**Solving problems of orphaned users when moving a database from source to destination (log shipping, mirroring, attach)**

--drop database orphans

--STEP 1. CREATE A DATABASE AND TABLE, POPULATE TABLE

--Create a database

Use master

go

CREATE DATABASE [Orphans]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'Orphans', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\DATA\Orphans.mdf',

SIZE = 8192KB ,

FILEGROWTH = 65536KB )

LOG ON

( NAME = N'Orphans\_log',

FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\DATA\Orphans\_log.ldf',

SIZE = 8192KB ,

FILEGROWTH = 65536KB )

GO

--create table cars

USE [Orphans]

GO

CREATE TABLE [dbo].[Cars](

[cars] [varchar](50) NULL

) ON [PRIMARY]

GO

--Insert data into table cars

USE [Orphans]

GO

insert into Cars values('Rolls Royce'),('Benz'),('Bently'),('Porche'),('Jag')

select \* from Cars

--STEP 2.CREATE A SQL LOGIN SANDY

USE [master]

GO

CREATE LOGIN [SANDY]

WITH PASSWORD=N'password123',

DEFAULT\_DATABASE=[master],

CHECK\_EXPIRATION=OFF,

CHECK\_POLICY=OFF

GO

--STEP 4. FIND ALL THE USERS AND LOGINS IN DATABASE (RUN ON SERVER1 THEN ON SERVER 2)

use Orphans

go

Select LOGINNAME, SID--<< FIND ALL LOGINS (TOM AND SANDY)

from sys.syslogins

ORDER BY 1 DESC

Select NAME,sid --<< FIND ALL USERS

from sys.sysusers

ORDER BY 1 DESC

--5. CREATE AND MAP SANDY SQL LOGIN TO DATABASE ORPHAN

USE [Orphans]

GO

CREATE USER [SANDY] FOR LOGIN [SANDY]

GO

--FIND SPECIFICALLY SANDY'S SID

Select LOGINNAME, SID--<< FIND LOGINS SID FOR (SANDY)

from sys.syslogins WHERE loginname = 'SANDY'

ORDER BY 1 DESC

Select NAME,sid --<< FIND SID FOR SANDY

from sys.sysusers WHERE NAME = 'SANDY'

ORDER BY 1 DESC

--NOTICE BOTH THE SIDS ARE THE SAME

--STEP 6. MOVE DATABASE ORPHANS FROM SERVER1 TO SERVER2 USING BACKUP AND RESTORE

BACKUP DATABASE [Orphans]

TO DISK = N'C:\s\ORPHANS.BAK'

WITH NOFORMAT,

NOINIT,

NAME = N'Orphans-Full Database Backup',

SKIP,

NOREWIND,

NOUNLOAD,

STATS = 10

GO

--STEP 6. COPY PASTE THE BACKUP AND RESTORE ON SERVER2 AND NOTICE WHICH SQL LOGINS AND USERS MOVED WITH THE DATABASE

--STEP 7.

--(RUN RESTORE ON SERVER2)

--(RUN RESTORE ON SERVER2)

USE [master]

RESTORE DATABASE [Orphans]

FROM DISK = N'\\SERVER2\D\ORPHANS.BAK' WITH FILE = 1,

MOVE N'Orphans'

TO N'C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Orphans.mdf',

MOVE N'Orphans\_log'

TO N'C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Orphans\_log.ldf',

NOUNLOAD, STATS = 5

GO

--STEP 7.

--RUN THIS ON SERVER2

--RUN THIS ON SERVER2

--FIND SPECIFICALLY SANDY'S SID

USE ORPHANS

GO

Select LOGINNAME, SID--<< FIND LOGINS SID FOR (SANDY)

from sys.syslogins WHERE loginname = 'SANDY'

ORDER BY 1 DESC

Select NAME,sid --<< FIND SID FOR SANDY

from sys.sysusers WHERE NAME = 'SANDY'

ORDER BY 1 DESC

--To detect orphaned users in a given database, simply run the following.

