SSIS Blog

Helpful information and examples on how to use SQL Server Integration Services.

SSIS Catalog - Backup and Restore

Kaarthik Sivashanmugam 23 Mar 2012 7:35 AM 3

Integration Services catalog (SSISDB) uses the encryption mechanism available in SQL Server to protect its sensitive data. So backup and restore of SSIS catalog across machines need some extra steps in addition to the usual backup and restore steps. This post describes all the steps involved in backup and restore of SSIS catalog.

Backup

1. Backup the master key used for encryption in SSISDB database and protect the backup file using a password. This is done using BACKUP MASTER KEY statement.

```
 USE SSISDB
 BACKUP MASTER KEY TO FILE = 'c:\DMK\SSISDB\key'
 ENCRYPTION BY PASSWORD = 'SS1SC@talogMKBKUP'
```

This step is not necessary every time you do backup unless you have lost the file or the password or if you have changed the master key of the SSISDB database.

2. Backup SSISDB database using SQL Server Management Studio or BACKUP DATABASE statement in TSQL.

3. If SSISDB database will be restored in a SQL Server instance that never had SSIS catalog, follow steps 3.1. & 3.2. listed below in additional backup steps section.

Restore

4. If SSISDB database will be restored in a SQL Server instance that never had any SSIS catalog, follow preparatory steps 4.1. & 4.2. listed below in additional restore steps section.

5. Restore SSISDB database using SQL Server Management Studio or RESTORE DATABASE statement in TSQL.

6. If SSISDB is restored in a SQL Server instance that never had any SSIS catalog, follow steps 6.1. thru 6.3. listed below in additional restore steps section.

7. Restore backup of the master key from the source server using the backup file created in step 1 in Backup section above.

```
 USE SSISDB
 RESTORE MASTER KEY FROM FILE = 'D:\MasterKeyBackup\SSIS-
Server1234\key'
 DECRYPTION BY PASSWORD = 'SS1SC@talogMKBKUP'
 ENCRYPTION BY PASSWORD = 'NewC@talogPassw0rd'
 FORCE
```

"SS1SC@talogMKBKUP" is the password used to protect the file containing backup of the master key and "NewC@talogPassw0rd" is the new password to encrypt database master key.

The warning reported when carrying out this step *"The current master key cannot be decrypted. The error was ignored because the FORCE option was specified."* can be ignored.

Decrypting all keys using the currently active master key and re-encrypting using the restored master key is a resource-intensive operation should be scheduled when server usage is low.

SSIS catalog should be fully functional in the restored server now.

Additional Backup Steps

Following steps are necessary during backup if you will be restoring SSISDB database to a SQL Server instance where the catalog was never created.

3.1. Generate create script for sp_ssis_startup stored procedure which is created when SSIS catalog is created in a SQL Server. The script is generated using SQL Server Management Studio. Right-click on the stored procedure in master database in SSMS Object Explorer, select "Script Stored Procedure as" -> "CREATE TO" -> "File" options to save the script to a file (for example, sp_ssis_startup.sql).

3.2. Generate create script for Agent job [SSIS Server Maintenance Job] which is created in SQL Server Agent automatically when SSIS catalog is created in a SQL Server. The script is generated using SQL Server Management Studio. Right-click on the Agent job in SSMS Object Explorer, select "Script Job as" -> "CREATE TO" -> "File" options to save the job to a file (for example, ssis_server_maintenance_job.sql).

Additional Restore Steps

Following steps are necessary during restore if you are restoring SSISDB database to a SQL Server instance where the catalog was never created.

4.1. Enable CLR functionality in SQL Server that SSISDB database depends on.

```
 USE MASTER
 EXEC sp_configure 'clr enabled', 1
 RECONFIGURE
```

4.2. Create asymmetric key and unsafe assembly loading principal that SSISDB database depends on. The login is used only for granting permission and hence does not have to be mapped to a database user.

```
 USE MASTER
 CREATE ASYMMETRIC KEY MS_SQLEnableSystemAssemblyLoadingKey FROM
EXECUTABLE FILE = 'C:\Program Files\Microsoft SQL
Server\110\DTS\Binn\ISServerExec.exe'
 CREATE LOGIN MS_SQLEnableSystemAssemblyLoadingUser FROM ASYMMETRIC KEY
MS_SQLEnableSystemAssemblyLoadingKey
 GRANT UNSAFE ASSEMBLY TO MS_SQLEnableSystemAssemblyLoadingUser
```

You may need to change the file system path to ISServerExec.exe depending on your install location.

6.1. Create login [##MS_SSISServerCleanupJobLogin##] using CREATE LOGIN TSQL statement. This login is for internal use in SSISDB maintenance.

6.2. Map SSISDB user ##MS_SSISServerCleanupJobUser## to server login ##MS_SSISServerCleanupJobLogin##

```
 USE SSISDB
 ALTER USER [##MS_SSISServerCleanupJobUser##] with LOGIN =
[##MS_SSISServerCleanupJobLogin##]
```

6.3. Create startup stored procedure and agent job by executing scripts (sp_ssis_startup.sql & ssis_server_maintenance_job.sql) created in steps 3.1 & 3.2 under additional backup steps section above in the master database in the SQL Server instance in which SSISDB database is restored. You need to update @servername parameter for sp_add_jobserver step in ssis_server_maintenance_job.sql with the name of the new server replacing old server name.

Tweet 6 Like 2 Share 1 □ Save this on Delicious

Comments

26 Mar 2012 1:53 PM

ArthurZ

The password used to encrypt the catalog with (at its creation time), if not available anymore (e.g. employee left) what are the implications of not having this password?

5 Apr 2012 7:36 PM

Kaarthik Sivashanmugam

For restore operation, you do not need the password used when creating the catalog. Backup of the master key and the password used to protect it should be enough.

9 Apr 2012 9:12 AM

ArthurZ

Turned out there is no backup for either. What to do next?