HKEY_LOCAL_MACHINE\SOFTWARE\Idera\SQLsafe named AzureFileSize.
- Update the AzureFileSize key to a decimal value of your desired file size. For example, if you want the default file size to be 100MB, the key value should be set to 100.
- Restart all SQL Safe services on the machine.

If there are network connection problems, the backup eventually fails, but files remain on Windows Azure. SQL Safe will not delete these partial backup files. If you want to delete those, you need to do it manually on your Azure account.

✓ When Azure Blob Storage settings are defined in **Configure General Preferences** section, they can be reused later through different backups and restores.

⚠ **Failed Backup**

If the backup fails before creating all blobs and only a couple of blobs are created, these blobs will remain in the container unless you manually delete them.

Network Resiliency

Take into account that if you do not enable the network resiliency settings for your backup operations and the network goes down, the operation fails and no retry is executed. When enabling the network resiliency settings and using Azure Blob for backup operations, only the following parameters are applicable:

- **Retry Interval** - the waiting period before retrying the backup operation.
- **Total retry interval** - the total time for retrying the backup operation before stopping it.

In restore operations, the resiliency settings remain enabled.

Naming conventions for containers

Taken into account the following naming conventions for your container:

- The container name should be a valid DNS name.
- Names have to start with a letter or number and can contain only letters, numbers, and the dash (-) character.
- Every dash (-) character must be immediately preceded and followed by a letter or number. Consecutive dashes are not permitted in container names.
- All letters should be in lowercase.
- Names can be from three to sixty-three characters long.

Naming blobs and blob sizes

The maximum size for a blob is 70MB. If a backup file is, for example, 160 MB, multiple blobs will be created using the following naming format:

- <blobname>.safe_<i> where <i> is the blob counter

If the blob name is testdb for example, the three blobs created in the container will have the following names: testdb.safe_1 (70MB), testdb.safe_2(70MB), testdb.safe_3(20MB).

5.12.4 Manage License Keys

You can manage your License Keys settings by going to the **Administrator** tab and clicking **Manage Licenses**.

> This option is only available to Administrators.

The **License Key Manager** provides an intuitive, simple-to-use interface for SQL Safe license key management. On this window, you can view two tabs: Instances and License Keys.

> You should use licenses valid for the Centralized Licensing model. When Centralized Licensing is enabled, all licensing information is managed using the SQL Safe Management Service and each of the individual SQL Safe Backup Agent licenses are removed.

Instances

In this tab, you can view all instances that are licensed for backup operations. Use this section to select/unselect to move backup licenses between registered instances. Additionally, you can select to **Automatically license instances for scripted agent deployment**.

If you have to save the information displayed on the instances tab, select **Save to File.**

License Keys

In this tab, you can view all your purchased license keys. For each License Key, you can view the following information:

- License Type
- Number of instances
- Expiration Date
- Key

Additionally, you can perform the following actions in this section**:**

- **Remove** - use this option to delete licenses.
- **Add** - type one license per line and click **Add**.
- **Save to file** - use this option to save the **License Key Manager** information to a file.

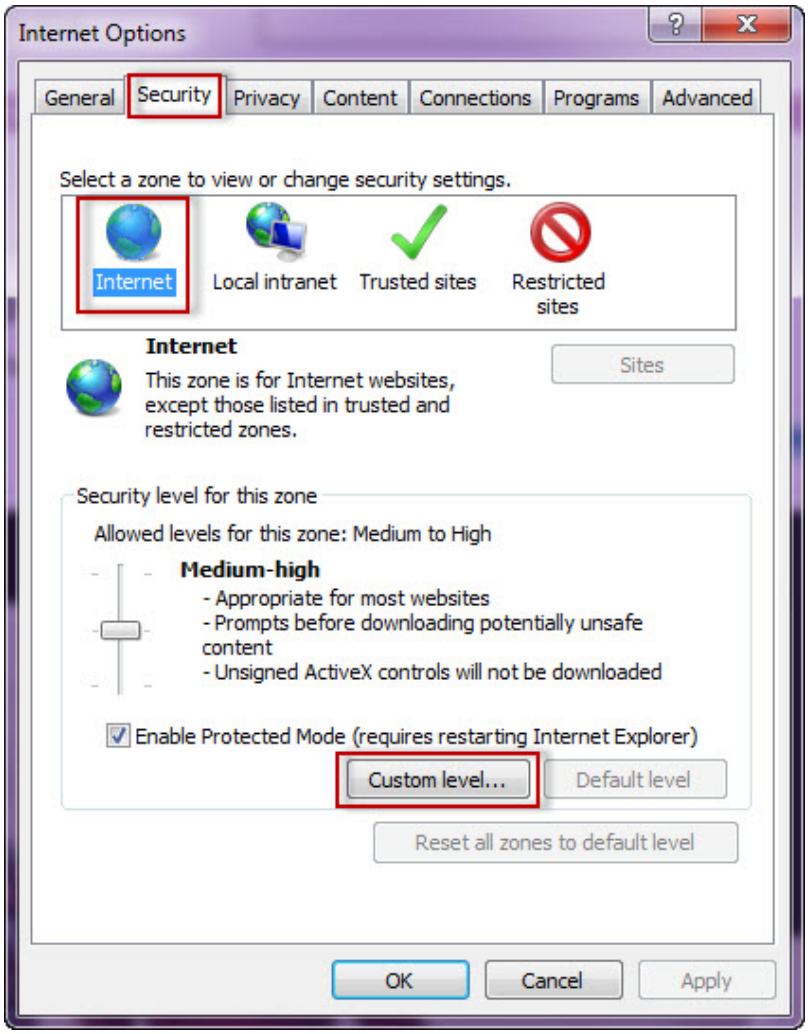
5.13 Configuring your browser for Windows Authentication

In order to be able to use Windows Authentication to log into SQL Safe, you have to configure your browser settings.

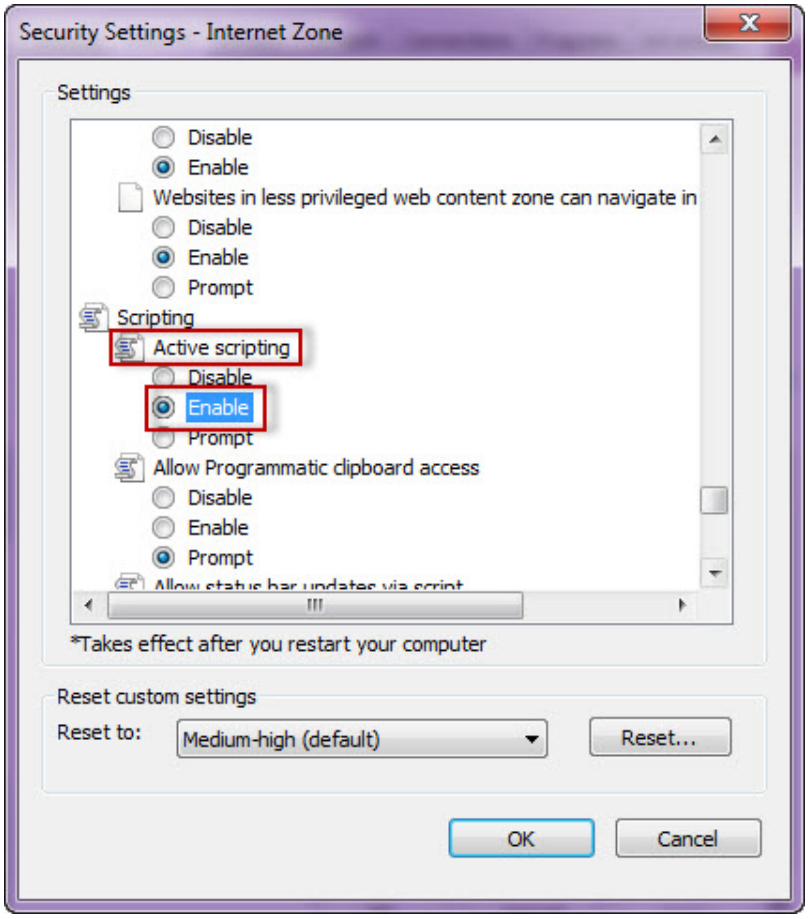
5.13.1 Internet Explorer 9,10,11

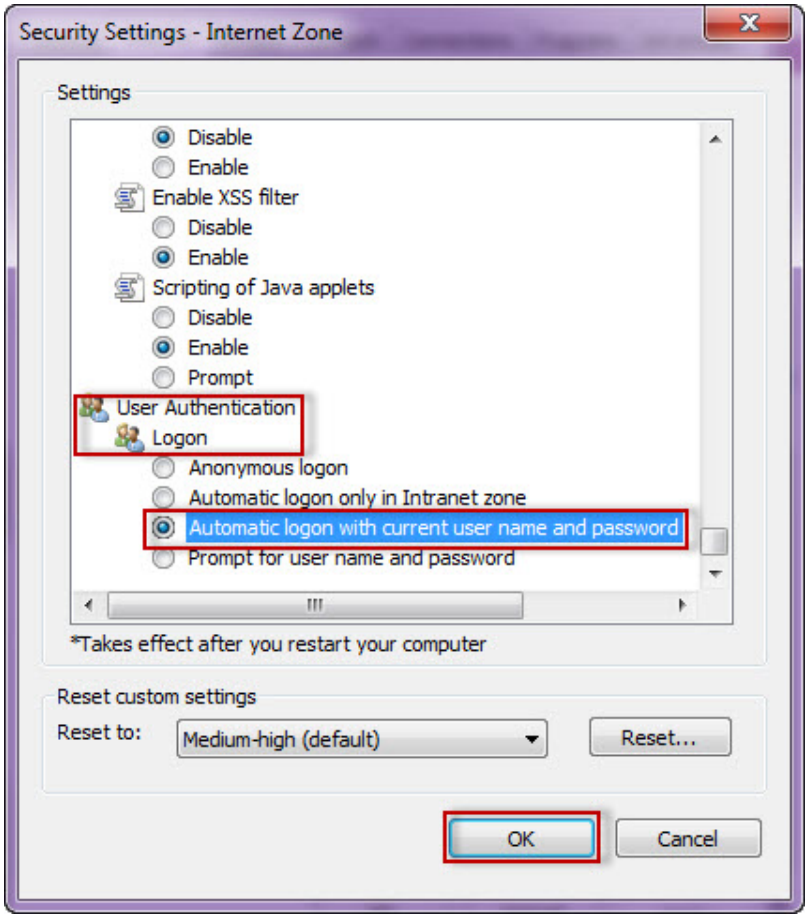
To configure these versions of **Internet Explorer**, follow these steps:

1. Go to **Internet Options**.
2. Click **Security**.
3. Click **Internet** and then **Custom Level**.

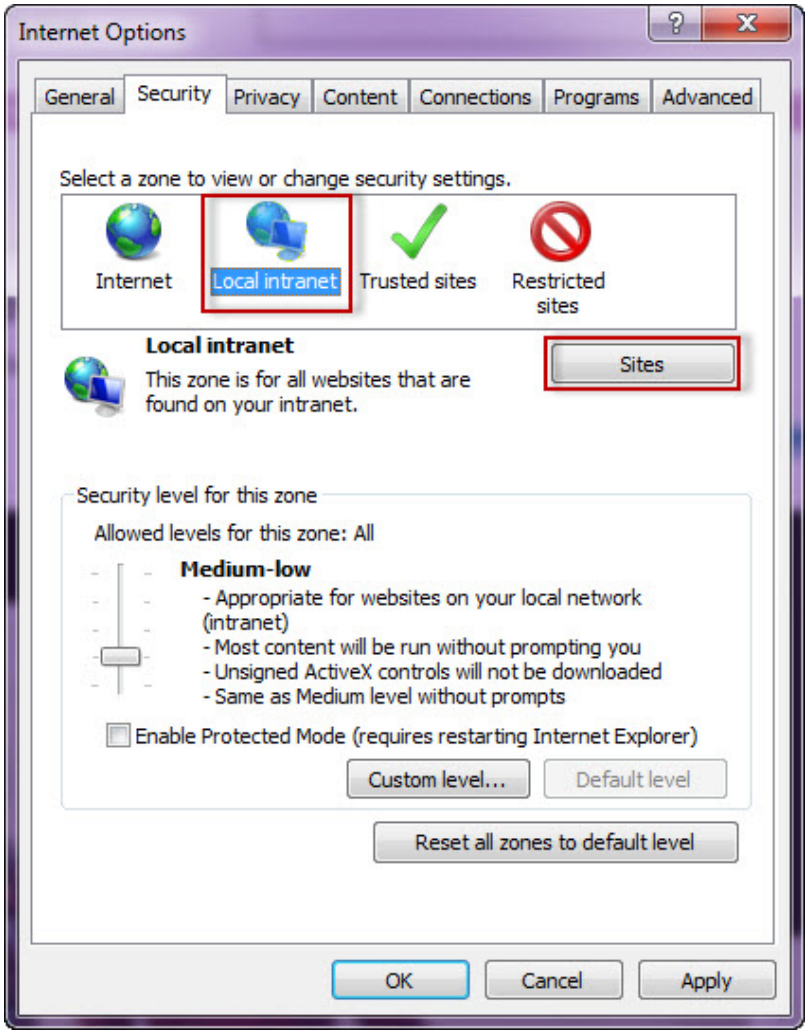


4. Go to Scripting/Active scripting and make sure **Enable** is selected.
5. Then go to User Authentication/Logon and make sure **Automatic logon with current user name and password** is selected.
6. Click **OK**.

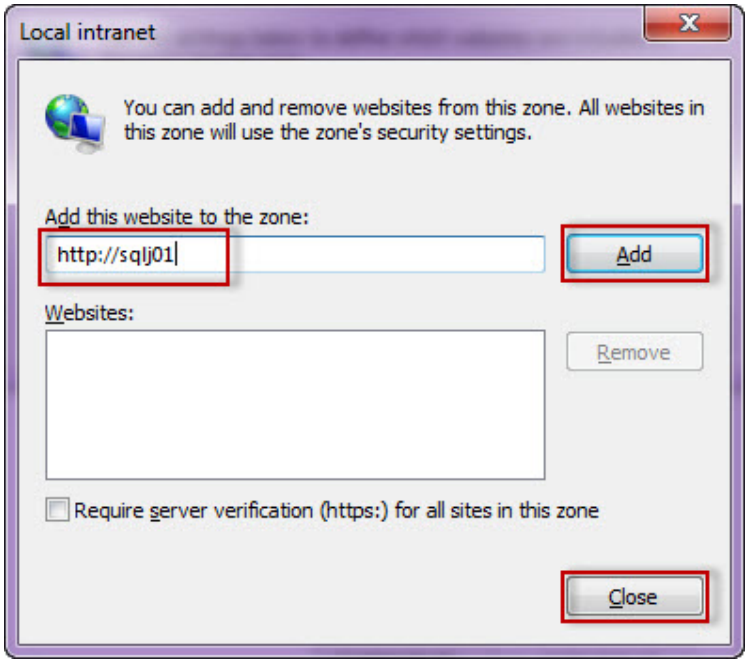

Security Settings - Internet Zone
Settings
Disable
Enable
Websites in less privileged web content zone can navigate in
Disable
Enable
Prompt
Scripting
Active scripting
Disable
Enable
Prompt
Allow Programmatic clipboard access
Disable
Enable
Prompt
*Takes effect after you restart your computer
Reset custom settings
Reset to:
Medium-high (default)
Reset...
OK
Cancel



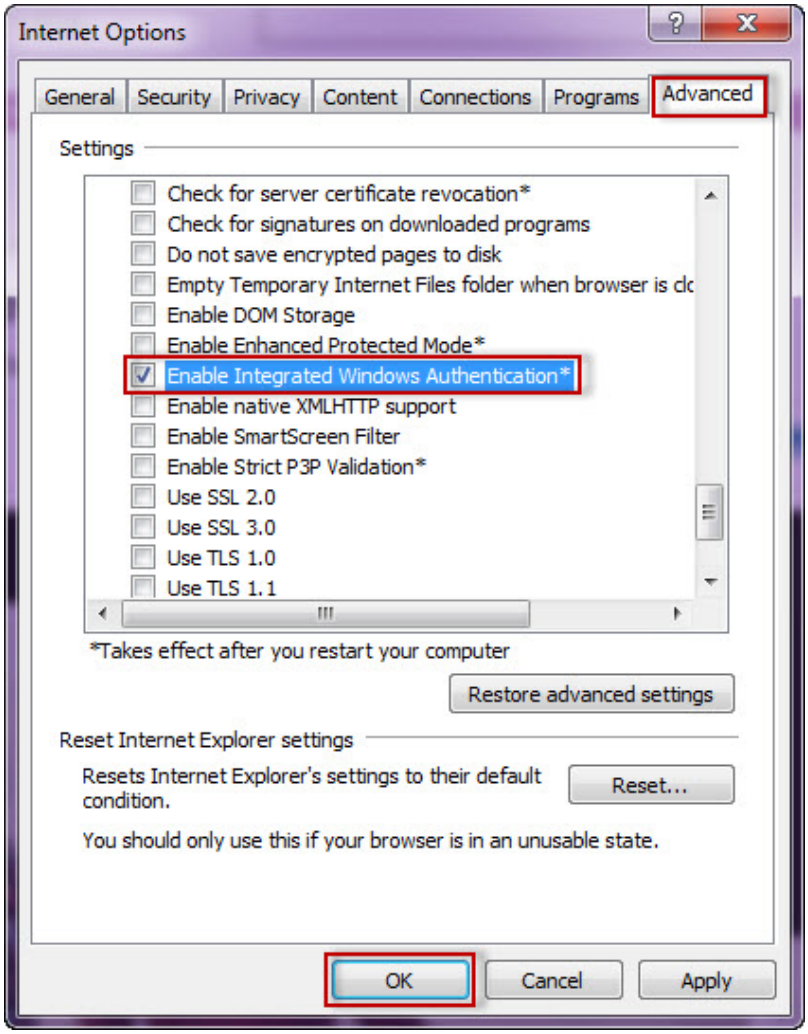
7. Click **Local intranet** and then **Sites**.



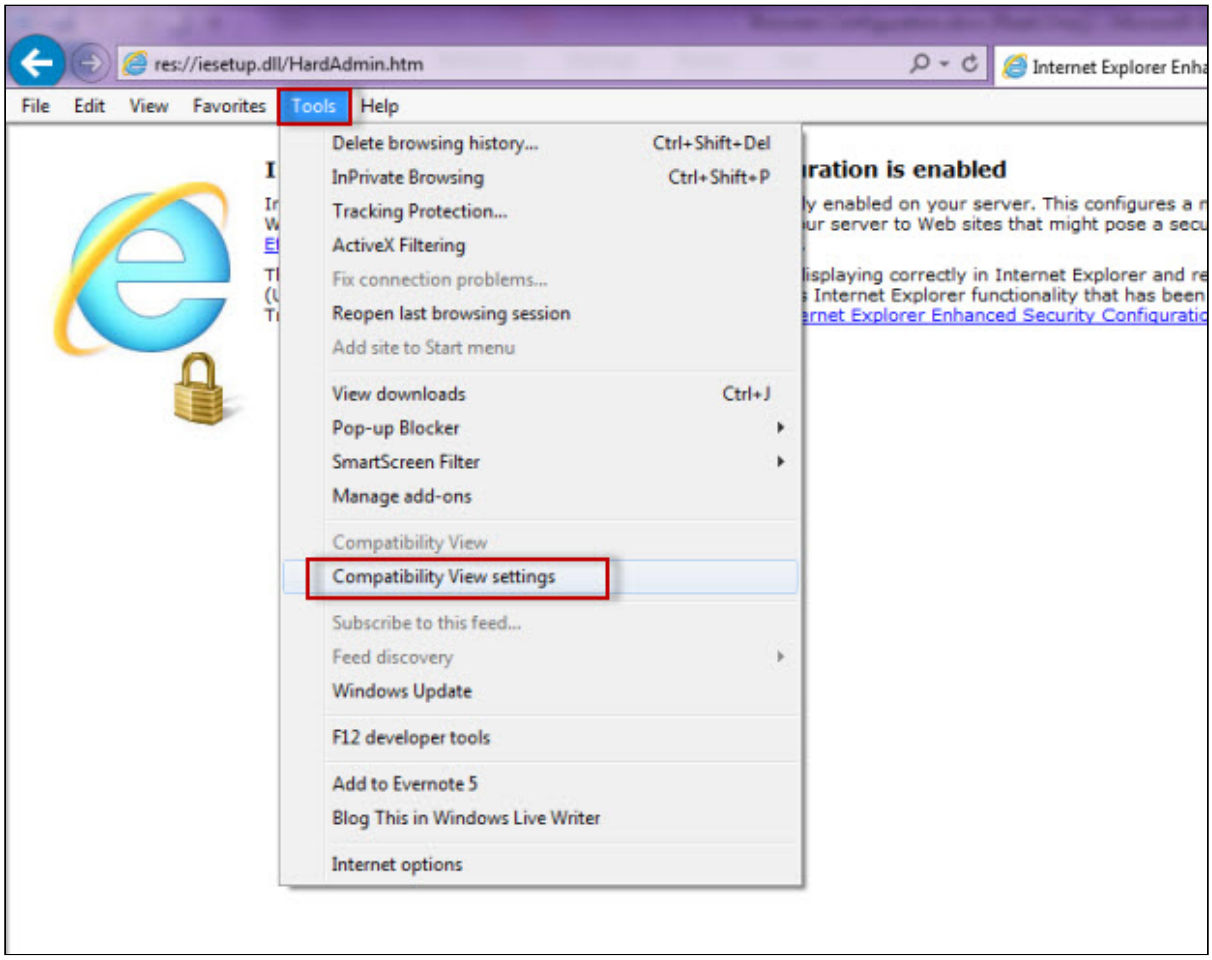
8. Type the following site **http://sqlj01** and click **Add**.
9. Click **Close**.



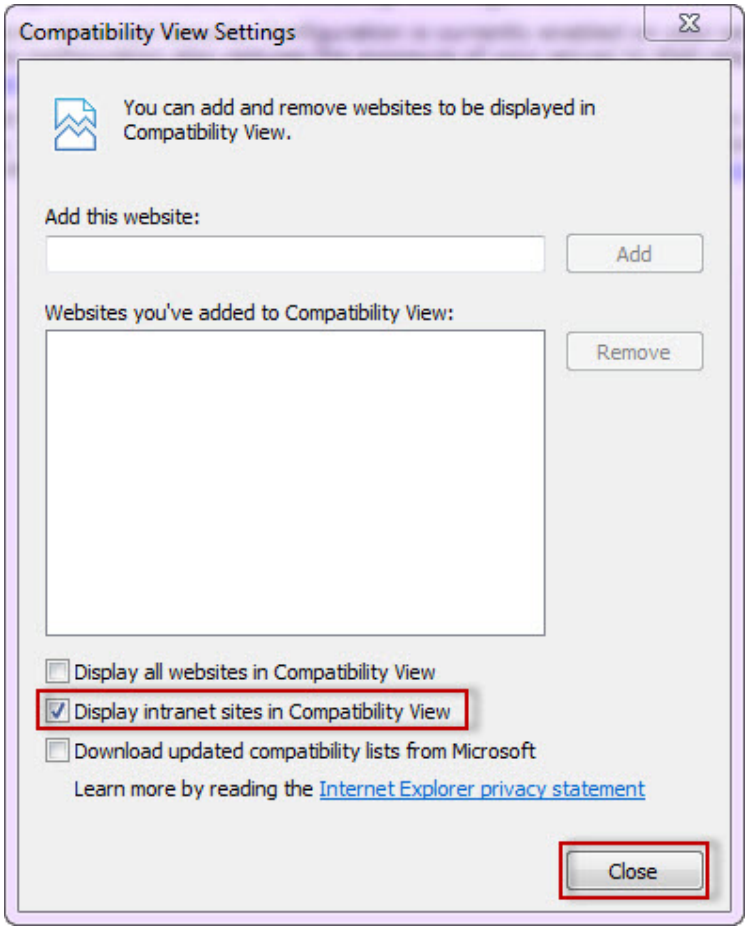
10. Go to **Advanced** and make sure **Enable Integrated Windows Authentication** is selected.
11. Click **OK**.



12. Go to **Tools** in the menu bar, and select **Compatibility View Settings**.



13. In the **Compatibility View Settings** window, make sure **Display intranet sites in Compatibility View** is selected.
14. Click **Close.**

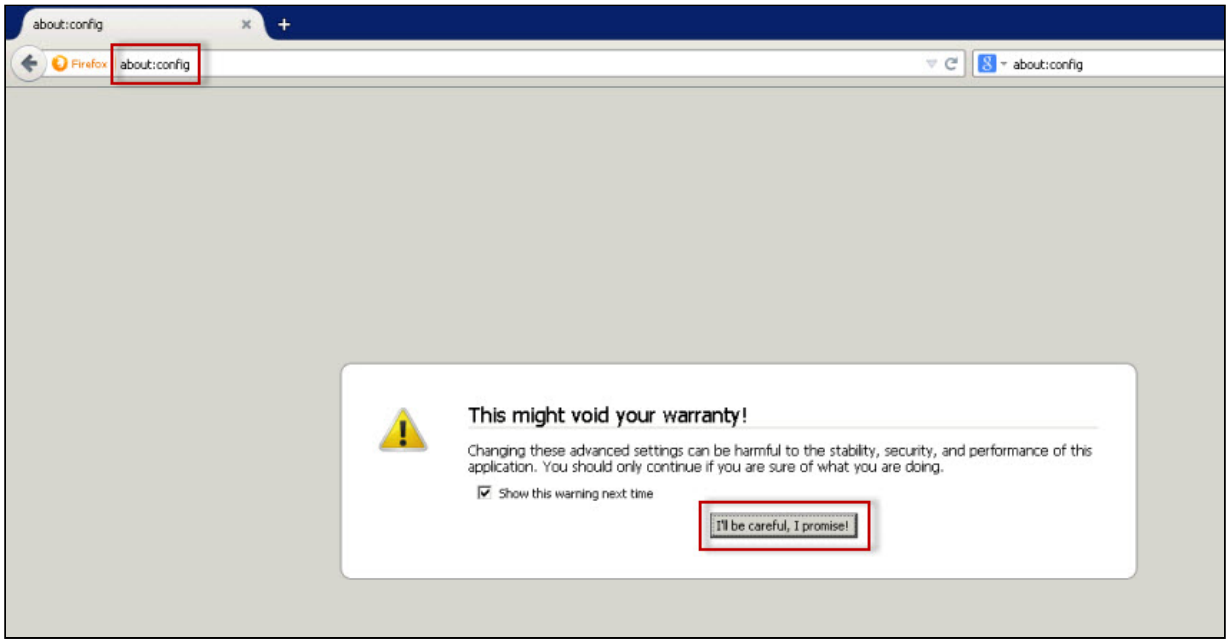


5.13.2 Configuring Google Chrome and Mozilla Firefox

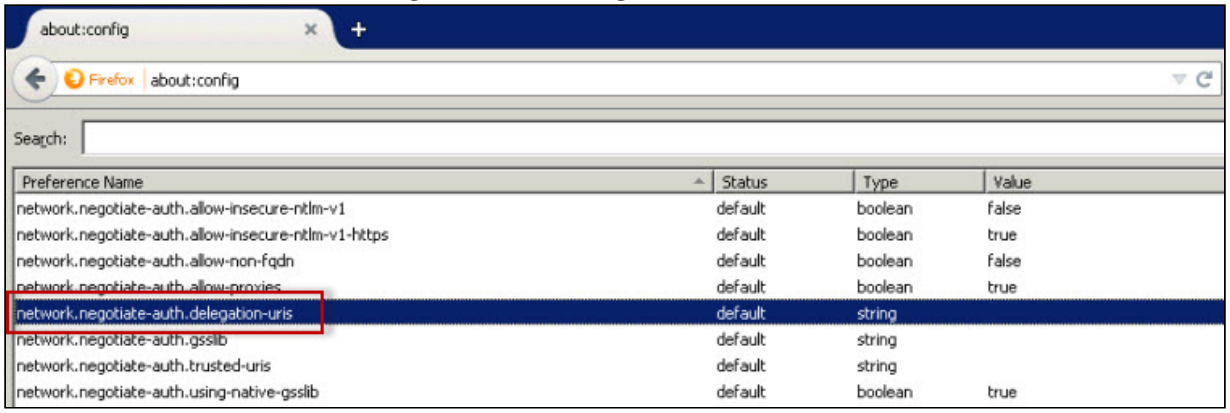
To configure **Google Chrome**, you can go to the following link: http://dev.chromium.org/developers/design-documents/http-authentication.

To configure **Mozilla Firefox**, view the following steps:

1. Open your Mozilla Firefox browser and type **about:config**.
2. Click **I´ll be careful, I promise!**.



3. Search and double click **network.negotiate-auth.delegation-uris**.



4. Type https://localhost:9291. Click **OK**.
5. Search **network.automatic-ntlm-auth.trusted-uris**, double-click and type the same value as before: https://localhost:9291. Click **OK**.

Close all instances of the Firefox browser to make the changes effective. Launch the browser again and access the application.

6 Navigate the Desktop Console

Learn the basics for navigating the Desktop Console, the following table contains the steps and main features of SQL Safe desktop console:

☑	**Follow these steps:**
☐	Register the SQL Server instances you want to back up and restore.
☐	Deploy Backup Agents to your registered instances.
☐	***If you want to receive alert notifications through e-mail***, configure the e-mail settings.
☐	***If you want the option to use the InstantRestore feature to restore a database***, enable the InstantRestore service.
☐	Determine what your backup and recovery strategy will be.
☐	Determine whether you will perform manual backups of create policies to automate your backups.
☐	Determine which types of compression and encryption are best for your environment.
☐	Determine whether you will use log shipping policies as part of your recovery strategy.
☐	Ensure that you are able to restore the database. You can perform a full backup or you can test your configuration by running a verify-only backup. To perform a verify-only backup, run the Restore Wizard and select the **Verify only** option on the Recovery tab.

6.1 What information is available in the SQL Safe Today view?

The **SQL Safe Today** view provides a high-level record of backup and recovery operations across your enterprise.

Use this view to monitor the status of backup and restore operations and easily access the most commonly used tasks. This view automatically displays when you start the Management Console.

6.1.1 How do you access SQL Safe Today?

To use SQL Safe Today, click the **SQL Safe Today** globe icon in the task bar, or click **View** > **SQL SafeToday** from the menu.

6.1.2 What is the Status Summary?

The Status Summary provides a simple indicator to tell you at a glance whether backup and restore operations across your enterprise have been successful. The green check icon indicates success, and the red X icon indicates errors have occurred.

6.1.3 What are the Status Details?

The statistics pane shows the values for the following metrics:

- Number of policies whose status is OK (all operations have completed successfully)
- Number of policies whose status is not OK (includes a failed, skipped, or canceled operation)
- Number of successful operations
- Number of operation that failed (returned errors)

For more information, see how policies work.

6.1.4 Why are SQL Safe Today statistics different than Server statistics?

The filter used by the SQL Safe Today statistics is different than that used by the Servers statistics. SQL Safe Today shows the status for all instances and databases. Servers shows status based on your filter settings plus the currently selected node and the databases under it.

6.1.5 What is Disk Space Savings?

The disk space savings pane shows the disk space savings achieved by using compression on your backup sets, and, using the TCO/GB parameter set in the SQL Safe Preferences, calculates your return on investment (ROI) using SQL Safe for your SQL Server instance backups. For more information on calculating TCO, see modify Total Cost of Operation (TCO) preferences.

6.1.6 What tabs are available on the SQL Safe Today view?

There are two tabs available for you to choose on the SQL Safe Today view:

- Policies
- Backup and restore operations

6.1.7 What can you find on the Policies status tab?

The Policies tab displays the current status of your Backup and Log Shipping policies.

When the Policies tab is selected, all existing policies are displayed with the following columns:

Column Header	Definition
Status	Status can be ok, or display a warning or error state.
Name	Displays the policy name.
Databases Covered	Displays the number of databases covered by the policy.
Last Backup Time	Displays the start date and time of the last backup executed by the policy.
Last Backup Failure Time	Displays the start time of the last failed backup executed by the policy.

For more information, see backup policies and log shipping policies.

6.1.8 What can you find on the Backup & Restore Operation status tab?

The Backup & Restore Operations tab displays the current status of the backup and restore operations that were scheduled to run today.

When the Backup & Restore Operations tab is selected, all existing operations are displayed with the following columns:

Column Header	Definition
Progress	During an operation, the progress bar will denote the percentage of the operation completed. When the operation is complete, it will display a green bar labeled 100%. When an operation completed with errors, this column will display a red bar labeled Error. This column also indicates when the backup file has been deleted (groomed), and therefore is no longer available to be restored.
Instance	Displays the instance name that was backed up or restored by this operation.
Icon	Displays an icon if the backup includes maps containing metadata for InstantRestore and SQL virtual database. For more information about InstantRestore, see how InstantRestore works. For information about SQL virtual database, see recover objects using Virtual Database.
Database	Displays the database name that was backed up or restored by this operation.

Column Header	Definition
Operation	Displays the operation performed. The options are Backup, Restore, and Verify.
Backup Type	Displays the type of the backup performed by the operation. The options are Full, Log, Differential, and File.
Start Time	Displays the start date and time of the operation.
Duration	Displays the number of seconds required to complete the operation.

6.1.9 Can you customize the columns in the grid?

You can sort by the content of any of the columns by clicking on the column header.

6.1.10 How do you refresh the operations status?

If a recent operation does not appear in the status view, you can refresh the status of this pane by clicking on the refresh icon in the pane title bar.

6.1.11 What are the Common tasks?

The Common Tasks are shortcuts to some of the more frequently performed actions in SQL Safe.

Task	Definition
Register SQL Server	Allows you to add a new SQL Server instance. For more information about Registering SQL Server Instances, see register an instance
Backup Database	Starts the SQL Safe Backup Wizard, which allows you to back up one or more databases. For more information about the Backup Wizard, see perform a manual backup.
Restore Database	Starts the SQL Safe Restore Wizard, which allows you to restore one or more databases. For more information about the Restore Wizard, see restore a database.
Create Backup Policy	Starts the SQL Safe Backup Policy Wizard, which allows you to create a new backup policy. For more information about the Backup Policy Wizard, see how backup policies work.
Create Restore Policy	Starts the SQL Safe Restore Policy Wizard, which allows you to create a new restore policy. For more information about the Restore Policy Wizard, see

Task	Definition
Manage SQL Safe Backup Agents	Changes the Console Display to the Settings View. From this view, you can manage your SQL Safe Backup Agent Settings. For more information about the SQL Safe Backup Agents, see install and configure the SQL Safe Backup Agent.

6.2 Register an instance

SQL Safe provides the ability to add, remove, and group SQL Server instances from within the tree pane.

6.2.1 How do you register an instance?

To register a SQL Server instance with SQL Safe:

1. In the navigation pane, click **Servers**.
2. Right-click on the Server Group to which you want to add the SQL Server instance.
3. Select **Register SQL Server** from the context menu.
4. In the Available Servers list of the Register SQL Servers dialog, select the instance you want to add to the Server Group.
5. Click **Add >** and SQL Safe moves the selected server to the Added Servers list.
6. Select the required authentication method used to log in to the SQL Server instance, and then click **OK**.

or

1. Go to the Common Tasks section of the SQL Safe Today view, click **Register SQL Server**.
2. Select the SQL Server Instance from the  add more option available on the Server name.
3. Assign a Friendly name for your SQL Server.
4. Select the Server Group from the drop-down option.
5. Determine the SQL Server Authentication method (Windows or SQL Server).
6. Use the respective user name and password.

> You can also register an instance "on the fly" when you back up the hosted databases through the Management Console.

6.2.2 How do you group SQL Server instances?

To group SQL Server instances, right-click the SQL Server instances folders on the pane tree of the **Servers** View and select **Add Group**, then you can Register SQL Servers to add them to the respective group.

Organizing your SQL Server instances into related groups can help you verify the backup status of specific types of SQL Server instances. For example, you can categorize servers based on location, purpose, importance, platform, or any other logical category.

6.2.3 What other options do you have available when registering an instance?

To see more available options for your instance, right-click the respective instance and select one of the following options:

- **Backup Database(s) -** opens the SQL Safe Backup Wizard. For more information about backups, go to perform a manual backup.
- **Restore Database(s) -** opens the SQL Safe Database Restore Wizard. For more information about restores, go to perform a manual restore.
- **Restore Database(s) Files -** opens the SQL Safe Database File Restore Wizard. For more information about restore database files, go to perform a manual restore.
- **Browse TSM Backups -** you must install the TSM client before browsing TSM Backups. For more information, go to integrate SQL Safe with TSM.

- **Install SQL Safe Backup Agent / Upgrade SQL Safe Backup Agent -** opens a dialog box to install or upgrade the Backup Agent. You can select the checkbox option to include to your installation the SQL Safe Extended Procedures or include the Backup Service Install log. For more information, go to install and configure the SQL Safe Backup Agent.
- **Install SQL Safe Extended Stored Procedures -** opens a dialog box to install SQL Safe Extended Stored Procedures. You can select the checkbox option to include to your installation the Backup Service Install log. For more information, go to deploy the SQL Safe Extended Stored Procedures.
- **Enable/Disable SQL Safe InstantRestore -** select this option to enable the InstantRestore and bring the database online quickly while the restore occurs in the background. You can also disable it by selecting this option. For more information, go to enable/disable SQL Safe InstantRestore.
- **SQL Safe Agent Properties -** opens a dialog box where you can edit the SQL Safe Agent properties. For more information, go to SQL Safe Backup Agent properties.
- **Remove SQL Server -** opens a dialog box where you can confirm you want to remove the SQL Server instance.
  Keep in mind, in the **Servers tree**, the SQL Server Instances node lists all the SQL Server instances you have registered with SQL Safe. However, this list may not reflect all registered SQL Server instances across your environment. For example, when your backup or log shipping policy contains instances registered by other database administrators, SQL Safe lists these instances in the Discovered Instances node. Although you can delete this node, SQL Safe recreates the node after a policy status is refreshed.
- **Refresh SQL Server -** automatically refreshes the SQL Server Instance Information page.
- **Properties -** use this option to edit general and advanced properties of a registered Instance.
  The General tab, allows you set different credentials to connect to the SQL Server Instance. For more information about the credentials needed to monitor an instance, go to add new SQL Server instances.
  The Advanced tab, allows you specify a Network Name to connect to the SQL Server Instance and/or the SQL Safe Agent Components.

6.3 Configure your deployment

After initially installing and setting up SQL Safe, there are several tasks you might want to do in order to further customize and streamline your install. Review the following sections to get a good understanding and make the best of your SQL Safe installation.

- Configure your Console preferences
- Manage licenses
- Configure the Management Service
- Configure e-mail settings
- Import archived backup sets
- Understand the total cost of your operations (TCO)

6.3.1 Configure Console preferences

SQL Safe allows you to modify many of the default settings of the application, and you can change your **Management Console Preferences** at any time. To access this option, go to the **Tools** Menu and select the respective option. The window for **Management Console Preferences** allows you to modify settings in the following categories:

- Backup
- Agent Deployment
- User Experience
- Policy Data
- Amazon Settings
- Azure Settings

What Backup settings can you change?

On the **Backup** tab, you can set the default parameters that appear on the **Backup Wizard**. Set the default parameters to the values you typically use. If you want to use different settings on any given backup, you can still make changes on the wizard itself.

The parameters you can set include the following:

- Backup archives location.
- Tivoli Storage Manager backup archives location.
- Default compression and encryption algorithms.
- Generating maps containing metadata for use with InstantRestore and SQL virtual database.
- Auto-generated backup file names.
- Number of threads employed in a backup.

What Agent Deployment settings can you change?

On the **Agent Deployment** tab, you can identify service account used to run the agents. You can also choose whether or not you want to automatically upgrade the Backup Agents and the XSP if you upgrade to a new SQL Safe version.

What User Experience settings can you change?