USE Orphans

GO

EXEC sp\_change\_users\_login 'report'

--DESTINATION DATABASE

USE MASTER

GO

SELECT name as SQL\_LogIn,SID as SQL\_SID FROM sys.syslogins

WHERE [name] = 'sandy'

GO

USE Orphans

GO

SELECT name DataBase\_User,SID as Database\_SID FROM sysusers

WHERE [name] = 'sandy'

GO

-----------------------------------------------------------------------------------

---------------------------------------------------------------

--STEP 7. EXECUTE RESTORE

USE [master]

go

RESTORE DATABASE [Orphans]

FROM DISK = N'\\SERVER2\D\ORPHANS.BAK' WITH FILE = 1,

MOVE N'Orphans'

TO N'C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Orphans.mdf',

MOVE N'Orphans\_log'

TO N'C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Orphans\_log.ldf',

NOUNLOAD,

REPLACE,

STATS = 5

GO

--STEP 8 RERUN THE SCRIPTSS

--NOTICE THAT THE DATABASE USER SID IS MISSING EVEN THOUGH THE DATABASE HAS BEEN MOVED AND SHE WAS PART OF THE SOURCE DATABASE (ORPHANS)

--WE HAVE ORPHANED THIS SQL LOGIN

USE ORPHANS

GO

Select LOGINNAME, DBNAME, SID--<< FIND LOGINS SID FOR (SANDY)

from sys.syslogins WHERE loginname = 'SANDY'

ORDER BY 1 DESC

Select NAME,sid --<< FIND SID FOR SANDY

from sys.sysusers WHERE NAME = 'SANDY'

ORDER BY 1 DESC

--STEP 9.

--CREATE AND CORRECT THE MISSING SID FOR LOGIN SANDY (ON SERVER2)

USE [master]

GO

CREATE LOGIN [SANDY]

WITH PASSWORD=N'password123',

DEFAULT\_DATABASE=[master],

CHECK\_EXPIRATION=OFF,

CHECK\_POLICY=OFF

GO

--CHECK

USE ORPHANS

GO

Select LOGINNAME, DBNAME, SID--<< FIND LOGINS SID FOR (SANDY)

from sys.syslogins WHERE loginname = 'SANDY'

ORDER BY 1 DESC

Select NAME,sid --<< FIND SID FOR SANDY

from sys.sysusers WHERE NAME = 'SANDY'

ORDER BY 1 DESC

--STEP 10

--To resolve an orphaned user, resync the SID of the user to map to the login

USE Orphans

GO

EXEC sp\_change\_users\_login 'update\_one', 'SANDY', 'SANDY'

--STEP 11. VERIFY

USE ORPHANS

GO

Select LOGINNAME, DBNAME, SID--<< FIND LOGINS SID FOR (SANDY)

from sys.syslogins WHERE loginname = 'SANDY'

ORDER BY 1 DESC

Select NAME,sid --<< FIND SID FOR SANDY

from sys.sysusers WHERE NAME = 'SANDY'

ORDER BY 1 DESC

-- Great script by Microsoft

USE master

GO

IF OBJECT\_ID ('sp\_hexadecimal') IS NOT NULL

DROP PROCEDURE sp\_hexadecimal

GO

CREATE PROCEDURE sp\_hexadecimal

@binvalue varbinary(256),

@hexvalue varchar (514) OUTPUT

AS

DECLARE @charvalue varchar (514)

DECLARE @i int

DECLARE @length int

DECLARE @hexstring char(16)

SELECT @charvalue = '0x'

SELECT @i = 1

SELECT @length = DATALENGTH (@binvalue)

SELECT @hexstring = '0123456789ABCDEF'

WHILE (@i <= @length)

BEGIN

DECLARE @tempint int

DECLARE @firstint int

DECLARE @secondint int

SELECT @tempint = CONVERT(int, SUBSTRING(@binvalue,@i,1))

SELECT @firstint = FLOOR(@tempint/16)

SELECT @secondint = @tempint - (@firstint\*16)

SELECT @charvalue = @charvalue +

SUBSTRING(@hexstring, @firstint+1, 1) +

SUBSTRING(@hexstring, @secondint+1, 1)

SELECT @i = @i + 1

END

SELECT @hexvalue = @charvalue

GO

IF OBJECT\_ID ('sp\_help\_revlogin') IS NOT NULL

DROP PROCEDURE sp\_help\_revlogin

GO

CREATE PROCEDURE sp\_help\_revlogin @login\_name sysname = NULL AS

DECLARE @name sysname

DECLARE @type varchar (1)

DECLARE @hasaccess int

DECLARE @denylogin int

DECLARE @is\_disabled int

DECLARE @PWD\_varbinary varbinary (256)

DECLARE @PWD\_string varchar (514)

DECLARE @SID\_varbinary varbinary (85)

DECLARE @SID\_string varchar (514)

DECLARE @tmpstr varchar (1024)

DECLARE @is\_policy\_checked varchar (3)

DECLARE @is\_expiration\_checked varchar (3)

DECLARE @defaultdb sysname

IF (@login\_name IS NULL)

DECLARE login\_curs CURSOR FOR

SELECT p.sid, p.name, p.type, p.is\_disabled, p.default\_database\_name, l.hasaccess, l.denylogin FROM

sys.server\_principals p LEFT JOIN sys.syslogins l

ON ( l.name = p.name ) WHERE p.type IN ( 'S', 'G', 'U' ) AND p.name <> 'sa'

ELSE

DECLARE login\_curs CURSOR FOR

SELECT p.sid, p.name, p.type, p.is\_disabled, p.default\_database\_name, l.hasaccess, l.denylogin FROM

sys.server\_principals p LEFT JOIN sys.syslogins l

ON ( l.name = p.name ) WHERE p.type IN ( 'S', 'G', 'U' ) AND p.name = @login\_name

OPEN login\_curs

FETCH NEXT FROM login\_curs INTO @SID\_varbinary, @name, @type, @is\_disabled, @defaultdb, @hasaccess, @denylogin

IF (@@fetch\_status = -1)

BEGIN

PRINT 'No login(s) found.'

CLOSE login\_curs

DEALLOCATE login\_curs

RETURN -1

END

SET @tmpstr = '/\* sp\_help\_revlogin script '

PRINT @tmpstr

SET @tmpstr = '\*\* Generated ' + CONVERT (varchar, GETDATE()) + ' on ' + @@SERVERNAME + ' \*/'

PRINT @tmpstr

PRINT ''

WHILE (@@fetch\_status <> -1)

BEGIN

IF (@@fetch\_status <> -2)

BEGIN

PRINT ''

SET @tmpstr = '-- Login: ' + @name

PRINT @tmpstr

IF (@type IN ( 'G', 'U'))

BEGIN -- NT authenticated account/group

SET @tmpstr = 'CREATE LOGIN ' + QUOTENAME( @name ) + ' FROM WINDOWS WITH DEFAULT\_DATABASE = [' + @defaultdb + ']'

END

ELSE BEGIN -- SQL Server authentication

-- obtain password and sid

SET @PWD\_varbinary = CAST( LOGINPROPERTY( @name, 'PasswordHash' ) AS varbinary (256) )

EXEC sp\_hexadecimal @PWD\_varbinary, @PWD\_string OUT

EXEC sp\_hexadecimal @SID\_varbinary,@SID\_string OUT

-- obtain password policy state

SELECT @is\_policy\_checked = CASE is\_policy\_checked WHEN 1 THEN 'ON' WHEN 0 THEN 'OFF' ELSE NULL END FROM sys.sql\_logins WHERE name = @name

SELECT @is\_expiration\_checked = CASE is\_expiration\_checked WHEN 1 THEN 'ON' WHEN 0 THEN 'OFF' ELSE NULL END FROM sys.sql\_logins WHERE name = @name

SET @tmpstr = 'CREATE LOGIN ' + QUOTENAME( @name ) + ' WITH PASSWORD = ' + @PWD\_string + ' HASHED, SID = ' + @SID\_string + ', DEFAULT\_DATABASE = [' + @defaultdb + ']'

IF ( @is\_policy\_checked IS NOT NULL )

BEGIN

SET @tmpstr = @tmpstr + ', CHECK\_POLICY = ' + @is\_policy\_checked

END

IF ( @is\_expiration\_checked IS NOT NULL )

BEGIN

SET @tmpstr = @tmpstr + ', CHECK\_EXPIRATION = ' + @is\_expiration\_checked

END

END

IF (@denylogin = 1)

BEGIN -- login is denied access

SET @tmpstr = @tmpstr + '; DENY CONNECT SQL TO ' + QUOTENAME( @name )

END

ELSE IF (@hasaccess = 0)

BEGIN -- login exists but does not have access

SET @tmpstr = @tmpstr + '; REVOKE CONNECT SQL TO ' + QUOTENAME( @name )

END

IF (@is\_disabled = 1)

BEGIN -- login is disabled

SET @tmpstr = @tmpstr + '; ALTER LOGIN ' + QUOTENAME( @name ) + ' DISABLE'