On the **User Experience** tab, you can:

- Enable automatic refresh on screen and set the number of seconds between each refresh.
- Change display settings for server status.
- Set the Total Cost of Ownership parameter necessary to calculate your return on investment.
- Configure troubleshooting settings.

What Policy Data settings can you change?

On the **Policy Data** tab you can define the location of your policy data files.

By default, policy data files are stored in the C:\InstallPath\SQL Safe\PolicyData folder, but you can use this option to specify a different location. Click **Local Path** and browse the folder where you want to store your policy data files or select the respective option for creating a new folder.

By default,each agent uses its own installation directory to store policy data. If a custom location cannot be created on a specific server, the Backup Agent will use <InstallPath>\PolicyData.

What Amazon Settings can you change?

On the **Amazon Settings** tab, you can:

- Specify the Access and Secret Key of your cloud credentials.
- Select the Region where your information will be stored.
- Set the Bucket Name where your backup will be stored.
- Specify the Part Size in megabytes of the backup file that will be sent to the bucket simultaneously.

> The minimal value for File Size is 10 MB. (1,000,000 bytes).

- Specify the Temporary Download Location to improve resiliency and performance when downloading large backup files from Amazon S3 Cloud Storage. This location will be displayed when the "Download File from Cloud" option is checked in a restore wizard or restore policy.

> When Amazon cloud settings are set in the preferences, they can be used in backup and restore operations.

- When performing backup and restore operations, you will see the option to specify Subfolder(s) (optional) if you want to save your backup file or restore it from there. Consider the following situations:
  - If the Subfolder field is left empty, the backup file will be saved in the root of the specified container.
  - If the Subfolder field is populated with the name of a folder which does not exist on the storage container \ bucket, the folder will be created and the backup file will be saved to the specified Subfolder.
  - Multiple Subfolders can be specified by separating each folder with a forward slash: MyFolder/MySubfolder/MyNestedFolder.

What Azure Settings can you change?

On the **Azure Settings** tab, you can:

- Specify the name of the Azure Container where the new blob will be created and the backup stored.
- Specify the Azure Storage Account Name and Access Key.
- Define the Azure Sector Type:
  - *Public* - commercial cloud storage solution.
  - *Government* - cloud storage solution offered to US government customers and their partners.
- Specify the Part Size in megabytes of the backup file that will be sent to the bucket simultaneously.

> The minimal value for File Size is 10 MB. (1,000,000 bytes).

- Specify the Temporary Download Location to improve resiliency and performance when downloading large backup files from Microsoft Azure Blob Storage. This location will be displayed when the "Download File from Cloud" option is checked in a restore wizard or restore policy.

> When Azure settings are set in the preferences, they can be used in backup and restore operations.

6.3.2 Manage licenses

The License Key Manager provides an intuitive, simple-to-use interface for SQL Safe license key management. You can:

- View the instances licensed with your current license.
- Select which instances you want SQL Safe to take into account for your license key instance count.
- See how many available instances you can still license for backup operations.
- Edit your license key.
- Save your license information to a file.

The Management Service manages your SQL Safe license and receives requests from the respective Backup Agents to license your instances.

How do you manage your licenses?

You may need to edit your current license if you exhaust your trial license, or if you need to replace it with one that allows you to license more instances.

To access these options, on the **Tools** menu, click **License Key Manager** . You can also click the key icon located on the task bar to access the same option.

What information can you see on the License Key Manager window?

On the License Key Manager window you can find:

- Instances licensed for backup operations.
- Your license keys.

On the Instances tab you can find:

- A list of the instances licensed for backup operations and all available registered instances.
- An option to license All or None of them.
- An option for automatically licensing instances for scripted agent deployments.
- An option to save this information to a file.

On the License Keys tab you can find:

- Your current licences keys, type, the number of instances allowed for each key, and the expiration date.
- Options for removing and adding license keys.
- The number of used licenses.
- An option for automatically licensing instances for scripted agent deployments.
- An option to save this information to a file.

To upgrade a trial license to a permanent license:

1. On the **License Key Manager** window, on the License Keys tab, click **Add**.
2. Enter the respective license key.
3. Click **OK** . The license key will be displayed in the License Key Manager window.
4. If you want to save the list to a file, click **Save to a File** and save the file to your desired location.

What is a multi-instance license key?

A multi-instance license key allows you to centralize the license management with the SQL Safe Management Service. SQL Safe Backup Agents configured to this Management Service will use this licensing management

method and enable you to enter a license key through the Management Console and support the licensing of a certain number of instances.

> Please take into account that a multi-instance key will replace any single instance keys previously installed and the user will be prompted to switch to centralized license management.

> Standalone SQL Safe Backup Agents not configured to a SQL Safe Management Service will be treated as standalone installations and therefore use the old licensing model.

How can you license your instances for backup operations?

On the **License Key Manager** window, you can see the list of all your registered instances under the Instances tab. Check or uncheck those instances that you want SQL Safe to license to perform backup operations.The number of available licenses will be updated according to your choices.Then click **OK**.

What are the terms of the trial license?

By default, SQL Safe installs with a limited 14-day time, unlimited instances trial license key. After you install the SQL Safe components using the Typical or Custom setups, the Management Console lists your trial license in the **License Key Manager**. This license key is stored in the SQL Safe Management Server.

What are the terms of the production license?

SQL Safe licenses are issued per SQL Server instance and for a specific time period. You can manage this license with the License Key Manager. The SQL Safe production license gives you full access to the Backup Agent through the Management Console, including operation status information.

What is the SQL Safe Lite license?

When you have different versions of SQL Safe deployed in your environment, one or more registered SQL Server instances may be running SQL Safe Lite.

SQL Safe Lite does not support backup and restore operations through the Management Console. For example, you cannot create a backup policy for a SQL Server instance running SQL Safe Lite.

If you want to manage all registered SQL Server instances through the Management Console, you can upgrade the SQL Safe Lite Backup Agents to the enterprise version of SQL Safe.

How do you upgrade the SQL Safe Lite license?

You can temporarily upgrade a SQL Safe Lite license to an enterprise edition license by installing a SQL Safe trial license. Note that, when the trial period has expired, your license will revert back to SQL Safe Lite.

You can then permanently upgrade a SQL Safe Lite license to an enterprise edition by purchasing a production license key and entering it in the License Key Manager.

To upgrade a SQL Safe Lite license:

1. In the Servers tree, expand the SQL Server Instances node, and then select the instance that is running the SQL Safe Freeware Edition Backup Agent.
2. Click **Enable Trial License** in the Backup/Restore Operation Status pane.

How do you upgrade the SQL Safe Freeware Edition?

You can upgrade SQL Safe Freeware Edition to SQL Safe enterprise edition by upgrading the Backup Agent on the corresponding SQL Server computer.

This installs the SQL Safe enterprise edition trial license. You can then permanently upgrade a SQL Safe Lite license to an enterprise edition by purchasing a production license key and entering it in the License Key Manager.

To upgrade a SQL Safe Freeware Edition Backup Agent:

1. Navigate to the **SQL Safe Agents** view.
2. Right-click the target SQL Server computer, and then select **Install SQL Safe Backup Agent** on the context menu.

How do you save the license keys to a file?

1. On the **Tools** menu, click **License Key Manager** .
2. Click **Save to File** , and browse to the location to which you want to save the file.
3. Enter the file name, and click **Save**.

6.3.3 Configure the Management Service

You can specify the location and authentication credentials necessary to access the SQL Safe Repository. You can connect to the Repository database using Window Authentication or SQL Server Authentication.

You can also change the port assignment for the Management Service.

What are the available fields?

Computer

Allows you to select the computer where the Management Service is located.

SQL Server

Specify the SQL Server instance that currently hosts the SQL Safe Repository.

Database

Allows you to specify the name of the SQL Safe Repository.

Windows Authentication

Allows you to specify Windows Authentication for accessing the selected SQL Server instance. Selecting this option uses the credentials of the Management Service to log on to the SQL Safe Repository.

SQL Server Authentication

Allows you to specify SQL Server Authentication for accessing the selected SQL Server instance. Selecting this option allows you to specify the SQL Server login ID and password you want to use to access the target SQL Server instance.

Test Connection

Allows you to verify that the Management Service can use the specified account to connect to the Repository database.

Configure E-mail Notifications

Allows you to configure the settings for sending email alerts. In this section you can also enable or disable the option for sending these notifications to the Windows Application Event Log on the Management Service Computer.

Repository Grooming

Allows you to specify how long (in days) you want to keep operational history, such as status messages for backup and restore operations. By default, the Repository is groomed every 30 days. Operational history older than 30 days is permanently deleted.

How do you configure the Management Service?

To configure the Management Settings you can use any of the following paths:

- Go to the **Tools** Menu and click **Repository and Management Service Settings**.

- Select the  icon on the bar menu.

6.3.4 Configure e-mail settings

You can enable SQL Safe to send email notifications about the current status of your backup and restore operations.

Access these settings by clicking **Configure E-mail Notifications** on the **Repository and Management Service Settings** window, accessible from the bar menu  icon, or by selecting **E-mail Notification Settings** in the **Tools** menu.

What email settings can I change?

If you enable **E-mail Notifications**, you can configure how the email will appear in your Inbox.

Sender Name

Enter the name that will appear as the sender of the email.

Reply-to Address

Enter the email address that will appear as the sender, and where replies to the message will be sent.

Priority

Select low, normal, or high priority for the email alerts.

What mail server information is required?

You must specify the mail server information so that SQL Safe can send email notifications.

Server Address

Enter the address of your mail server.

Server Port

If you want to specify a port different from 25 (set by default), you can do that in this section. You can also enable SSL encrypted connection.

SMTP Authentication

If your SMTP server requires authentication, you must type a valid **User Name** and **Password** that SQL Safe should use to access to the mail server.

Test your settings

To be sure that your settings are correct, click **Test Settings** on the bottom section of the window, then check the test email sent to your email server.

6.3.5 Import archived backup sets

SQL Safe allows you to import archived backup sets into the SQL Safe Repository to manage all your backups from one place.

SQL Safe cannot import copies of backup files that have been previously deleted or groomed. You can still access the backup files from the alternate location through the Restore wizard.

How do you import archived backup sets?

You can find and add archive files created outside your current SQL Safe environment to the Repository. You can also use this feature to help you recreate the SQL Safe Repository in the event of a critical failure.

You can reach the **Locate Backup Sets** dialog from the **File** menu and then selecting **Import Backup Archive(s)**.

To import backup archives from a local folder:

1. In the **Locate Backup Sets** window, click **Browse Locally**.
2. Select the archive file to import.
3. Review the displayed backup set information and click **OK**.

To import backup archives from a remote share:

1. In the **Locate Backup Sets** window, click **Browse Remotely**.
2. Select the SQL Server instance from the drop-down menu.
3. Select the archive file to import.
4. Review the displayed backup set information and click **OK**.

To import backup archives from TSM tape backup:

1. In the **Locate Backup Sets** window, click **Browse TSM**.
2. Select the appropriate TSM options file.
3. Enter a High Level and Low Level search parameters and choose whether to include or not inactive files.
4. From the Results text box, select the found files to be imported.
5. Review the displayed backup set information and click **OK**.

6.3.6 Understand total cost of operation (TCO)

SQL Safe provides a built-in calculator to help you calculate your monetary return on your SQL Safe investment. You can view this calculator in the Disk Space Savings pane on the SQL Safe Today view.

The calculator attempts to measure the time and monetary savings you gain through using the SQL Safe compression scheme. The Return On Investment (ROI) calculator bases your ROI on the total cost of ownership of your storage devices multiplied by the amount of disk space savings you realize using SQL Safe. SQL Safe defaults to the commonly used estimate of $200 per GB of storage. You can change this estimate to reflect your particular hardware configuration.

6.4 Install and Configure the SQL Safe Backup Agent

The Backup Agent performs backup and restore operations. The agent is a service that runs on the target SQL Server instance. When you request a backup or restore operation, the Management Console wakes the previously deployed Backup Agent.

> ⚠ Take into account that the Backup Agent must be licensed to perform Backup, InstantRestore, Object Level Recovery, and Virtual Database restore operations. If it is not licensed yet, the Agent will contact the Management Service to request a license.

While executing the backup or restore operation, the agent periodically sends messages to the Management Service.

6.4.1 How do you install the Backup Agent?

You can install the Backup Agent locally using the setup program or deploy the Backup Agent remotely using the Management Console. To install the agent in an environment that does not contain a SQL Safe Management Service and Repository, use the Agent Only setup type provided in the setup program. The Backup Agent will contact the Management Service to request licensing. For more information, see install SQL Safe Backup Agent and license management.

6.4.2 Can you monitor the Backup Agent?

You can monitor and maintain the performance of each Backup Agent, click **SQL Safe Agents** in the navigation pane. For more information, see Backup Agent configuration.

6.4.3 Can you modify the Backup Agent properties?

You can modify many of the SQL Safe Backup Agent properties from the Management Console and adjust performance parameters to suit your system needs. For more information, see modify Backup Agent properties.

6.4.4 How do you upgrade your Backup Agent?

You can configure SQL Safe to automatically upgrade the Backup Agent to the current software version in the SQL Safe Preferences window. For more information,see configure your deployment.

6.4.5 How do you run the Backup Agent without receiving messages?

You can run the Backup Agent in silent mode. Silent mode allows you to use the Backup Agent in environments that do not require the Management Service or SQL Safe Repository.

When in silent mode, the Backup Agent does not return status information about backup and restore operations. Use this mode if you do not plan to track backup and restore status, or if you plan to perform backup and restores through the command-line interface only. This flexibility allows you to easily integrate SQL Safe into your existing backup and recovery infrastructure so you can take advantage of SQL Safe features without changing your established processes.

6.4.6 What do you do after installing the SQL Safe Backup Agent?

When deployment is complete, you can backup and recover databases hosted on your virtual instances. You do not need to install any other SQL Safe components on your clustered servers to implement a disaster recovery strategy for those virtual instances. If you have a clustered environment hosting multiple instances, you must manually deploy the SQL Safe Backup Agent on each node.

6.4.7 Install the SQL Safe Backup Agent

You can remotely deploy the SQL Safe Backup Agent from the Management Console to SQL Server instances across your enterprise.

To install a SQL Safe Backup Agent:

1. In the navigation pane, click **SQL Safe Agents**.
2. Right-click on the computer in question in the tree pane.
3. Click **Install SQL Safe Backup Agent** from the context menu.
4. Choose whether you want to install the SQL Safe XSP. You can also enable or disable the option for installing SQL Safe Agent Extended Stored Procedures.
5. Click **OK**.

6.4.8 Backup Agent configuration

How do you access the agent configuration information?

To manage your SQL Safe Backup Agents, click **SQL Safe Agents** in the navigation pane. To view information about a specific agent, click the corresponding SQL Server computer listed in the tree pane and you will be able to see the configuration information of the respective agent.

What agent configuration settings can you view?

The content pane in the SQL Safe Agents Settings view contains the agent configuration information. This information allows you to monitor and maintain the performance of each Backup Agent.

Column	Definition
Computer	Displays the name of the host computer.
Version	Displays the version number of the selected Backup Agent.
Management Server	Displays the location of the SQL Safe Management service that the Agent is configured to communicate with.
Max Load	Displays the maximum number of concurrent operations that the backup agent can perform.
Priority	Displays the Windows thread priority at which backup agent threads run.
Send Status	Displays the frequency that the agent is configured to communicate with the Management Server.
SQL Timeout	Displays the SQL DMO timeout value, which determines how long the Backup Agent will wait for a response from SQL Server before timing out.
VDI Trans. Limit	Displays the maximum size of a transfer block for the VDI operation.
VDI Buffers	Displays the number of buffers used for the VDI operation.
VDI Block Size	Displays the size of a VDI device block. All data transfers are integer multiples of this value.
VDI Timeout	Displays the timeout for configuring the VDI.

6.4.9 Modify the Backup Agent properties

You can modify many of the SQL Safe Backup Agent properties from the Management Console and adjust performance parameters to suit your system needs.

If the SQL Server instance is running SQL Safe Lite, the Send Status every *x* seconds option is ignored. SQL Safe displays operation status information only for Backup Agents running with an enterprise edition license.

If the SQL Server instance is running SQL Safe Freeware Edition, all settings are unavailable. You must upgrade the Backup Agent to either SQL Safe Lite or the enterprise edition to make changes to the Backup Agent properties. For more information, see manage licenses.

How do you access the Backup Agent properties?

To manage your SQL Safe Backup Agents:

1. In the navigation pane, click **SQL Safe Agents**.
2. Right-click the appropriate SQL Server instance.
3. Click **Properties** from the context menu.
4. Change the SQL Safe Agent properties to improve the performance of your backup and restore operations, or enable debug mode for troubleshooting an issue. For more information about SQL Safe Agent properties, see view agent settings.
5. Click **OK.**

You can also change the port assignment for the Backup Service.

Why should you enable troubleshooting?

Occasionally when you contact IDERA support for assistance, a representative will ask you to enable logging to get a better idea of what the issue is in your environment. SQL Safe allows you to customize your debug settings when troubleshooting an issue with your Backup Agent.

Is there a disadvantage if you leave debug mode enabled for a long period of time?

There is no disadvantage to leaving SQL Safe in debug mode for an extended period of time. If you experience an issue that occasionally and unexpectedly occurs, or you want to capture data over a long period of time, leave debug mode enabled. This settings gives you the advantage of already logging the data when the issue occurs.

6.4.10 Manage debug settings

The Advanced Debug Settings display additional debug selections for troubleshooting the Backup Agent running on the selected SQL Server computer. You can also set the log file characteristics and maximum size depending on what issue you want to troubleshoot. It is recommended that you change these settings based on guidance from IDERA support.

A rolling log allows you to create a log files that "roll" when they reach the maximum file size or the service is restarted, meaning that when the max size/restart occurs, it deletes the oldest information and logs the newest. This feature helps you avoid a large, cumbersome log file.

How do you access debug settings?

To manage your debug settings:

1. In the navigation pane, click **SQL Safe Agents**.
2. Right-click the appropriate SQL Server instance.
3. Click Properties from the context menu.
4. Enable Debug Mode.
5. Click **Advanced**.
6. Make the necessary changes, and then click **OK**.

What log file settings can you change?

If you enable debug mode, SQL Safe creates a log file based on the settings in the Roll Log fields. You can choose one of the following options for the log file

No

Maintains the debug log as a single file that is unlimited in size. This option allows SQL Safe to capture diagnostic information over an extended period of time. Your log file may become very large and should be monitored to avoid any issues with file size.

> Do not enable No unless absolutely necessary. Use No only when the problem being diagnosed occurs very infrequently and is not noticed in a timely manner. In almost all cases, increasing the size or number of files is sufficient for troubleshooting an issue and an unlimited file size is unnecessary.

Yes and keep 1, 6, or 12 files

Maintains the debug log as a series of files that are limited in size and quantity. This option allows SQL Safe to capture diagnostic information over a limited period of time, depending on the size and quantity of files kept. When a log file reaches the size limit or the service is restarted, SQL Safe renames the file and starts a new log file. The older the log file, the higher the digit that exists at the end of the file name. For example, file `x.log.3` is more recent than `x.log.4`. When the log file rolls over and reaches the maximum quantity, SQL Safe deletes the oldest file and the next oldest file takes its place. Because the amount of space used by the logs is limited, this setting does not require you to monitor the log files.

A Yes option is the recommended setting. Be aware that the amount of history retained is limited, and it may be possible for a problem being diagnosed to be missed. If this may be the case, first increase the quantity of files retained, then increase the size if necessary. Increasing quantity before size helps to maintain log files that are smaller and easier to view and send to IDERA support.

6.5 Define your Backup and Recovery Strategy

Before performing database backups within your SQL Server environment, establish a backup and restore strategy. Your strategy should consider the following points:

- Data availability needs.
- Data loss impact.
- Recovery model you want to use: Simple, Full, or Bulk-Logged.
- Restore process you want to use: InstantRestore or normal.
- Data storage space allotted to backup storage.

SQL Safe supports whatever strategy you decide to implement, while allowing you to take advantage of the fastest, most efficient SQL Server backup solution available. You can create custom backup and restore policies that ensure your data is archived and recovered according to your corporate standards and Service Level Agreements (SLAs).

If your strategy includes tape backup, SQL Safe also allows you to easily integrate the third party data-protection product, Tivoli Storage Manager (TSM), into your backup strategy. For more information, see integrate SQL Safe with TSM.

If your SQL Server environment requires FIPS compliance, see ensure FIPS compliance.

6.5.1 How do you define a backup and recovery strategy?

Use the following checklist to ensure you have everything in place to successfully implement your backup strategy.

☑	**Follow these steps ...**
☐	Determine the backup types you want to perform for your different SQL Server instances.
☐	Determine what type of compression you need.
☐	Determine the type of encryption you want to use.
☐	Identify which databases should be routinely archived using backup policies.
☐	Identify which databases should be routinely recovered using restore policies.

6.5.2 How can you get your database up and running quickly during a restore?

SQL Safe's InstantRestore feature is the fastest way to get your database back online. Under certain conditions, InstantRestore allows you to restore your database while providing your users with quick access to the database during this process. Note that you may experience some performance issues because the restore is still running while you attempt to use the database.

6.5.3 How to choose backup type

SQL Safe supports four standard database backup types:

- Full Backup.
- Differential Backup.
- Transaction Log Backup.
- File Backup.

You can use a backup type exclusively or combine types to fit your backup strategy.

What is a full backup?

A full backup creates a full copy of the data in a database. Full backups usually run at regularly scheduled intervals and require more storage space and time to complete. Full backups copy data and transaction log pages to the backup set. The backup is smaller than the database itself because unused space is not retained.

Full backups allow you to restore your database to its original state prior to backup. During the restoration of a full backup, the SQL Server instance being restored rolls back uncommitted transactions. Use transaction log backups to recover uncommitted transactions.

What is a differential backup?

Differential backups record only the data that changed since the last full backup. Consider using differential backups on active SQL Server instances where minimal database downtime is critical. Smaller and faster differential backups allow you to make more frequent backups with less impact on your server. Performing frequent backups helps maintain optimal database availability and minimizes data loss risks. Differential backups allow you to restore your database to the last completed differential backup.

What is a transaction log backup?

A transactional log backup creates a copy of the transaction log file. It sequentially records all database transactions that occurred since the last transaction log backup. In conjunction with a full or differential database restore, restoring a transaction log backup allows you to recover the database to the point of failure or a specific time.

Typically transaction log backups do not require intensive resource usage and can be scheduled more frequently than other backup types. Ensure you increase the frequency of your transaction log backups if your database has a high transaction rate. Also, consider storing critical transaction log backups on fault-tolerant storage devices.

While you cannot execute a transaction log backup during a full or differential backup, you can during a file backup. Ensure you create database or file backups before backing up the transaction log. The transaction log contains only the database changes made after the creation of the last backup.

What is a file or filegroup backup?

Backs up either individual files or all files in a filegroup within a database. Backing up single files or filegroups allows you to restore only corrupted files. Restoring only corrupted files increases recovery speed. Consider file and filegroup backups when your database has one or all of the following attributes:

- Database size hinders regular full or differential backups
- Database can be unavailable for short periods of time only
- Specific files are either regularly corrupted, are more critical, or change more frequently than others

You can backup files or filegroups and transaction logs at the same time.

6.5.4 How to choose compression and encryption

SQL Safe offers a unique combination of state-of-the-art compression and encryption technologies. These technologies set SQL Safe apart and make it unique in the SQL Server backup arena. You designate the compression rate necessary to match your storage needs, and you select the level of encryption you need to ensure data security within your environment.

For more information about the compression levels available, see understand compression levels and understand IntelliCompress options. For information on how selecting the appropriate compression scheme reduces your storage costs, see understand Total Cost of Operation (TCO)

For more information about the encryption levels available, see understand encryption levels.

SQL Safe automatically detects whether your environment requires compliance with the Federal Information Processing Standard (FIPS), and then chooses the appropriate encryption algorithm. For more information, see ensure FIPS compliance.

Understand compression levels

SQL Safe allows you to set the compression rate suited to your backup needs. You designate a default compression level during the initial setup of SQL Safe. Any time prior to initiating a backup, you can modify your compression level.

How do you choose the best compression level for my environment?

The compression level that is best for your environment depends on your storage and performance needs. Before you choose a compression level, determine whether you need maximum storage and compression (lower performance) or maximum performance (lower compression).

Compression rates and backup times depend on the following factors:

- Whether the SQL Server computer utilizes multiple processors
- Whether you are striping data to multiple backup files
- Available bandwidth on your network connections
- Current processing load, such as backing up multiple databases in the same job
- The type of data you are backing up (for example, text compresses to a smaller size than binary data)

Level 1

Low compression. Provides high execution speed and minimal server load. This compression level typically provides 75-90% compression rates on text data. This compression rate may significantly decrease if you are backing up a database that contains binary data or previously compressed data.Use this compression level if you want to perform fast backups, sometimes during business hours, at the expense of a larger size.

In environments with a slow write speed, this level will not produce backups as fast as higher levels of compression.

Level 2

Medium compression. Provides good data compression while maintaining high-speed execution. This compression level places a moderate load on your server to provide increased compression. This compression level works well in environments with a good balance between multi-processor servers (for example, a 4- to 6-way SMP server) and IO speed.

Use this compression level if your environment includes one or more of the following conditions:

- You want to increase compression without significantly impacting performance
- You can schedule backups during off-hours, if needed

Level 3

High compression. Provides a high level of compression while slightly decreasing execution speed. This compression level provides significant reduction in backed up data size, while placing a higher load on your server. This compression level works well for nightly backups in environments with a powerful multi-processor servers (for example, an 8-way SMP server) where saving space is a high priority.

Use this compression level if your environment includes one or more of the following conditions:

- You want to maximize compression without significantly impacting performance
- You can schedule backups during off-hours, if needed

Level 4

Ultra-high compression. Provides the highest level of compression, to be used when saving space is critical. This compression level places a high load on your server. To achieve acceptable run times, this level should be used on very powerful servers with 8 or more processors and generally only during off-peak periods.

Use this compression level if reduction in backed up data size is your primary objective.

Understand IntelliCompress

SQL Safe offers IntelliCompress compression levels to maximize compression performance for your backups. Each time you run a backup using an IntelliCompress compression level, SQL Safe analyzes your backup data and determines the best algorithm to use. This customization optimizes the performance, no matter how the backup data may have changed since the last backup. Analyzing the data each time you run a backup provides the best compression rate for each backup, so your data is compressed in the optimal way each time, saving you time and disk space.

IntelliCompress – Optimize for Speed (iSpeed)

Provides maximum performance by automatically optimizing for speed. At each backup, SQL Safe selects a compression ratio that provides the fastest backup in that environment. This compression level meets most storage and performance needs. We recommend this compression level, particularly if you are backing up databases that contain text data.

IntelliCompress – Optimize for Size (iSize)

Provides high compression by automatically optimizing for size. At each backup, SQL Safe selects the best mix of compression and speed based on CPU power and read/write speed.

Understand encryption levels

SQL Safe allows you to set the encryption level most appropriate for your backup needs. During the initial setup of SQL Safe, you can select a default encryption level. Any time before executing a backup, you can strengthen or lessen the encryption applied to the current backup.

You must have a password in order to restore an encrypted backup. For security reasons, when you generate a T-SQL or CLI script of an encrypted backup, SQL Safe does not write the specified password to the script. To successfully run the script, supply the appropriate password. SQL Safe also does not store encryption passwords and cannot recover lost or forgotten passwords.

SQL Safe automatically detects whether the target SQL Server instances require FIPS compliant encryption. When this security setting is detected, SQL Safe uses the FIPS-compliant AES encryption algorithms provided by Microsoft. For more information about FIPS compliance, see ensure FIPS compliance.

SQL Safe encryption offers you the following encryption methods, allowing you to choose based on your security needs:

None

Provides the fastest execution speed and does not encrypt backed up data.

Advanced Encryption Standard (AES) 128-bit

Provides a strong encryption. The AES algorithm encrypts data in 128-bit blocks using a 128-bit key.

Advanced Encryption Standard (AES) 256-bit

Provides a stronger encryption. The AES algorithm encrypts data in 128-bit blocks using a 256-bit key. This method provides more secure encryption than AES 128-bit.

Ensure FIPS compliance

You can use SQL Safe to back up and restore SQL Server databases in environments where Federal Information Processing Standard (FIPS) compliance is required. SQL Safe automatically detects whether the target SQL Server instances require FIPS compliant encryption. When this security setting is detected, SQL Safe uses the FIPS-compliant AES encryption algorithms provided by Microsoft.

For more information about FIPS compliance, see the corresponding Microsoft TechNet Web Article and Microsoft Knowledge Base Article.

How do you know whether your environment requires FIPS compliance?

Ask your Windows security administrator whether the FIPS system cryptography setting has been enabled in the Local Security Policy or a Group Policy that applies to the SQL Server computer.

Are there additional product requirements to support FIPS?

No, FIPS compliance for SQL Safe does not require any additional software to be installed.

6.5.5 How InstantRestore works

SQL Safe InstantRestore is a powerful new restore technology that allows you to bring a database online quickly while the restore occurs in the background. SQL Safe enables the SQL Server to immediately begin the transactional part of a database restore, deferring the data file (MDF) restoration until after the database is online. SQL Server continues to handle all transaction log (LDF) restoration activity.

When the restore process is complete and the database is online, SQL Safe takes over and restores the remaining data to the data files in the background. If SQL Server needs data not yet restored, SQL Safe delivers the data to SQL Server directly from the backup. Because SQL Safe never interferes in the SQL Server log operations, ACID (Atomicity, Consistency, Isolation and Durability) compliance for your databases is not affected. When SQL Safe completes data file restoration, it removes itself from all I/O activity of the database and leaves behind a database identical to one restored with a traditional restore process. As a result, SQL Safe is no longer required to access the database.

Beginning with version 7.0, SQL Safe includes a mini-filter driver to support the InstantRestore feature. The driver, named SQL SafeFilterDriver, allows SQL Server to access database data while SQL Safe is performing an instant restore. The driver is only used during an instant restore and is no longer necessary once the database is completely restored.

How to enable InstantRestore

You first must enable the InstantRestore feature. Because some users may feel uneasy installing a device driver on their systems, InstantRestore is disabled by default. You can enable or disable the InstantRestore feature quickly depending on what task you are performing:

- ***If you are viewing your SQL Server instances in the Servers tree***, right-click the instance you want to restore, and then select **Enable SQL Safe InstantRestore** or **Disable SQL Safe InstantRestore**.
- ***If you are in the SQL Safe Database Restore wizard***, complete the wizard up to the Restore type tab where you will find the option for enabling InstantRestore.

If an InstantRestore operation is in progress when a user attempts to disable these components, SQL Safe displays a warning message.

Eligible backups

The InstantRestore feature is available for only a database backup that is:

- **A SQL Safe backup archive with backup metadata (maps)**. Because InstantRestore allows SQL Server to immediately access the data in a backup, the process needs additional information about the backup which is not present in a native backup file. Please note that this information is also missing in SQL Safe backups that are written directly to Tivoli Storage Manager (TSM).
- **A complete database restore**. InstantRestore can restore a database using any normal restore chain starting with a full backup. InstantRestore does not support partial restores such as file restores or restoring a database with the NO RECOVERY or STANDBY options.

Monitoring your instant restores

As SQL Safe performs an instant restore, you can monitor its progress using the SQL Safe Management Console or via alerting. InstantRestore is a new type of restore operation and appears in the Management Console status grid like traditional backup or restore operations.

The InstantRestore operation is tracked with the following two operation types:

InstantRestore

The InstantRestore operation tracks the progress of the entire database restore process. The progress bar increments to 100% for the initial restore progress until the database comes online. When the initial restore completes and the database is online, the status changes to **Online** and the cell changes to light green. SQL Safe then displays a new line for the Hydrate operation.

Hydrate

The Hydrate operation tracks the progress of the background restore process. The progress bar increments to 100% for the background restore progress until the restore is complete. When the database restore is complete, the status of both the InstantRestore and Hydrate operations changes to **Complete** and the cell changes to dark green.

Instant Restore operations include the following two statuses to indicate important milestones of the operation:

Online

The Online status indicates that the database is online and ready for use.

Halted

The Halted status indicates that an event interrupted the InstantRestore process. A network issue between SQL Safe and the backup archive can interrupt an instant restore. Because InstantRestore allows changes to the database while the restore is occurring, the database is not deleted if an issue occurs during Hydration. If such an event occurs, the database transitions to a read-only state to prevent the system and users from writing additional data to the database. At this point, you can restore access to the backup archives and the instant restore can safely resume.

Handling errors during Hydration

If the hydration process is interrupted for any reason:

- The InstantRestore and Hydration operations transition to the Halted state.
- SQL Safe displays an error message stating that hydration is interrupted.

If an error occurs during the InstantRestore operation prior to the beginning of the Hydrate process, SQL Safe displays only the InstantRestore operation with an error status, and includes the error message for the failure.

> The InstantRestore operation has two phases. In the first phase, the T-SQL restore command runs and after the database is online, hydration starts. If an error is encountered in the first phase (i.e. the T-SQL restore command) and the database remains in SQL Server, SQL Safe does not delete the database.

SQL Safe includes the following failure scenarios that may occur during an instant restore.

Component	Failure	Resolution
Server	Crashes	If the server suffers a catastrophic crash and is no longer available, no recovery is available.
Server	Reboots	If the server reboots because of a power failure, automatic software update, or other similar situation, and comes back online correctly, the SQL Safe Filter Service restarts and then resumes hydration.
Server	Runs out of resources	If the server runs out of memory or other resources, and the SQL Safe Filter Service cannot allocate the additional resources during hydration, SQL Safe uses the following steps: 1. ***If the offending process is identified***: a. The user must stop the process. b. The user can manually restart the SQL Safe Filter Service (if stopped). c. If hydration does not resume once the SQL Safe Filter Service restarts, the user can manually restart the operation. 2. ***If the offending process is not identified***, the user may reboot the server.
Server	Suffers a disk failure	If the database or InstantRestore support files is corrupted by a disk failure, no recovery is available.
Service	Restarts	If one or all of the following items restarts, hydration should resume after the restart is complete: • SQL Server • SQL Safe Backup Agent • SQL Safe Filter Service If the database did not go into Suspect mode during this process, hydration resumes from the point where it left off when the SQL Safe Filter Service restarted. If the database is in Suspect mode, the SQL Safe Filter Service brings the database out of Suspect mode, and then resumes hydration.
Filter Service	Crashes	If the SQL Safe Filter Service crashes, hydration resumes once the service restarts. If the SQL Safe Filter Service crashes again, you may need to recover any new data added since the first crash.
Backup file	Is corrupt	If the backup file is corrupt or there is a read problem when accessing the network, the SQL Safe Filter Service fails to decompress during hydration. If the backup file is corrupt, and you have another copy of the backup file, you can restart hydration using the non-corrupt backup file. If the issue is a read problem when accessing the network, you can restart hydration once you address the network issue.

Component	Failure	Resolution
Backup file	Is inaccessible due to a network failure	If the backup file is inaccessible due to a network failure, InstantRestore attempts a retry. If the retry fails, and the maximum retry attempts is reached, the Hydration operation status transitions to the Failure state. Once you correct the network issue or relocate the backup file, you can resume hydration. If the Hydration fails, you may need to restart InstantRestore.
Backup file	Is inaccessible due to a lack of access permissions	If the SQL Safe Filter Service restarts and is unable to open the backup file because the account attempting to read the file does not have the proper permissions, you must provide the account permission, and then resume hydration.

Does SQL Safe include new characteristics specific to the InstantRestore feature?

Yes, there are new details in SQL Safe to support InstantRestore. For supported platforms, the following components were added to SQL Safe to support InstantRestore:

SQL Safe Filter Service (SQL SafeFilterService.exe)

The SQL Safe Filter Service is responsible for handling I/O requests from SQL Server and performing the background database restore (hydration).

SQL Safe Filter Driver (SQL SafeFilterDriver.sys)

The SQL Safe Filter Driver is responsible for intercepting I/O requests for databases that have active InstantRestore operations under way. When an instant restore completes the driver totally disengages from all I/O activity of the database and is no longer needed. This device driver utilizes the Microsoft mini-filter driver technology.

Do you have to use the console for InstantRestore?

No, the Console is not the only place where you can use the InstantRestore feature. You can execute an InstantRestore via T-SQL script using either the SQL Safe CLI or XSP commands. To use the XSP InstantRestore command, see the sample XSP scripts available from the Programs menu.

Example CLI code snippets that use the InstantRestore command

You can also perform an instant restore through the CLI. Additional options can be set in the SQL Safe Restore wizard, from which you can generate a CLI script that includes the specified wizard settings.

```
SQLsafeCmd.exe InstantRestore <database> <full_backup> -diff <diff_backup> -log
<log_backup>
```

The following three options are specific to a backup set:

- BackupFile (if the backup set is striped).
- BackupSet.
- Password (or EncryptedRestorePassword).

Where these options appear in the command determines to which backup set they are applied. When you encounter one of these options, it is applied to the full if no -Diff/-Log option is yet encountered, otherwise it is applied to the most recent -Diff/-Log. For example, if you want to instantly restore the following backups:

- Full backup, 2 stripes, backupset 2, encryption key "full".
- Diff backup, 2 stripes, backupset 3, encryption key "diff".
- Log backup, 2 stripes, backupset 4, encryption key "log".

Use the command:

```
SQLsafeCmd InstantRestore Northwind "C:\Backup\Northwind_Full (1 of 2).safe" -BackupFile
"C:\Backup\Northwind_Full (2 of 2).safe" -BackupSet 2 -Password "full" -Diff "C:
\Backup\Northwind_Diff (1 of 2).safe" -BackupFile "C:\Backup\Northwind_Diff (2 of
2).safe" -BackupSet 3 -Password "diff" -Log "C:\Backup\Northwind_Log (1 of 2).safe"
-BackupFile "C:\Backup\Northwind_Log (2 of 2).safe" -BackupSet 4 -Password "log"
```

For more information about available instant restore options, see the usage statements in the CLI Help. To view the CLI Help, type `SQLsafeCmd help instantrestore`.

6.5.6 How script generation works

You can generate CLI and T-SQL scripts for backup and restore operations through the Backup and Restore wizards. SQL Safe generates the CLI or T-SQL script using the settings you specified for the backup or restore operation.

CLI scripts can be run as a batch file from the command line prompt. Generated CLI scripts use supported options for the backup and restore actions.

T-SQL scripts can be run through Query Analyzer or as a scheduled SQL Server job. Generated T-SQL scripts leverage the SQL Safe XSP to execute backups and restores.

If you need a command line or T-SQL script for your backup or restore, SQL Safe provides the Generate Script button to let you generate CLI and T-SQL scripts for these operations. When you use a wizard to run a backup or restore, SQL Safe disables this button until sufficient criteria exists to generate a script. SQL Safe generates the CLI or T-SQL script using the settings you specified for the backup or restore operation.

How do you generate script?

You can generate script through the Backup Wizard or the Restore Wizard once your settings provide SQL Safe with enough information to create the script. Click Generate Script, and SQL Safe displays command line script by default. Click the T-SQL button and SQL Safe displays the script in T-SQL format.

To retain your script in either format, click the Save to a file or Copy script to clipboard icon. SQL Safe also allows you to use normal select, cut, copy, and paste functionality directly on the displayed script.

6.5.7 How threads affect backups and restores

By default, SQL Safe automatically calculates the optimal number of threads necessary to process a backup or restore operation. You can calculate the number of threads for your environment based on the processors available on the computer running the SQL Server databases you want to backup. Consider performing several backups to find the appropriate number of threads for your environment. To calculate the appropriate number of threads for your environment, use the following guidelines. Also consider other loads on the SQL Server computer that may affect CPU performance and availability.

Number of CPUs	Number of Threads
Single processor	1
Multiple processors	(number of CPUs)-1

You can set the appropriate number of threads when backing up a database through the Management Console. You can customize the number of threads you want SQL Safe to use when performing a backup or restore. A similar number of threads used in each operation ensures that you achieve the same performance optimization for your backups and restores.

For SQL Server 2000 instances, selecting 12 or more threads can cause the backup operation to fail.

6.5.8 Recover objects using Virtual Database

With Virtual Database, you can create and manage Virtual Databases from backups created by native SQL Server and SQL Safe Backup. You can attach full backups or a series of backups. Virtual Databases behave just like actual live physical databases.

Virtual Database allows you to:

- Recover any object from the backup file without having to restore the database.
- Analyze and report on objects and permissions in backup files without having to restore the database.
- Access backup files as though they were read-only databases.

When creating a backup or backup policy, you can check the option to generate maps for InstantRestore and SQL virtual database. This metadata includes data files for each database included in your backup. Generating this metadata is optional; SQL Safe can mount a Virtual Database without the metadata. However, these metadata improves the performance during the virtual database creation.

You can create Virtual Databases from a single full backup file or multiple backup files. You can also create multiple Virtual Databases from the same backup file, which allows you to make Virtual Databases that include data from different points in time. Once created, the Virtual Databases can be fully managed and queried using Microsoft SQL Server Management Studio or another database management tool.

What is a Virtual Database?

Virtual Database is a powerful one-of-a-kind solution that lets you attach SQL Server backup files and query them like real databases. With its revolutionary, patent-pending technology, you gain instant access to critical data in a backup file without spending the time and storage previously required for restore. In minutes, you can create a Virtual Database and then use any native SQL Server or third-party tools to query and extract the data you need.

For more information about SQL Virtual Database, see working with Virtual Database.

Are there disk space recommendations for the Virtual Database metadata?

Use the following table to help you set aside the appropriate amount of disk space for the Virtual Database metadata SQL Safe generates. Typically, this metadata requires only a fraction of the disk space consumed by a fully restored backup.

Size of Backup	Additional Disk Space
1 TB	105 MB
500 GB	51 MB
100 GB	10 MB
1 GB	105 KB
500 MB	51 KB

6.6 View SQL Server status

The Group, Instance, and Database views display backup and restore details and operation status for all SQL Server instances registered with SQL Safe, as well as at-a-glance summaries of important administrative information. You can view information about a group of SQL Server instances, a single instance, or a database.

For this node ...	You can view ...
A server group	Displays the total number of instances in the group, the number of successful and failed operations, and the number of instances up and the number of instances not connected at the time.
An instance	Displays whether the connection to SQL Server is active, the number of successful and failed operations, whether the SQL Safe Backup Agent is running, the SQL Safe version, the license status, the number of databases on the instance, and the SQL Server version it is running. You can also find the backup and restore operations status for the last 7 days.
A database	Displays whether the database is currently online, the number of successful and failed operations, and the date of the last backup performed on the database. You can also find the backup and restore operations status for the last 7 days.

You can re-run any previous backup operation from these views. To re-run a backup, right-click the appropriate operation, and then select Back Up Again (executes backup using previous settings) or Back Up with Different Options (opens the Backup wizard). You can also quickly restore the backup files associated with a specific operation.

6.6.1 How can you manage your SQL Servers?

To manage your SQL Server instances, click **Servers** in the navigation pane, and then click the appropriate node in the Servers tree.

6.6.2 What information is available for your SQL Servers?

On the **Instance** view, you can find the following information:

- The Instance Information.
- Your Operation Status summary.
- A list of your Backup/Restore Operation Status.
- Your SQL Server status details.

6.6.3 View the Instance Information

The **Instance Information** section, displays your SQL Server instance information:

- The **Status** of your SQL Server.
- The **SQL Safe Version** under which it is running.
- The **License** for your SQL Server.
- The number of **Databases**.
- The **SQL Server Version**.

How do you edit the Settings?

The **Settings** option, helps you to install any options you did not include during your initial SQL Safe installation. The following options are available depending on your SQL Server circumstances:

- **Install SQL Safe Backup Agent / Upgrade SQL Safe Backup Agent** - opens a dialog box to install or upgrade the Backup Agent. You can select the checkbox option to include to your installation the SQL Safe Extended Procedures or include the Backup Service Install log. For more information, see install and configure the SQL Safe Backup Agent.
- **Install SQL Safe Extended Stored Procedures** - opens a dialog box to install SQL Safe Extended Stored Procedures. You can select the checkbox option to include to your installation the Backup Service Install log. For more information, see install SQL Safe Extended Stored Procedures.
- **Update license** - opens a dialog box to update the SQL Safe license.
- **Enable/Disable SQL Safe InstantRestore** - select this option to enable the InstantRestore and bring the database online quickly while the restore occurs in the background. You can also disable it by selecting this option. For more information, see enable/disable SQL Safe InstantRestore.
- **SQL Safe Agent Properties** - opens a dialog box where you can edit the SQL Safe Agent properties. For more information, see SQL Safe Backup Agent properties.

How do you refresh the Instance Information?

If a recent operation does not appear in the Instance Information pane, you can refresh the status of this pane by clicking the **Refresh** icon in the pane title bar.

6.6.4 View operations status summary

The **Operations** status summary displays a green success icon if the most recent backup or restore operation for each of the databases in the group or instance have been completed with success. When a failed operation is followed by a successful operation on the same database, the status is given as success. The number of successes and errors noted in the Operation Status Summary will always add to the number of databases in the group or instance.

6.6.5 View Backup/Restore Operation Status

The **Backup/Restore Operation Status** area displays a listing of all backup and restore operations performed for the selected object for the last 7 days. To change how much status information you see, click **Filter** and then select a different date **Range** in the **Event Time** settings.

You can re-run any previous backup operation from this grid. To re-run a backup, right-click the appropriate operation, and then select Backup Again (executes backup using previous settings) or Backup with Different Options (opens the Backup wizard). You can also quickly restore the backup files associated with a specific operation.

What column information can you select?

Column	Definition
Progress	During an operation, the progress bar will denote the percentage of the operation completed. When the operation is complete, it will display a green bar labeled 100%. If an operation completed with errors, this column will display a red bar labeled Error. If an operation completed with warnings, this column will display a yellow bar labeled 100% with an asterisk. This column also indicates when the backup file has been deleted (groomed), and therefore is no longer available to be restored.
Instance	Displays the name of the SQL Server instance that was backed up or restored by this operation.
Icon (Enhanced Restorability)	Displays an icon if the backup includes maps containing metadata for InstantRestore and SQL virtual database. For more information about InstantRestore, see How InstantRestore works. For information about SQL virtual database, see recover objects using SQL virtual database.
Database	Displays the name of the database that was backed up or restored by this operation.
Operation	Displays the type of operation performed. The types are Backup, Restore, and Verify.
Backup Type	Displays the type of the backup performed by the operation. The types are Full, Log, Differential, and File.
Compressed	Displays the size of the backup file after compression.
Ratio	Displays the ratio of the Uncompressed size of the database reported by SQL Server to the resulting Compressed size of the backup file created by SQL Safe.
Compression	Displays the type of compression used for the backup.
Database Size	Displays the size of the original database.
Uncompressed	Displays the size of data contained in the database, as reported by SQL Server.

Column	Definition
Encryption	Displays the type of encryption SQL Safe used during the backup operation.
Duration	Displays the time (hours:minutes:seconds) required to complete the operation.
Start Time	Displays the start date and time of the operation.
End Time	Displays the end date and time of the operation.
Threads	Displays the number of threads SQL Safe used during the backup operation.
Format	Displays the backup format. SQL Safe backup (Safe) or native backup (Bak) format.

How do you customize the columns in the grid?

Task	Action
Add or remove columns in the grid	Click Filter in the pane title bar, then select the columns you want to display in the grid.
Sort the content of a column	Click on the column header to sort the column in ascending order; click again to sort the column in descending order.
Rearrange the order of the columns	Click on the column header and drag it to a new position in the grid.
Group column headings	Click on the column header and drag it to a position beneath the column header by which it will be grouped.

How do you refresh the operations status?

If a recent operation does not appear in the status view, you can refresh the status of this pane by clicking the **Refresh** icon in the pane title bar.

Why is the Backup/Restore Operation Status grid blank?

SQL Safe only displays operation status information for Backup Agents running with an enterprise edition license. If the Backup Agent has a SQL Safe Lite or SQL Safe Freeware Edition license, this pane will be blank.

You may view the operation status for SQL Safe Lite or SQL Safe Freeware Edition Backup Agents by installing a purchased license. To use a trial before purchase, click Enable Trial License. For more information, see manage license.

6.6.6 View server status details

To see the detailed results of a specific operation, click the operation in the Backup/Restore Operation Status grid, and the Details area displays below. The **Details** area provides the following information:

Information	Definition
Statistics	Displays the database size, the size of the uncompressed backup, the size of the compressed backup, and the compression ratio achieved with this backup. The ratio is a measure of the storage savings achieved with SQL Safe compression technology. For more information about the storage space savings you can realize using SQL Safe, see understand your total cost of operation (TCO).
Result Text	Displays text describing the result of the operation.
Files	Displays the complete path of the backup set file for the backup or restore.
Backup Set Description	Displays the description you specified for this backup.
Storage Options	Displays the storage options you specified for this backup.

6.7 Perform a Manual Backup

The SQL Safe Backup Wizard allows you to:

- Backup multiple databases on different SQL Server instance.
- Backup multiple databases on the same SQL Server instance.
- Backup individual databases.

SQL Safe executes all of these operations in parallel.

To successfully backup data on a SQL Server instance, SQL Safe requires that you deploy a Backup Agent to the target instance. You can remotely deploy a Backup Agent through the Backup Wizard by registering the target instance. For more information, see install and configure the SQL Safe Backup Agent.

If SQL Safe detects that your instances are not licensed, go to **License Key Manager** and enable the license for that instance.

To backup multiple databases on a routine schedule, use a backup policy to maintain an up-to-date archives of your databases.

6.7.1 How do you create an archive using the Backup Wizard?

The Backup Wizard guides you through the steps required to archive your database content into backup sets. You can back up a single database, multiple databases, or an entire SQL Server instance.

You can create a backup with the SQL Safe Backup Wizard by accessing it from the following paths:

- Go to the task bar, click **Backup**.
- From any tab, go to the **File** menu and select **Backup**.
- Go to the **Common Tasks** bar of the **SQL Safe Today** view and click **Backup Database.**
- In the **Servers tree**, select the SQL Server instance or specific database you want to backup. Right click on it and select **Backup Database(s)**.

To get started with the SQL Safe Backup Wizard, follow the steps:

1. Select the database to backup.
2. Specify the backup type.
3. Select a location for the backup.
4. Configure options for manual backup.
5. Configure notifications for manual backup.
6. Review the details of the backup.

6.7.2 Select databases for manual backup

The **Databases** tab of the SQL Safe Backup wizard allows you to specify the instance that hosts the databases, and the specific databases you want to back up.

> ⚠ If your instance is not licensed, SQL Safe displays a warning message. Go to **License Key Manager** to enable the license for your instance.

What can you do on the Databases tab?

You can select the instance that hosts your target databases.

After you select the instance, the database list is populated. From the database list, select the databases you want to back up.

If you want to ...	Select this option ...
Back up all databases on the selected SQL Server instance	All Databases
Back up only User databases on the selected SQL Server instance	All User Databases
Back up only System databases on the selected SQL Server instance	All System Databases (master, model, msdb, distribution)
Back up only the databases you specify on the selected SQL Server instance	Specific Databases, and then choose the appropriate databases

Why is the target instance not listed?

The instance list only includes SQL Server instances that have been registered with SQL Safe. If the instance is not in the drop-down list, you can choose to add a new instance by clicking **Register SQL Server** . For more information, see register an instance.

> ✓ You can click **Refresh** to update the list of your databases if you do not see the current information.

Once you select the databases for your backup, click **NEXT** to select the backup type.

6.7.3 Select backup type

The **General** tab of the SQL Safe Backup wizard allows you to specify the backup type, name, and description of the backup you are creating.

What types of backups can you choose?

SQL Safe supports the standard SQL Server database backup types:

- Full Backup.
- Differential Backup.
- Transaction Log Backup.
- File Backup.

What should you do for your initial backup?

If you are backing up the database for the first time, select **Full Backup**. A full backup will provide a comprehensive data set, and is required to perform differential backups or transaction log backups later on. For more information about backup types, understand backup types.

What to choose in the Format drop-down option?

By default, the SQL Safe format is displayed in this option. Backups created using the SQL Safe file format can only be restored by SQL Safe. The SQL Server format is the native SQL Server backup file format. Backups created using the SQL Server format can be restored using SQL Safe and native SQL Server tools.

When should you specify a description?

You should provide a description to identify important details about this operation so you can easily identify which backup sets should be restored later. The backup description will appear in the status view of past and current backups, and will allow you to more easily identify problems when they occur.

How do you verify the integrity of your backup?

You can choose to **Verify Backup**. When this option is selected, SQL Safe performs a data integrity check after the backup has been created. SQL Safe only verifies the integrity of the data files in the backup set created by this backup.

Verifying the backup helps identify potential issues that could occur when restoring these data files.

What is a copy-only backup?

A **copy-only backup** is a copy of the database, not a true backup, and cannot be used as a part of a restore strategy or restore chain. It is a backup that does not affect the log sequence numbers (LSN) of the database.

Once you choose your backup type, click **NEXT** to select the location of your backup files.

6.7.4 Select location for manual backup

The **Locations** tab of the SQL Safe Backup wizard allows you to specify the backup location you want to use to store the backup set.

For a TSM backup, you can change the TSM connections settings to override the values set in the client options file if you need to write the backup files to a TSM Server other than the TSM Server already specified in the dsm.opt file.

Where can you store your backup set?

SQL Safe supports the following location types:

- Single File.
- Striped Files.
- Tape (using Tivoli Storage Manager).
- Data Domain.
- Amazon S3 Cloud.
- Microsoft Azure Cloud.
- Tape (using Tivoli Storage Manager) Striped Files.

What actions can you take with the location types?

Location Type	Actions
Single File	Enter the filename of the primary archive and select if you want to mirror archives.
Striped Files	Enter as many backup archive names as the number of striped files you want.
Tape (Tivoli Storage Manager)	Enter the filename for the High Level, Low Level, and the Management Class of the primary archive. Specify the TSM Client configuration file and its connection settings.
Data Domain	Enter the filename of the primary archive and select if you want to mirror archives.
Amazon S3 Cloud	Specify your Amazon S3 Cloud storage options to be used for your backup. For more information, go to Amazon Settings.
Microsoft Azure Cloud	Specify your Microsoft Azure cloud storage options to be used for your backup. For more information, go to Azure Settings.
Tape (using Tivoli Storage Manager) Striped Files	Enter the High Level filename and the Management Class for the primary archive. Specify the TSM Client configuration file and its connection settings. Also, the number of striped files you want to backup. ⚠ Take into account that if the number of stripes is greater than the available sessions on TSM server, the backup fails with a message "sessions are not available on TSM". There is no available way for the TSM client to find out available sessions on the TSM server.

What are striped files?

If you want to take advantage of distributing I/O overhead for a large database, select striped files, and select backup locations on different local disks.

How do you handle errors encountered while writing to the network during a backup?

Select **Enable network resiliency** and then click **Configure** to change the default settings. By default, SQL Safe will retry the backup operation every 10 seconds and then fail after 5 minutes (300 seconds) of continuous errors. Also, over the course of the operation, SQL Safe allows a total of 60 minutes in which to retry the backup before stopping the operation.

> This option is not available when backing up to tape using Tivoli Storage Manager or Amazon S3 Cloud.

What do you do if you do have an existing archive?

You can **Append** to an existing archived backup set or choose to **Overwrite** it.

What do you do if you do not have an existing archive file?

If you do not have an existing archive file, SQL Safe creates a new archive file that includes this backup set, using the name you specified.

> Keep in mind, the filename extension for all backups performed on SQL Safe format are .safe and for all backups performed on SQL server format are .bak.

How do you specify a UNC path?

To specify a UNC path, type the UNC path directly in the Filename field. You cannot specify a UNC path when using the  browse option.

> Using a UNC path allows you to restore backups to a different or new server from the original archive.

How do you mirror your backups?

Click **Mirror Archives**, and then specify where you want the mirror copies to be stored.

For each mirror, SQL Safe creates a copy of the backup archive set. You can specify up to 2 mirrors for each backup operation.

> Keep in mind that creating mirrors can impact the performance of your backup operation.

If you want to stop the backup operation when mirror location is unavailable, select the **Abort backup if a mirror location reports a failure**.

> This option is only available when backing up to Single File and Data Domain.

What do you specify when backing up to a TSM Server?

When a TSM location is selected, you must specify the following settings:

- High Level file path.
- Low Level file path.
- Management Class (optional).
- the location of the TSM Client options file that enables generate password for authorization.

You can change the TSM connections settings to override the values set in the client options file. You can also configure SQL Safe to mark these files as inactive after a specified age.

Note that SQL Safe accepts up to 260 characters for the TSM file path name.

Can you store backup files in a different TSM Client file space?

Yes. Under **TSM Client Settings**, specify the name of the node you want to use and the password required to access the node.

What settings can you change in the TSM Client options file?

Click **Change** to specify the node name, the password required to access the node, and the TCP/IP Server address and port.

What do you specify when backing up to Amazon S3 Cloud?

When the Amazon S3 Cloud location is selected, you must specify the following settings:

- Access Key.
- Secret Key.
- Region.
- Bucket Name.
- Subfolder(s) (optional).
- Filename.
- Part Size.

For more information, go to Amazon Settings.

What do you specify when backing up to Microsoft Azure Cloud?

When the Microsoft Azure location is selected, you must specify the following settings:

- Azure Storage Account Name.
- Azure Access Key.
- Sector Type.
- Container Name.
- Subfolder(s) (optional).
- Filename.
- Part Size.

For more information, go to Azure Settings.

Once you select the location of your backup files, click **NEXT** to configure options for your backup.

6.7.5 Configure options for manual backup

The **Options** tab of the SQL Safe Backup wizard allows you to select additional options, such as compression and encryption, to use for the current backup operation.

What types of compression algorithms are available?

- None.
- IntelliCompress, optimize for size (iSize).
- IntelliCompress, optimize for speed (iSpeed).
- Levels 1, 2, 3, 4.

A backup operation using Level 1 completes fastest but achieves the least amount of compression. Level 4 achieves maximum compression but the backup operation may take longer.

For more information about backup compression, see how to choose compression and encryption.

When performing a backup on a SQL Server format, the compression options available are: none and compress backup.

What types of encryption algorithms are available?

- None.
- AES (128-bit).
- AES (256-bit).

If your SQL Server environment requires FIPS compliance, use the AES encryption option. For more information, see ensure FIPS compliance.

When performing a backup on a SQL Server format, the encryption option is not available.

Does encryption require a password?

When you choose to encrypt an archive, you must designate a password. For security reasons, SQL Safe does not store this password. Ensure you remember the password you select.

What are the advanced options?

The following options are available as Advanced Options:

Options	Description
Number of threads	Allows you to specify how many threads you want SQL Safe to use to distribute the backup operation across multiple processors on the target SQL Server computer. Use this setting to optimize backup performance. Select **Auto** to have SQL Safe determine the optimal thread count for your environment.

Options	Description
Remove inactive transaction log entries	Removes all completed transactions from the transaction log after SQL Safe finishes the backup. This option is only available for Log backups.
Generate maps	Generates maps containing metadata for each database included in your backup file. Depending on the number of transactions completed since your last backup, generating maps may impact the performance of the backup operation. Generating maps is optional, but must exist in the backup file for InstantRestore to accept and restore that file. SQL virtual database can attach SQL Safe backup files without the metadata, but the data files improve SQL vdb performance during creation of the virtual database. For more information, see recover objects using SQL virtual database. This option is selected by default.
Include database logins in backup file	Copies SQL login information for the selected databases, including credentials and privileges, when the backup files are written. To help ensure the security of your SQL Server database, SQL Safe encrypts the login information. This option is available for full backups only.

When performing a backup on a SQL Server format, some advanced options are not available for this type of format.

What are the advanced options for SQL Server 2005 and later?

The following options are available as Advanced Options for SQL Server 2005 and later:

Options	Description
Generate checksums	Generates a checksum for the backup file.
Ignore checksum errors	Select this option to ignore any errors from the generated checksum. ***If checksum errors are encountered***, this option indicates that SQL Safe should continue to back up this database.
Read-write filegroups	Specifies a partial backup, which includes the primary filegroup and any read-write secondary filegroups. Read-write filegroups are not supported by SQL virtual database. ***If this option is selected***, the **Generate metadata** option will be disabled. Additionally, backups created with the read-write filegroups option cannot be used by SQL virtual database to create virtual databases.

What does the option Generate Script do?

You can generate a T-SQL or CLI script that will execute the backup you have defined in the wizard. For more information about generating scripts, see how script generation works.

Once you configure the options for your backup, click **NEXT** to configure notifications.

6.7.6 Configure notifications for manual backup

The **Notifications** tab of the SQL Safe Backup wizard allows you to email a status notification to the appropriate database administrators about this backup. Email notifications let you, and your staff, remotely monitor the status of your backups.

Choose the status you want to monitor, type the email address of each recipient, and then click **Next**.

You must configure your mail server settings before SQL Safe can send email notifications. Click **Configure E-mail** to check your settings. For more information, see configure e-mail settings for status notifications.

When is the email sent?

SQL Safe sends an email to the specified recipients only when the selected backup status occurs.

For example, if you chose to monitor whether the backup fails, you will not be emailed if the backup is skipped. Because you are performing a manual backup, you will receive one status notification.

Once you configure notifications for your backup, click **NEXT** to review details.

6.7.7 Review details for manual backup

The **Summary** tab of the SQL Safe Backup wizard provides the summary of specified values and options you have selected in the Backup Wizard.

What do you do next?

After you have reviewed the information on the Summary tab, click **Backup** to submit the backup job immediately, or click **Generate Script** to create a script you can use to run the job at a later time. For more information about generating scripts, see how script generation works.

How do you verify the status of your backup?

If you chose to run the backup job immediately, and want to verify a successful run, you can view its status using the **Instance View**. For more information, see view Backup/Restore operation status.

What actions can you perform on the Summary tab?

Action	Steps
Immediately backup databases	Click **Backup**, and then highlight the instance or database in the tree pane to see the status of the operation.
Create a CLI backup script	Click **Generate Script**, and then click **Command Line**. To **Save the script to a file**, click the saving icon or the **Copy script to clipboard** icon respectively. SQL Safe creates a backup script using the settings you specified for the selected databases. You can use this script to perform future backups of any system or user database you selected. Click **Close** to return to the Backup Wizard.
Create a T-SQL backup script	Click **Generate Script**, and then click **T-SQL**. To **Save the script to a file**, click the saving icon or the **Copy script to clipboard** icon respectively. SQL Safe creates a backup script using the settings you specified. You can use this script to perform future backups of any system or user database you selected. Click **Close** to return to the Backup Wizard. This script requires the SQL Safe XSP. For more information on installing the SQL Safe XSP, see deploy the SQL Safe XSP. For more information about how to use the SQL Safe XSP, see the sample scripts available from the Programs menu.

6.8 Perform a Manual Restore

SQL Safe allows you to restore multiple databases or files to any SQL Server instance you have registered. Ensure each registered instance is running the SQL Safe Backup Agent. Depending on your needs, you can restore databases from specific backup sets, or use the intuitive user interface to select specific points in time for each database you want to restore. When you select a time, SQL Safe automatically selects the appropriate backup sets that contain the data to be restored.

The Restore wizard will walk you through the restore process. Use the following checklist to ensure you have everything in place to restore your databases to the correct locations and to the correct points in time.

☑	**Follow these steps ...**
☐	Determine the location of the databases you want to restore to the SQL Server instance in question.
☐	Determine which SQL Server instance should host the recovered databases.
☐	For each database you need to recover, decide whether you will be restoring data from a specific backup set, or if you will restore data to a specific point in time. You can select the specific one in the Restore wizard.
☐	Determine whether you want and are able to use InstantRestore.

To restore a database on a routine schedule, use a restore policy to maintain an up-to-date copy of your database.

6.8.1 What does the Restore wizard do?

The SQL Safe Restore wizard allows you to simultaneously restore multiple databases on different SQL Server instances, restore multiple databases on the same SQL Server instance, or restore individual databases. SQL Safe also allows you to verify the integrity of a backed up database without restoring it.

6.8.2 What is InstantRestore?

InstantRestore allows you to quickly come back on line while restoring your database. It is important to understand InstantRestore fully before undertaking this type of database restore. Make sure you know the supported restore information before attempting to use the InstantRestore feature. Note that InstantRestore supports only complete database restores and does not support file or filegroup restores.

6.8.3 How do you restore a backup using the Restore Wizard?

You can restore your backups with the SQL Safe Restore Wizard by accessing it from the following paths:

- Go to the task bar, click **Restore**, and select Database(s) or Database files.
- From any tab, go to the **File** menu, select **Restore**, and choose Database(s) or Database files.
- Go to the **Common Tasks** bar of the **SQL Safe Today** view and click **Restore Database.**

- In the **Servers tree**, select the SQL Server instance or specific database you want to restore. Right click on it and select according to your needs either:
  - Restore Database(s) or
  - Restore Database Files

6.8.4 Restore Databases

When you select to **Restore Database(s)**, the SQL Safe Database Restore Wizard opens with the following sections:

- Target
- Databases
- Backup Sets
- Database Files
- Recovery State
- Restore Type
- Notifications
- Summary

Select target instance for restore

The **Target** tab of the Restore wizard allows you to you select the SQL Server instance where you want to restore the database(s). In this section you can follow these steps:

1. Select from the drop-down list the SQL Server instance where you want to restore your database(s). If your instance is not displayed, register it by using the Add Instance Wizard.
2. Choose one of the following actions:
    - **Restore** - select this option to restore your databases. You can select the option **Disconnect users before restore** to instruct SQL Safe to disconnect users from databases before performing the restore operation.
    - **Verify** - this restore option helps you ensure your backup operations are successful without actually restoring your data. Consider using this restore verification option on all critical backups after executing the backup operation.

What other Advanced Options do you have on this section?

You can specify the number of threads for compressing data or you can choose the **Auto** option so that SQL Safe calculates the optimum number of threads for your operation.

Once you select the target instance for your restore, click **NEXT** to select databases.

Select the databases you want to restore

The **Databases** tab of the Restore wizard allows you to specify the databases you want to restore and the general location of the corresponding archive files. You can select:

- **Repository** - use this option when the backup files reside in your repository. Choose the SQL Server where the database(s) to be restored were backed up, then select the databases you want to restore.
- **File System** - use this option when the archive file was written to the local File System. Type the path from the network share or local drive and click **ADD**. This path must be accessible by the Backup Agent installed on the Agent Computer.
- **Target Server** - use this option when a network share is available on a remote file system (Target Server). Type the respective path and click **ADD**.
- **Tivoli Storage Manager** - use this option when the backup was performed using TSM. Specify the TSM Path. Use the Browse option to find the correct database archive file. Specify the TSM Client Connection Settings. Click Change to override the values set in the client options file. Specify a different Node name and password, Server address and port. Specify the High and Low Level archive files. You can select the checkbox to include inactive files. You can also, find archives.
- **Amazon S3 Cloud** - use this option when the backup file is stored in Amazon S3 Cloud. You have to specify the Access Key, Secret Key, Region, Bucket Name, and Subfolder(s) (optional). For more information, go to Cloud Settings. You can also, click **Load Bucket** and **Load Backups** to access your files in your cloud storage account. Then select the files and/or databases you want to restore.
- **Microsoft Azure Cloud** - use this option when the backup file is stored in the Microsoft Azure Cloud. You have to specify the Azure Storage Account Name, Azure Access Key, Sector Type, Container Name, and Subfolder(s) (optional). For more information, go to Azure Settings. You can also, click **Load Files** and **Load Databases** to access your files in your cloud storage account. Then select the files and/or databases you want to restore.

Once you select databases for your restore, click **NEXT** to select Backup Sets.

Select the backup set for the restore

The **Backup Sets** tab of the Restore wizard allows you to choose which backup sets you want to use to restore the selected databases.

What information is on the Backup Sets tab?

For each database you have selected to restore, the backup set listing is populated with the available backup sets from the Repository. You can easily identify your SQL Safe backups or native backups (SQL Server) in the format column. You can choose the specific backup sets you want to restore, or you can select a point in time to which you want to restore data. To restore more than one database, select a backup set for each database.

If your backups have been done locally in a server that is different from the restore server, you may need to specify manually the location of these backup sets.

What do you do on the Backup Sets tab?

For each database you are restoring, you can choose one of three methods to select the appropriate backup sets:

- Time slider
- Manual selection of a specific point-in-time
- Manual selection of the backup set

What is the benefit of using the point-in-time slider?

When you select a point in time by clicking in the time slider, the corresponding backup sets are automatically selected. This ensures you are using the appropriate backup sets and files to restore the data you need. SQL Safe does not restore data time-stamped with dates later than the point in time you specify.

How do you use the point-in-time slider?

Depending on where you click, you can control which backups are used in this restore.

Click location	Result
Within a full backup marker	Selection of the full backup set.
Within a differential backup marker	Selection of the last known full backup, as well as the differential backup.
Above a transactional log marker	Slider snaps to the end of that log file and you will select the last known full backup and the entire transactional log file.
Within a transactional log marker	Selection of the last known full backup and all the transactions up to the specific point-in-time you clicked on.

You can set the amount of information displayed on the point-in-time slider by choosing information from the Last 30 days, Last 14 days, Last 7 days, Yesterday & Today, Today, and Custom Date Range.

Why would you select a specific point-in-time instead of using the slider?

Because the precision of the slider may not suit your needs, you can manually enter the specific point in time to restore in the **Selection Options**. This automatically selects the appropriate backup sets while providing pin-point time accuracy.

How do you manually select backup sets?

You can also choose to manually select which backup sets to restore. If you choose this option, you must also select the specific backup set to restore. To pick specific backup files, click the Backup Files from the list of Backup Set Names.

How do you select where your backup files are located?

Click **Edit File Paths** to display a list of backup files that you can use to restore the selected backup set. The Backup File Locations window displays the list of backup files SQL Safe uses to restore your backup sets. By clicking on the filename, you can edit it. You can also edit the path by clicking the button to the right of the path you want to change. Mirrored files also appear in the drop-down list if available. Click **OK** to apply your changes.

What are the encryption settings?

The option for **Encryption Settings** will be available when you choose a backup set where you have used the encryption option during its backup. Use this option to access these encryption settings.

How do you keep your restores running despite network errors?

Select **Enable network resiliency**, and then click **Configure** to change the default settings. By default, SQL Safe will retry the restore operation every 10 seconds and then fail after 5 minutes (300 seconds) of continuous errors. Also, over the course of the operation, SQL Safe allows a total of 60 minutes in which to retry the restore before stopping the operation.

Why is the Generate Script button enabled?

You can generate a T-SQL or CLI script that will execute the restore operation with the settings you have defined in this wizard. For more information about generating scripts, see how script generation works.

How do you use a mirrored file?

In some situations, SQL Safe creates a mirrored version of a backup file. You can use this file when the database you want is mirrored, and you cannot find the original because it was moved or deleted.

When SQL Safe tests file paths for a restore, and does not find the information it is looking for, a dialog box opens asking if you want to choose the same path and try again, or if you want to choose a different location for these files. If you choose to find a different path, SQL Safe displays the path of the failed file and allows you to choose another location. SQL Safe displays an icon to the left of the file name that alerts you of any file that failed the access test. Select the appropriate path, and then click**OK.**

Once you select the Backup Sets for you restore, click **NEXT** to specify your database files.

Specify a file name and location

The **Database Files** section of the Restore wizard allows you to rename a database by changing the name or move the location by editing the data file locations of the restored files.

What information is on the Database Files tab?

For each database you have selected to restore, you are required to specify the name for the restored database, and the filename to which you will restore the database.

What do you do on the Database Files tab?

For each database you are restoring, you have several ways to select the restored database name and path. You can:

- Select target database from drop-down list of existing databases
- Enter a new database name
- Enter a new database path
- Select restore options for these files
- Edit the filename of the restoring file

When you select a database name from the drop-down list, or edit the field, the **Restore As Filename** is automatically updated to reflect the new name, but you can edit this field by directly typing on the grid. You can also change the database path by simply editing the filename in the grid.

What actions can you perform on the Database Files tab?

Action	Steps
Create a new database to restore.	Type a new database in the **Restore As** text box.
Change the path of the target database	Enter a new path in the **Change path** field.
Ensure the selected backup files are restored, even if that means overwriting an existing database.	Select the **Force Restore** option (replace).
Restore the SQL logins associated with the selected databases	Select the **Restore database logins** option. This option is available when you are restoring a full backup that contains the database login information.
Restore databases that uses Microsoft SQL Server Change Data Capture (CDC) feature.	Select the **Keep CDC** option.
Ignore any errors from the generated checksum. ***If checksum errors are encountered***, SQL Safe should continue to restore the backup file.	Select the **Ignore checksum** errors option.
Retain the settings used when the selected databases were replicated.	Select the **Preserve replication** settings option.

Action	Steps
Include a temporary download location to restore very large files from cloud storage. Instead of restoring on the fly, this option allows you to download the file completely first and then perform a normal restore. This may help if you are running into memory or streaming issues when working with very large files. The file location is relative to the target server where the database will be restored to.	Select the **Download File from Cloud** option.

Can you overwrite an existing database?

To restore a database over an existing database, select the **Force Restore** option to ensure SQL Safe writes the selected backup files over the existing database.

Once you specify your database files, click **NEXT** to select the recovery state.

Select recovery state

The **Recovery State** tab of the Restore wizard allows you to choose the recovery state each database should be left in after the restore.

Fully accessible is the only recovery state supported by the InstantRestore feature. If you choose a partial recovery state, you cannot restore your database using InstantRestore. For additional information about performing an InstantRestore, see how InstantRestore works .

Which recovery states are supported?

SQL Safe supports the following recovery states:

- **Fully Accessible**. Leaves the database operational. No additional transaction logs can be restored. Note that you must use this fully-operational recovery state to use InstantRestore.
- **Not Accessible** (no recovery mode). Leaves the database non-operational, but able to restore additional transaction logs.
- **Accessible but read-only** (standby mode). Leaves the database read-only and able to restore additional transaction logs. You can specify an **Undo file** for this option.

Once you select the recovery state, click **NEXT** to select the restore type.

Select restore type

The **Restore Type** section of the Restore wizard allows you to choose whether you want to perform the Normal SQL Safe Restore or the SQL Safe InstantRestore when restoring your database.

⚠ Take into account when **InstantRestore** can or cannot be used:

- **InstantRestore** is not available for all restores as not all properties are supported; for example, you cannot restore a SQL Safe backup from a TSM Server.
- **InstantRestore** supports only complete database restores and does not support file or filegroup restores.
- **InstantRestore** can only work when you choose a Fully Accessible recovery state.
- **InstantRestore** does not support native format files.

What is the benefit of using InstantRestore?

In most cases, InstantRestore allows you to use the database almost immediately after starting the restore. If you have large databases that you need to access very quickly, this may be the best option for you, but take into account that there may be some performance issues if your users are making changes to the database while the restore is in progress.

InstantRestore will bring the database online quickly allowing you to access your data while SQL Safe continues to restore the database in the background.

What options do you have available on this section?

You can choose between a Normal SQL Safe Restore and a SQL Safe InstantRestore. If you choose the first one, SQL Safe will restore the database using the traditional restore engine and the database will become available when the restore completes. If you choose the InstantRestore option, the database will become available in a fraction of the time that a normal restore normally takes.

✓ Keep in mind, the SQL Safe InstantRestore option is disabled for restore operations where at least one backup file in the SQL Server file format is selected.

Once you select the restore type, click **NEXT** to configure notifications.

Configure notifications for manual restore

The **Notifications** section of the Restore wizard allows you to email a status notification to the appropriate database administrators about the restore operation. Email notifications let you, and your staff, remotely monitor the status of your restores.

Choose the status you want to monitor, type the email address of each recipient, and then click **Next**.

> You must configure your mail server settings before SQL Safe can send email notifications. Click **Configure E-mail** to check your settings. For more information, see configure e-mail settings for status notifications.

When is the email sent?

SQL Safe sends an email to the specified recipients only when the selected restore status occurs.

Once you configure notifications, click **NEXT** to review details.

Review details for manual restore

The **Summary** tab of the restore wizard provides the summary of specified values and options you have selected in the Restore Wizard.

What do you do next?

After you have reviewed the information on the Summary tab, click **Restore** to submit the restore job immediately, or click **Generate Script** to create a script you can use to run the job at a later time.

If you choose to run the restore job immediately, and want to verify a successful run, you can view its status in the Instance view. For more information, see view Backup/Restore operation status.

What actions can you perform on the Summary tab?

Action	Steps
To immediately restore databases	Click **Restore**, and then highlight the instance or database in the tree pane to see the status of the operation.
To create a CLI restore script	Click **Generate Script**, and then click **Command Line**. To save the script to a file, click the save icon or the copy script to clipboard icon. SQL Safe creates a restore script using the settings you specified for the selected databases. You can use this script to perform future restores of any system or user database you selected. Click **Close** to return to the Restore Wizard.
To create a T-SQL restore script	Click **Generate Script**, and then click **T-SQL**. To save the script to file, click the save to a file icon or the copy script to clipboard icon. SQL Safe creates a restore script using the settings you specified. You can use this script to perform future restores of any system or user database you selected. Click **Close** to return to the Restore Wizard. This script requires the SQL Safe XSP. For more information on installing the SQL Safe XSP, see deploy the SQL Safe XSP. For more information about how to use the SQL Safe XSP, see the sample scripts available from the Programs menu.

How do you verify the status of your restore?

If you want to verify that your restore operation successfully ran, you can view its status using the Instance view. For more information, see view Backup/Restore operation status.

6.8.5 Restore Database Files

When you select to perform a **Restore** to your **Database Files**, the SQL Safe Database File Restore Wizard opens with the following options:

- **Target** - in this section, you can select the SQL Server instance where you want to restore. Go to select target instance for more detailed information.
- **Databases** - specify the databases you want to restore and the general location of the corresponding archive files. Go to select databases for more detailed information.
- **Backup Sets** - choose which backup sets you want to use for restore. Go to select the backup set for more detailed information.
- **Database Files** - you can rename a database by changing the name or move the location by editing the data file locations of the restored files. Go to specify database files for more detailed information.
- **Recovery State** - choose the recovery state each database should be left in after the restore. Go to select the recovery state for more detailed information.
- **Notifications** - email a status notification to the appropriate database administrators about the restore operation. Go to configure notifications for more detailed information.
- **Summary** - before performing your **Restore**, you can go **Back** to review and make any necessary modifications or **Generate a Script**. Go to review details for more detailed information.

6.9 Automate Backups and Restores

A SQL Safe policy consists of a set of databases for which a set of disaster recovery operations will be performed according to a defined schedule. You can use policies to enforce corporate standards or Service Level Agreement (SLA) requirements.

SQL Safe allows you to perform the following policies:

- Create a backup policy - backup policies allow you to quickly and easily schedule backups for large sets of databases that have similar needs.
- Create a log shipping policy - log shipping policies allow you to schedule the synchronization of transaction logs between a primary database and one or more secondary databases.
- Create a restore policy - restore policies allow you to schedule the routine recovery of a specific database.

6.9.1 How do you access the Policies status?

To view any policy status, click **Policies** in the navigation pane, and then select the appropriate policy listed in the tree pane. SQL Safe provides an at-a-glance record of your policy statuses.

You can view information about all your policies (per type) or view the status of an individual policy. You can also create new policies, edit existing policies, view your policy settings, or disable them.

6.9.2 Backup policies

Backup policies allow you to define backup maintenance plans across multiple SQL Server instances in your enterprise. These instances can reside on one or more physical servers.

What is a backup policy?

A backup policy consists of a list of databases you want to back up, a set of backup operations to be performed on those databases, and a set of schedules according to which the backups will be performed. You can choose to create the associated jobs to run on a specific schedule, run on demand (execute the jobs manually from the Management Console), or you can choose to define your policy for monitoring purposes only. You can then monitor the status of each backup, all from a single point of contact in the Management Console.

How do you incorporate backup strategies in your policies?

Implementing a policy requires that you have a clear understanding of your backup strategy. To determine a backup strategy to use, consider the following recovery model requirements.

Model	Full Backup?	Differential Backup?	Transaction Log Backup?	File or Filegroup Backup?
Simple Model	Required	Optional	N/A	N/A
Full Model	Required	Optional	Required	Optional
Bulk-Logged Model	Required	Optional	Required	Optional

What constitutes a good backup strategy?

Consider using all four backup types to maximize your recovery and minimize your data loss. A basic backup strategy fulfills the following needs:

1. Creation of regularly scheduled database backups.
2. Creation of frequent differential backups between full backups.
3. Creation of transaction log backups more frequently than differential backups.

Database backup creation depends on server activity and data sensitivity. Ensure you implement a strategy and create policies that back up both user databases and system databases.

How do backup policies help you?

Backup policies allow you to plan and schedule your SQL Server backup maintenance, as well as monitor its success and failures, all from a single point of contact at the Management Console. By allowing the application and scheduling of a set of backup operations across all of your SQL Server instances enterprise-wide, SQL Safe policies make updating your maintenance plans a quick and easy process.

To create backup policies, see create a backup policy.

To view the status of your policies, see view the status of all backup policies or view status of a specific backup policy.

Create a backup policy

The SQL Safe Backup Policy Wizard allows you to create backup maintenance plans across your enterprise. A SQL Safe Backup Policy is defined as a set of databases for which a set of backup operations will be performed according to a defined schedule. If you choose to create backup jobs for this policy, SQL Safe creates SQL Server jobs for the specified backups.

How do you access the Backup Policy Wizard?

You can access the Backup Policy Wizard from any of the following paths:

- Go to the task bar, click **Create Policy** and then choose **Backup Policy**.
- On the Policies tab, right-click the Backup Policies folder and select **Create Backup Policy**.
- On the Policies tab, right-click one of your Backup Policies and select **Create Backup Policy**.
- On the Policies tab, click **Create New Policy** located on the **Operation Summary** section of the **Backup Policies Status** window. This option is only available before you create your first backup policy.
- From any tab, go to the **File** menu, select **Create Policy** and then **Backup Policy**.
- You can also find this option on the **SQL Safe Today** view, by going to the **Common tasks** and then selecting **Create Backup Policy**.

To get started with the Backup Policy Wizard:

1. Name the policy.
2. Select the databases you want to back up.
3. Select backup options.
4. Specify where you want to store the backup files.
5. Schedule when and how often you want the backup to occur.
6. Get notifications about the policy status.
7. Review details.

Name the policy

The **General** tab of the SQL Safe Backup Policy Wizard allows you to specify the basic properties of the backup policy.

Why should you specify a name or description?

You are required to enter a unique name for each policy.

Both the name and description will appear in the status messages for your policies. Using a meaningful name and description will allow you to more easily identify problems when they occur. For example, consider specifying a description that will help you later choose the correct backup to restore during a disaster recovery situation.

Which format should you choose to perform your backup?

Depending on the needs of the backup policy. The option to select between SQL Safe file format and native SQL Server file format is available. The *Format* option provides two options, "SQL Safe" and "SQL Server". By default, SQL Safe option is selected.

For native SQL Server file format select "SQL Server". For SQL Safe file format, which is compatible with InstantRestore, select "SQL Safe".

Which policy action should you choose?

Choose the action that best reflects how you want to use this policy. You have multiple options in this section:

- Monitor and automatically create backup jobs using the SQL Server Agent
- Monitor and automatically create backup jobs using the SQL Safe Backup Agent
- Monitor only

The first option allows you to create the policy for monitoring database backups and automatically creates the backup jobs using the SQL Server Agent on your SQL Server instances. Creating jobs allows to enforce consistent backup settings across your environment.

The second option allows you to create the policy for monitoring database backups and automatically creates backup jobs using the SQL Safe Backup Agent which is in charge of executing and scheduling these policies.

The third option creates the policy only to monitor database backups and no jobs are created. By default, SQL Safe will monitor the status of any backup operation that meets the criteria of your policy.

Note that SQL Server Express does not support the SQL Server Agent. As a result, you must use the second option if you add any instance with SQL Server Express in your policy. This option allows the SQL Safe Backup Agent (second option) to create your policy backup jobs.

If you choose to use the SQL Safe Backup Agent, policy data files will be stored by default at *C:\Program Files\IDERA\SQL Safe\PolicyData*. You can change these settings by going to the **Policy Data** tab on the management console preferences and selecting or creating the folder directory where you want to store these files.

Once you define some policy settings, click **NEXT** to select your databases.

Select databases

Use the **Membership** tab of the SQL Safe Backup Policy Wizard to select which SQL Server instances and databases you want to monitor with this policy. You can also exclude one or more databases from this policy.

To make your selections, click **Add/Remove Instances**, and then choose the instances from where you want to backup databases. Then select from the different options for choosing databases (**All Databases**, **All User Databases**, **All System Databases**, **Specific Databases**). By choosing one of the "All" database options, the policy will automatically include the relevant databases as they are added or removed on the server.

- You may select databases from different servers for one policy.

- When you choose the databases you need to backup you can also specify those ones that you want to exclude from your policy, for this option move the ones you want to exclude to the list on the left.

Once you select your databases, click **NEXT** to configure your backup options.

Configure options

The **Options** tab of the SQL Safe Backup Policy Wizard allows you to enter the backup types and options for each operation included the backup policy.

What information is on this tab?

For each backup operation you include in the backup policy, you can select compression, encryption, and verification options, enable object-level recovery, and set additional advanced options such as removing inactive entries from the transaction log.

What types of backup can you choose?

On the Options tab you can choose the types of backups you want for your backup policy. You can specify one, two or the three types of backup: **Full**, **Differential**, **Log**, just select the respective backup types and provide their settings.

Why can't you see the options for all the backup types?

The options for each backup type are hidden unless the backup type is selected for the policy. For more information about backup types, see understand backup types.

What types of compression algorithms are available?

- None
- IntelliCompress, optimize for size (iSize)
- IntelliCompress, optimize for speed (iSpeed)
- Levels 1, 2, 3, 4

Backup operations using Level 1 complete fastest but achieve the least amount of compression. Level 4 achieves maximum compression but the backup operation may take longer.

For more information about backup compression, see how to choose compression and encryption.

When performing a backup on a SQL Server format, the compression options available are: none and compress backup.

What types of encryption algorithms are available?

- None
- AES (128-bit)
- AES (256-bit)

If your SQL Server environment requires FIPS compliance , use the AES encryption option. For more information, see ensure FIPS compliance.

When you choose to encrypt an archive, you must designate a password. For security reasons, SQL Safe does not store this password. Ensure you remember the password you select.

When performing a backup on a SQL Server format, the encryption option is not available.

What additional options are available?

For each type of backup you select, you can also specify the following advanced options:

Options	SQL Safe	SQL Server	Description
Verify the integrity of the backup when complete	Available	Available	• Performs a data integrity check after the backup is created. SQL Safe verifies the integrity of the data files in the backup set created by this backup. • Verifying the backup helps identify potential issues that could occur when restoring these data files.
Generate maps	Available	Not available	• Generates maps containing metadata for each database included in your backup file. • Depending on the number of transactions completed since your last backup, generating maps may impact the performance of the backup operation. • Generating maps is optional, but must exist in the backup file for InstantRestore to accept and restore that file. • SQL virtual database can attach SQL Safe backup files without the metadata, but the data files improve SQL vdb performance during creation of the virtual database. • For more information, see recover objects using SQL virtual database. This option is selected by default.
Report T-Log operations that are skipped as SUCCESS	Available	Available	• Allows SQL Safe to report SKIPPED T-Log operations as SUCCESS. • Avoids backup policies from reporting a warning status when T-Log operations are skipped for databases that are in simple recovery.

Options	SQL Safe	SQL Server	Description
Include database logins in backup file	Available	Not available	• Copies SQL login information for the selected databases, including credentials and privileges, when the backup files are written. • To help ensure the security of your SQL Server database, SQL Safe encrypts the login information. This option is available for full backups only.
Thread Count	Available	Not available	• Allows you to specify how many threads you want SQL Safe to use to distribute the backup operation across multiple processors on the target SQL Server computer. Use this setting to optimize backup performance. • When the resultant backup file is restored, SQL Safe uses the same thread setting to ensure consistent performance. • Select **Auto** to have SQL Safe determine the optimal thread count for your environment.
Transaction Log	Available	Available	• Removes all completed transactions, inactive entries, from the transaction log after SQL Safe finishes the backup.

Options	SQL Safe	SQL Server	Description
Checksum: Generate	Available	Available	• Generates a checksum for the backup file.
Checksum: Ignore Errors	Available	Available	• Ignores any errors from the generated checksum. • If checksum errors are encountered, this option indicates that SQL Safe should continue to back up this database.
Backup: Copy only	Available	Available	Specifies a copy-only backup. This is a copy of the database and cannot be used as part of a restore strategy. It will allow you to take a "snapshot" backup of your database without interferring the LSN (log sequence number) order of your backup strategy.
Backup: Read-write filegroups	Available	Available	• Specifies a partial backup, which includes the primary filegroup and any read-write secondary filegroups. Read-write filegroups are not supported by SQL virtual database. • If this option is selected, the **Generate metadata** option (Generate maps for InstantRestore and SQL virtual database) will be disabled. • Backups created with the read-write filegroups option cannot be used by SQL virtual database to create virtual databases.

Once you configure options for your backup, click **NEXT** to select your backup location.

Select location

The **Locations** tab of the SQL Safe Backup Policy Wizard allows you to specify the backup location for each operation you include in the backup policy.

What information is on the Locations tab?

For each operation you have included in the backup policy, you can specify the location type, full path in which to store the backup file, an optional housecleaning schedule for existing disk archives, and the backup file extension.

What types of backup locations can you use?

SQL Safe supports the following location types:

- Backup to a single file on the local computer or a network share.
- Backup to multiple striped files on the local computer or a network share.
- Backup to tape using Tivoli Storage Manager.
- Backup to Data Domain.
- Backup to Amazon S3 Cloud.
- Backup to Microsoft Azure Cloud.
- Backup to Tape using Tivoli Storage Manager striped files.

What do you do if you don't have an existing archive?

If you do not specify an existing archive, SQL Safe creates a new backup set with the name you specify. The location entered for each backup type must be valid for all SQL Server instances.You can choose to **Append** or **Overwrite** if the archive already exists.

What accounts can you specify to access the backup files location?

Depending whether you selected to use the SQL Server Agent or the SQL Safe Backup Agent for the scheduling of your policy, on this section you have the option to select between SQL Server Agent service account/SQL Safe Backup Agent service account respectively or a Windows account. Click **Account** and select your preferred option.

> ⚠ The account specified must have read and write privileges on the directory selected for your backup file location.

How do you keep my backups running despite network errors?

Select **Enable network resiliency** and then click **Configure** to change the default settings. By default, SQL Safe will retry the backup operation every 10 seconds and then fail after 5 minutes (300 seconds) of continuous errors. Also, over the course of the operation, SQL Safe allows a total of 60 minutes in which to retry the backup before stopping the operation.

This option is not available when backing up to tape using Tivoli Storage Manager.

Can you change the default file locations?

SQL Safe automatically populates the path using several available variables, depending on location type. You can modify this path to suit your needs, taking advantage of all the available variables.

For a disk backup, browse for or enter the directory in which to store the backup file. You can use the supplied macros in the way best suited to your storage needs. If you want to limit the lifetime of your backup sets created by the policy, you can select the option that removes files older than the specified time.

For a TSM backup, browse for or enter the high level directory for the tape file. You can use the supplied macros in the way best suited to your storage needs. Browse for or enter the location of the TSM configuration file.

Keep in mind, the filename extension for all backups performed under the SQL Safe format are .safe and for all backups performed under the SQL server format are .bak.

What does removing old files do?

For backups written to a single file or mirrored files, you can choose to remove old files to prevent disk space limitations. When you select to remove files older than the specified time, backup files created with names of the same format will be deleted from that directory. You can configure SQL Safe to delete old backup files from the primary archive as well as from your mirror archives.

For backups written to a TSM Server, you can configure SQL Safe to mark these files as inactive after a specified age.

How do you mirror the backups this policy creates?

Click **Mirror Archives** , and then specify where you want the mirrored files to be stored.

For each mirror, SQL Safe creates a copy of the backup archive set. You can specify up to 2 mirrors for each backup operation.

Keep in mind that creating mirrors can impact the performance of your backup operation.

If you want to stop the backup operation when mirror location is unavailable, select **Abort backup if a mirror location reports a failure**.

What do you specify when backing up to a TSM Server?

When a TSM location is selected, you must specify the following settings:

- the location of the TSM Client configuration file that enables generate password for authorization
- High Level file path
- Low Level file path
- Management Class (optional)

You can also configure SQL Safe to mark these files as inactive after a specified age.

Note that SQL Safe accepts up to 260 characters for the TSM file path name.

SQL virtual database is not available when backing up to a TSM Server.

What do you specify when backing up to Amazon S3 Cloud?

When the Amazon S3 Cloud location is selected, you must specify the following settings:

- Access Key
- Secret Key
- Region
- Bucket Name
- Subfolder(s) (optional)
- Filename

- Part Size

For more information, go to Amazon Settings.

What do you specify when backing up to Microsoft Azure?

When the Microsoft Azure location is selected, you must specify the following settings:

- Azure Storage Account Name
- Azure Access Key
- Sector Type
- Container Name
- Subfolder(s) (optional)
- Filename
- Part Size

For more information, go to Azure Settings.

Once you determine your backup location, click **NEXT** to configure a schedule for your backup operations.

Configure schedule

The **Schedules** tab of the SQL Safe Backup Policy Wizard allows you to schedule the frequency and duration of your backup operations. For each backup type, enter the appropriate information into each of the schedule fields to satisfy your backup requirements.

What information is on the Schedules tab?

For each operation you have included in the policy, you can specify when your operation will begin, how frequently backup jobs will be executed, and the respective duration for these operations. You can also choose to run the operation "On Demand," allowing you to easily manually execute the associated jobs according to your preset options.

How do you know what frequency to set?

The schedule of your operations should be determined by how much data you can afford to lose in the event of a catastrophic failure. The schedule should be developed in concert with your backup strategy. For example, for lab or development instances, you may want to schedule on-demand or weekly backups whereas for critical production instances you may want to schedule full backups every day with transaction log backups every hour.

Can you set a different schedule for each backup operation?

Each backup operation can have a different schedule. For instance, perhaps you decide you want to run full backups monthly, differential backups once a week, and transaction logs during business hours every day.

How do you set the schedule?

You can set up a schedule by defining the following options:

Field	Description
Occurs	Unit of Frequency: • On Demand • Daily • Weekly • Monthly
Daily Frequency	Time of day: • Occurs once at `HH:MM:SS AM/PM` • Occurs every `N` Hours/Minutes starting at `HH:MM:SS AM/PM`, ending at `HH:MM:SS AM/PM`
Duration	Length of time: • Start date `mm/dd/yyyy` • End date `mm/dd/yyyy` • No end date

When an operation does not occur as scheduled, the backup policy will consider it "missed" and SQL Safe can notify you about it if you configured it to do so.

Once you configure the schedule of your backup operations, click **NEXT** to configure notifications.

Configure notifications

The **Notifications** tab of the SQL Safe Backup Policy Wizard allows you to choose the backup statuses about which you want to receive alert notifications through email. Email notifications let you, and your staff, remotely monitor the status of the backups you have automated with this policy.

The status of the backup operations determine the status of your policy.When your backups are successfully completed on scheduled, the policy is considered ok.

Choose the status you want to monitor, type the email address of each recipient, select the desired alert frequency for each operation, and then click **Next**.

You must configure your mail server settings before SQL Safe can send e-mail notifications. Click **Configure E-mail** to check your settings. For more information, see configure e-mail settings for alert notifications.

When is the email sent?

SQL Safe sends an email to the specified recipients when the selected operation status occurs. Because SQL Safe checks the status of your backup operations every minute, your alert notifications provide a real-time indication of the health of your service level agreements and disaster recovery plans for the SQL Server instances covered by this policy.

However, how often you are emailed about a specific status update depends on the notification frequency setting you select. For example, if you want to receive an email whenever a backup fails, even when the failures occur sequentially, choose to receive notifications every time the event occurs.

Once you configure notifications, click **NEXT** to review details.

Review details

The **Summary** tab of the SQL Safe Backup Policy Wizard provides the summary of specified values and options you have selected in the Backup Policy wizard. After you review the information on the Summary tab, click **Finish** to create the policy and corresponding backup jobs.

If you want to create the policy but not the backup jobs, return to the General tab and select the **Monitor Only** action.

View the status of all backup policies

When **Backup Policies** is selected in the **Policies** tree pane, the content pane displays information describing the overall status of all of these policies. Use this view to quickly determine whether your servers are in compliance with your corporate backup policies.

What does the Current Status mean?

The **Current Status** displays the most recent, combined status of all operations performed by your backup policies. Even though there are multiple operation statuses, the overall policy status reflects the most critical operation status. When all backups have been completed successfully according to the policy schedule, a green OK icon is displayed.

What is the Last Operation Status?

The **Last Operation Status** shows an overview of the most recent backup, restore, or log shipping operations that occurred across your enterprise. Use this view to quickly determine whether your servers are in compliance with your corporate policies.

What is the Operation Summary?

The **Operation Summary** displays a list of all policies, providing information in the following columns:

Column Header	Definition
Status	Displays either a green OK status bar, a yellow warning status bar, or a red error status bar.
Name	Displays the policy name.
Databases Covered	Displays the number of databases being backed up by the policy.
Last Operation	Display the date and time of the most recent backup operation (of any type defined by the policy).
Last Operation Failure	Displays the date and time of the most recent backup failure (of any type defined by the policy).

How do you get details about a specific policy?

You can get more details about the status of a specific policy by double-clicking on one of the policy operations in the **Operation Summary** grid or by choosing the respective policy on the Backup Policies folder tree node.

Can you customize the columns in the grid?

You can sort the content of any of the columns by clicking on the column header.

How do you refresh the operations status?

If a recent operation does not appear in the operation summary view, you can refresh the status of this pane by clicking the **Refresh** icon in the pane title bar.

View status of a specific backup policy

When a specific backup policy is selected from the **Backup Policies** tree node, the content pane displays information describing the status of that policy. Use this view to determine which backup operations initiated by the policy have succeeded or failed.

What actions can you perform?

From the Policies tree

By right-clicking on a policy under the **Backup Policies** node, you can access the following shortcuts:

Action ...	**What it does ...**
Create Backup Policy	Opens the Backup Policy wizard, allowing you to create a new policy.
Edit Policy	Opens the Backup Policy wizard (with all options pre-set to the values used for this operation), allowing you to edit any of the options.
Delete Policy	Allows you to delete the policy. Although backup operations associated with this policy will no longer be performed, the previous backup files and status messages created by this policy will continue to be stored in the SQL Safe Repository.
Disable Policy	Allows you to disable the selected policy. Backup operations associated with this policy will no longer be performed and will turn off any email notifications you have configured to alert on the backup or restore status.
Start Jobs for Policy	Allows you to run the backup jobs associated with this policy, performing an ad-hoc backup with the options already set by the policy.
View Out of Date Jobs	Allows you to view jobs that are out of date.
Update Out of Date Jobs	Allows you to update the list of out of date jobs.
Refresh Policy List	Updates the Backup Policies node with the latest policies.

From the Current Status pane

By clicking the links available in the **Current Status** pane, you can access the following shortcuts:

Action ...	What it does ...
View Policy Settings	Allows you to view a summary of the policy settings.
Edit Policy	Opens the Backup Policy wizard, allowing you to change your policy settings.
Disable Policy	Disables the selected policy. Once a policy is disabled, it will no longer perform backup operations for the associated databases. To back up a database that belongs to a disabled policy, perform a manual backup using the Backup Wizard.
Start Full Backups	Performs a full backup of all databases that belong to this policy by running the corresponding job. This action applies your previously defined policy settings, and is only available when your policy includes a full backup operation.
Start Diff Backups	Performs a differential backup of all databases that belong to this policy by running the corresponding job. This action applies your previously defined policy settings, and is only available when your policy includes a differential backup operation.
Start Log Backups	Performs a transaction log backup of all databases that belong to this policy by running the corresponding job. This action applies your previously defined policy settings, and is only available when your policy includes a log backup operation.

From the Operation Details grid

By right-clicking a backup operation, you can access the following shortcuts:

Action ...	What it does ...
Cancel	Allows you to cancel the backup policy.
View Details	Shows the Details pane, providing additional information about the selected backup operation such as Statistics, Result text, Files, Backup Set Description, Storage Options.
Back up again	Runs the backup operation again, using the same settings.
Back up with Different Options	Opens the Backup wizard (with all options pre-set to the values used for this operation), allowing you to specify different options before running the operation.
Verify backup	Verifies that the backup file is "good" and can be restored with all data intact. When you perform this operation a new operation is listed. If you right click Verify of the Operation column, you can Verify Again, Verify with Different Options and Set Progress to different statuses. When you select Verify with Different Options, the SQL Safe Database Restore Wizard opens with the Verify option enabled. Use this wizard to set specific settings for this verify operation.
Restore database	Opens the Restore wizard, allowing you to restore this backup file.

Action ...	What it does ...
Set Progress To	Allows you to change the status of the selected operation to Complete, Complete, Warning, Error, Canceled, Skipped or Deleted.

What does the Current Status mean?

The **Current Status** displays the most recent, combined status of the backup operations performed by this policy. When there are multiple operation statuses, the policy status reflects the most critical operation status. When all backups have been completed successfully according to the policy schedule, a green ok icon is displayed.

What is the Last Operation Status?

The **Last Operation Status** shows an overview of the most recent backup occurred across your enterprise. Use this view to quickly determine whether your servers are in compliance with your corporate policies. The operation status is limited to backup operations performed by this policy. Click any backup operation to see more details about it, including the reasons for failed or skipped backup operations if applicable.

What are the Operation Details?

The **Operation Details** grid displays a listing of all backup and restore operations performed for the databases included in the selected policy for the last 7 days. This grid includes the following columns:

Column	Definition
Progress	During an operation, the progress bar will denote the percentage of the operation completed. When the operation is complete, it will display a green bar labeled 100%. If an operation completed with errors, this column will display a red bar labeled Error. If an operation completed with warnings, this column will display a yellow bar labeled 100% with an asterisk. This column also indicates when the backup file has been deleted (groomed), and therefore is no longer available to be restored.
Instance	Displays the name of the SQL Server instance that was backed up or restored by this operation.
Icon (Enhanced Restorability)	Displays an icon if the backup includes maps containing metadata for InstantRestore and SQL virtual database. For more information about InstantRestore, see How InstantRestore works. For information about SQL virtual database, see recover objects using SQL virtual database.
Database	Displays the name of the database that was backed up or restored by this operation.
Operation	Displays the type of operation performed. The types are Backup, Restore, and Verify.
Backup Type	Displays the type of the backup performed by the operation. The types are Full, Log, Differential, and File.
Compressed	Displays the size of the backup file after compression.
Ratio	Displays the ratio of the Uncompressed size of the database reported by SQL Server to the resulting Compressed size of the backup file created by SQL Safe.

Column	Definition
Compression	Displays the type of compression used for the backup.
Database Size	Displays the size of the original database.
Uncompressed	Displays the size of data contained in the database, as reported by SQL Server.
Encryption	Displays the type of encryption SQL Safe used during the backup operation.
Duration	Displays the time (hours:minutes:seconds) required to complete the operation.
Start Time	Displays the start date and time of the operation.
End Time	Displays the end date and time of the operation.
Threads	Displays the number of threads SQL Safe used during the backup operation.
Format	Displays the backup format. SQL Safe backup (Safe) or native backup (Bak) format.

Can you customize the columns in the Operation Details grid?

You can sort the content of any of the columns by clicking the column header.

You can select which columns are visible in this grid, and enable column grouping, by clicking the **Filter** icon in the pane title bar.

How do you refresh the data displayed in the Operation Details grid?

Yes. If a recent operation does not appear in the status view, you can refresh the status of this pane by clicking the **Refresh** icon in the pane title bar.

What are the details?

To see the detailed results of a specific operation, click the operation in the **Operation Details** grid.
The **Details** pane displays below. By default, this pane is hidden.

The **Details** pane provides the following information about the selected backup operation:

Tab	Description
Statistics	Displays the database size, the size of the uncompressed backup, the size of the compressed backup, and the compression ratio achieved with this backup. The ratio is a measure of the storage savings achieved with SQL Safe compression technology. For more information about the storage space savings you can realize using SQL Safe, see understand your total cost of operation (TCO).
Result Text	Displays text describing the result of the backup.
Files	Displays the complete path of the backup set file for the backup.

Tab	Description
Backup Set Description	Displays the description you specified for this backup.
Storage Options	Displays which locations were chosen to store the backup files.

6.9.3 Log shipping policies

Log shipping policies allow you to ship transaction logs between multiple SQL Server instances in your enterprise, on a scheduled basis. These instances can reside on one or more physical servers.

What is a log shipping policy?

A log shipping policy consists of primary and secondary databases you want to synchronize, a set of transaction log backup and restore operations to be performed on those databases, and a set of schedules according to which these operations will be performed. You can also choose to mirror the backup files, storing copies of the transaction logs in multiple secured locations. You can then monitor the policy status, all from a single point of contact in the Management Console.

How do you log shipping policies help you?

Log shipping policies allow you to implement a disaster recovery strategy for your entire SQL Server environment. You can use log shipping polices to synchronize, or back up and restore, one database to another. Using a log shipping policy to synchronize databases also helps you save disk space and network bandwidth, and comply with security requirements, because each transaction log backup can be compressed and encrypted.

To create log shipping policies, see create a log shipping policy.

To view the status of your log shipping policies, see view status of all log shipping policies or view status of a specific log shipping policy.

Create a log shipping policy

The SQL Safe Log Shipping Policy Wizard allows you to create log shipping maintenance plans across your enterprise. A SQL Safe log shipping policy is defined as a set of primary and secondary databases whose data is synchronized by shipping transaction log backups according to a defined schedule.

How do you access the Log Shipping Wizard?

You can access the Log Shipping Policy Wizard from any of the following paths:

- Go to the task bar, click **Create Policy** and then choose **Log Shipping Policy**.
- On the Policies tab, right-click the Log Shipping Policies folder and select **Create Log Shipping Policy**.
- On the Policies tab, right-click one of your Log Shipping Policies and select **Create Log Shipping Policy**.
- On the Policies tab, click **Create New Policy** located on the **Operation Summary** section of the **Log Shipping Policies Status** window. This option is only available before you create your first log shipping policy.
- From any tab, go to the **File** menu, select **Create Policy** and then **Log Shipping Policy**.

To get started with the Log Shipping Policy Wizard:

1. Name the policy.
2. Select the primary database that you want to back up.
3. Specify where these transaction log files should be stored.
4. Select the secondary database you want to synchronize with the primary.
5. Get e-mail notifications about the policy status.
6. Review details.

Name the log shipping policy

The **General** tab of the Log Shipping Policy Wizard allows you to specify the basic properties of the log shipping policy.

Why should you specify a name or description?

You are required to enter a unique name for each policy.

Both the name and description will appear in the status messages for your policies. Using a meaningful name and description will allow you to more easily identify problems when they occur. For example, consider specifying a description that will help you later choose the correct backup to restore during a disaster recovery situation.

What options are available for creating a log shipping policy?

When you create a log shipping policy, you can choose between the two following actions:

- Create Backup and Restore Jobs using the SQL Server Agent.
- Create Backup and Restore Jobs using the SQL Safe Backup Agent.

How does SQL Safe determine that a log shipping policy is okay?

SQL Safe determines that the policy is okay by looking at the following statuses:

- Whether the transaction log backup on the primary database has completed on schedule.
- Whether the transaction log restore on the secondary database has completed without warnings or errors.
- Whether the data on the secondary database is stable.

How do you control when a log shipping policy is compliant?

You can control how SQL Safe determines a missed backup by changing these options:

- Select a time limit for the log backup to occur. This is the leeway time allowed for the log backup to occur. If the log backup occurs within this period from the scheduled time, the policy is still compliant.
- Select an age limit for the secondary's data. This is the tolerance level for how old the data in the secondary database can be.

Once you define some policy settings, click **NEXT** to select the primary database.

Select the primary database

Use the **Primary** tab of the Log Shipping Policy Wizard to select which SQL Server instance will be the primary source for the log files. This is the database you will be backing up using log shipping.

What information is required on this tab?

On this tab, you have to specify the following fields:

- **SQL Server** - Select the SQL Server that contains the database to be backed up or click **Register** to register a new instance.
- **Database** - Select the database from which you will ship the backup logs.
- **Backup Job** - This schedule defines how often the backup job occurs. By default, SQL Safe schedules this job to occur every day, every 15 minutes between 12:00 AM and 11:59 PM and to start on the current date. Click **Schedule** to change the frequency and start date.
- **Backup Options** - These options allow you to change the methods used for compression, encryption, and the number of threads used when performing a backup.

Log shipping cannot be performed on a database configured to use the simple recovery model. Your database should use the Full or the Bulk-logged recovery model. SQL Safe prompts you to change the recovery model if the simple recovery model is currently used at the database.

What types of compression algorithms are available?

- None
- IntelliCompress, optimize for size (iSpeed)
- IntelliCompress, optimize for speed (iSize)
- Levels 1, 2, 3, 4

Backup operations using Level 1 complete fastest but achieve the least amount of compression. Level 4 achieves maximum compression but the backup operation may take longer.

For more information about backup compression and encryption, see how to choose compression and encryption.

What types of encryption algorithms are available?

- None
- AES (128-bit)
- AES (256-bit)

If your SQL Server environment requires FIPS compliance, use the AES encryption option. For more information, see ensure FIPS compliance.

When performing a backup, ensure the backup does not truncate the transaction logs of the database. Truncating the log will cause this log shipping policy to fail.

Once you select the primary database, click **NEXT** to select the location.

Select location for log shipping

The **Location** tab of the Log Shipping Policy Wizard allows you to specify the location for the backups you are creating with this log shipping policy. Backups must be stored to a network path that all servers in the policy can write to.

What options can you set on this tab?

- **Access Filesystem As**

This is the account SQL Safe will use to access the specified primary and mirror locations. Depending whether you selected to use the SQL Server Agent or the SQL Safe Backup Agent for the scheduling of your log policy, on this section you have the option to select between SQL Server Agent service account/SQL Safe Backup Agent service account respectively or a Windows account. Click **Account** and select your preferred option.

> ⚠ Enter a user account that has access rights to the target locations. The user account used must have read and write permissions to the specified resource.

You can also choose how to handle errors encountered while writing to the network during a backup by selecting **Enable network resiliency**.By default, SQL Safe will retry the backup operation every 10 seconds and then fail after 5 minutes (300 seconds) of continuous errors. Also, over the course of the operation, SQL Safe allows a total of 60 minutes in which to retry the backup before stopping the operation.

- **Primary Location**

This is the first location where the backup files will be stored. By default, SQL Safe will ship the backup files from this location to your secondary server. When you configure the secondary database settings, you can specify an alternate location.

Enter the network path or click **Browse** to select the location of where you want the log backup archive to be kept. The destination folder must be configured as a network share.

> ⓘ SQL Safe detects if the Computer Browser service is not running on your computer. This service enables Windows to list other computers on the network, if this service is not running, Windows may not be able to list the computers on your network. SQL Safe allows you to start this service but keep in mind that it may take several minutes for computers to become visible.

You can also specify how long you want to keep old backup files. By default, SQL Safe will delete files older than three (3) days.

- **Mirror Locations**

These are the locations where copies, or "mirrors", of the backup files will be saved. For each mirror location, SQL Safe creates and stores a copy of the backup files. You can specify up to 2 mirrors for each log shipping operation. Keep in mind that creating mirrors can impact the performance of your log shipping operation.

You can also specify:

- How long you want to keep old backup files. By default, SQL Safe delete files older than three (3) days.

- Whether SQL Safe should cancel the backup when one of the specified mirror locations reports a failure, such as connection timeout.

How do you keep your backups running despite network errors?

Select Retry writing backup files after network errors, and then click Configure to change the default settings. By default, SQL Safe will retry the backup operation every 10 seconds and then fail after 5 minutes (300 seconds) of continuous errors. Also, over the course of the operation, SQL Safe allows a total of 60 minutes in which to retry the backup before stopping the operation.

This option is not available when backing up to tape using Tivoli Storage Manager.

Once you determine the location, click **NEXT** to select secondary databases.

Select secondary databases

Use the **Secondary(s)** tab of the Log Shipping Policy Wizard to select the SQL Server instances and databases where the log backups will be restored.

From here, you can add, edit, or remove secondary databases. Each database can be restored with different options, schedule, recovery mode, etc. For more information about adding secondary databases, see specify secondary database settings for the log shipping policy.

Once you select secondary databases, click **NEXT** to configure notifications.

Configure secondary options

Use this window to select the SQL Server instances you want to synchronize with the log backups from the primary database.

What options are available in this window?

- **SQL Server**

The SQL Server that contains the database to be restored. Select a registered SQL Server or click **Register** to register a new instance.

- **Database**

Create a new database or select the database that you want to receive the transaction log restores. To create a new database, type directly the database name in the **Database** field. If you want to select an existing database, click **Select** to access the list of databases available on the instance you have selected.

- **Initialization**

Specifies the initial state of the secondary database that receives the transaction log restores. Click **Change** to modify the type of initialization that will be performed.

By default, SQL Safe initializes the database with a newly-generated full backup.

> SQL Safe detects when the primary database was previously configured to use the simple recovery model and requires for a new full backup to be performed to initialize the secondary database for the new log shipping policy. Full backups of a database using the simple recovery model lack log checkpoint information necessary for subsequent log restores.

When you click **Change**, a window for Database Initialization options opens where you can choose:

- Do not initialize. Database exists and has received most recent backup of primary database.
- Initialize database with a newly generated full backup. This will be the only option available when SQL Safe detects that the primary database was previously configured to use the simple recovery model and requires for a new full backup to initialize the secondary database.
- Initialize database with these backups. If you enable this option, you can specify the location of the backups and add encryption settings.

You can also click **Database File Locations** in this window to choose where to store your database files.

- **Database State**

Select the recovery mode the secondary database is left in after each log restore.

This setting affects how the status appears for the secondary database. If you select **Not Accessible** (No recovery mode) , then the secondary database shows the status as "Restoring" and it is unusable. If you select **Accessible but read-only** (Standby mode) , then the database is in a read-only state. In the latter option, you can choose to disconnect users when performing restore.

- **Restore Job**

This is how often the restore will occur. Click **Schedule** to change the frequency. By default, the restore occurs every 15 minutes every day, but you can specify other settings for your required daily frequency and duration of the job.

You can also choose to delay the restores by a number of minutes or hours. This represents the minimum time within which a secondary can be synced. For example, setting this value to 15 minutes would mean that the secondary will always be at least 15 minutes out of sync.

- **Restore From**

Specify the location that will contain the transaction log backup files you want to restore (ship) to this secondary database.

To use the network path you previously specified for the transaction log backup location, click **Same location as backup**.

To restore from a different location, click **Different location**, and then specify the appropriate network path.

> ⚠ Take into account that to restore from a different location, the database must already be initialized.

Configure notifications for log shipping

The **Notifications** tab of the Log Shipping Policy Wizard allows you to choose the log shipping statuses about which you want to receive alert notifications through email. Email notifications let you, and your staff, remotely monitor the status of the backups and restores you have automated with this policy.

The status of the log shipping operations determine the status of your policy. When your backups and restores are successfully completed on schedule, the policy is considered okay.

Choose the status you want to monitor, type the email address of each recipient, select the desired alert frequency for each operation, and then click **Next**.

> You must configure your mail server settings before SQL Safe can send email notifications. Click **Configure E-mail** to check your settings. For more information, see configure e-mail settings for alert notifications.

When is the email sent?

SQL Safe sends an email to the specified recipients when the selected operation status occurs. Because SQL Safe checks the status of your backup and restore operations every minute, your alert notifications provide a real-time indication of the health of your log shipping policy and your primary and secondary servers.

However, how often you are emailed about a specific status update depends on the notification frequency setting you select. For example, if you want to receive an email whenever a backup fails, even when the failures occur sequentially, choose to receive notifications every time the event occurs.

Once you configure notifications, click **NEXT** to review details.

Review details for log shipping

The **Summary** tab of the Log Shipping Policy Wizard provides the summary of specified values and options you have selected in the Log Shipping Policy Wizard.

What do you do next?

After you have reviewed the information on the Summary tab, click **Finish** to create the policy and corresponding log shipping schedule. SQL safe opens a window with the list of tasks for your policy and verifies them.

View status of all log shipping policies

When **Log Shipping Policies** is selected in the Policies tree pane, the content pane displays information describing the overall status of all of these policies. Use this view to quickly determine whether your servers are in compliance with your corporate log shipping policies.

What does the Current Status mean?

The **Current Status** area displays the most recent, combined status of all backup and restore operations performed by your log shipping policies. Even though there are multiple operation statuses, the overall policy status reflects the most critical operation status. When all backups and restores have been completed successfully according to the policy schedule, a green okay icon is displayed.

What is the Last Operation Status?

The **Last Operation Status** shows an overview of the most recent backup, restore, or log shipping operations that occurred across your enterprise. Use this view to quickly determine whether your servers are in compliance with your corporate policies. Click the status to see more detail about your operations.

What is the Operation Summary?

The **Operation Summary** area displays a list of all policies, providing information in the following columns:

Column Header	Definition
Status	Displays either a green Compliant status bar, or a red Non-Compliant status bar.
Name	Displays the policy name.
Databases Covered	Displays the number of databases being backed up by the policy.
Last Operation	Display the date and time of the most recent operation (of any type defined by the policy).
Last Operation Failure	Displays the date and time of the most recent failure (of any type defined by the policy).

How do you get details about a specific policy?

You can get more details about the status of a specific policy by double-clicking the policy operation in the Operation Summary grid.

Can you customize the columns in the grid?

You can sort by the content of any of the columns by clicking the column header.

How do you refresh the operations status?

If a recent operation does not appear in the status view, you can refresh the status of this pane by clicking the **Refresh** icon in the pane title bar.

View status of a specific log shipping policy

When a specific log shipping policy is selected in the tree pane, the content pane displays information describing the status of that policy. Use this view to determine which backup or restore operations initiated by the policy have succeeded or failed.

What actions can you perform?

From the Log Shipping Policies tree

By right-clicking a policy under the Log Shipping Policies node, you can access the following shortcuts:

Action ...	**What it does ...**
Create Log Shipping Policy	Opens the Log Shipping Policy wizard, allowing you to create a new policy.
Edit Policy	Opens the Log Shipping wizard (with all options pre-set to the values used for this operation), allowing you to edit any of the options.
Delete Policy	Allows you to delete the policy. Although backup operations associated with this policy will no longer be performed, the previous backup files and status messages created by this policy will continue to be stored in the SQL Safe Repository.
Disable Policy	Allows you to disable of the selected policy. Backup and restore operations associated with this policy will no longer be performed and email notifications configured on these jobs will be turned off.
Re-Initialize Secondary Database	Allows you to re-initialize any of the secondary databases associated with the selected policy.
View Out of Date Jobs	Allows you to view jobs that are out of date.
Update Out of Date Jobs	Allows you to update the list of out of date jobs.
Refresh Policy List	Updates the Log Shipping Policies node with the latest policies and their statuses.

From the Current Status pane

By clicking the links available in the Current Status pane, you can access the following shortcuts:

Action ...	What it does ...
View Policy Settings	Allows you to view a summary of the policy settings.
Edit Policy	Opens the Log Shipping Policy wizard, allowing you to change your policy settings.
Disable Policy	Disables the selected policy. Once a policy is disabled, it will no longer ship transaction logs to the secondary databases and it will no longer send email notifications configured for the respective jobs.
Start Log Backups	Performs a transaction log backup of the primary database that belongs to this policy by running the corresponding job. This action applies your previously defined policy settings.
Start Log Restores	Performs a transaction log restore on the secondary databases that belongs to this policy by running the corresponding job. This action applies your previously defined policy settings.

What does the Current Status mean?

The **Current Status** displays the most recent, combined status of the backup and restore operations performed by this policy. When there are multiple operation statuses, the policy status reflects the most critical operation status. When all backups and restores have been completed successfully according to the policy schedule, a green okay icon is displayed.

What is the Last Operation Status?

The **Last Operation Status** shows an overview of the most recent backup, restore, or log shipping operations that occurred across your enterprise. Use this view to quickly determine whether your servers are in compliance with your corporate policies. The operation status is limited to the operations performed by this policy.

What are the Operations Details?

The **Operation Details** graphic provides the following status details for the primary and secondary databases that belong to the selected policy:

Detail	Description
Icons	Indicate whether primary and secondary databases remain compliant with the backup and restore schedules defined by the policy. When compliant, the database icons are marked with a green check and the arrow icons pointing from database to database are green.
State	Indicates whether the database is online or off-line, as well as whether the database is read-only.
Last Backup	Displays the date and time when the last successful backup was performed. This detail displays for the primary database only.

Detail	Description
Last Restore	Displays the date and time when the last successful restore was performed. This detail displays for each secondary database.
Latency	Latency measures the amount of time that has elapsed since the last successful restore operation.
Threshold (primary database)	Indicates how much time can elapse before a scheduled backup is considered missed. When a backup is missed, the policy status is non-compliant. You can change the threshold setting by editing your log shipping policy.
Threshold (secondary database)	Indicates how much time can elapse after the last successful restore operation before the database is considered stale. When the database becomes stale, the policy status is non-compliant. You can change the threshold setting by editing your log shipping policy.
Schedule	Displays how often backups are performed on the primary database and how often restores are performed on the secondary databases.

How do you refresh the Operation Details?

If a recent operation does not appear in the status view, you can refresh the status of this pane by clicking the **Refresh** icon in the pane title bar.

6.9.4 Restore policies

Restore policies allow you to define backup maintenance plans across multiple SQL Server instances in your enterprise. These instances can reside on one or more physical servers.

What is a restore policy?

A restore policy consists of a list of databases you want to restore, a source backup archive, and a schedule according to when the restores will be performed. You can then monitor the status of each recurring restore, all from a single point of contact in the Management Console.

How does restore policies help you?

Restore policies allow you to plan and schedule your SQL Server restore maintenance, as well as monitor its success and failures, all from a single point of contact at the Management Console.

Can you select InstantRestore for your restore policy?

No. InstantRestore is available only when performing a manual restore.

To create restore policies, see create a restore policy.

To view the status of your restore policies, see view status of all restore policies or view status of a specific restore policy.

Create a restore policy

The SQL Safe Restore Policy Wizard allows you to create restore maintenance plans across your enterprise. A SQL Safe restore policy is defined as a set of databases for which restore operations will be performed according to a defined schedule. By default, SQL Safe creates the SQL Server jobs for the specified restores.

You can create a restore policy for any database that belongs to a backup policy and has a full backup.

How do you access the Restore Policy Wizard?

You can access the Restore Policy Wizard from any of the following paths:

- Go to the task bar, click **Create Policy** and then choose **Restore Policy**.
- On the Policies tab, right-click the Restore Policies folder and select **Create Restore Policy**.
- On the Policies tab, right-click one of your Restore Policies and select **Create Restore Policy**.
- On the Policies tab, click **Create New Policy** located on the **Operation Summary** section of the **Restore Policies Status** window. This option is only available before you create your first restore policy.
- From any tab, go to the **File** menu, select **Create Policy** and then **Restore Policy**.
- You can also find this option on the **SQL Safe Today** view, by going to the **Common tasks** and then selecting **Create Restore Policy**.

To get started with the Restore Policy Wizard:

1. Name the policy.
2. Select the source database which contains the data you want to restore.
3. Select the target database where the data will be restored.
4. Get email notifications about the policy status.
5. Review details.

Name the restore policy

The **General** tab of the Restore Policy Wizard allows you to specify the basic properties of the restore policy.

Why should you specify a name or description?

You are required to enter a unique name for each policy.

Both the name and description will appear in the status messages for your policies. Using a meaningful name and description will allow you to more easily identify problems when they occur. For example, consider specifying a description that will help you later choose the correct restore operation to monitor during a disaster recovery situation.

What options do you have for creating a restore policy?

When you create a restore policy, you can choose from between the two following options:

- Create Restore Jobs using the SQL Server Agent.
- Create Restore Jobs using the SQL Safe Backup Agent.

Once you define some policy settings, click **NEXT** to select your databases.

Select the database you want to restore

The **Source** tab of the Restore Policy Wizard allows you to specify the database you want to restore, the location of the corresponding backups, and which account SQL Safe should use to access these files.

To choose the location of your backup, click **Select** and choose one of the locations displayed. SQL Safe will automatically restore the latest backup found in that location each time your restore policy runs.

SQL Safe requires the selected database to belong to a backup policy since it will get the backup file location from a corresponding backup policy for the source database. ***If you choose a database that does not have a backup policy***, SQL Safe will prompt you to create a new backup policy for this database.

How do you keep your restores running despite network errors?

Select **Enable Network Resiliency** and then click **Configure** to change the default settings. By default, SQL Safe will retry the restore operation every 10 seconds and then fail after 5 minutes (300 seconds) of continuous errors. Also, over the course of the operation, SQL Safe allows a total of 60 minutes in which to retry the restore before stopping the operation. You can change these settings according to your requirements.

What accounts can you specify to access the backup files?

Depending whether you selected to use the SQL Server Agent or the SQL Safe Backup Agent for the scheduling of your restore policy, on this section you have the option to select between SQL Server Agent service account/SQL Safe Backup Agent service account respectively or another account with the respective credentials.

The specified user account must have read and write privileges on the selected directory for the backup file location.

Once you specify the databases you want to restore, click **NEXT** to select the target database.

Select the target database

The **Target** tab of the Restore Policy Wizard allows you to specify the database that you want to keep updated with routine restores.

What can you do on this tab?

You can perform the following actions:

- Select the instance where your target database is.
- Select the database you want to update using this restore operation.
- Specify the location of the data and log files associated with this database.
- Choose the appropriate recovery state for the database (Fully Accessible, Accessible but read-only or Not Accessible).
- Schedule when the Backup Agent should execute the restore job.
- Select the applicable restore options.

What do you do if your instance is not listed?

If your instance is not displayed in the SQL Server drop-down list, you can choose to add a new instance by clicking the **Register** SQL Server button. For more information, see register an instance.

How do you change the location of your database files?

If SQL Safe does not display the correct path for the location where you want to restore a file, click **Select** in the Database File Locations section, and then select the proper location.

The Database File Locations window allows you to manage the paths where SQL Safe restores new data files and log files. SQL Safe creates the file name automatically using the file type and destination database name for easy identification.

How do you set the restore schedule?

You can click **Schedule** on the Restore Job option and set the frequency and the duration of your restore policy job.

How do you restore the SQL logins for this database?

You can recover SQL logins associated with this database by selecting the **Restore database logins** option in the Restore options. SQL Safe applies this option when the source backup files contain login information. To capture login information, configure your backup policy to include the database logins.

What do you do if you have users connected to the database?

You can instruct SQL Safe to disconnect users from the database before performing the restore. To do so, select the **D isconnect users** option from the restore options.

What other additional options do you have?

Additionally, you have the following options when performing your restore:

- **Ignore checksum errors** - Select this option to ignore any errors from the generated checksum. If checksum errors are encountered, this option indicates that SQL Safe should continue to back up this database.
- **Preserve replication settings** - Choose this option to retain the settings used when the selected databases were replicated.
- **Keep CDC**- choose this option to restore databases that uses Microsoft SQL Server Change Data Capture (CDC) feature.

Once you select the target database, click **NEXT** to configure notifications.

Configure notifications for restore policy

The **Notifications** tab of the Restore Policy Wizard allows you to choose the restore statuses from which you want to receive alert notifications through email. Email notifications let you, and your staff, remotely monitor the status of the restores you have automated with this policy.

The status of the restore operations determine the status of your policy.When your restores are successfully completed on scheduled, the policy is considered okay.

Choose the status you want to monitor, type the email address of each recipient, select the desired alert frequency for each operation, and then click **Next**.

You must configure your mail server settings before SQL Safe can send email notifications. Click **Configure E-mail** to check your settings. For more information, see configure e-mail settings for alert notifications.

When is the email sent?

SQL Safe sends an email to the specified recipients when the selected operation status occurs. Because SQL Safe checks the status of your restore operations every minute, your alert notifications provide a real-time indication of the health of your service level agreements and disaster recovery plans for the SQL Server instances covered by this policy.

However, how often you are emailed about a specific status update depends on the notification frequency setting you select. For example, if you want to receive an email whenever a restore fails, even when the failures occur sequentially, choose to receive notifications every time the event occurs.

Once you configure notifications, click **NEXT** to review details.

Review details for restore policy

The **Summary** tab of the Restore Policy Wizard provides the summary of specified values and options you have selected in the Restore Policy wizard.

What do you do next?

After you have reviewed the information on the Summary tab, click **Finish** to create the policy and corresponding restore jobs.

View status of all restore policies

When **Restore Policies** is selected in the Policies tree pane, the content pane displays information describing the overall status of all of these policies. Use this view to quickly determine whether your servers are in compliance with your corporate restore policies.

What does the Current Status mean?

The **Current Status** displays the most recent, combined status of all operations performed by your restore policies. Even though there are multiple operation statuses, the overall policy status reflects the most critical operation status. When all restores have been completed successfully according to the policy schedule, a green okay icon is displayed.

What is the Last Operation Status?

The **Last Operation Status** shows an overview of the most recent backup, restore, or log shipping operations that occurred across your enterprise. Use this view to quickly determine whether your servers are in compliance with your corporate policies. Click the status to see more detail about your operations.

What is the Operation Summary?

The **Operation Summary** displays a listing of all policies, providing information in the following columns:

Column Header	Definition
Status	Displays either a green okay status bar, a yellow warning status bar, or a red error status bar.
Name	Displays the policy name.
Databases Covered	Displays the number of databases being restored by the policy.
Last Operation	Display the date and time of the most recent restore operation.
Last Operation Failure	Displays the date and time of the most recent restore failure.

How do you get details about a specific policy?

You can get more details about the status of a specific policy by double-clicking a policy operation in the **Operation Summary** grid.

Can you customize the columns in the grid?

You can sort by the content of any of the columns by clicking the column header.

How do you refresh the operations status?

If a recent operation does not appear in the status view, you can refresh the status of this pane by clicking the **Refresh** icon in the pane title bar.

View status of a specific restore policy

When a specific restore policy is selected from the Restore Policies tree node, the content pane displays information describing the status of that policy. Use this view to determine which restore operations initiated by the policy have succeeded or failed.

What actions can you perform?

From the Policies tree

By right-clicking a policy under the Restore Policies node, you can access the following shortcuts:

Action ...	**What it does ...**
Create Restore Policy	Opens the Restore Policy wizard, allowing you to create a new policy.
Edit Policy	Opens the Restore Policy wizard (with all options pre-set to the values used for this operation), allowing you to edit any of the options.
Delete Policy	Allows you to delete the policy. Although restore operations associated with this policy will no longer be performed, the previous status messages returned by this policy will continue to be stored in the SQL Safe Repository.
Disable Policy	Allows you to disable of the selected policy. Restore operations associated with this policy will no longer be performed.
Start Jobs for Policy	Allows you to run the restore job associated with this policy, performing an ad-hoc restore with the options already set by the policy.
View Out of Date Jobs	Allows you to view jobs that are out of date.
Update Out of Date Jobs	Allows you to update the list of out of date jobs.
Refresh Policy List	Updates the Restore Policies node with the latest policies.

From the Current Status pane

By clicking the links available in the Current Status pane, you can access the following shortcuts:

Action ...	What it does ...
View Policy Settings	Allows you to view a summary of the policy settings.
Edit Policy	Opens the Restore Policy wizard, allowing you to change your policy settings.
Disable Policy	Disables the selected policy. Once a policy is disabled, it will no longer perform restore operations for the associated databases. To recover a database that belongs to a disabled policy, perform a manual restore using the Restore Wizard.
Start Full Restore	Performs a full restore of all databases that belong to this policy by running the corresponding job. This action applies your previously defined policy settings.

From the Operation Details grid

By right-clicking on an operation, you can access the following shortcuts:

Action ...	What it does ...
Cancel	Allows you to cancel the backup policy.
View Details	Shows the Details pane, providing additional information about the selected operation.
Restore again	Runs the restore operation again, using the same settings.
Restore with different options	Opens the Restore wizard (with all options pre-set to the values used for this operation), allowing you to specify different options before running the operation.
Set Progress To	Allows you to change the status of the selected operation.
Close Details	Hides the Details pane.

What does the Current Status mean?

The **Current Status** displays the most recent, combined status of the restore operations performed by this policy. When there are multiple operation statuses, the policy status reflects the most critical operation status. When all restores have been completed successfully according to the policy schedule, a green OK icon is displayed.

What is the Last Operation Status?

The **Last Operation** Status shows an overview of the most recent backup, restore, or log shipping operations that occurred across your enterprise. Use this view to quickly determine whether your servers are in compliance with your corporate policies. The operation status is limited to restore operations performed by this policy.

What are the Operation Details?

The **Operation Details** grid displays a listing of all restore operations performed for the databases included in the selected policy for the last 7 days. This grid includes the following columns:

Column	Definition
Progress	During an operation, the progress bar will denote the percentage of the operation completed. When the operation is complete, it will display a green bar labeled 100%. If an operation completed with errors, this column will display a red bar labeled Error. If an operation completed with warnings, this column will display a yellow bar labeled 100% with an asterisk. This column also indicates when the backup file has been deleted (groomed), and therefore is no longer available to be restored.
Instance	Displays the name of the SQL Server instance that was backed up or restored by this operation.
Icon (Enhanced Restorability)	Displays an icon if the backup includes maps containing metadata for InstantRestore and SQL virtual database. For more information about InstantRestore, see How InstantRestore works. For information about SQL virtual database, see recover objects using SQL virtual database.
Database	Displays the name of the database that was backed up or restored by this operation.
Operation	Displays the type of operation performed. The types are Backup, Restore, and Verify.
Backup Type	Displays the type of the backup performed by the operation. The types are Full, Log, Differential, and File.
Compressed	Displays the size of the backup file after compression.
Ratio	Displays the ratio of the Uncompressed size of the database reported by SQL Server to the resulting Compressed size of the backup file created by SQL Safe.
Compression	Displays the type of compression used for the backup.
Database Size	Displays the size of the original database.

Column	Definition
Uncompressed	Displays the size of data contained in the database, as reported by SQL Server.
Encryption	Displays the type of encryption SQL Safe used during the backup operation.
Duration	Displays the time (hours:minutes:seconds) required to complete the operation.
Start Time	Displays the start date and time of the operation.
End Time	Displays the end date and time of the operation.
Threads	Displays the number of threads SQL Safe used during the backup operation.
Format	Displays the backup format. SQL Safe backup (Safe) or native backup (Bak) format.

Can you customize the columns in the Operation Details grid?

You can sort by the content of any of the columns by clicking the column header.

You can select which columns are visible in this grid, and enable column grouping, by clicking the **Filter** icon in the pane title bar.

How do you refresh the data displayed in the Operation Details grid?

If a recent operation does not appear in the status view, you can refresh the status of this pane by clicking the **Refresh** icon in the pane title bar.

What are the details?

To see the detailed results of a specific operation, right-click the operation in the Operation Details grid and select **View Details**. The Details pane displays below. By default, this pane is hidden.

The Details pane provides the following information about the selected backup operation:

Tab	Description
Statistics	Displays the database size, the size of the uncompressed backup, the size of the compressed backup, and the compression ratio achieved with this backup. The ratio is a measure of the storage savings achieved with SQL Safe compression technology. For more information about the storage space savings you can realize using SQL Safe, see understand your total cost of operation (TCO).

Tab	Description
Result Text	Displays text describing the result of the restore.
Files	Displays the complete path of the backup set file for the restore.
Backup Set Description	Displays the description you specified for this restore.
Storage Options	Displays which locations were chosen to store the backup files associated with the restore.

6.9.5 View last operation status

After a policy runs, the **Policies tree** content pane displays information describing the overall status of all of these operations. The Last Operation Status shows an overview of the most recent backup, restore, or log shipping operations that occurred across your enterprise. Use this view to quickly determine whether your servers are in compliance with your corporate policies.

If your operations are successful, SQL Safe displays a list of these operations and their associated messages. If an operation was unsuccessful for any reason, the Last Operation Status also allows you to rerun the operation from this view instead of requiring you to access the appropriate wizard.

You can rerun any previous operation from this grid. To rerun an operation, check the box to the left of the appropriate operation, and then click Run Selected Operations Again . This request executes the backup or restore using the previous settings.

How do you rerun your operations?

You can go to the **Last Operation Status** section, click a topic listed under this section, and a wizard appears displaying a list of all operations for the selected topic, such as Backups were canceled by user. SQL Safe provides information in the following columns:

Column Header	Definition
Instance	Displays the instance name.
Database	Displays the database name.
Operation	Displays the type of operation performed.
Start Time	Displays the start date and time of the most recent operation.

Can you rerun more than one operation?

Yes, you can quickly rerun one or more operations through the Last Operation Status. Check the boxes next to all appropriate operations, and then click **Run Selected Operations Again**. To select all of the displayed operations, check the box next to the Instance column title.

How can you capture the provided information?

SQL Safe allows you to copy the list summary and the related messages using the copy buttons to the left of the operation summary box and the messages box. You can then paste this information in another application, such as Notepad. Use the upper copy button to capture the status, instance, database, operation, and start time. Use the lower copy button to capture the same information, but also include the message.

6.9.6 Deploy maintenance plans using SQL Safe

You can use the SQL Safe log shipping, backup, and restore policies to automate and enforce your disaster recovery requirements or Service Level Agreements (SLAs).

> ⚠ SQL Safe no longer provides the Maintenance Plan Conversion Utility.

How to use log shipping policies as maintenance plans

Use the following steps to create a new log shipping policy based on an existing SQL Server Log Shipping maintenance plan.

To create a log shipping policy based on a maintenance plan:

1. Use SQL Server Management Studio to connect to the SQL Server instances for which you have a native maintenance plan.
2. Disable the backup jobs associated with the primary instance.
3. Allow all restore jobs associated with the secondary instances to complete, and then disable these jobs.
4. Start the SQL Safe Management Console, and start the Log Shipping Policy Wizard.
5. On each wizard window, specify the appropriate options, and then click **Finish**. Because the selected instances were previously included in a native maintenance plan, you do not need to initialize the secondary instances.
6. Test your new log shipping policy by tracking the policy status.
7. Delete your native maintenance plan.

How to use backup policies as maintenance plans

Use the following steps to create a new backup policy based on an existing SQL Server maintenance plan.

To create a backup policy based on a maintenance plan:

1. Use SQL Server Management Studio to connect to the SQL Server instances for which you have a native maintenance plan.
2. Disable the backup jobs associated with this plan.
3. Start the SQL Safe Management Console, and start the Backup Policy Wizard.
4. On each wizard window, specify the appropriate options, and then click **Finish**.
5. Test your new backup policy by tracking the policy status.
6. Open your maintenance plan and delete the tasks that performed your backups, and then enable the appropriate jobs associated with this plan. ***If your maintenance plan performed backups only***, delete your native maintenance plan.

How to use restore policies as maintenance plans

Use the following steps to create a new restore policy based on an existing SQL Server maintenance plan.

1. Create a backup policy that archives the databases you want to routinely restore.
2. Test your backup policy by running the policy jobs. This action also creates source backup files for your new restore policy.
3. Use SQL Server Management Studio to connect to the SQL Server instances that have a native maintenance plan you want to replace.
4. Disable the backup and restore jobs associated with this plan.
5. Start the SQL Safe Management Console, and start the Restore Policy Wizard.

6. On each wizard window, specify the appropriate options, and then click **Finish**.
7. Test your new restore policy by tracking the policy status.
8. Open your maintenance plan and delete the tasks that performed your backups and restores, and then enable the appropriate jobs associated with this plan. ***If your maintenance plan performed backups and restores only***, delete your native maintenance plan.

7 Navigate the IDERA Dashboard

7.1 What is the IDERA Dashboard?

The **IDERA Dashboard** is a common technology framework designed to support the entire IDERA product suite. The **IDERA Dashboard** allows you to get an overview of the status of your SQL Servers and hosted databases in a consolidated view while providing the means to drill into individual product dashboards for details.

The **IDERA Dashboard** is comprised of the following tabs:

- Overview
- Details View
- Alerts
- Administration

7.2 What information provides the Overview tab of the IDERA Dashboard?

In the **IDERA Dashboard** users can glance at their registered products overall status through the **Overview** tab. If you have several products registered with the Dashboard, use the drop down option from the Filters section to type the products and tags for which you want widgets to display information. Click **Clear Filters** to remove them or **Export** to save the displayed information.

By default the **Overview** tab displays the following widgets for SQL Safe:

- **Safe Disk Space Savings** - displays the total of savings achieved by using SQL Safe and the respective ROI (Return on Investment). You can set this value in the basic configurations of the **General Preferences** window of your **SQL Safe Administration** tab.
- **Safe Status Details** - displays a summary of the SQL Safe policies and operations in your environment.

You can customize this tab by dragging and dropping widgets, expanding  , collapsing  , closing  or limiting the amount of information displayed  .

Customize the Overview tab

To add widgets and further customize this tab, go to **Configure Dashboard views** in **Administration** view.

7.3 What actions can be performed in the Details tab of the IDERA Dashboard?

This tab allows you to see specific metrics by selected SQL Server Instance and SQL Safe widget information across all instances. To change the instance details, click the drop-down option at the top of this tab and select the instance for which you want to see its alert information.

You can expand or collapse a widget using , view full size , remove , or configure its options (limiting the number of alerts displayed).

Customize the Details view tab

You can further customize this view in the **Configure Dashboard Views** section of the **Administration** view.

7.4 What actions can be performed in the Alerts view of the IDERA Dashboard?

In the **Alerts tab,** you can view alerts for all products instances registered with the Dashboard. Users can filter this information by product, alert category, severity, specific metrics, and tags.

If you want changes to be applied as you select filters, select **Apply filter as it changes**. If this option is not selected, SQL Safe applies the filters after you finish selecting them.

What actions can be performed in the Administration view of the IDERA Dashboard?

In the **IDERA Dashboard**, all products show a common **Administration** tab when the logged-in user has administrator privileges. Selecting this tab displays the Administration view which hosts a range of sections for performing administration-related actions.

In this tab you can perform the following options:

- Manage Users
- Manage Instances
- Manage Products
- Manage Tags
- Customize Main Navigation Tab Order
- Customize DBA Dashboard
- Send Notifications
- Manage Licenses

Most of these options can also be accessed from the gear icon located at all tabs on the top right corner of the IDERA Dashboard.

7.5 Managing users in the IDERA Dashboard

The **Users** section of the **Administration** view allows users to grant access to other team members or groups and manage their roles. Users with administrative privileges are divided in:

- **Dashboard administrators** - with capability to manage access on Dashboard functions as well as individual products functions.
- **Product administrators** - with capability to grant access to individual products for which they have administrative rights.

To add new users, edit their details (name, subscription, or email address), or remove them, select **Manage Users** in the Administration view.

7.5.1 Adding a user in the IDERA Dashboard

In the **IDERA Dashboard**, access is granted to Windows users or groups. To add users follow these steps:

1. Click the **Add User / Group** option and the Add User/Group dialog displays.
2. Type the name of the user you want to grant access to. You should enter a Windows user name in the following format: **<domain\user>**.
3. Type the name with which the user will be displayed.
4. Select **User** or **Group** in the Account Details field.
5. Check the **Do not timeout the browser session for this account** check box to stay logged in.
6. In the **Product** field, you can select to add users to the **IDERA Dashboard** or to a specific product like SQL Safe.
7. If you select the first option (IDERA Dashboard), in the **Role** field you can assign the **Dashboard Administrator** or **Dashboard Guest** roles.
8. If you select the second option (SQL Safe or respective product), in the **Role** field you can assign the Administrator, User, or Guest roles.
9. Click **Add More** to assign more user roles to the different available products.
10. Click **SAVE.**

7.5.2 Editing a user in the IDERA Dashboard

This option allows you to edit the account name, display name, disable his/her account, and add new permissions. To edit a user or group follow these steps:

1. Click the user you want to edit to access its details.
2. Change the necessary settings and/or click **Add New Permission** to grant the user access to the IDERA Dashboard or to assign different roles to other products. You can also remove roles by clicking the X icon next to the role.
3. Click **SAVE**.

> You cannot edit the logged in user credentials.

7.5.3 Removing a user from the IDERA Dashboard

This option allows you to remove a user from access to the IDERA Dashboard. To delete a user or group follow these steps:

1. Click the user you want to remove from the list of available users.

2. Click **Delete** in the user details window.
3. A warning that requires a confirmation whether you want to delete the selected user or group displays.

7.6 Managing Instances in the IDERA Dashboard

The **IDERA Dashboard** tracks all SQL Server instances, including not only instances managed or registered with the products, but also those discovered on the network. You can review your SQL Server environment and its coverage in areas such as backup, security, and performance. Additionally, with the Dashboard you can remove registered instances that no longer exist in your SQL Server environment.

To access these options, select **Manage Instances** in the **Administration** view. The **Manage Instances** view allows you to select the following filters by clicking **Options:**

- **Instance Name** - select from all available SQL Server instances in your environment.
- **MSSQL Version** - select the SQL Version for which you want to view instances information
- **Tags** - select the tags for which you want to view instances information
- **Discovered** - choose a period of time when instances were discovered
- **Last seen** - select a period of time when IDERA products last monitored the instances.
- **Idera Product** - specify the IDERA product for which you want view instances
- **Status** - select if you want to view Managed (Registered), Unmanaged (discovered on the network but not registered), Archived, Discovered, Ignored or Unsupported instances.

To remove all filters, click **Clear Filters**.

When selecting any of the instances displayed on the list, you can:

- View its details (products where they are registered, SQL Server version, Status, and Available License).
- Change the instance status by product and then click **Save** to keep the changes.

7.7 Managing products in the IDERA Dashboard

The **IDERA Dashboard** hosts IDERA products that are registered with the dashboard. The **Products** section of the **Administration** tab allows users to register, view, and manage products.

7.7.1 Register a product

To register a product, follow these steps:

1. Click **Register a Product** on the top section of the Manage Products tab.
2. In the **Add a Product to IDERA Dashboard** window, fill in the required fields:
   - **Product:** select or enter *SQLSafeRestService*
   - **Display name:** specify a unique name.
   - **Host:** specify the host on which the SQL Safe Rest Service has been installed on.
   - **Port:** specify the port on which the SQL Safe Rest Service is listening on. The default port is 9998.
   - **Username:** specify the user with Dashboard Administrator privileges.
   - **Password:** enter the password for the user.
3. Click the **Register** button.

7.7.2 Editing a product

To edit a product, follow these steps:

1. Click the product from the list of available products.
2. In the **Edit Product** window, you can change the location and the connection credentials.
3. Click **Save** to keep changes.
4. To migrate the product to a different IDERA Dashboard, click **Migrate product**. In the new window, specify the host computer, port, and the administrator credentials to access the new location.

7.7.3 Removing a product

To remove a product, follow these steps:

1. Click the product from the list of available products.
2. Select **Delete** on the **Edit Product** window.
3. A window will appear asking for confirmation for the removal of the product. Click **Yes** to continue with the removal of the product. Click **No** to cancel.

7.8 Manage Tags

Tags allow you to group instances according to your work environment preferences. The IDERA Dashboard allows you to create, edit, and manage tags available in all your IDERA products. Tags can be assigned to individual instances or databases. Click the **Manage Tags** option in the **Administration** tab to access these options.

> Although you can add Tags through the IDERA Dashboard options, SQL Safe currently does not support tags in this version.

To create a new tag follow these steps:

- Click **Add Tag**
- Type the Tag name
- Click **Add Instance** to select the instances where the tag will be assigned
- Click **Add Database** to select the databases where the tag will be assigned
- Click **Save** to add the new tag

To edit tags, follow these steps:

- Click the tag you want to edit from the list of available tags
- Edit the tag name or change the instances and databases where it is assigned
- Click **Save** to keep the changes

To delete tags, follow these steps:

- Click the tag you want to remove from the list of available tags
- In the tag details view, select **Delete** to remove the tag. You can also find the same option at the top of this view.
- A warning message verifies if you want to delete the selected tag

7.8.1 How can you filter tags?

If you have multiple tags, you can apply filters to view only the ones you require. To filter your tags follow these steps:

- Click **Options** from the **Manage Tags** view
- Specify the products for which you want to view tags
- Select the instances and databases for which you want to view tags
- Click **Search**
- To specify different search criteria, click **Clear Filters**

7.9 Configure navigation order in the IDERA Dashboard

The **Configure Navigation Order** section of the **Administration** view allows users to customize the order of the different IDERA products on the navigation tab. When you access this option, you can drag and drop tabs to change the order. To go to previous settings, click **Reset to the default order**.

You can also modify the order of the products by accessing the top menu  and clicking  **Customize** to enable the drag and drop feature. Select a product and release it in the desired position. Click **Save** when you are done.

7.10 Configure IDERA Dashboard views

The **Configure Dashboard Views** section of the **Administration** tab allows you to customize which product widgets are shown in the **Overview** and **Details** view tabs. Click **Customize DBA Dashboards** and access these options:

- **Hide/Show DBA Dashboard** - You can choose to show the DBA Dashboard to all users, hide it, or show it to users with permissions in two or more products.
- **Select View** - Select if you want to apply the settings of this window to your **Overview** or **Details** tab.
- **Add or remove columns** - Select how many widget columns you want to view in your **Overview** or **Details** tabs.
- **Add a Widget** - Use the drop-down option to select the widget you want to add to your view. You can specify in which column you want to add the widget. Take into account that the option **Total Widgets** automatically updates the number of widgets selected for your view.
- **Set column widths** - Specify the width percentage of each column and click **Apply**. Note that all your columns widths should sum up 100%.
- **Replace/delete/move product widgets** - On each widget you can perform the following actions:
  - **Choose Another** - Use the drop-down options to select another available widget to be displayed instead of the current one. You can only replace your current widget with other that is not currently displayed on your view.
  - **Delete** - Click the **X** icon on the upper right corner of each widget to remove it.
  - **Settings** - Use this option to specify the default setting for your widget: **Expanded** or **Collapsed** and limit the number of alerts the **Dashboard** displays.
  - **Move -** Drag and drop widgets to order them according to your view requirements.

7.11 Sending notification

If you want to send a message to all users notifying an important status, upgrade, or other news, follow these steps:

- Go to the **Administration** tab and click **Send Notification**.
- Type your message and click **Send**.
- The message will be sent to all users registered within the IDERA Dashboard.

7.12 Managing Licenses

To view the license status of your IDERA products, go to the **Administration** tab and click **Manage Licenses**.

The new view allows you to check all the licenses used in your IDERA products.

8 Availability Groups

AlwaysOn Availability Groups are part of an integrated solution, introduced in SQL Server 2012 with the goal of achieving the highest level of data availability and disaster recovery for organizations.

Availability Groups grant DBAs the ability to automatically or manually failover a group of databases as a single unit with support for up to four secondary replicas.

For additional information on availability groups, see the Microsoft document, Overview of AlwaysOn Availability Groups (SQL Server).

SQL Safe supports SQL Server Availability Groups and allows you to perform backup and recovery strategies on your primary and secondary replicas.

> Keep in mind that an agent needs to be installed on all nodes in the availability group that will be considered for backups.

8.1 Backup policies with Availability Groups

SQL Safe allows you to configure backup policies with availability groups. The policy includes all the databases in the Availability Group that you want to backup and you can determine which Availability Group members you would like to be taken into account.

Keep in mind that in Availability Groups:

- On primary replicas you can perform full, differential, and log backups.
- On secondary replicas, full backups must be copy only, differential backups are not supported, and log backups must NOT be copy only.

When SQL Safe performs the backup operation, it detects the preferred replica for the backup operation and it skips the non-preferred ones.

> Please take into account that if you add a new node as the preferred replica and it is not part of the policy, you will need to add the respective database to the policy.

8.2 Restores on Availability Groups

SQL Safe assembles all the necessary files for a restore operation even though the backups were taken from different Availability Group members. You can select which one of the Availability Group members (primary or replica) will be restored.

The **Backup Sets** section in the Restore wizard shows you backups taken from primary and secondary replicas for an Availability Group database.

9 Integrate SQL Safe with TSM

Use this TSM Guide to integrate SQL Safe into your existing TSM-based backup and recovery processes. SQL Safe interfaces with the TSM Client API, allowing you to backup and restore directly to the TSM Server while using the SQL Safe user interfaces. By integrating SQL Safe with your TSM deployment, you can immediately receive the benefits of fast, compressed, secure backups as well as several enterprise storage management features – without retooling your current archival workflow.

9.1 TSM integration checklist

☑	**Follow these steps ...**
☐	Install the SQL Safe components, and review the supported TSM Client versions.
☐	Install the SQL Safe Backup Agent on each SQL Server instance on which backups will be performed.
☐	Install the TSM Client on the same SQL Server instances, and then configure each TSM Client to connect to the TSM Server. For more information, see the IBM TSM Backup-Archive Clients Installation and User's Guide.
☐	Ask your TSM Administrator to create a new management class for SQL Safe. For more information, see the IBM Tivoli Storage Manager for Windows Administrator's Guide.
☐	Perform a test backup and restore using SQL Safe with TSM settings.
☐	Create policy jobs to enforce consistent backup operations across your TSM environment.

9.2 How SQL Safe works with TSM

When you perform a TSM backup, the SQL Safe Backup Agent sends the data files directly to the TSM Server using the TMS Client API, which handles the final storage and management of the backup archives. The SQL Safe Backup Agent uses the TSM Client API options file to locate the TSM Server.

During the backup operation, SQL Safe compresses and optionally encrypts the data files before sending them to the TSM Server, minimizing the impact on your network and storage requirements. The SQL Safe user interfaces expose several TSM parameters, such as retention periods and storage locations, that can help you and the TSM Administrator more easily manage your backup archives. For more information about specific TSM functions, see the IBM TSM Backup-Archive Clients Installation and User's Guide.

9.3 How TSM data retention works

Because both SQL Safe and TSM play very important but different roles in determining data file retention, some advance planning and configuration is required before you begin using SQL Safe with TSM.

First, determine your data retention requirements. For example, you may want to groom (delete) backup archive sets after 7 days. Your TSM Administrator will need to create a new management class for SQL Safe and configure the desired data retention period in TSM (such as, retain all backup files for one week).

When configuring TSM, note that SQL Safe accepts up to 260 characters for the TSM file path name.

Once TSM has been configured, backups can now be performed automatically using SQL Safe policies or manually using the SQL Safe CLI or XSP. When you perform a backup, you can define how long SQL Safe should retain a backup archive. Backup archives that age beyond the specified time period will not be available to restore through the SQL Safe user interfaces. However, keep in mind that these data files will be available on the TSM Server until TSM grooms the backup archives according to the retention time.

9.3.1 Setting data retention in SQL Safe Backup Policy jobs

When using the Management Console, you can configure the SQL Safe retention period (Remove files older than) option on the Location tab of the Policy wizard. This option specifies how long SQL Safe should keep this backup archive available for restore. The SQL Safe Backup Agent will automatically mark these data files as inactive on the TSM Server, which allows the files to be groomed (deleted) by TSM. This expiration processing is based on the retention values configured for the SQL Safe management class.

9.3.2 Setting data retention through the CLI

When using the SQL Safe CLI, use the -delete backup option to specify when SQL Safe should make a backup archive as inactive. For example:

```
SQLsafecmd backup northwind TSM -tsmhighlevel BACKUP -tsmmanagementclass SQL Safe -delete 1weeks
```

You do not need to specify the Low Level filename when grooming files through SQL Safe. The Low Level filename is automatically generated in the following format: `<instance name>_<database name>_<backup type>_<timestamp>.safe`

After the designated time period, SQL Safe marks the backup archives as "deleted" in the Management Console, signaling they are no longer available to restore. However, these backup archives are simply inactive until they are permanently groomed from the TSM Server. You can manually browse inactive backup archives using the TSM Browse command and then manually restore the backup archive.

9.3.3 Setting Data Retention through the XSP

When using the XSP, use the `@delete` parameter. This parameter functions in the same way as the `-delete` backup option in the CLI. For more information, see the sample XSP scripts available from the Programs menu.

9.4 Backup to the TSM Server

You can use the SQL Safe Backup wizard in the Management Console or the backup command in the CLI or XSP to send a backup directly to your TSM Server.

TSM is case sensitive so special care should be taken when specifying the High Level and Low Level settings. Note that SQL Safe accepts up to 260 characters for the TSM file path name.

9.4.1 Backup wizard

Start the Backup Wizard and follow the tabs, setting the appropriate options. On the Locations tab, select **Tape** (Tivoli Storage Manager), and then specify the following required fields:

- High Level
- Low Level
- Management class (optional)
- Client Options File (dsm.opt file)
- Connection Settings

If your backup set is located on a TSM Server other than the server included in the dsm.opt file, you can change the TSM connections settings to override the values set in the client options file. Click **Change** on this option and specify: Node Name, Node Password, Server Address and Server Port.

You can also configure SQL Safe to mark these files as inactive after a specified age.

9.4.2 Example CLI code snippets that use the backup command

You can also perform a backup through the CLI. Additional backup options can be set in the SQL Safe Backup wizard, from which you can generate a CLI script that includes the specified wizard settings. For example:

```
SQLsafeCmd Backup Northwind TSM -TsmClientOwnerName tsmclient -TsmClientOwnerPassword
password -TsmConfigFile "C:\Program Files\Tivoli\TSM\baclient\dsm.opt" -TsmHighLevel
Backup -TsmLowLevel %instance%_%database%_%backuptype%_%timestamp%.safe
-TsmManagementClass mclass1 -TsmTcpServerAddress tsmserver -TsmTcpPort 1500
```

You can use the CLI to change the TSM connections settings for the client options file. The `TsmTcpServerAddress` and `TsmTcpPort` options are compatible with any command that supports TSM.

For more information about available backup options, see the usage statements in the CLI Help. To view the CLI Help, type `SQLsafeCmd help backup`.

9.4.3 XSP

You can perform backups using the XSP. The XSP backup parameters function in the same way as backup options in the CLI. For more information, see the sample XSP scripts available from the Programs menu.

9.5 Restore a backup from a TSM Server

You can use the SQL Safe Restore wizard in the Management Console or the restore command in the CLI or XSP to restore a backup file directly from your TSM Server.

TSM is case sensitive so special care should be taken when specifying the High Level and Low Level settings. Note that SQL Safe accepts up to 260 characters for the TSM file path name

9.5.1 Restore wizard

Start the Restore Wizard and follow the tabs, setting the appropriate options. On the Sources tab, specify where the backup files are located. You can specify the location using either action:

- On the Repository tab, select the database you originally backed up.
- On the Tivoli Storage Manager tab, select the backup files.

9.5.2 Example CLI code snippets that use the restore command

You can also perform a restore through the CLI. Additional restore options can be set in the SQL Safe Restore wizard, from which you can generate a CLI script that includes the specified wizard settings.

```
SQLsafeCmd Restore Northwind TSM -InstanceName SQL2000 -TsmHighLevel Backup -TsmLowLevel
SQL SafeDEV01_SQL2000_Northwind_Full_200805301028.safe
```

```
SQLsafeCmd Restore Northwind Tsm -InstanceName SQL2000 -TsmHighLevel Backup -TsmLowLevel
SQL SafeDEV01_SQL2000_Northwind_Log_200805301030.safe
```

You can use the CLI to change the TSM connections settings for the client options file. The `TsmTcpServerAddress` and `TsmTcpPort` options are compatible with any command that supports TSM.

For more information about available restore options, see the usage statements in the CLI Help. To view the CLI Help, type `SQLsafeCmd help restore`.

9.5.3 XSP

You can restore data files using the XSP. The XSP restore parameters function in the same way as restore options in the CLI. For more information, see the sample XSP scripts available from the Programs menu.

9.6 Automate backups to your TSM Server

Once the SQL Server computer has been properly configured to send and receive information from the TSM Server, you can then create policy jobs that instruct SQL Safe to write backup files directly to the TSM Server.

Start the Backup Policy Wizard and follow the tabs, setting the appropriate options. On the Locations tab, select **Tape** (Tivoli Storage Manager).

SQL Safe will skip any invalid backup types or options. For example, SQL Safe will skip databases that are off-line, will not perform T-Log backups of databases that are in simple mode, and will ignore the object level recovery option when backing up system databases.

9.7 Browse archives on the TSM Server

The available backup archives can be viewed (browsed) by right-clicking the target SQL Server instance in the **Servers** navigation pane and selecting **Browse TSM Archives**.You can also view this information through the SQL Safe Restore wizard, CLI, or XSP. You can view a list of all available files including those flagged as inactive.

To browse the TSM Server through the Restore wizard, select **Tivoli Storage Manager** on the Source tab, and then click **Browse**.

To use the XSP browse command, see the sample XSP scripts available from the Programs menu.

9.7.1 Example CLI code snippets that use the browse command

To browse all active files:

```
SQLsafeCmd Browse TSM
```

To browse all active and inactive files:

```
SQLsafeCmd Browse TSM -TSMIncludeInactive
```

To browse all active and inactive files in a Highlevel called BACKUP:

```
SQLsafeCmd Browse TSM -TSMIncludeInactive -TSMHighLevel BACKUP
```

TSM is case sensitive. Be careful when specifying the High Level and Low Level file set.

For more information about available browse TSM options, see the usage statements in the CLI Help. To view the CLI Help, type `SQLsafeCmd help browse TSM`.

9.8 Extract archives from the TSM Server

You can extract any active backup archive from the TSM Server using the command line interface (CLI).

9.8.1 An example CLI code snippet that uses the extract command

```
SQLsafeCmd extract TSM -BackupFile c:\NW_full.safe -TSMHighLevel Backup -TSMLowLevel
SQLSAFEDEV01_SQL2000_Northwind_Full_2005300847.safe
```

TSM is case sensitive. Be careful when specifying the High Level and Low Level file set.

For more information about available extract TSM options, see the usage statements in the CLI Help. To view the CLI Help, type `SQLsafeCmd help extract TSM`.

9.9 Mark SQL Safe backup files inactive

You can manually mark any SQL Safe backup that is stored on your TSM Server as inactive using the command line interface (CLI).

9.9.1 An example CLI code snippet that uses the expire command

```
SQLsafeCmd expire TSM -BackupFile c:\NW_full.safe -TSMHighLevel Backup -TSMLowLevel SQL
SafeDEV01_SQL2000_Northwind_Full_2005300847.safe -age 7 days.
```

TSM is case sensitive. Be careful when specifying the High Level and Low Level file set.

For more information about available expire TSM options, see the usage statements in the CLI Help. To view the CLI Help, type `SQLsafeCmd help expire TSM`.

10 SafeToSQL Utility

Use this SafeToSQL Guide to install and use the SafeToSQL utility. This utility helps you easily convert your SQL Safe archive files for use with Microsoft SQL Server. With SafeToSQL, you can use a simple command line statement to save your archive files in `.bak` format, ensuring they are stored in an industry-wide format.

10.1 How SafeToSQL works

SafeToSQL provides a quick and easy way to convert SQL Safe archive files to the format used by Microsoft SQL Server. The conversion SafeToSQL performs is useful when you are sending your archive files to someone who may not have SQL Safe installed, and needs to access the archive files for troubleshooting or data migration purposes.

You can execute this conversion though a simple command-line statement from the command shell or in a batch file. When SafeToSQL converts the `.safe` file, the utility appends the name of the backup set index to the `.bak` file name.

10.1.1 How does the utility handle multi-threaded backup sets?

SQL Safe has the capability of creating multi-threaded backup set to multiple virtual devices. When that backup set is converted using SafeToSQL, multiple SQL Server backup files will be created, one for each virtual device.

10.1.2 Why does the utility output multiple backup files from a single SQL Safe archive?

SQL Safe uses multiple processing threads to apply compression and encryption settings during a backup operation. During the backup, SQL Server divides the data between separate backup devices. SQL Safe then writes the finished backup from all of these devices into a single archive file. When SafeToSQL converts the `.safe` file to native format, a separate `.bak` file is created for each device or thread that was used during the original backup. This approach ensures the information can be restored to SQL Server using the same number of devices that were used for the backup. This is a SQL Server requirement.

10.2 Deploy the SafeToSQL utility

Use the following information and instructions to successfully deploy the **SafeToSQL** utility in your SQL Server environment.

10.2.1 Requirements

Installing and running **SafeToSQL** requires .NET Framework 4.0 or higher.

The **SafeToSQL** utility will convert your SQL Safe backup files to uncompressed native backup files. You should ensure adequate disk space is available for the converted backups.

10.2.2 How to install SafeToSQL

When you install the SQL Safe Full version, the **SafeTOSQL** utility is installed automatically. The utility is located in the SafeToSQL sub-directory of the main SQL Safe installation directory (e.g. *C:\Program Files\IDERA\SQLsafe\SafeToSQL*).

If you want to install **SafeToSQL** on different servers, locate the **SafeToSQL** installation file ("Idera SafeToSQL x64.exe") in the installation directory of the SQL Safe Management Console (e.g. *C:\Program Files\IDERA\SQLsafe*).

Then, follow these steps:

1. Log on with an administrator account to the computer on which you want to install **SafeToSQL**.
2. Copy the "Idera SafeToSQL x64.exe" file onto the server and run it.
3. Read the Welcome window, and then click **Next**.
4. Review and accept the license agreement by selecting **I accept the terms in the license agreement**, then click **Next**.
5. Choose the destination folder, then click **Next**.
6. Click **Install**.

10.3 Create the SafeToSQL command

To use the SafeToSQL utility, run the Command Prompt, and then type the appropriate command syntax for the conversion you need to execute. Use the following descriptions to choose the options you need.

10.3.1 Command syntax

Use the following syntax when converting a SQL Safe archive file:

```
SafeToSQL source_file_path [-backupfile file_name] [ -backupset #] [-password pwd] [-list]
```

Where the following option is mandatory:

source_file_path

Defined as the complete directory path and file name of the SQL Safe archive containing the backup set to convert to Microsoft SQL Server backup format.

10.3.2 Options

The SafeToSQL utility provides the following options:

-backupfile filename

Provides the names of additional files in multi-file archives. You must specify each file in a multi-file archive and provide the complete path to the file.

-backupset #

Specifies the index (1-based) of the backup set in an archive containing multiple backup sets. ***If you do not specify the backup set index,*** the backup set defaults to 1, the index of the first backup set in the archive.

-password pwd

Specifies the password for decrypting an encrypted backup set. ***If the backup set is encrypted,*** provide the password you specified during backup.

-list

Prints out the complete contents of the archive specified by source_file_path.

10.3.3 Output file name format

SafeToSQL uses the following file naming convention for SQL Safe backup files it converts to Microsoft SQL Server backup files:

```
filename_#.bak
```

Where the file name components are as follows:

filename

Specifies the name of the source archive file.

#

Specifies the name of the backup set index.

10.4 Example SafeToSQL commands

Use the following example scenarios to create SafeToSQL commands that fit your conversion needs.

10.4.1 Convert an archive with a single backup set

To convert an encrypted archive that contains the *pubs* database as the single backup set, type the following command at the command prompt:

```
SafeToSQL "d:\sqlsafe_backup\LT1_pubs_01_enc.safe" -password my_password
```

The output path and name of the converted file is:

```
d:\sqlsafe_backup\LT1_pubs_01_enc_1.bak
```

10.4.2 Convert an archive with multiple backup sets

To convert an archive that contains the *northwind* database as the second backup set in the archive, type the following command at the command prompt:

```
SafeToSQL "d:\sqlsafe_backup\LT1_multi_01.safe" -backupset 2
```

The output path and name of the converted file is:

```
d:\sqlsafe_backup\LT1_multi_01_2.bak
```

To list the contents of the archive, type the following command at the command prompt:

```
SafeToSQL "d:\sqlsafe_backup\LT1_multi_01.safe" -list
```

10.4.3 Convert an archive saved across multiple files

To convert an archive saved in the directory in two files, type the following command at the command prompt:

```
SafeToSQL "d:\sqlsafe_backup\pubs_a.safe" -backupfile "d:\sqlsafe_backup\pubs_b.safe"
```

The output path and names of the converted files are:

```
d:\sqlsafe_backup\pubs_a_1.bak
d:\sqlsafe_backup\pubs_b_1.bak
```

11 Report on Backup and Restore Operations

SQL Safe Reports (Reports) provides several built-in reports that allow you to quickly and access backup and restore information. Each report gives detailed information about backups and restores performed by SQL Safe.

11.1 How reports work

Reports seamlessly integrates into Microsoft SQL Server Reporting Services (Reporting Services). For each built-in SQL Safe report, the Reports Installer utility deploys Report Definition Language (RDL) files to the Report Server computer. The RDL files define the report layout and parameters, using the data source (SQL Safe Repository) you specified during install. Reporting Services automatically acknowledges these files, allowing you to immediately generate and view reports on imported data using the Report Manager Web interface.

For more information about the Reporting Services architecture, see the Reporting Services Books Online.

11.2 Available reports

You can generate any of the following reports. These reports help you better understand and track backup and restore operations, such as measuring the performance of individual backups or monitoring storage consumption. Use these reports to proactively identify and meet the needs of your changing SQL Server environment.

11.2.1 SQL Safe – Backup Owners

This report lists the backups and associated owners for all SQL Server instances and databases registered with SQL Safe. Use this report to monitor backup activity and enforce database security. For example, you can verify that the appropriate database administrators are executing backup operations on the databases they own.

11.2.2 SQL Safe – Backup Performance

This report provides backup performance statistics for the selected SQL Server instances and databases. Use this report to track and compare backup performance.

11.2.3 SQL Safe – Backup Size Chart

This report charts SQL Safe backup sizes for a specific database over time. Use this report to track how the backup size for a particular database has changed.

11.2.4 SQL Safe – Backup Store Utilization

This report indicates how much backup storage space is currently used by the selected SQL Server instances and databases.

11.2.5 SQL Safe – Large Backup

This report lists any SQL Safe backup that is larger than the specified size. Use this report to identify backups that could compromise storage policies.

11.2.6 SQL Safe – Last Backup

This report provides details about the last database backup executed on the selected SQL Server instances and databases. Use this report to monitor or troubleshoot recent backup operations.

11.2.7 SQL Safe restored databases

This report provides details about all database restore activity that occurred in a given date range. Use this report to track restore operations across your SQL Server environment.

11.2.8 SQL Safe storage savings report

This report summarizes the storage cost savings you have gained by using SQL Safe. This savings is calculated using the following formula: (saved MB)/(1024 x cost per MB). Use this report to understand one of the many benefits of using SQL Safe.

11.3 Customize reports

You can customize any of the built-in SQL Safe reports or develop new reports that fit your unique needs. If you decide to customize these reports, consider the following best practices:

- Saving your new and modified reports to a separate folder
- Using a different file name for modified reports

For more information about developing custom reports, see the Reporting Services Books Online.

11.4 How to run reports

With the appropriate permissions, you can generate and view data directly from the Report Manager Web interface.

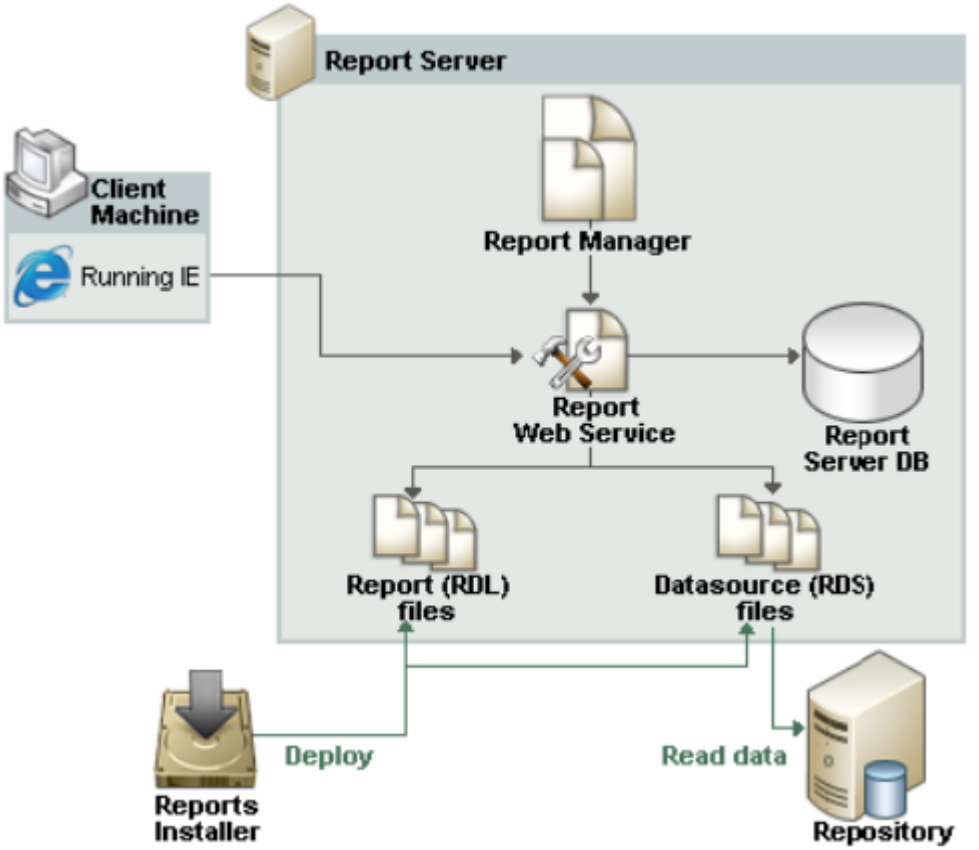
To use Report Manager to run reports:

1. Start Internet Explorer, and then type `http://ReportServerName/Reports` in the Address field. For example, if you are running Reporting Services on the `lab01` server computer, type `http://lab01/Reports`.
2. On the Reporting Services home page, click **SQL Safe**.
3. Click the report you want to run.

> In order to configure Microsoft SQL Server Reporting Services on your computer, you can refer to Reporting Services Configuration Manager (Native Mode).

11.5 Deploy reports

You can implement Reports on any computer running Microsoft SQL Server 2005 Reporting Services (Reporting Services) or later. The following installation scenario illustrates how you can implement Reports in an existing SQL Server environment that uses a dedicated Report Server.



11.5.1 Reports requirements

SQL Safe leverages Microsoft SQL Server Reporting Services (Reporting Services) to provide on-the-spot reporting on your backup and restore data. The Report Server computer should meet or exceed the hardware and software requirements recommended by Microsoft to run and manage the Reporting Services components.

SQL Safe supports Reporting Services version 1.0 or later and requires Microsoft Internet Explorer version 6.0 or later. You can generate SQL Safe reports on both 32-bit and 64-bit Report Servers.

11.5.2 Reports permissions and requirements

Reports leverages the existing role-based security model provided with Reporting Services. Reports supports Windows Authentication only (mixed mode on SQL Server).

Reports requires the following permissions and rights to successfully generate reports on your backup and restore data. Assign the appropriate role on the SQL Safe folder in the Report Manager. The individual report files inherit the permissions you set.

By default, the Reports Installer utility deploys the report files to `Program Files\IDERA\SQLsafe Reporting Services` on the Report Server computer.

Account	Action	Requirements
Proxy user (specified in data source)	Connect to the SQL Safe Repository and read data per report parameters	Read access to the SQL Safe Repository database
Administrator	Configure reports and set security, run Installer utility	Content Manager role
End user (auditor or manager)	Generate and view reports	Browse role

For more information, on how to access the Report Manager, click Report Manager (SSRS Native Mode).

11.5.3 Install reports

The Reports setup program installs the **SQL Safe Reports** application. The Reports Installer utility allows you to specify the proxy user account credentials and select which built-in SQL Safe reports you want to deploy.

When you install the SQL Safe Full version, the **SQL Safe Reports** utility is installed automatically. Locate the utility in the *SQLSafe Reports* sub-directory of the main SQL Safe installation directory (e.g. C:\Program Files\IDERA\SQLsafe\SQLSafe Reports).

If you want to install **SQL Safe Reports** locally, locate the **SQL Safe Reports** installation file ("Idera SQLsafe Reporting Services.exe") in the installation directory of the SQL Safe Management Console (e.g. C:\Program Files\IDERA\SQLsafe).

The following procedure guides you through a local installation of Reports. However, you can also deploy Reports remotely.

To install reports:

1. Log on with an administrator account to the Report Server computer.
2. Copy the "Idera SQLsafe Reporting Services.exe" file onto the server and run it.
3. Read the Welcome window, and then click **Next**.
4. Review and accept the license agreement by selecting **I accept the terms in the license agreement**, and then click **Next**.
5. Accept the default folder for your Reports installation, or click **Change** to specify a different folder, and then click **Next**.
6. Click **Install**. When the installation is completed, the *Idera SQL Safe Reports Installer* window will appear.
7. On the **IDERA SQL Safe Reports** Installer window, specify the following configuration settings, and then click **Next**.
    - **Report Server Web Service URL:** URL for the SQL Server Reporting Service.
    - **Target SQL Server:** SQL Server instance hosting the SQL Safe repository database.
    - **SQL Database:** Name of the SQL Safe repository database.
    - **Login ID** and **Passwords:** Credentials of the Windows account the Report Server should use to connect to the repository database. The specified account should have read permissions on this database.
8. Select which SQL Safe reports you want to deploy to this Report Server, then click **Next**.
9. Click **Install**.

11.5.4 Change the report data source

You can change the Reports data source to pull data from a different Repository location. Use the Report Manager to change the data source connection link. For more information, see the Reporting Services Books Online.

12 Use Command Line Interface (CLI) to automate SQL Safe Backup functions

SQL Safe Backup provides you with a Command Line Interface (CLI) to automate its functions. The CLI commands allow you to make changes to your SQL Safe Backup configurations across multiple SQL Server instances in a few lines of text, saving you time.

12.1 About Command Line Interface (CLI)

A Command Line Interface (CLI) is a user interface that allows you to interact with the operating system. The interaction is performed through commands where you type an specific command and the operating system performs an action.

Make sure you are aware of proper command format for each action you want to perform.

12.2 SQL Safe Backup CLI Usage

The following commands help you automate your SQL Safe Backup functions:

Usage	Description
SQLsafeCmd <action> [options]	Perform an action.
SQLsafeCmd help <action>	Display detailed help for an action.
SQLsafeCmd -ArgsFile <filename>	Perform the action defined within an argument file.

where:

Action	Description
<action>	A keyword that tells SQL Safe what to do.
<filename>	Specifies the file that contains command line arguments.

When you use the *SQLsafeCmd* command, the following actions can be performed:

Actions	Description	Example
Help	Display detailed descriptions and available options.	SQLsafeCmd help <action>
Backup	Backup a database.	SQLsafeCmd help backup
Restore	Restore a database.	SQLsafeCmd help restore

Actions	Description	Example
RestoreLast	Restore the most recent full backup of a database from the specified directory.	SQLsafeCmd help restorelast
InstantRestore	Instant Restore a database.	SQLsafeCmd help instantrestore
Verify	Verify a database backup.	SQLsafeCmd help verify
RestoreFileListOnly	List files for database in backup set.	SQLsafeCmd help restorefilelistonly
RestoreHeaderOnly	List backup sets in an archive.	SQLsafeCmd help restoreheaderonly
Browse TSM	Browse Tivoli Storage Management.	SQLsafeCmd help browse TSM
Expire TSM	Expire Tivoli Storage Management backup archives.	SQLsafeCmd help expire TSM
Extract TSM	Extract a file from Tivoli Storage Management.	SQLsafeCmd help extract TSM
License	Register a Backup Agent license.	SQLsafeCmd help license
EncryptBackupPassword	Encrypt plain-text password for encrypted backups.	SQLsafeCmd help encryptbackuppassword
EncryptRestorePassword	Encrypt plain-text password for encrypted restores.	SQLsafeCmd help encryptrestorepassword
EncryptSqlPassword	Encrypt plain-text password for SQL Server logins.	SQLsafeCmd help encryptsqlpassword
EncryptWindowsPassword	Encrypt plain-text password for Windows logins.	SQLsafeCmd help encryptwindowspassword
InstallXsp	Install SQL Server extended stored procedures for SQLsafe.	SQLsafeCmd help installxsp
Delete	Delete backup archives older than the specified date.	SQLsafeCmd help delete
LogShipBackup	Log Shipping primary backup.	SQLsafeCmd help logshipbackup
LogShipRestore	Log Shipping secondary restore.	SQLsafeCmd help logshiprestore

Actions	Description	Example
ObjectLevelRecovery	Restore database objects.	SQLsafeCmd help objectlevelrecovery
Create-Policy	Create a new policy.	SQLsafeCmd help create-policy
Edit-Policy	Edit existing policy.	SQLsafeCmd help edit-policy
AddDatabase	Deploy new database for existing policy (LogShipping or Backup policy).	SQLsafeCmd help adddatabase
Virtual Database	Display detailed descriptions and available options.	SQLvdbCmd help

The examples in the above table provides you with commands to use in Command Prompt. Those commands will display detailed descriptions and available options for each action.

12.3 Add Database CLI Commands

SQL Safe provides you with CLI commands to help you manually add databases.

To add databases use the following commands:

- SQLsafeCmd AddDatabase <policy_name> <db_name>
- SQLsafeCmd AddDatabase <policy_guid> <db_name>
- SQLsafeCmd AddDatabase <policy_name> <instance_name> <db_name> [primary|secondary]
- SQLsafeCmd AddDatabase <policy_guid> <instance_name> <db_name> [primary|secondary]

Where:

Action	Description
<policy_name>	Name of the policy.
<policy_guid>	GUID of the policy.
<db_name>	Name of the database.
<instance_name>	Specifies the instance name. {server, instance}. Note: this parameter is optional for Backup policy and required for LogShipping policy.
[primary|secondary]	Specifies the location. Note: the default value is "secondary" database. this parameter is optional for LogShipping policy.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help AddDatabase`).

12.4 Backup Agent License CLI Commands

SQL Safe provides you with CLI commands to help you manually register a backup agent license.

To perform License operations use the following command:

- SQLsafeCmd License <sql_server> <license_key> [options]

Where:

Action	Description
<sql_server>	The full name of the SQL server instance to license. E.g. HOSTNAME\INSTANCE
<license_key>	The license key.

And the [options] are:

Option	Description
-ForceLocal	Forces the installation of a license to the local machine, although <sql_server> might refer to another machine. Note: it is necessary in clustered environments where a clustered SQL Server instance name differs from the physical node hosting it.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help License`).

12.5 Backup CLI Commands

SQL Safe provides you with CLI commands to help you manually backup a database.

To backup databases use the following commands:

- SQLsafeCmdBackup <db_name> [<db_name>] <backup_archive> [options]
- SQLsafeCmdBackup <db_name> [<db_name>] TSM [options]
- SQLsafeCmdBackup <db_name> [<db_name>] TRUNCATEONLY [options]

Where:

Action	Description
<db_name>	One or more names of database(s) to backup. If a database name contains a space, the database name should be surrounded with "double quotes". Special keywords may be used to backup multiple databases. These keywords are: {all}, {allsystem}, {alluser}.
<backup_archive>	Path to the backup archive.
Tsm	Use Tivoli Storage Manager (see below for special options).
TruncateOnly	Use this keyword to truncate the transaction log of the SQL Server database only.

12.5.1 Common Options

The following options help you perform backup operations:

Options	Description
-BackupDescription <desc>	<desc> - the description of the backup.
-BackupName <name>	<name> - the name of the backup.
-BackupType <type>	The backup type required. <type> - {full, differential, log, file}.

<table>
<tr><th>Options</th><th>Description</th></tr>
<tr><td>-BckDstType <type></td><td>The backup destination type is used for the backup.
<type> - {0, 1, 2, 3, 4, 5, 6, 7}.
where:
<table>
<tr><th>Type</th><th>Description</th></tr>
<tr><td>0</td><td>Single File</td></tr>
<tr><td>1</td><td>Stripped Files</td></tr>
<tr><td>2</td><td>Mirrored</td></tr>
<tr><td>3</td><td>TSM</td></tr>
<tr><td>4</td><td>Data Domain</td></tr>
<tr><td>5</td><td>Amazon S3</td></tr>
<tr><td>6</td><td>Azure Blob</td></tr>
<tr><td>7</td><td>TSM Stripes</td></tr>
</table>
Note: the default value is "0".</td></tr>
<tr><td>-CompressionLevel <level></td><td>The compression level used for the backup.
<level> - {ispeed, isize, 0, 1, 2, 3, 4}.
Note: if the compression level is not specified, ispeed is the default.</td></tr>
<tr><td>-Exclude <db_name> [<db_name> ...]</td><td><db_name> [<db_name> ...] - one or more names of database(s) to not backup.</td></tr>
<tr><td>-InstanceName <name></td><td><name> - the SQL server instance name.
Note: it is not required if the instance is set as a default on the target server.</td></tr>
<tr><td>-NoTruncate</td><td>Do not truncate the transaction log. (Log backup only).</td></tr>
<tr><td>-Overwrite</td><td>Overwrite existing archive if one exists.
Note: if this option is omitted, the default behavior is to append.</td></tr>
</table>

Options	Description
-Server <hostname>	<hostname> - the hostname of the server hosting the SQL Server where the operation should be performed. Note: this option is required for accessing remote or clustered SQL Servers (where applicable).
-Verify	Verify the backup set after backup is complete.
-SectorType	Public or Government based on Azure Sector.

12.5.2 Encryption Options

Encrypt your backups with the following options:

Options	Description
-EncryptionType <type>	The type of encryption used to encrypt the backup. <type> - {AES128, AES256}.
-Password <pwd>	<pwd> - the non-encrypted password used to encrypt the backup.
-EncryptedBackupPassword <pwd>	<pwd> - the encrypted password used to encrypt the backup. (Used with EncryptionType).

12.5.3 Security Options

Secure your backups with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-SecurityModel <model>	The security model used to log into SQL Server. <model> - {Integrated, SQL}. Note: Integrated (Windows authentication) is the default.
-SqlUsername <username>	<username> - the SQL Server username. (SQL SecurityModel).
-SqlPassword <pwd>	<pwd> - the SQL Server password. (SQL SecurityModel).
-EncryptedSqlPassword <pwd>	<pwd> - the encrypted SQL Server password generated by EncryptSqlPassword action. (SQLSecurityModel).

Options	Description
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

12.5.4 Advanced Options

The following advanced options help you perform backup operations:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.
-BackupFile <filename>	Specifies additional backup archive files to be used for striping backups. <filename> - specifies the backup archive files. Note: use once for each additional stripe.
-Checksum	Instructs SQL Server to generate backup checksums during a backup, or verify backup checksums during a verify or restore. Note: for SQL 2005 and later only.
-ContinueAfterError	Instructs SQL Server to continue the operation despite encountering errors such as invalid checksums. Note: for SQL 2005 and later only.
-CopyOnly	Specifies that the backup does not affect the normal sequence of backups. Note: for SQL 2005 and later only.
-DatabaseFilegroup <filegroup>	<filegroup> - the database logical filegroup. Note: for file BackupType only.
-DatabaseFilename <filename>	<filename> - the database logical filename. Note: for file BackupType only.

Options	Description
-Delete <n><time_period>	After a backup successfully completes, delete archives that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -delete 2hours. Note: if you use the space between <n> and <time_period>, the backup archive filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_ <timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.
-DeleteMirror <n><time_period>	After a backup successfully completes, delete mirrors that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -deletemirror 2hours. Note: if you use the space between <n> and <time_period>, the mirror filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_<timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.
-FailOnMirrorError	To abort a backup if an error is encountered while writing to a mirror backup archive. Note: the default behavior is to abort only if an error is encountered while writing to a primary backup archive.
-IncludeLogins	For backup, includes the database logins in the backup file. For restore, creates the logins from the backup file on the destination server.
-MailTo <email_address>	<email_address> - an email address(es) to send the notification via SMTP. Note: multiple addresses may be separated by spaces, semicolons, or commas.

Options	Description
-MaxTransferSize	Specifies the largest unit of transfer in bytes to be used between SQL Safe and the backup media. The possible values are multiples of 65536 bytes (64 KB) ranging up to 4194304 bytes (4 MB). This parameter is used to enable compression on TDE enabled databases only when the MaxTransferSize value is set to 65537 or higher. If omitted, the MaxTransferSize will be taken from the 'Transfer Limit' value set in the SQL Safe agent properties.
-MirrorFile <filename>	Specifies additional backup archive files to be used for mirroring backups. <filename> - specifies the backup archive files. Note: use once for each additional mirror. Up to two mirrors may be specified.
-NoSkip	Disables automatic skipping of databases that cannot be backed up, such as offline databases.
-NoStatus	Prevents status messages from being cached or sent to the Repository.
-ReadWriteFileGroups	Instructs SQL Server to perform a partial backup, which includes the primary filegroup and any read/write secondary filegroups. Note: SQL 2005 and later only.
-RecoveryMode <mode> [-UndoFile<filename>]	Specifies the mode in which to leave the database after the operation is completed. <mode> - NoRecovery, Standby. Note: for Standby mode an undo file may be specified with the -UndoFile option.
ReportTLog	For backup, 'Yes' reports Skipped T-Log backups against databases that are in simple mode with a SUCCESS status rather than SKIPPED.
-UndoFile <filename>	<filename> - specifies the ABSOLUTE path to the undo filename. Note: for Standby recovery mode only.
-RetryWrites <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.
-Threads <number>	<number> - specifies the number of threads that should be used to distribute the backup process across multiple processors.
-VDB Off	Do not optimize for quick access by SQL virtual database.

Options	Description
-Verbose	Displays SQL Server result text for both successful and failed backups. Note: by default, it displays for failed backups only.

12.5.5 Tivoli Storage Manager (TSM) Options

There are TSM options for your backup operations:

Options	Description
-TsmClientOwnerName <name>	<name> - the client owner name.
-TsmClientOwnerPassword <pwd>	<pwd> - the client owner password.
-EncryptedTsmClientOwnerPassword <pwd>	<pwd> - the encrypted TSM client owner password.
-TsmConfigFile <filename>	<filename> - the configuration file location.
-TsmHighLevel <name>	<name> - the high level file specification (path).
-TsmLowLevel <name>	<name> - the low level file specification (file name).
-TsmTcpServerAddress <address>	<address> - the TCP/IP address for the TSM server.
-TsmTcpPort <port>	<port> - the TCP/IP port address for the TSM server.
-TsmManagementClass <name>	<name> - the management class to which the backup file will be bound.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Backup`).

12.6 Delete Backups CLI Commands

SQL Safe provides you with CLI commands to help you manually delete backup archives older than the specified date.

To perform delete operations use the following command:

- SQLsafeCmd Delete <filename> [options]

Where the required [options] are:

Options	Descriptions
<filename>	The path and filename to be deleted. An asterisk (*) can be used in the filename to delete multiple files at once.
-Age <n><time_period>	Delete/Expire archives that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. Note: There must be NO SPACE between <n> and <time_period>. E.g., -age 2hours.

12.6.1 Security Options

Secure your delete operations with the following options:

Options	Description
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

12.6.2 Advanced Options

The following advanced options help you perform your delete operations:

Options	Description
-IncludeSubdirectories	If set, includes subdirectories when finding files to delete.

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.
-No Prompt	Do not prompt for confirmation before deleting files.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Delete`).

12.7 Encrypt Passwords CLI Commands

SQL Safe provides you with CLI commands to help you manually encrypt plain-text passwords.

The following topics are included in this section:

- Encrypt Backup Password CLI Commands
- Encrypt Restore Password CLI Commands
- Encrypt SQL Password CLI Commands
- Encrypt Windows Password CLI Commands

12.7.1 Encrypt Backup Password CLI Commands

SQL Safe provides you with CLI commands to help you manually encrypt plain-text password for encrypted backups.

To perform EncryptBackupPassword operations use the following command:

- SQLsafeCmd EncryptBackupPassword <password>

Where:

Action	Description
<password>	Plain-text password to encrypt.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help EncryptBackupPassword`).

12.7.2 Encrypt Restore Password CLI Commands

SQL Safe provides you with CLI commands to help you manually encrypt plain-text password for encrypted restores.

To perform EncryptRestorePassword operations use the following command:

- SQLsafeCmd EncryptRestorePassword <password>

Where:

Action	Description
<password>	Plain-text password to encrypt.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help EncryptRestorePassword`).

12.7.3 Encrypt SQL Password CLI Commands

SQL Safe provides you with CLI commands to help you manually encrypt plain-text password for SQL Server logins.

To perform EncryptSqlPassword operations use the following command:

- SQLsafeCmd EncryptSqlPassword <password>

Where:

Action	Description
<password>	Plain-text password to encrypt.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help EncryptSqlPassword`).

12.7.4 Encrypt Windows Password CLI Commands

SQL Safe provides you with CLI commands to help you manually encrypt plain-text password for Windows logins.

To perform EncryptWindowsPassword operations use the following command:

- SQLsafeCmd EncryptWindowsPassword <password>

Where:

Action	Description
<password>	Plain-text password to encrypt.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help EncryptWindowsPassword`).

12.8 Install Extended Stored Procedures (XSP) CLI Commands

SQL Safe provides you with CLI commands to help you manually install SQL Server extended stored procedures (XSP) for SQLsafe.

To perform the installation use the following command:

- SQLsafeCmd InstallXsp [options]

12.8.1 Common Options

The following options help you perform the installation:

Options	Descriptions
-InstanceName	SQL server instance name. To specifiy all instances, use an asterisk (*) for the instance name. Note: it is not required if the instance is set as a default on the target server.
-Server	The remote server that hosts the SQL Server instance(s) where the extended stored procedures should be installed (or uninstalled).

12.8.2 Security Options

Secure your installation with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-SecurityModel <model>	The security model used to log into SQL Server. <model> - {Integrated, SQL}. Note: Integrated (Windows authentication) is the default.
-SqlUsername <username>	<username> - the SQL Server username. (SQL SecurityModel).
-SqlPassword <pwd>	<pwd> - the SQL Server password. (SQL SecurityModel).
-EncryptedSqlPassword <pwd>	<pwd> - the encrypted SQL Server password generated by EncryptSqlPassword action. (SQLSecurityModel).
-WindowsUsername	The Windows user that will be used during the installation process.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.

Options	Description
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

12.8.3 Advanced Options

The following advanced options help you perform your installation:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.
-Remove	Remove all SQLsafe extended stored procedures from the specified instance(s).

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help InstallXsp`).

12.9 InstantRestore CLI Commands

SQL Safe provides you with CLI commands to help you manually perform an Instant Restore to a database.

To perform InstantRestore operations use the following commands:

- SQLsafeCmd InstantRestore <db_name> <backup_archive> [options]
- SQLsafeCmd InstantRestore <db_name> <point_in_time> [options]

Where:

Action	Description
<database_name>	Name of the database.
<backup_archive>	Path to the backup archive.
<point_in_time>	Date/Time {"MM/dd/yyyy hh:mm:ss"} to restore to.

12.9.1 Common Options

The following options help you perform InstantRestore operations:

Options	Descriptions
-BackupSet <index>	<index> - the index of the backup set within the backup archive. (1-based).
-DisconnectUsers	Disconnects all users from the target database before the restore operation begins.
-EncryptedRestorePassword <pwd>	<pwd> - the encrypted password used to decrypt the backup.
-InstanceName <name>	<name> - SQL server instance name. Note: it is not required if the instance is set as a default on the target server.
-Move <logical_filename> <target_filename>	To move the database logical database file to the physical target file. <logical_filename> - the database logical database file. <target_filename> - the physical target file. Corresponds to the WITH MOVE option in the RESTORE DATABASE T/SQL command.
-Password <pwd>	<pwd> - the non-encrypted password used to encrypt the backup.

Options	Descriptions
-Server <hostname>	<hostname> - the hostname of server hosting the SQL Server where the operation should be performed. Note: this option is required for accessing remote or clustered SQL Servers (where applicable).
-Replace	Override database if exists.

12.9.2 Security Options

Secure your InstantRestore operations with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-SecurityModel <model>	The security model used to log into SQL Server. <model> - {Integrated, SQL}. Note: Integrated (Windows authentication) is the default.
-SqlUsername <username>	<username> - the SQL Server username. (SQL SecurityModel).
-SqlPassword <pwd>	<pwd> - the SQL Server password. (SQL SecurityModel).
-EncryptedSqlPassword <pwd>	<pwd> - the encrypted SQL Server password generated by EncryptSqlPassword action. (SQL SecurityModel).
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

12.9.3 Advanced Options

The following advanced options help you perform InstantRestore operations:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.

Options	Description
-Diff <filename>	The differential backup. <filename> - the file path to the differential backup. This can be followed by -BackupFile, -BackupSet, or -Password to set individual options for this backup set.
-Log <filename>	The log backup. <filename> the file path to the log backup. This can be followed by -BackupFile, -BackupSet, or -Password to set individual options for this backup set.
-BackupFile <filename>	Specifies additional backup archive files to be used for striping backups. <filename> - specifies the backup archive files. Note: use once for each additional stripe.
-IncludeLogins	For backup, includes the database logins in the backup file. For restore, creates the logins from the backup file on the destination server.
-KeepReplication	Preserves replication settings when restoring a published database to a server other than that on which it was created.
-NoChecksum	Disables the validation of any checksums by the restore operation. Note: for SQL 2005 and later only.
-MailTo <email_address>	<email_address> - an email address(es) to send the notification via SMTP. Note: multiple addresses may be separated by spaces, semicolons, or commas.
-MaxTransferSize	Specifies the largest unit of transfer in bytes to be used between SQL Safe and the backup media. The possible values are multiples of 65536 bytes (64KB) ranging up to 4194304 bytes (4 MB). This parameter is used to enable compression on TDE enabled databases only when the MaxTransferSize value is set to 65537 or higher. If omitted, the MaxTransferSize will be taken from the 'Transfer Limit' value set in the SQL Safe agent properties.
-NoStatus	Prevents status messages from being cached or sent to the Repository.

Options	Description
-RecoveryMode <mode> [-UndoFile <filename>]	Specifies the mode in which to leave the database after the operation is completed. <mode> - {NoRecovery, Standby}. Note: for Standby mode an undo file may be specified with the -UndoFile option.
-ReportTLog	For backup, 'Yes' reports Skipped T-Log backups against databases that are in simple mode with a SUCCESS status rather than SKIPPED.
-StopAt <datetime>	Specifies the database to be restored to the state it was in as of the specified date and time. <datetime> - {"mm/dd/yyyy hh:mm:ss"}. Note: for Log BackupType only.
-StopAtMark <mark> [-After <datetime>]	Specifies recovery to the specified <mark>, including the transaction that contains the <mark>. Note: for Log BackupType only.
-StopBeforeMark <mark> [-After <datetime>]	Specifies recovery to the specified <mark> but does not include the transaction that contains the <mark>. Note: for Log BackupType only.
-After <datetime>	Recovery stops at the first <mark> having the specified name exactly at or after <datetime>. Note: only valid with -StopAtMark/-StopBeforeMark options. For Log BackupType only.
-Wait	Wait for hydration to complete before returning result of operation. Note: by default, it is set to return result of bringing database online.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help InstantRestore`).

12.10 Log Shipping CLI Commands

SQL Safe provides you with CLI commands to help you manually perform log shipping operations.

The following topics are included in this section:

- Log Shipping Backup CLI Commands
- Log Shipping Restore CLI Commands

12.10.1 Log Shipping Backup CLI Commands

SQL Safe provides you with CLI commands to help you manually perform a log shipping primary backup.

To perform LogShipBackup operations use the following command:

- SQLsafeCmd LogShipBackup [options]

Where the required [options] are:

Options	Descriptions
-BackupLocation <path>	<path> - the network path where backups will be stored.
-DatabaseName <name>	<name> - name of the primary database.

Common Options

The following options help you perform LogShipBackup operations:

Options	Descriptions
-BckDstType <type>	The backup destination type is used for the backup. <type> - {0, 1, 2, 3, 4}.
-CompressionLevel <level>	The compression level used for the backup. <level> - {ispeed, isize, 0, 1, 2, 3, 4}. Note: if the compression level is not specified, ispeed is the default.
-EncryptionType <type>	The type of encryption used to encrypt the backup. <type> - {AES128, AES256}.
-EncryptedBackupPassword <pwd>	<pwd> - the encrypted password used to encrypt the backup. (Used with EncryptionType).

Options	Descriptions
-Delete <n><time_period>	After a backup successfully completes, delete archives that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -delete 2hours. Note: if you use the space between <n> and <time_period>, the backup archive filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_ <timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.
-InstanceName <name>	<name> - SQL server instance name. Note: it is not required if the instance is set as a default on the target server.

Security Options

Secure your LogShipBackup operations with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-SecurityModel <model>	The security model used to log into SQL Server. <model> - {Integrated, SQL}. Note: Integrated (Windows authentication) is the default.
-SqlUsername <username>	<username> - the SQL Server username. (SQL SecurityModel).
-SqlPassword <pwd>	<pwd> - the SQL Server password. (SQL SecurityModel).
-EncryptedSqlPassword <pwd>	<pwd> - the encrypted SQL Server password generated by EncryptSqlPassword action. (SQLSecurityModel).
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

Advanced Options

The following advanced options help you perform LogShipBackup operations:

Options	Description
-MailTo <email_address>	<email_address> - an email address(es) to send the notification via SMTP. Note: multiple addresses may be separated by spaces, semicolons, or commas.
-Threads <number>	<number> - specifies the number of threads that should be used to distribute the backup process across multiple processors.
-RetryWrites <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help LogShipBackup`).

12.10.2 Log Shipping Restore CLI Commands

SQL Safe provides you with CLI commands to help you manually perform a log shipping secondary restore.

To perform a LogShipRestore operations use the following command:

- SQLsafeCmd LogShipRestore [options]

Where the required [options] are:

Options	Descriptions
-BackupLocation <path>	<path> - the network path where backups will be stored.
-DatabaseName <name>	<name> - name of the primary database.

Common Options

The following options help you perform LogShipRestore operations:

Options	Descriptions
-DisconnectUsers	Disconnects all users from the target database before the restore operation begins.
-EncryptedRestorePassword <pwd>	<pwd> - the encrypted password used to decrypt the backup.
-InstanceName <name>	<name> - SQL server instance name. Note: it is not required if the instance is set as a default instance on the target server.
-LoadDelay <minutes>	<minutes> - the number of minutes to delay the restore.

Security Options

Secure your LogShipRestore operations with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-SecurityModel <model>	The security model used to log into SQL Server. <model> - {Integrated, SQL}. Note: Integrated (Windows authentication) is the default.
-SqlUsername <username>	<username> - the SQL Server username. (SQL SecurityModel).
-SqlPassword <pwd>	<pwd> - the SQL Server password. (SQL SecurityModel).
-EncryptedSqlPassword <pwd>	<pwd> - the encrypted SQL Server password generated by EncryptSqlPassword action. (SQLSecurityModel).
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

Advanced Options

The following advanced options help you perform LogShipRestore operations:

Options	Description
-MailTo <email_address>	<email_address> - an email address(es) to send the notification via SMTP. Note: multiple addresses may be separated by spaces, semicolons, or commas.
-RecoveryMode <mode> [-UndoFile<filename>]	Specifies the mode in which to leave the database after the operation is completed. <mode> - {NoRecovery, Standby}. Note: for Standby mode an undo file may be specified with the -UndoFile option.
-Threads <number>	<number> - specifies the number of threads that should be used to distribute the backup process across multiple processors.
-UndoFile <filename>	<filename> - specifies the ABSOLUTE path to the undo filename. Note: for Standby recovery mode only.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help LogShipRestore`).

12.11 Object Level Recovery CLI Commands

SQL Safe provides you with CLI commands to help you manually restore database objects.

To perform ObjectLevelRecovery operations use the following commands:

- SQLsafeCmd ObjectLevelRecovery <db_name> <backup_archive> [options]
- SQLsafeCmd ObjectLevelRecovery <db_name> <point_in_time> [options]

Where:

Action	Description
<db_name>	Name of the database.
<backup_archive>	Path to the backup archive.
<point_in_time>	Date/Time {"MM/dd/yyyy hh:mm:ss"} to restore to.

12.11.1 Common Options

The following options help you perform ObjectLevelRecovery operations:

Options	Descriptions
-Async	Performs Action in background. Returns command control as soon as possible.
-InstanceName <name>	<name> - SQL server instance name. Note: it is not required if the instance is set as a default on the target server.
-Server <hostname>	<hostname> - the hostname of server hosting the SQL Server where the operation should be performed. Note: this option is required for accessing remote or clustered SQL Servers (where applicable).
-BackupFile <filename>	Specifies additional backup archive files to be used for striping backups. <filename> - specifies the backup archive files. Note: use once for each additional stripe.
-BackupSet <index>	<index> - the index of the backup set within the backup archive. (1-based).

Options	Descriptions
-RestoreAs <db_name>	<db_name> - a database new name. Note: the default is "<dbName>OLR" name.
-DatabaseFilesLocation <path>	<path> - specifies the database files location.
-TemporaryVdbLocation <location_path>	<location_path> - location where save VDB files.
-TemporaryVdbServerName <hostname>	<hostname> - host where create VDB for OLR Restore action.
-Password <pwd>	<pwd> - the non-encrypted password used to encrypt the backup.

12.11.2 Advanced Options

The following advanced options help you perform ObjectLevelRecovery operations:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.
-Keep CDC	Flag to indicate whether the restore will support the Microsoft SQL Server Change Data Capture (CDC) feature. The possible values are 1 (CDC restore will be supported) or 0 (CDC restore will not be supported). If the Keep CDC parameter is set to 1 then the CDC enabled database will be restored along with the CDC related artifacts and the Capture and Cleanup jobs will be created with the default options. If the parameter is omitted, CDC restore will not be supported.
-KeepReplication	Preserves replication settings when restoring a published database to a server other than that on which it was created.
-IncludeLogins	For backup, includes the database logins in the backup file. For restore, creates the logins from the backup file on the destination server.
-ReporTLog	For backup, 'Yes' reports skipped t-log backups against databases that are in simple mode with a SUCCESS status rather than SKIPPED.

Options	Description
-MaxTransferSize	Specifies the largest unit of transfer in bytes to be used between SQL Safe and the backup media. The possible values are multiples of 65536 bytes (64KB) ranging up to 4194304 bytes (4 MB). This parameter is used to enable compression on TDE enabled databases only when the MaxTransferSize value is set to 65537 or higher. If omitted, the MaxTransferSize will be taken from the 'TransferLimit' value set in the SQL Safe agent properties.
-DisconnectUsers	Disconnects all users from the target database before the restore operation begins.
-ContinueAfterError	Instructs SQL Server to continue the operation despite encountering errors such as invalid checksums. Note: SQL 2005 and later only.
-VerifyOnly	Perform Verify Only Action.
-MailTo <email_address>	<email_address> - an email address(es) to send the notification via SMTP. Note: multiple addresses may be separated by spaces, semicolons, or commas.
-Threads <number>	<number> - specifies the number of threads that should be used to distribute the backup process across multiple processors.

12.11.3 Options for Objects to Recovery

The following options are required to perform ObjectLevelRecovery operations:

Options	Description
-SchemaObjects <name1> [<name2> ...]	<name1> [<name2> ...] - the list of database schema objects. * (select all).
-UserDefinedTypeObjects <name1> [<name2> ...]	<name1> [<name2> ...] - the list of database user defined type objects. * (select all).
-XmlCollectionObjects <name1> [<name2> ...]	<name1> [<name2> ...] - the list of database xml collection objects. * (select all).
-TableObjects <name1> [<name2> ...]	<name1> [<name2> ...] - the list of database table objects. * (select all).
-FunctionObjects <name1> [<name2> ...]	<name1> [<name2> ...] - the list of database function objects. * (select all).

Options	Description
-ViewObjects <name1> [<name2> ...]	<name1> [<name2> ...] - the list of database view objects. * (select all).
-StoredProcedureObjects <name1> [<name2> ...]	<name1> [<name2> ...] - the list of database stored procedure objects. * (select all).
-ConstraintObjects <name1> [<name2> ...]	<name1> [<name2> ...] - the list of database constraint objects. * (select all).

12.11.4 Security Options

Secure your ObjectLevelRecovery operations with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-SecurityModel <model>	The security model used to log into SQL Server. <model> - {Integrated, SQL}. Note: Integrated (Windows authentication) is the default.
-SqlUsername <username>	<username> - the SQL Server username. (SQL SecurityModel).
-SqlPassword <pwd>	<pwd> - the SQL Server password. (SQL SecurityModel).
-EncryptedSqlPassword <pwd>	<pwd> - the encrypted SQL Server password generated by EncryptSqlPassword action. (SQLSecurityModel).
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help ObjectLevelRecovery`).

12.12 Policies CLI Commands

SQL Safe provides you with CLI commands to help you manually create and edit policies.

The following topics are included in this section:

12.12.1 Create Policies CLI Commands

SQL Safe provides you with CLI commands to help you manually create new policies.

To create policies use the following command:

- SQLsafeCmd Create-Policy <policy_type> <policy_name> [options]

Where:

Action	Description
<policy_type>	Type of the policy. Valid types: Restore, Backup, LogShipping.
<policy_name>	Name of the policy.

Advanced Options

The following advanced options help you create policies:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.

The following topics are included in this section:

- Create Backup Policies CLI Commands
- Create Log Shipping Policies CLI Commands
- Create Restore Policies CLI Commands

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Create-Policy`).

Create Backup Policies CLI Commands

SQL Safe provides you with CLI commands to help you manually create backup policies.

Options to Create Backup Policies

To create backup policies with minimal required options use the following command:

- SQLsafeCmd Create-Policy Backup <policy_name> -IncludeInstance <name> -IncludeDatabases <db_name> [<db_name> ...] [options] [-BackupType <type> [backup options] [-BackupType <type> [backup options] ...]]

The following options can also be used to create backup policies:

Option	Description
-ActionType <action>	Specifies the policy action. <action> - SqlAgent, SqlSafeBackupAgent, MonitorOnly. Note: the default is the SqlAgent action.
-RestrictRun [on|off]	[on|off] - Enable/Disable the policy.
-Description <description>	<description> - specifies the policy description.
-IncludeInstance <instance_name>	<instance_name> - the instance name to include in policy. Option can be used one or more times. There must be NO OPTIONS between -IncludeInstance and -IncludeDatabases options. Note: this parameter is required.
-IncludeDatabases <db_name> [<db_name> ...]	<db_name> [<db_name> ...] - one or more names of database(s) to include in policy. Special keywords: {All}, {AllSystem}, {AllUser}. Note: it is required for each -IncludeInstance option.
-Exclude <db_name> [<db_name> ...]	<db_name> [<db_name> ...] - one or more names of database(s) to not backup.

Email Options

You can set email notifications for your backup policy creations by using the following options:

Options	Description
-MailTo <email> [<email> ...] <on_event> [<on_event> ...] [<frequency>]	<email> - specifies where to send the notification. <on_event> - specifies the event on which to send the notification. Valid events are: OnError, OnSkip, OnCancel, OnWarning, OnSuccess. The default is no event. <frequency> - specifies the notification frequency. Valid values: Once, Always, Never. The default is never.

Note: you can optionally specify email settings.

Backup Options

The following backup option help you create backup policies:

Option	Description
-BackupType <type>	Specifies backup type for policy. <type> - {Full, Differential, Diff, Log}. Option can be used one or more times. Note: the default type is "Full" with default settings. This parameter is optional.

There are available options for -BackupType option:

Option	Description
-CompressionLevel <level>	The compression level used for the backup. <level> - {ispeed, isize, 0, 1, 2, 3, 4}. Note: if the compression level is not specified, ispeed is the default.
-EncryptionType <type>	The type of encryption used to encrypt the backup. <type> - {AES128, AES256}.
-BackupEncryptionPassword <pwd>	<pwd> - specifies the password for encrypted backup file.
-Verify <yes|no>	Verifies the backup set after backup is complete.

Option	Description
-IncludeLogins <yes|no>	For backup, includes the database logins in the backup file. For restore, creates the logins from the backup file on the destination server. Note: this parameter is optional.
-GenerateMap <yes|no>	Generates maps. Note: for InstantRestore and SQL virtual database.
-Threads <number>	<number> - specifies the number of threads that should be used to distribute the backup process across multiple processors.
-Checksum <yes|no>	Instructs SQL Server to generate backup checksums during a backup, or verify backup checksums during a verify or restore. Note: for SQL 2005 and later only. This parameter is optional.
-ContinueAfterError <yes|no>	Instructs SQL Server to continue the operation despite encountering errors such as invalid checksums. Note: for SQL 2005 and later only. This parameter is optional.
-CopyOnly <yes|no>	Specifies that the backup does not affect the normal sequence of backups. Note: for SQL 2005 and later only. This parameter is optional.
-ReadWriteFileGroups <yes|no>	Instructs SQL Server to perform a partial backup, which includes the primary filegroup and any read-write secondary filegroups. Note: for SQL 2005 and later only. This parameter is optional.
-TruncateTransactionLog <yes|no>	Removes inactive entries for transaction log in "Log" backup option. Optional. Note: by default, this option is enabled.

Option	Description
-Location <type>	Specifies backup storage type. <type> - {Single File, TSM, Striped Files, Data Domain, S3 Cloud}. Note: The default value is "Single File". This parameter is optional.
-Overwrite <yes|no>	Overwrites existing archive if one exists. Note: The default value is "no". This parameter is optional.
-RetryWrites <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.
-FailOnMirrorError <yes|no>	To abort backup if a mirror location reports a failure. Note: by default, this option is disabled. This parameter is optional.
-MirrorFile <filename>	Specifies additional backup archive files to be used for mirroring backups. <filename> - specifies the backup archive files. Note: use once for each additional mirror. Up to two mirrors may be specified.
-DeleteMirror <n><time_period>	After a backup successfully completes, delete mirrors that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -deletemirror 2hours. Note: if you use the space between <n> and <time_period>, the mirror filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_<timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.

Option	Description
-BackupFile <filename>	Specifies additional backup archive files to be used for striping backups. <filename> - specifies the backup archive files. Note: use once for each additional stripe.
-Delete <n><time_period>	After a backup successfully completes, delete archives that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -delete 2hours. Note: if you use the space between <n> and <time_period>, the backup archive filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_ <timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.
-UseAgentAccount <yes|no>	Specifies Agent account for accesing to files. Note: by default, the policy uses Agent account. This parameter is optional.
-WindowsUsername <domain\user>	<domain\user> - specifies user name, used when writing to remote files during backup.
-WindowsPassword <pwd>	<pwd> - specifies password.
-BucketName <bucket_name>	<bucket_name> - specifies the cloud bucket name.
-SecretKey <key>	<key> - specifies the cloud secret name.
-AccessKey <key>	<key> - specifies the cloud access key.
-FileSize <file_size>	<file_size> - specifies the cloud file size.
-Region <region>	<region> - specifies the cloud region.

Schedule Options

You can schedule the creation of backup policies by using the following options:

Options	Description
-Schedule <occurs_type>	Specifies the schedule type (occurs type). <occurs_type> - OnDemand, Daily, Weekly, Monthly.
-Every <n>	Specifies how often the policy runs. It depends on the schedule type. For Daily: every <n> day(s). For Weekly: every <n> week(s). For Monthly: every <n> month(s). Note: the default value is "1".
-Day <day>	<day> - specifies a day of the week or a day of a month. Note: it is valid only with Weekly, Monthly schedule types. Valid values for Weekly: MON - SUN and * (every day). SUN is default. Valid values for Monthly: MON - SUN, Weekday, WeekendDay, Day or number 1 - 31. Is required when -MonthDay option is used.
-MonthDay <month_day>	Specifies how often the policy runs. <month_day> - FIRST, SECOND, THIRD, FOURTH, LAST use with -Day option. LASTDAY - the policy runs on the last day of the month.
-StartTime <time>	<time> - specifies the time of day that the policy runs. Note: it is required with a "once" frequency. The default value is the current local time when policy creates.
-EndTime <time>	<time> - specifies the time of day that the policy ends. Note: it is is not valid in a "once" frequency. The default value is the "23:59:59".

Options	Description
-Frequency <n><time_period>	Specifies daily frequency (how often the policy runs within a day). <n> - amount of time. <time_period> - {hour, minute}. E.g., -Frequency 2hours. Special keyword: Once. E.g., -Frequency Once.
-StartDate <date>	<date> - specifies the date that the policy starts. Note: the default value is the current date. It is optional.
-EndDate <date>	<date> - specifies the last date that the policy is scheduled to run. Note: by default, the schedules have no ending date. It is optional.

Note: you can optionally specify backup job occur schedule.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Create-Policy`).

Create Log Shipping Policies CLI Commands

SQL Safe provides you with CLI commands to help you manually create log shipping policies.

Options to Create Log Shipping Policies

To create log shipping policies with minimal required options use the following command:

- SQLsafeCmd Create-Policy LogShipping <policy_name> -InstanceName <name> -DatabaseName <primary_db_name> -NetworkPath <path> [options] -SecondaryDatabase <secondary_db_name> -TargetInstance <name> [secondary database options] [-SecondaryDatabase <secondary_db_name> -TargetInstance <name> [secondary database options] ...]

The following options can also be used to create log shipping policies:

Option	Description
-ActionType <action>	Specifies the policy action. <action> - SqlAgent, SqlSafeBackupAgent, MonitorOnly. Note: the default is the SqlAgent action.
-RestrictRun [on|off]	[on|off] - Enable/Disable the policy.
-Description <description>	<description> - specifies the policy description.
-RestoreGracePeriod <n><time_period>	Specifies when SQLsafe should consider this log shipping policy compliant. <time_period> - {hour, minute} A log backup must occur within "n" minutes/hours of the scheduled run time.
-BackupGracePeriod <n><time_period>	Specifies when SQLsafe should consider this log shipping policy compliant. <time_period> - {hour, minute} A secondary's data is no more than "n" minutes/hours older the n the most recent log backup.

Email Options

You can set email notifications for your log shipping policy creations by using the following options:

Options	Description
-MailTo <email> [<email> ...] <on_event> [<on_event> ...] [<frequency>]	<email> - specifies where to send notification. <on_event> - specifies the event on which send notification. Valid events are: OnError, OnSkip, OnCancel, OnWarning, OnSuccess. The default is no event. <frequency> - specifies the notification frequency. Valid values: Once, Always, Never. The default is never.

Note: you can optionally specify email settings.

Primary Database Options

The following primary database options help you create log shipping policies:

Options	Description
-InstanceName <instance_name>	<instance_name> - specifies the instance on the primary database location. Note: this parameter is required.
-DatabaseName <db_name>	<db_name> - specifies the primary database. Note: this parameter is required.
-Server <server_name>	<server_name> - specifies the server on the primary database location. Note: this parameter is optional.

Schedule Options

You can schedule the primary database options to create log shipping policies by using the following options:

Options	Description
-CompressionLevel <level>	The compression level used for the backup. <level> - {ispeed, isize, 0, 1, 2, 3, 4}. Note: if the compression level is not specified, ispeed is the default.
-EncryptionType <type>	The type of encryption used to encrypt the backup. <type> - {AES128, AES256}.

Options	Description
-BackupEncryptionPassword <pwd>	<pwd> - specifies the password for encrypted backup file.
-Threads <number>	<number> - specifies the number of threads that should be used to distribute the backup process across multiple processors.
-UseAgentAccount <yes|no>	Specifies the Agent account for accessing to files. Note: this parameter is optional. By default, the policy uses Agent account.
-WindowsUsername <domain\user>	<domain\user> - specifies the user name, used when writing to remote files during backup.
-WindowsPassword <pwd>	<pwd> - specifies the password.
-NetworkPath <path>	<path> - specifies where to store your backup files for primary database.
-Delete <n><time_period>	After a backup successfully completes, delete archives that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -delete 2hours. Note: if you use the space between <n> and <time_period>, the backup archive filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_ <timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.
-RetryWrites <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.
-FailOnMirrorError <yes|no>	To abort backup if a mirror location reports a failure. Note: by default, this option is disabled. This parameter is optional.

Options	Description
-MirrorFile <filename>	Specifies additional backup archive files to be used for mirroring backups. <filename> - specifies the backup archive files. Note: use once for each additional mirror. Up to two mirrors may be specified.
-DeleteMirror <n><time_period>	After a backup successfully completes, delete mirrors that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -deletemirror 2hours. Note: if you use the space between <n> and <time_period>, the mirror filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_<timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.

Note: you can optionally specify backup job occur schedule for primary database.

Secondary Database (s) Options

The following secondary database options help you create log shipping policies:

Options	Description
-SecondaryDatabase <db_name>	<db_name> - specifies the "Secondary Database" name. Note: it requires -TargetInstance option.
-TargetInstance <instance_name>	<instance_name> - specifies the target instance. Note: this parameter is required.
-TargetServer <server_name>	<server_name> - specifies the target server. Note: this parameter is optional.

Options	Description
-Inatialization <type> [-Backupfile <file_name>]	Specifies the secondary database initialization. <type> - {None, Latest Backup, Create Copyonly, Selection} "none" - Do not initialize. The database exists and has received most recent full backup of primary database. "latest_backup" - Initialize with a newly generated full backup. "create_copyonly" - Initialize with a newly generated copy-only full backup. "selection" - Initialize with selected backups, option -Backupfile is required for this initialization type. Note: The default value is "Create Copyonly". This parameter is optional.
-BackupFile <file_name>	<file_name> - specifies backups for "Selection" database initialization type.
-RecoveryMode <mode>	Specifies the secondary database state after restore. <mode> {NoRecovery, Standby}. Option -DisconnectUsers is applicable with "Standby" recovery mode. Note: the default value is "Standby" with enabled -DisconnectUsers option. This parameter is optional.
-UndoFile <filename>	<filename> - specifies the ABSOLUTE path to the undo filename. Note: for Standby recovery mode only.
-DisconnectUsers <yes|no>	Disconnects all users from the target database before the restore operation begins. Note: this parameter is optional.

Schedule Options

You can schedule the secondary database options to create log shipping policies by using the following options:

Options	Description
-Schedule <occurs_type>	Specifies the schedule type (occurs type). <occurs_type> - OnDemand, Daily, Weekly, Monthly.

Options	Description
-Every <n>	Specifies how often the policy runs. It depends on the schedule type. For Daily: every <n> day(s). For Weekly: every <n> week(s). For Monthly: every <n> month(s). Note: the default value is "1".
-Day <day>	<day> - specifies a day of the week or a day of a month. Note: it is valid only with Weekly, Monthly schedule types. Valid values for Weekly: MON - SUN and * (every day). SUN is default. Valid values for Monthly: MON - SUN, Weekday, WeekendDay, Day or number 1 - 31. Is required when -MonthDay option is used.
-MonthDay <month_day>	Specifies how often the policy runs. <month_day> - FIRST, SECOND, THIRD, FOURTH, LAST use with -Day option. LASTDAY - the policy runs on the last day of the month.
-StartTime <time>	<time> - specifies the time of day that the policy runs. Note: it is required with a "once" frequency. The default value is the current local time when policy creates.
-EndTime <time>	<time> - specifies the time of day that the policy ends. Note: it is is not valid in a "once" frequency. The default value is the "23:59:59".
-Frequency <n><time_period>	Specifies daily frequency (how often the policy runs within a day). <n> - amount of time. <time_period> - {hour, minute}. E.g., -Frequency 2hours. Special keyword: Once. E.g., -Frequency Once.
-StartDate <date>	<date> - specifies the date that the policy starts. Note: the default value is the current date. This parameter is optional.

Options	Description
-EndDate <date>	<date> - specifies the last date that the policy is scheduled to run. Note: by default, the schedules have no ending date. This parameter is optional.
-SetRestoreLocation <path>	<path> - specifies different location for restore. Special keyword: "AsBackup" sets location the same as backup. Note: the default value is the same location as backup. This parameter is optional.

Note: you can optionally specify restore job occur schedule for secondary database.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Create-Policy`).

Create Restore Policies CLI Commands

SQL Safe provides you with CLI commands to help you manually create restore policies.

Options to Create Restore Policies

To create restore policies with minimal required options use the following command:

- SQLsafeCmd Create-Policy Restore <policy_name> -InstanceName <name> -DatabaseName <source_db_name> -SourceBackupPolicy <policy_name|GUID> -TargetInstance <name> -TargetDatabase <target_db_name> [options]

The following options can also be used to create restore policies:

Option	Description
-ActionType <action>	Specifies the policy action. <action> - SqlAgent, SqlSafeBackupAgent, MonitorOnly. Note: the default is the SqlAgent action.
-RestrictRun [on|off]	[on|off] - Enable/Disable the policy.
-Description <description>	<description> - specifies the policy description.
-InstanceName <instance_name>	<instance_name> - specifies the source instance. Note: this parameter is required.
-DatabaseName <db_name>	<db_name> - specifies the database to restore. Note: this parameter is required.
-Server <server_name>	<server_name> - specifies the source server. Note: this parameter is optional.
-RetryWrites <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.
-SourceBackupPolicy <policy_name|GUID>	<policy_name|GUID> - specifies the backup policy for source database.
-BackupEncryptionPassword <pwd>	<pwd> - specifies the password for encrypted source backup file.

Account Options

The following account options help you create restore policies:

Options	Description
-UseAgentAccount <yes|no>	Specifies the Agent account for accessing to files. Note: by default, the policy uses Agent account. This parameter is optional.
-WindowsUsername <domain\user>	<domain\user> - specifies the user name, used when writing/reading to files during restore.
-WindowsPassword <pwd>	<pwd> - specifies the password.

Target Options

The following target options help you create restore policies:

Options	Description
-TargetInstance <instance_name>	<instance_name> - specifies the target instance. Note: this parameter is required.
-TargetDatabase <db_name>	<db_name> - specifies the new database. Note: this parameter is required.
-TargetServer <server_name>	<server_name> - specifies the target server. Note: this parameter is optional.
-LogLocation <path>	<path> - specifies where to store the database log files. Note: the default value is the current target database log files location.
-DataLocation <path>	<path> - specifies where to store database data files. Note: the default value is the current target database data files location.
-RecoveryMode <mode>	Specifies the database state after restore action. <mode> {NoRecovery, Standby, Recovery}.
-DisconnectUsers <yes|no>	Disconnects all users from the target database before the restore operation begins. Note: this parameter is optional.

Options	Description
-ContinueAfterError <yes|no>	Instructs SQL Server to continue the operation despite encountering errors such as invalid checksums. Note: For SQL 2005 and later only. This parameter is optional.
-IncludeLogins <yes|no>	For backup, includes the database logins in the backup file. For restore, creates the logins from the backup file on the destination server. Note: this parameter is optional.
-KeepReplication <yes|no>	Preserves replication settings when restoring a published database to a server other than that on which it was created. Note: this parameter is optional.

Schedule Options

You can schedule the creation of restore policies by using the following options:

Options	Description
-Schedule <occurs_type>	Specifies the schedule type (occurs type). <occurs_type> - OnDemand, Daily, Weekly, Monthly.
-Every <n>	Specifies how often the policy runs. It depends on the schedule type. For Daily: every <n> day(s). For Weekly: every <n> week(s). For Monthly: every <n> month(s). Note: the default value is "1".
-Day <day>	<day> - specifies a day of the week or a day of a month. Note: it is valid only with Weekly, Monthly schedule types. Valid values for Weekly: MON - SUN and * (every day). SUN is default. Valid values for Monthly: MON - SUN, Weekday, WeekendDay, Day or number 1 - 31. Is required when -MonthDay option is used.
-MonthDay <month_day>	Specifies how often the policy runs. <month_day> - FIRST, SECOND, THIRD, FOURTH, LAST use with -Day option. LASTDAY - the policy runs on the last day of the month.

Options	Description
-StartTime <time>	<time> - specifies the time of day that the policy runs. Note: it is required with a "once" frequency. The default value is the current local time when policy creates.
-EndTime <time>	<time> - specifies the time of day that the policy ends. Note: it is is not valid in a "once" frequency. The default value is the "23:59:59".
-Frequency <n><time_period>	Specifies daily frequency (how often the policy runs within a day). <n> - amount of time. <time_period> - {hour, minute}. E.g., -Frequency 2hours. Special keyword: Once. E.g., -Frequency Once.
-StartDate <date>	<date> - specifies the date that the policy starts. Note: The default value is the current date. This parameter is optional.
-EndDate <date>	<date> - specifies the last date that the policy is scheduled to run. Note: by default, the schedules have no ending date. This parameter is optional.

Note: you can optionally specify restore job occur schedule.

Email Options

You can set email notifications for your restore policy creations by using the following options:

Options	Description
-MailTo <email> [<email> ...] <on_event> [<on_event> ...] [<frequency>]	<email> - specifies where to send notification. <on_event> - specifies the event on which send notification. Valid events are: OnError, OnSkip, OnCancel, OnWarning, OnSuccess. The default is no event. <frequency> - specifies the notification frequency. Valid values: Once, Always, Never. The default is never.

Note: you can optionally specify email settings.
For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Create-Policy`).

12.12.2 Edit Policies CLI Commands

SQL Safe provides you with CLI commands to help you manually edit your policies.

To edit policies use the following commands:

- SQLsafeCmd Edit-Policy <policy_name> [options]
- SQLsafeCmd Edit-Policy <policy_GUID> [options]

Where:

Action	Description
<policy_name>	Name of the policy.
<policy_GUID>	GUID of the policy.

Advanced Options

The following advanced options help you edit policies:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.

The following topics are included in this section:

- Edit Backup Policies CLI Commands
- Edit Log Shipping Policies CLI Commands
- Edit Restore Policies CLI Commands

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Edit-Policy`).

Edit Backup Policies CLI Commands

SQL Safe provides you with CLI commands to help you manually edit your backup policies.

Options to Edit Backup Policies

To edit backup policies with minimal required options use the following command:

- SQLsafeCmd Create-Policy Backup <policy_name> -IncludeInstance <name> -IncludeDatabases <db_name> [<db_name> ...] [options] [-BackupType <type> [backup options] [-BackupType <type> [backup options] ...]]

The following options can also be used to edit backup policies:

Option	Description
-ActionType <action>	Specifies the policy action. <action> - SqlAgent, SqlSafeBackupAgent, MonitorOnly. Note: the default is the SqlAgent action.
-RestrictRun [on|off]	[on|off] - Enable/Disable the policy.
-Description <description>	<description> - specifies the policy description.
-NewName <policy_name>	<policy_name> - specifies a new name of the policy.
-IncludeInstance <instance_name>	<instance_name> - the instance name to include in policy. Option can be used one or more times. There must be NO OPTIONS between -IncludeInstance and -IncludeDatabases options. Note: this parameter is required.
-IncludeDatabases <db_name> [<db_name> ...]	<db_name> [<db_name> ...] - one or more names of database(s) to include in policy. Special keywords: {All}, {AllSystem}, {AllUser}. Note: it is required for each -IncludeInstance option.
-Exclude <db_name> [<db_name> ...]	<db_name> [<db_name> ...] - one or more names of database(s) to not backup.

Email Options

You can set email notifications for your backup policy editions by using the following options:

Options	Description
-MailTo <email> [<email> ...] <on_event> [<on_event> ...] [<frequency>]	<email> - specifies where to send the notification. <on_event> - specifies the event on which to send the notification. Valid events are: OnError, OnSkip, OnCancel, OnWarning, OnSuccess. The default is no event. <frequency> - specifies the notification frequency. Valid values: Once, Always, Never. The default is never.

Note: you can optionally specify email settings.

Backup Options

The following backup option help you edit backup policies:

Option	Description
-BackupType <type> <on|off>	Specifies backup type for policy and Enable/Disable it. <type> - {Full, Differential, Diff, Log}. Option can be used one or more times. Note: the default type is "Full" with default settings. This parameter is optional.

There are available options for -BackupType option:

Option	Description
-CompressionLevel <level>	The compression level used for the backup. <level> - {ispeed, isize, 0, 1, 2, 3, 4}. Note: if the compression level is not specified, ispeed is the default.
-EncryptionType <type>	The type of encryption used to encrypt the backup. <type> - {AES128, AES256}.
-BackupEncryptionPassword <pwd>	<pwd> - specifies the password for encrypted backup file.
-Verify <yes|no>	Verifies the backup set after backup is complete.

Option	Description
-IncludeLogins <yes|no>	For backup, includes the database logins in the backup file. For restore, creates the logins from the backup file on the destination server. Note: this parameter is optional.
-GenerateMap <yes|no>	Generates maps. Note: for InstantRestore and SQL virtual database.
-Threads <number>	<number> - specifies the number of threads that should be used to distribute the backup process across multiple processors.
-Checksum <yes|no>	Instructs SQL Server to generate backup checksums during a backup, or verify backup checksums during a verify or restore. Note: for SQL 2005 and later only. This parameter is optional.
-ContinueAfterError <yes|no>	Instructs SQL Server to continue the operation despite encountering errors such as invalid checksums. Note: for SQL 2005 and later only. This parameter is optional.
-CopyOnly <yes|no>	Specifies that the backup does not affect the normal sequence of backups. Note: for SQL 2005 and later only. This parameter is optional.
-ReadWriteFileGroups <yes|no>	Instructs SQL Server to perform a partial backup, which includes the primary filegroup and any read-write secondary filegroups. Note: for SQL 2005 and later only. This parameter is optional.
-TruncateTransactionLog <yes|no>	Removes inactive entries for transaction log in "Log" backup option. Optional. Note: by default, this option is enabled.
-Location <type>	Specifies backup storage type. <type> - {Single File, TSM, Striped Files, Data Domain, S3 Cloud}. Note: The default value is "Single File". This parameter is optional.

Option	Description
-Overwrite <yes|no>	Overwrites existing archive if one exists. Note: The default value is "no". This parameter is optional.
-RetryWrites <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.
-FailOnMirrorError <yes|no>	To abort backup if a mirror location reports a failure. Note: by default, this option is disabled. This parameter is optional.
-EnableMirroring <yes|no>	<yes|no> - Enable/Disable mirroring. Note: this parameter is optional.
-MirrorFile <filename>	Specifies additional backup archive files to be used for mirroring backups. <filename> - specifies the backup archive files. Note: use once for each additional mirror. Up to two mirrors may be specified.
-DeleteMirror <n><time_period>	After a backup successfully completes, delete mirrors that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -deletemirror 2hours. Note: if you use the space between <n> and <time_period>, the mirror filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_<timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.
-BackupFile <filename>	Specifies additional backup archive files to be used for striping backups. <filename> - specifies the backup archive files. Note: use once for each additional stripe.

Option	Description
-Delete <n><time_period>	After a backup successfully completes, delete archives that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -delete 2hours. Note: if you use the space between <n> and <time_period>, the backup archive filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_ <timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.
-UseAgentAccount <yes|no>	Specifies Agent account for accesing to files. Note: by default, the policy uses Agent account. This parameter is optional.
-WindowsUsername <domain\user>	<domain\user> - specifies user name, used when writing to remote files during backup.
-WindowsPassword <pwd>	<pwd> - specifies password.
-BucketName <bucket_name>	<bucket_name> - specifies the cloud bucket name.
-SecretKey <key>	<key> - specifies the cloud secret name.
-AccessKey <key>	<key> - specifies the cloud access key.
-FileSize <file_size>	<file_size> - specifies the cloud file size.
-Region <region>	<region> - specifies the cloud region.

Schedule Options

You can schedule the edition of backup policies by using the following options:

Options	Description
-Schedule <occurs_type>	Specifies the schedule type (occurs type). <occurs_type> - OnDemand, Daily, Weekly, Monthly.

Options	Description
-Every <n>	Specifies how often the policy runs. It depends on the schedule type. For Daily: every <n> day(s). For Weekly: every <n> week(s). For Monthly: every <n> month(s). Note: the default value is "1".
-Day <day>	<day> - specifies a day of the week or a day of a month. Note: it is valid only with Weekly, Monthly schedule types. Valid values for Weekly: MON - SUN and * (every day). SUN is default. Valid values for Monthly: MON - SUN, Weekday, WeekendDay, Day or number 1 - 31. Is required when -MonthDay option is used.
-MonthDay <month_day>	Specifies how often the policy runs. <month_day> - FIRST, SECOND, THIRD, FOURTH, LAST use with -Day option. LASTDAY - the policy runs on the last day of the month.
-StartTime <time>	<time> - specifies the time of day that the policy runs. Note: it is required with a "once" frequency. The default value is the current local time when policy creates.
-EndTime <time>	<time> - specifies the time of day that the policy ends. Note: it is is not valid in a "once" frequency. The default value is the "23:59:59".
-Frequency <n><time_period>	Specifies daily frequency (how often the policy runs within a day). <n> - amount of time. <time_period> - {hour, minute}. E.g., -Frequency 2hours. Special keyword: Once. E.g., -Frequency Once.
-StartDate <date>	<date> - specifies the date that the policy starts. Note: the default value is the current date. It is optional.

Options	Description
-EndDate <date>	<date> - specifies the last date that the policy is scheduled to run. Note: by default, the schedules have no ending date. It is optional.

Note: you can optionally specify backup job occur schedule.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Edit-Policy`).

Edit Log Shipping Policies CLI Commands

SQL Safe provides you with CLI commands to help you manually edit log shipping policies.

Options to Edit Log Shipping Policies

To edit log shipping policies with minimal required options use the following command:

- SQLsafeCmd Create-Policy LogShipping <policy_name> -InstanceName <name> -DatabaseName <primary_db_name> -NetworkPath <path> [options] -SecondaryDatabase <secondary_db_name> -TargetInstance <name> [secondary database options] [-SecondaryDatabase <secondary_db_name> -TargetInstance <name> [secondary database options] ...]

The following options can also be used to edit log shipping policies:

Option	Description
-ActionType <action>	Specifies the policy action. <action> - SqlAgent, SqlSafeBackupAgent, MonitorOnly. Note: the default is the SqlAgent action.
-RestrictRun [on|off]	[on|off] - Enable/Disable the policy.
-Description <description>	<description> - specifies the policy description.
-NewName <policy_name>	<policy_name> - specifies a new name of the policy.
-RestoreGracePeriod <n><time_period>	Specifies when SQLsafe should consider this log shipping policy compliant. <time_period> - {hour, minute} A log backup must occur within "n" minutes/hours of the scheduled run time.
-BackupGracePeriod <n><time_period>	Specifies when SQLsafe should consider this log shipping policy compliant. <time_period> - {hour, minute} A secondary's data is no more than "n" minutes/hours older the n the most recent log backup.

Email Options

You can set email notifications for your log shipping policy editions by using the following options:

Options	Description
-MailTo <email> [<email> ...] <on_event> [<on_event> ...] [<frequency>]	<email> - specifies where to send notification. <on_event> - specifies the event on which send notification. Valid events are: OnError, OnSkip, OnCancel, OnWarning, OnSuccess. The default is no event. <frequency> - specifies the notification frequency. Valid values: Once, Always, Never. The default is never.

Note: you can optionally specify email settings.

Primary Database Options

The following primary database options help you edit log shipping policies:

Options	Description
-InstanceName <instance_name>	<instance_name> - specifies the instance on the primary database location. Note: this parameter is required.
-DatabaseName <db_name>	<db_name> - specifies the primary database. Note: this parameter is required.
-Server <server_name>	<server_name> - specifies the server on the primary database location. Note: this parameter is optional.

Schedule Options

You can schedule the primary database options to edit log shipping policies by using the following options:

Options	Description
-CompressionLevel <level>	The compression level used for the backup. <level> - {ispeed, isize, 0, 1, 2, 3, 4}. Note: if the compression level is not specified, ispeed is the default.
-EncryptionType <type>	The type of encryption used to encrypt the backup. <type> - {AES128, AES256}.

Options	Description
-BackupEncryptionPassword <pwd>	<pwd> - specifies the password for encrypted backup file.
-Threads <number>	<number> - specifies the number of threads that should be used to distribute the backup process across multiple processors.
-UseAgentAccount <yes|no>	Specifies the Agent account for accessing to files. Note: this parameter is optional. By default, the policy uses Agent account.
-WindowsUsername <domain\user>	<domain\user> - specifies the user name, used when writing to remote files during backup.
-WindowsPassword <pwd>	<pwd> - specifies the password.
-NetworkPath <path>	<path> - specifies where to store your backup files for primary database.
-Delete <n><time_period>	After a backup successfully completes, delete archives that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -delete 2hours. Note: if you use the space between <n> and <time_period>, the backup archive filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_ <timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.
-RetryWrites <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.
-FailOnMirrorError <yes|no>	To abort backup if a mirror location reports a failure. Note: by default, this option is disabled. This parameter is optional.
-EnableMirroring <yes|no>	Enable/Disable mirroring. Note: This parameter is optional.

Options	Description
-MirrorFile <filename>	Specifies additional backup archive files to be used for mirroring backups. <filename> - specifies the backup archive files. Note: use once for each additional mirror. Up to two mirrors may be specified.
-DeleteMirror <n><time_period>	After a backup successfully completes, delete mirrors that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. There must be NO SPACE between <n> and <time_period>. E.g., -deletemirror 2hours. Note: if you use the space between <n> and <time_period>, the mirror filename will be automatically generated with the following pattern: <instancename>_<databasename>_<backuptype>_<timestamp>.safe where the <timestamp> is in UTC time and in the form of YYYYMMDDHHMM.

Note: you can optionally specify backup job occur schedule for primary database.

Secondary Database (s) Options

The following secondary database options help you edit log shipping policies:

Options	Description
-SecondaryDatabase <db_name>	<db_name> - specifies the "Secondary Database" name. Note: it requires -TargetInstance option. Second parameter can be used for removing "Secondary Database" from policy. Note: by default, this parameter is "On".
-TargetInstance <instance_name>	<instance_name> - specifies the target instance. Note: this parameter is required.
-TargetServer <server_name>	<server_name> - specifies the target server. Note: this parameter is optional.

Options	Description
-Inatialization <type> [-Backupfile <file_name>]	Specifies the secondary database initialization. <type> - {None, Latest Backup, Create Copyonly, Selection} "none" - Do not initialize. The database exists and has received most recent full backup of primary database. "latest_backup" - Initialize with a newly generated full backup. "create_copyonly" - Initialize with a newly generated copy-only full backup. "selection" - Initialize with selected backups, option -Backupfile is required for this initialization type. Note: The default value is "Create Copyonly". This parameter is optional.
-BackupFile <file_name>	<file_name> - specifies backups for "Selection" database initialization type.
-RecoveryMode <mode>	Specifies the secondary database state after restore. <mode> {NoRecovery, Standby}. Option -DisconnectUsers is applicable with "Standby" recovery mode. Note: the default value is "Standby" with enabled -DisconnectUsers option. This parameter is optional.
-UndoFile <filename>	<filename> - specifies the ABSOLUTE path to the undo filename. Note: for Standby recovery mode only.
-DisconnectUsers <yes|no>	Disconnects all users from the target database before the restore operation begins. Note: this parameter is optional.

Schedule Options

You can schedule the secondary database options to edit log shipping policies by using the following options:

Options	Description
-Schedule <occurs_type>	Specifies the schedule type (occurs type). <occurs_type> - OnDemand, Daily, Weekly, Monthly.

Options	Description
-Every <n>	Specifies how often the policy runs. It depends on the schedule type. For Daily: every <n> day(s). For Weekly: every <n> week(s). For Monthly: every <n> month(s). Note: the default value is "1".
-Day <day>	<day> - specifies a day of the week or a day of a month. Note: it is valid only with Weekly, Monthly schedule types. Valid values for Weekly: MON - SUN and * (every day). SUN is default. Valid values for Monthly: MON - SUN, Weekday, WeekendDay, Day or number 1 - 31. Is required when -MonthDay option is used.
-MonthDay <month_day>	Specifies how often the policy runs. <month_day> - FIRST, SECOND, THIRD, FOURTH, LAST use with -Day option. LASTDAY - the policy runs on the last day of the month.
-StartTime <time>	<time> - specifies the time of day that the policy runs. Note: it is required with a "once" frequency. The default value is the current local time when policy creates.
-EndTime <time>	<time> - specifies the time of day that the policy ends. Note: it is is not valid in a "once" frequency. The default value is the "23:59:59".
-Frequency <n><time_period>	Specifies daily frequency (how often the policy runs within a day). <n> - amount of time. <time_period> - {hour, minute}. E.g., -Frequency 2hours. Special keyword: Once. E.g., -Frequency Once.
-StartDate <date>	<date> - specifies the date that the policy starts. Note: the default value is the current date. This parameter is optional.

Options	Description
-EndDate <date>	<date> - specifies the last date that the policy is scheduled to run. Note: by default, the schedules have no ending date. This parameter is optional.
-SetRestoreLocation <path>	<path> - specifies different location for restore. Special keyword: "AsBackup" sets location the same as backup. Note: the default value is the same location as backup. This parameter is optional.

Note: you can optionally specify restore job occur schedule for secondary database.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Edit-Policy`).

Edit Restore Policies CLI Commands

SQL Safe provides you with CLI commands to help you manually edit your restore policies.

Options to Edit Restore Policies

To edit restore policies with minimal required options use the following command:

- SQLsafeCmd Create-Policy Restore <policy_name> -InstanceName <name> -DatabaseName <source_db_name> -SourceBackupPolicy <policy_name|GUID> -TargetInstance <name> -TargetDatabase <target_db_name> [options]

The following options can also be used to edit restore policies:

Option	Description
-ActionType <action>	Specifies the policy action. <action> - SqlAgent, SqlSafeBackupAgent, MonitorOnly. Note: the default is the SqlAgent action.
-RestrictRun [on|off]	[on|off] - Enable/Disable the policy.
-Description <description>	<description> - specifies the policy description.
-NewName <policy_name>	<policy_name> - specifies the new name of the policy.
-InstanceName <instance_name>	<instance_name> - specifies the source instance. Note: this parameter is required.
-DatabaseName <db_name>	<db_name> - specifies the database to restore. Note: this parameter is required.
-Server <server_name>	<server_name> - specifies the source server. Note: this parameter is optional.
-RetryWrites <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.
-SourceBackupPolicy <policy_name|GUID>	<policy_name|GUID> - specifies the backup policy for source database.
-BackupEncryptionPassword <pwd>	<pwd> - specifies the password for encrypted source backup file.

Account Options

The following account options help you edit restore policies:

Options	Description
-UseAgentAccount <yes|no>	Specifies the Agent account for accessing to files. Note: by default, the policy uses Agent account. This parameter is optional.
-WindowsUsername <domain\user>	<domain\user> - specifies the user name, used when writing/reading to files during restore.
-WindowsPassword <pwd>	<pwd> - specifies the password.

Target Options

The following target options help you edit restore policies:

Options	Description
-TargetInstance <instance_name>	<instance_name> - specifies the target instance. Note: this parameter is required.
-TargetDatabase <db_name>	<db_name> - specifies the new database. Note: this parameter is required.
-TargetServer <server_name>	<server_name> - specifies the target server. Note: this parameter is optional.
-LogLocation <path>	<path> - specifies where to store the database log files. Note: the default value is the current target database log files location.
-DataLocation <path>	<path> - specifies where to store database data files. Note: the default value is the current target database data files location.
-RecoveryMode <mode>	Specifies the database state after restore action. <mode> {NoRecovery, Standby, Recovery}.
-DisconnectUsers <yes|no>	Disconnects all users from the target database before the restore operation begins. Note: this parameter is optional.

Options	Description
-ContinueAfterError <yes|no>	Instructs SQL Server to continue the operation despite encountering errors such as invalid checksums. Note: For SQL 2005 and later only. This parameter is optional.
-IncludeLogins <yes|no>	For backup, includes the database logins in the backup file. For restore, creates the logins from the backup file on the destination server. Note: this parameter is optional.
-KeepReplication <yes|no>	Preserves replication settings when restoring a published database to a server other than that on which it was created. Note: this parameter is optional.

Schedule Options

You can schedule the edition of restore policies by using the following options:

Options	Description
-Schedule <occurs_type>	Specifies the schedule type (occurs type). <occurs_type> - OnDemand, Daily, Weekly, Monthly.
-Every <n>	Specifies how often the policy runs. It depends on the schedule type. For Daily: every <n> day(s). For Weekly: every <n> week(s). For Monthly: every <n> month(s). Note: the default value is "1".
-Day <day>	<day> - specifies a day of the week or a day of a month. Note: it is valid only with Weekly, Monthly schedule types. Valid values for Weekly: MON - SUN and * (every day). SUN is default. Valid values for Monthly: MON - SUN, Weekday, WeekendDay, Day or number 1 - 31. Is required when -MonthDay option is used.
-MonthDay <month_day>	Specifies how often the policy runs. <month_day> - FIRST, SECOND, THIRD, FOURTH, LAST use with -Day option. LASTDAY - the policy runs on the last day of the month.

Options	Description
-StartTime <time>	<time> - specifies the time of day that the policy runs. Note: it is required with a "once" frequency. The default value is the current local time when policy creates.
-EndTime <time>	<time> - specifies the time of day that the policy ends. Note: it is is not valid in a "once" frequency. The default value is the "23:59:59".
-Frequency <n><time_period>	Specifies daily frequency (how often the policy runs within a day). <n> - amount of time. <time_period> - {hour, minute}. E.g., -Frequency 2hours. Special keyword: Once. E.g., -Frequency Once.
-StartDate <date>	<date> - specifies the date that the policy starts. Note: The default value is the current date. This parameter is optional.
-EndDate <date>	<date> - specifies the last date that the policy is scheduled to run. Note: by default, the schedules have no ending date. This parameter is optional.

Note: you can optionally specify restore job occur schedule.

Email Options

You can set email notifications for your restore policy editions by using the following options:

Options	Description
-MailTo <email> [<email> ...] <on_event> [<on_event> ...] [<frequency>]	<email> - specifies where to send notification. <on_event> - specifies the event on which send notification. Valid events are: OnError, OnSkip, OnCancel, OnWarning, OnSuccess. The default is no event. <frequency> - specifies the notification frequency. Valid values: Once, Always, Never. The default is never.

Note: you can optionally specify email settings.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Edit-Policy`).

12.13 Restore CLI Commands

SQL Safe provides you with CLI commands to help you manually restore a database.

To restore a database use the following commands:

- SQLsafeCmd Restore <db_name> <backup_archive> [options]
- SQLsafeCmd Restore <db_name> <point_in_time> [options]
- SQLsafeCmd Restore <db_name> TSM [options]

Where:

Action	Description
<db_name>	Name of the database.
<backup_archive>	Path to the backup archive.
<point_in_time>	Date/Time {"MM/dd/yyyy hh:mm:ss"} to restore to.
Tsm	Use Tivoli Storage Manager (see below for special options).

12.13.1 Common Options

The following options help you perform restore operations:

Options	Descriptions
-BackupSet <index>	<index> - the index of the backup set within the backup archive. (1-based).
-DisconnectUsers	Disconnects all users from the target database before the restore operation begins.
-DownloadFileFromCloud	False.
-TempDownloadLocation	Temporary Download Location For cloud.
-PartSize	10.
-EncryptedRestorePassword <pwd>	<pwd> - the encrypted password used to decrypt the backup.
-InstanceName <name>	<name> - SQL server instance name. Note: it is not required if the instance is set as a default on the target server.

Options	Descriptions
-Move <logical_filename> <target_filename>	To move the database logical database file to the physical target file. <logical_filename> - the database logical database file. <target_filename> - the physical target file. Corresponds to the WITH MOVE option in the RESTORE DATABASE T/SQL command.
-Password <pwd>	<pwd> - the non-encrypted password used to encrypt the backup.
-Server <hostname>	<hostname> - the hostname of server hosting the SQL Server where the operation should be performed. Note: this option is required for accessing remote or clustered SQL Servers (where applicable).
-Replace	Overrides database if exists.
-SectorType	Public or Government based on Azure Sector.

12.13.2 Security Options

Secure your restore operations with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-SecurityModel <model>	The security model used to log into SQL Server. <model> - {Integrated, SQL}. Note: Integrated (Windows authentication) is the default.
-SqlUsername <username>	<username> - the SQL Server username. (SQL SecurityModel).
-SqlPassword <pwd>	<pwd> - the SQL Server password. (SQL SecurityModel).
-EncryptedSqlPassword <pwd>	<pwd> - the encrypted SQL Server password generated by EncryptSqlPassword action. (SQLSecurityModel).
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.

Options	Description
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

12.13.3 Advanced Options

The following advanced options help you perform restore operations:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.
-BackupFile <filename>	Specifies additional backup archive files to be used for striping backups. <filename> - specifies the backup archive files. Note: use once for each additional stripe.
-ContinueAfterError	Instructs SQL Server to continue the operation despite encountering errors such as invalid checksums. Note: for SQL 2005 and later only.
-IncludeLogins	For backup, includes the database logins in the backup file. For restore, creates the logins from the backup file on the destination server.
-ReportTLog	For backup, 'Yes' reports Skipped T-Log backups against databases that are in simple mode with a SUCCESS status rather than SKIPPED.
-KeepCDC	Flag to indicate whether the restore will support the Microsoft SQL Server Change Data Capture (CDC) feature. The possible values are 1 (CDC restore will be supported) or 0 (CDC restore will not be supported). If the KeepCDC parameter is set to 1 then the CDC enabled database will be restored along with the CDC related artifacts and the Capture and Cleanup jobs will be created with the default options. If the parameter is omitted, CDC restore will not be supported.
-KeepReplication	Preserves replication settings when restoring a published database to a server other than that on which it was created.

Options	Description
-MailTo <email_address>	<email_address> - an email address(es) to send the notification via SMTP. Note: multiple addresses may be separated by spaces, semicolons, or commas.
-NoStatus	Prevents status messages from being cached or sent to the Repository.
-RecoveryMode <mode> [-UndoFile <filename>]	Specifies the mode in which to leave the database after the operation is completed. <mode> - {NoRecovery, Standby}. Note: for Standby mode an undo file may be specified with the -UndoFile option.
-UndoFile <filename>	<filename> - specifies the ABSOLUTE path to the undo filename. Note: for Standby recovery mode only.
-RetryReads <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.
-StopAt <datetime>	Specifies the database to be restored to the state it was in as of the specified date and time. <datetime> - {"mm/dd/yyyy hh:mm:ss"}. Note: for Log BackupType only.
-StopAtMark <mark> [-After <datetime>]	Specifies recovery to the specified <mark>, including the transaction that contains the <mark>. Note: for Log BackupType only.
-StopBeforeMark <mark> [-After <datetime>]	Specifies recovery to the specified <mark> but does not include the transaction that contains the <mark>. Note: for Log BackupType only.
-After <datetime>	Recovery stops at the first <mark> having the specified name exactly at or after <datetime>. Note: only valid with -StopAtMark/-StopBeforeMark options. For Log BackupType only.

12.13.4 Tivoli Storage Manager Options

There are TSM options for your restore operations:

Options	Description
-TsmClientOwnerName <name>	<name> - the client owner name.
-TsmClientOwnerPassword <pwd>	<pwd> - the client owner password.
-EncryptedTsmClientOwnerPassword <pwd>	<pwd> - the encrypted TSM client owner password.
-TsmConfigFile <filename>	<filename> - the configuration file location.
-TsmHighLevel <name>	<name> - the high level file specification (path).
-TsmLowLevel <name>	<name> - the low level file specification (file name).
-TsmTcpServerAddress <address>	<address> - the TCP/IP address for the TSM server.
-TsmTcpPort <port>	<port> - the TCP/IP port address for the TSM server.
-TsmId <object_id>	<object_id> - The TSM object id. Note: if the object id is provided, high level and low level are not needed.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Restore`).

12.14 Restore File List Only CLI Commands

SQL Safe provides you with CLI commands to help you manually list files for database in backup set.

To perform RestoreFileListOnly operations use the following commands:

- SQLsafeCmd RestoreFileListOnly <backup_archive> [options]
- SQLsafeCmd RestoreFileListOnly TSM [options]

Where:

Action	Description
<backup_archive>	Path to the backup archive.
Tsm	Use Tivoli Storage Manager (see below for special options).

12.14.1 Common Options

The following options help you perform RestoreFileListOnly operations:

Options	Descriptions
-BackupSet <index>	<index> - the index of the backup set within the backup archive. (1-based).

12.14.2 Security Options

Secure your RestoreFileListOnly operations with the following options:

Options	Description
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/ write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

12.14.3 Advanced Options

The following advanced options help you perform RestoreFileListOnly operations:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.

12.14.4 Tivoli Storage Manager Options

There are TSM options for your RestoreFileListOnly operations:

Options	Description
-TsmClientOwnerName <name>	<name> - the client owner name.
-TsmClientOwnerPassword <pwd>	<pwd> - the client owner password.
-EncryptedTsmClientOwnerPassword <pwd>	<pwd> - the encrypted TSM client owner password.
-TsmConfigFile <filename>	<filename> - the configuration file location.
-TsmHighLevel <name>	<name> - the high level file specification (path).
-TsmLowLevel <name>	<name> - the low level file specification (file name).
-TsmTcpServerAddress <address>	<address> - the TCP/IP address for the TSM server.
-TsmTcpPort <port>	<port> - the TCP/IP port address for the TSM server.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help RestoreFileListOnly`).

12.15 Restore Header Only CLI Commands

SQL Safe provides you with CLI commands to help you manually list backup sets in an archive.

To perform RestoreHeaderOnly operations use the following commands:

- SQLsafeCmd RestoreHeaderOnly <backup_archive> [options]
- SQLsafeCmd RestoreHeaderOnly TSM [options]

Where:

Action	Description
<backup_archive>	Path to the backup archive.
Tsm	Use Tivoli Storage Manager (see below for special options).

12.15.1 Security Options

Secure your RestoreHeaderOnly operations with the following options:

Options	Description
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/ write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

12.15.2 Advanced Options

The following advanced options help you perform RestoreHeaderOnly operations:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.

12.15.3 Tivoli Storage Manager Options

There are TSM options for your RestoreHeaderOnly operations:

Options	Description
-TsmClientOwnerName <name>	<name> - the client owner name.
-TsmClientOwnerPassword <pwd>	<pwd> - the client owner password.
-EncryptedTsmClientOwnerPassword <pwd>	<pwd> - the encrypted TSM client owner password.
-TsmConfigFile <filename>	<filename> - the configuration file location.
-TsmHighLevel <name>	<name> - the high level file specification (path).
-TsmLowLevel <name>	<name> - the low level file specification (file name).
-TsmTcpServerAddress <address>	<address> - the TCP/IP address for the TSM server.
-TsmTcpPort <port>	<port> - the TCP/IP port address for the TSM server.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help RestoreHeaderOnly`).

12.16 RestoreLast CLI Commands

SQL Safe provides you with CLI commands to help you manually restore the most recent full backup of a database from the specified directory.

To perform RestoreLast operations use the following command:

- SQLsafeCmd RestoreLast [options]

Where the required [options] are:

Option	Description
-BackupLocation <filename>	<filename> - the filename pattern to restore. The most recent backup file matching this pattern will be restored. Set this to TSM for Tivoli Storage Manager.
-DatabaseName <db_name>	<db_name> - the database to restore.

12.16.1 Common Options

The following options help you perform RestoreLast operations:

Options	Descriptions
-DisconnectUsers	Disconnects all users from the target database before the restore operation begins.
-EncryptedRestorePassword <pwd>	<pwd> - the encrypted password used to decrypt the backup.
-InstanceName <name>	<name> - SQL server instance name. Note: it is not required if the instance is set as a default on the target server.
-Move <logical_filename> <target_filename>	To move the database logical database file to the physical target file. <logical_filename> - the database logical database file. <target_filename> - the physical target file . Corresponds to the WITH MOVE option in the RESTORE DATABASE T/SQL command.
-Password <pwd>	<pwd> - the non-encrypted password used to encrypt the backup.

12.16.2 Security Options

Secure your RestoreLast operations with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-SecurityModel <model>	The security model used to log into SQL Server. <model> - {Integrated, SQL}. Note: Integrated (Windows authentication) is the default.
-SqlUsername <username>	<username> - the SQL Server username. (SQL SecurityModel).
-SqlPassword <pwd>	<pwd> - the SQL Server password. (SQL SecurityModel).
-EncryptedSqlPassword <pwd>	<pwd> - the encrypted SQL Server password generated by EncryptSqlPassword action. (SQLSecurityModel).
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

12.16.3 Advanced Options

The following advanced options help you perform RestoreLast operations:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.
-ContinueAfterError	Instructs SQL Server to continue the operation despite encountering errors such as invalid checksums. Note: for SQL 2005 and later only.
-IncludeLogins	For backup, includes the database logins in the backup file. For restore, creates the logins from the backup file on the destination server.
-ReportTLog	For backup, 'Yes' reports skipped t-log backups against databases that are in simple mode with a SUCCESS status rather than SKIPPED.

Options	Description
-MaxTransferSize	Specifies the largest unit of transfer in bytes to be used between SQL Safe and the backup media. The possible values are multiples of 65536 bytes (64KB) ranging up to 4194304 bytes (4 MB). This parameter is used to enable compression on TDE enabled databases only when the MaxTransferSize value is set to 65537 or higher. If omitted, the MaxTransferSize will be taken from the 'TransferLimit' value set in the SQL Safe agent properties.
-KeepReplication	Preserves replication settings when restoring a published database to a server other than that on which it was created.
-MailTo <email_address>	<email_address> - an email address(es) to send the notification via SMTP. Note: multiple addresses may be separated by spaces, semicolons, or commas.
-RecoveryMode <mode> [-UndoFile <filename>]	Specifies the mode in which to leave the database after the operation is completed. <mode> - NoRecovery, Standby. Note: for Standby mode an undo file may be specified with the -UndoFile option.
-RetryReads <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.
-UndoFile <filename>	<filename> - specifies the ABSOLUTE path to the undo filename. Note: for Standby recovery mode only.
-KeepCDC	Flag to indicate whether the restore will support the Microsoft SQL Server Change Data Capture (CDC) feature. The possible values are 1 (CDC restore will be supported) or 0 (CDC restore will not be supported). If the KeepCDC parameter is set to 1 then the CDC enabled database will be restored along with the CDC related artifacts and the Capture and Cleanup jobs will be created with the default options. If the parameter is omitted, CDC restore will not be supported.

12.16.4 Tivoli Storage Manager Options

There are TSM options for your RestoreLast operations:

Options	Description
-TsmClientOwnerName <name>	<name> - the client owner name.
-TsmClientOwnerPassword <pwd>	<pwd> - the client owner password.
-EncryptedTsmClientOwnerPassword <pwd>	<pwd> - the encrypted TSM client owner password.
-TsmConfigFile <filename>	<filename> - the configuration file location.
-TsmHighLevel <name>	<name> - the high level file specification (path).
-TsmLowLevel <name>	<name> - the low level file specification (file name).
-TsmTcpServerAddress <address>	<address> - the TCP/IP address for the TSM server.
-TsmTcpPort <port>	<port> - the TCP/IP port address for the TSM server.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help RestoreLast`).

12.17 Tivoli Storage Manager (TSM) CLI Commands

SQL Safe provides you with CLI commands to help you manually perform Tivoli Storage Management (TSM) operations.

The following topics are included in this section:

12.17.1 Browse Tivoli Storage Manager (TSM) CLI Commands

SQL Safe provides you with CLI commands to help you manually browse tivoli storage management (TSM).

To perform Browse TSM operations use the following command:

- SQLsafeCmd Browse TSM [options]

Tivoli Storage Manager Options

There are TSM options for your Browse TSM operations:

Options	Description
-TsmClientOwnerName <name>	<name> - the client owner name.
-TsmClientOwnerPassword <pwd>	<pwd> - the client owner password.
-EncryptedTsmClientOwnerPassword <pwd>	<pwd> - the encrypted TSM client owner password.
-TsmConfigFile <filename>	<filename> - the configuration file location.
-TsmHighLevel <name>	<name> - the high level file specification (path).
-TsmLowLevel <name>	<name> - the low level file specification (file name).
-TsmTcpServerAddress <address>	<address> - the TCP/IP address for the TSM server.
-TsmTcpPort <port>	<port> - the TCP/IP port address for the TSM server.
-TsmIncludeInactive	If set, includes the inactive file copies in the browse results.

Advanced Options

The following advanced options help you perform Browse TSM operations:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.
-NoPrompt	Never prompt for credentials even if necessary.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Browse TSM`).

12.17.2 Expire Tivoli Storage Manager (TSM) CLI Commands

SQL Safe provides you with CLI commands to help you manually expire tivoli storage management backup archives.

To perform Expire TSM operations use the following command:

- SQLsafeCmd Expire TSM [options]

Where the required [options] are:

Option	Description
-Age <n><time_period>	Delete/Expire archives that are older than the specified amount of time. <n> - amount of time. <time_period> - {minutes, hours, days, weeks, months}. Note: There must be NO SPACE between <n> and <time_period>. E.g., -age 2hours.

Tivoli Storage Manager Options

There are TSM options for your Expire TSM operations:

Options	Description
-TsmClientOwnerName <name>	<name> - the client owner name.
-TsmClientOwnerPassword <pwd>	<pwd> - the client owner password.
-EncryptedTsmClientOwnerPassword <pwd>	<pwd> - the encrypted TSM client owner password.
-TsmConfigFile <filename>	<filename> - the configuration file location.
-TsmHighLevel <name>	<name> - the high level file specification (path).
-TsmLowLevel <name>	<name> - the low level file specification (file name).
-TsmTcpServerAddress <address>	<address> - the TCP/IP address for the TSM server.
-TsmTcpPort <port>	<port> - the TCP/IP port address for the TSM server.

Advanced Options

The following advanced options help you perform Expire TSM operations:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.
-NoPrompt	Do not prompt for confirmation before expiring files.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Expire TSM`).

12.17.3 Extract Tivoli Storage Manager (TSM) CLI Commands

SQL Safe provides you with CLI commands to help you manually extract a file from tivoli storage management.

To perform Extract TSM operations use the following command:

- SQLsafeCmd Extract TSM [required_options] [options]

Where the [required options] are:

Option	Description
-BackupFile <filename>	<filename> - the filename to which the extracted file will be written.
-TsmHighLevel <name>	<name> - the high level file specification (path).
-TsmLowLevel <name>	<name> - the low level file specification (file name).

Tivoli Storage Manager Options

There are TSM options for your Extract TSM operations:

Options	Description
-TsmClientOwnerName <name>	<name> - the client owner name.
-TsmClientOwnerPassword <pwd>	<pwd> - the client owner password.
-EncryptedTsmClientOwnerPassword <pwd>	<pwd> - the encrypted TSM client owner password.
-TsmConfigFile <filename>	<filename> - the configuration file location.

Security Options

Secure your Extract TSM operations with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

Advanced Options

The following advanced options help you perform Extract TSM operations:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Extract TSM`).

12.18 Verify Backups CLI Commands

SQL Safe provides you with CLI commands to help you manually verify a database backup.

To verify a database backup use the following commands:

- SQLsafeCmd Verify <backup_archive> [options]
- SQLsafeCmd Verify <db_name> <point_in_time> [options]
- SQLsafeCmd Verify TSM [options]

Where:

Action	Description
<backup_archive>	Path to the backup archive.
<point_in_time>	Date/Time {"MM/dd/yyyy hh:mm:ss"} to restore to.
Tsm	Use Tivoli Storage Manager (see below for special options).

12.18.1 Common Options

The following options help you verify a database backup:

Options	Descriptions
-BackupSet <index>	<index> - the index of the backup set within the backup archive. (1-based).
-EncryptedRestorePassword <pwd>	<pwd> - the encrypted password used to decrypt the backup.
-InstanceName <name>	<name> - SQL server instance name. Note: it is not required if the instance is set as a default on the target server.
-Password <pwd>	<pwd> - the non-encrypted password used to encrypt the backup.
-Server <hostname>	<hostname> - the hostname of server hosting the SQL Server where the operation should be performed. Note: this option is required for accessing remote or clustered SQL Servers (where applicable).

12.18.2 Security Options

Secure your verification operations with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-SecurityModel <model>	The security model used to log into SQL Server. <model> - {Integrated, SQL}. Note: Integrated (Windows authentication) is the default.
-SqlUsername <username>	<username> - the SQL Server username. (SQL SecurityModel).
-SqlPassword <pwd>	<pwd> - the SQL Server password. (SQL SecurityModel).
-EncryptedSqlPassword <pwd>	<pwd> - the encrypted SQL Server password generated by EncryptSqlPassword action. (SQL SecurityModel).
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

12.18.3 Advanced Options

The following advanced options help you verify a database backup:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.
-BackupFile <filename>	Specifies additional backup archive files to be used for striping backups. <filename> - specifies the backup archive files. Note: use once for each additional stripe.
-ContinueAfterError	Instructs SQL Server to continue the operation despite encountering errors such as invalid checksums. Note: SQL 2005 and later only.

Options	Description
-MailTo <email_address>	<email_address> - an email address(es) to send the notification via SMTP. Note: multiple addresses may be separated by spaces, semicolons, or commas.
-NoStatus	Prevents status messages from being cached or sent to the Repository.
-RetryReads <interval> <retry_time> <total_time>	On a network file error, retry every <interval> seconds for up to <retry_time> seconds. Total retry time allowed is <total_time> minutes.

12.18.4 Tivoli Storage Manager (TSM) Options

There are TSM options for your verification operations:

Options	Description
-TsmClientOwnerName <name>	<name> - the client owner name.
-TsmClientOwnerPassword <pwd>	<pwd> - the client owner password.
-EncryptedTsmClientOwnerPassword <pwd>	<pwd> - the encrypted TSM client owner password.
-TsmConfigFile <filename>	<filename> - the configuration file location.
-TsmHighLevel <name>	<name> - the high level file specification (path).
-TsmLowLevel <name>	<name> - the low level file specification (file name).
-TsmTcpServerAddress <address>	<address> - the TCP/IP address for the TSM server.
-TsmTcpPort <port>	<port> - the TCP/IP port address for the TSM server.

For detailed descriptions and available options, see the CLI Help (`SQLsafeCmd help Verify`).

12.19 Virtual Database CLI Commands

SQL Safe provides you with CLI commands to help you manually create virtual databases.

To perform operations use the following commands:

Commands	Description
SQLvdbCmd <action> [options]	Perform an action.
SQLvdbCmd help <action>	Display detailed help for an action.
SQLvdbCmd -ArgsFile <filename>	Perform the action defined within an argument file.

Where:

Action	Description
<action>	A keyword that tells SQLsafe what to do.
<filename>	Specifies the file that contains command line arguments.

When you use the *sqlvdbcmd* command, the following actions can be performed:

Action	Description
Create	Create a new virtual database.
Remove	Delete a virtual database.
Cleanup	Cleanup unused virtual database temporary files.
EncryptWindowsPassword	Encrypt plain-text password for Windows logins.
EncryptSqlPassword	Encrypt plain-text password for SQL Server logins.
EncryptRestorePassword	Encrypt plain-text password for encrypted restores.
Map	Generate maps (for InstantRestore or SQL virtual database).
Help	Display more detailed help.

For detailed descriptions and available options, see the CLI Help (`SQLvdbCmd help`).

12.19.1 Cleanup Virtual Database CLI Commands

SQL Safe provides you with CLI commands to help you manually cleanup unused virtual database temporary files.

To cleanup unused virtual database temporary files use the following command:

- SQLvdbCmd Cleanup

For detailed descriptions and available options, see the CLI Help (`SQLvdbCmd help Cleanup`).

12.19.2 Create a Virtual Database CLI Commands

SQL Safe provides you with CLI commands to help you manually create a new virtual database.

To create a new virtual database use the following command:

- SQLvdbCmd Create <db_name> <backup_archive> [options]

Where:

Action	Description
<db_name>	Name of the database.
<backup_archive>	Path to the backup archive.

Common Options

The following options help you create a new virtual database:

Options	Descriptions
-BackupSet <index>	<index> - the index of the backup set within the backup archive. (1-based).
-DisconnectUsers	Disconnects all users from the target database before the restore operation begins.
-EncryptedRestorePassword <pwd>	<pwd> - the encrypted password used to decrypt the backup.
-InstanceName <name>	<name> - SQL server instance name. Note: it is not required if the instance is set as a default on the target server.
-Move <logical_filename> <target_filename>	To move the database logical database file to the physical target file. <logical_filename> - the database logical database file. <target_filename> - the physical target file. Corresponds to the WITH MOVE option in the RESTORE DATABASE T/SQL command.
-Password <pwd>	<pwd> - the non-encrypted password used to encrypt the backup.
-Server <hostname>	<hostname> - the hostname of server hosting the SQL Server where the operation should be performed. Note: this option is required for accessing remote or clustered SQL Servers (where applicable).

Options	Descriptions
-Replace	Overrides database if exists.

Security Options

Secure your operations with the following options:

Options	Description
-NoPrompt	Never prompt for credentials even if necessary.
-SecurityModel <model>	The security model used to log into SQL Server. <model> - {Integrated, SQL}. Note: Integrated (Windows authentication) is the default.
-SqlUsername <username>	<username> - the SQL Server username. (SQL SecurityModel).
-SqlPassword <pwd>	<pwd> - the SQL Server password. (SQL SecurityModel).
-EncryptedSqlPassword <pwd>	<pwd> - the encrypted SQL Server password generated by EncryptSqlPassword action. (SQLSecurityModel).
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

Advanced Options

The following advanced options help you create a new virtual database:

Options	Description
-ArgsFile <filename>	The path to a file containing command-line arguments. <filename> - specifies the file that contains the command line arguments.
-Diff <filename>	The differential backup. <filename> - the file path to the differential backup. This can be followed by -BackupFile, -BackupSet, or -Password to set individual options for this backup set.

Options	Description
-Log <filename>	The log backup. <filename> the file path to the log backup. This can be followed by -BackupFile, -BackupSet, or -Password to set individual options for this backup set.
-BackupFile <filename>	Specifies additional backup archive files to be used for striping backups. <filename> - specifies the backup archive files. Note: use once for each additional stripe.
-IncludeLogins	For backup, includes the database logins in the backup file. For restore, creates the logins from the backup file on the destination server.
-KeepReplication	Preserves replication settings when restoring a published database to a server other than that on which it was created.
-NoChecksum	Disables the validation of any checksums by the restore operation. Note: for SQL 2005 and later only.
-RecoveryMode <mode> [-UndoFile <filename>]	Specifies the mode in which to leave the database after the operation is completed. <mode> - {NoRecovery, Standby}. Note: for Standby mode an undo file may be specified with the -UndoFile option.
-StopAt <datetime>	Specifies the database to be restored to the state it was in as of the specified date and time. <datetime> - {"mm/dd/yyyy hh:mm:ss"}. Note: for Log BackupType only.
-StopAtMark <mark> [-After <datetime>]	Specifies recovery to the specified <mark>, including the transaction that contains the <mark>. Note: for Log BackupType only.
-StopBeforeMark <mark> [-After <datetime>]	Specifies recovery to the specified <mark> but does not include the transaction that contains the <mark>. Note: for Log BackupType only.

Options	Description
-After <datetime>	Recovery stops at the first <mark> having the specified name exactly at or after <datetime>. Note: only valid with -StopAtMark/-StopBeforeMark options. For Log BackupType only.

For detailed descriptions and available options, see the CLI Help (`SQLvdbCmd help Create`).

12.19.3 EncryptRestorePassword Virtual Database CLI Commands

SQL Safe provides you with CLI commands to help you manually encrypt plain-text password for encrypted restores.

To encrypt plain-text password for encrypted restores use the following command:

- SQLvdbCmd EncryptRestorePassword <password>

Where:

Action	Description
<password>	Plain-text password to encrypt.

For detailed descriptions and available options, see the CLI Help (`SQLvdbCmd help EncryptRestorePassword`).

12.19.4 EncryptSqlPassword Virtual Database CLI Commands

SQL Safe provides you with CLI commands to help you manually encrypt plain-text password for SQL Server logins.

To encrypt plain-text password for SQL Server logins use the following command:

- SQLvdbCmd EncryptSqlPassword <password>

Where:

Action	Description
<password>	Plain-text password to encrypt.

For detailed descriptions and available options, see the CLI Help (`SQLvdbCmd help EncryptSqlPassword`).

12.19.5 EncryptWindowsPassword Virtual Database CLI Commands

SQL Safe provides you with CLI commands to help you manually encrypt plain-text password for Windows logins.

To encrypt plain-text password for Windows logins use the following command:

- SQLvdbCmd EncryptWindowsPassword <password>

Where:

Action	Description
<password>	Plain-text password to encrypt.

For detailed descriptions and available options, see the CLI Help (`SQLvdbCmd help EncryptWindowsPassword`).

12.19.6 Map Virtual Database CLI Commands

SQL Safe provides you with CLI commands to help you manually generate maps (for IstantRestore or SQL virtual database).

To generate maps use the following command:

- SQLvdbCmd Map <backup_archive> [options]

Where:

Action	Description
<backup_archive>	Path to the backup archive.

Common Options

The following options help you generate maps:

Options	Descriptions
-BackupFile <filename>	Specifies additional backup archive files to be used for striping backups. <filename> - specifies the backup archive files. Note: use once for each additional stripe.
-Password <pwd>	<pwd> - the non-encrypted password used to encrypt the backup.
-EncryptedRestorePassword <pwd>	<pwd> - the encrypted password used to decrypt the backup.
-Trace	Creates a diagnostic trace <log> to help troubleshoot issues.

Security Options

Secure your maps with the following options:

Options	Description
-WindowsUsername <domain\user>	<domain\user> - the Windows user that will be used to read/write the backup archive.
-WindowsPassword <pwd>	<pwd> - the password for the Windows user.
-EncryptedWindowsPassword <pwd>	<pwd> - the encrypted password for the Windows user generated by EncryptWindowsPassword action.

For detailed descriptions and available options, see the CLI Help (`SQLvdbCmd help Map`).

12.19.7 Remove a Virtual Database CLI Commands

SQL Safe provides you with CLI commands to help you manually delete a virtual database.

To delete a virtual database use the following command:

- SQLvdbCmd Remove <db_name> [options]

Where:

Action	Description
<db_name>	Name of the database.

Common Options

The following options help you delete a virtual database:

Options	Descriptions
-InstanceName <name>	<name> - SQL server instance name. Note: it is not required if the instance is set as a default on the target server.

For detailed descriptions and available options, see the CLI Help (`SQLvdbCmd help Remove`).

12.19.8 Virtual Database Help CLI Commands

To find out detailed descriptions and available options for your Virtual Database operations, type: `SQLvdbCmd help <action>`.

12.20 Help CLI Commands

To find out detailed descriptions and available options for your SQL Safe operations, type: `SQLsafeCmd help <action>`.