END

PRINT @tmpstr

END

FETCH NEXT FROM login\_curs INTO @SID\_varbinary, @name, @type, @is\_disabled, @defaultdb, @hasaccess, @denylogin

END

CLOSE login\_curs

DEALLOCATE login\_curs

RETURN 0

GO

exec sp\_help\_revlogin

Exec sp\_change\_users\_login ‘autofix’, ‘tom’

Select \* from sys.syslogins

Select \* from sys.sysusers

--drop database orphans

--STEP 1. CREATE A DATABASE AND TABLE, POPULATE TABLE

--Create a database

Use master

go

CREATE DATABASE [Orphans]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'Orphans', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\DATA\Orphans.mdf',

SIZE = 8192KB ,

FILEGROWTH = 65536KB )

LOG ON

( NAME = N'Orphans\_log',

FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\DATA\Orphans\_log.ldf',

SIZE = 8192KB ,

FILEGROWTH = 65536KB )

GO

--create table cars

USE [Orphans]

GO

CREATE TABLE [dbo].[Cars](

[cars] [varchar](50) NULL

) ON [PRIMARY]

GO

--Insert data into table cars

USE [Orphans]

GO

insert into Cars values('Rolls Royce'),('Benz'),('Bently'),('Porche'),('Jag')

select \* from Cars

--STEP 2. FIRST CREATE A WINDOWS DOMAIN USER 'TOM' IN ACTIVE DIRECTORY THEN CREATE SQL LOGIN AND MAP THE LOGIN TO ORPHANS DATABASE AS AN USER

--TOM

USE [master]

GO

CREATE LOGIN [SQL\tom] FROM WINDOWS WITH DEFAULT\_DATABASE=[master] --<< CREATE SQL LOGIN

GO

USE [Orphans]

GO

CREATE USER [SQL\tom] FOR LOGIN [SQL\tom] --<< CREATE SQL USER TOM

GO

USE [Orphans]

GO

ALTER ROLE [db\_datareader] ADD MEMBER [SQL\tom] --<< ADD SQL LOGIN TOM TO DB\_DATABASE ROLE (GROUP)

GO

--STEP 3. CREATE A SQL LOGIN SANDY

USE [master]

GO

CREATE LOGIN [SANDY]

WITH PASSWORD=N'password123',

DEFAULT\_DATABASE=[master],

CHECK\_EXPIRATION=OFF,

CHECK\_POLICY=OFF

GO

--STEP 4. FIND ALL THE USERS AND LOGINS IN DATABASE (RUN ON SERVER1 THEN ON SERVER 2)

Select LOGINNAME, DBNAME, SID --<< FIND ALL LOGINS (TOM AND BILLY)

from sys.syslogins

ORDER BY 1 DESC

Select NAME,sid --<< FIND ALL USERS

from sys.sysusers

ORDER BY 1 DESC

--5. CREATE AND MAP SANDY SQL LOGIN TO DATABASE ORPHAN

USE [Orphans]

GO

CREATE USER [SANDY] FOR LOGIN [SANDY]

GO

--FIND SPECIFICALLY SANDY'S SID

Select LOGINNAME, DBNAME, SID--<< FIND LOGINS SID FOR (SANDY)

from sys.syslogins WHERE loginname = 'SANDY'

ORDER BY 1 DESC

Select NAME,sid --<< FIND SID FOR SANDY

from sys.sysusers WHERE NAME = 'SANDY'

ORDER BY 1 DESC

--NOTICE BOTH THE SIDS ARE THE SAME

--STEP 6. MOVE DATABASE ORPHANS FROM SERVER1 TO SERVER2 USING BACKUP AND RESTORE

BACKUP DATABASE [Orphans]

TO DISK = N'C:\s\ORPHANS.BAK'

WITH NOFORMAT,

NOINIT,

NAME = N'Orphans-Full Database Backup',

SKIP,

NOREWIND,

NOUNLOAD,

STATS = 10

GO

--STEP 6. COPY PASTE THE BACKUP AND RESTORE ON SERVER2 AND NOTICE WHICH SQL LOGINS AND USERS MOVED WITH THE DATABASE

--STEP 7.

--(RUN RESTORE ON SERVER2)

--(RUN RESTORE ON SERVER2)

USE [master]

RESTORE DATABASE [Orphans]

FROM DISK = N'\\SERVER2\D\ORPHANS.BAK' WITH FILE = 1,

MOVE N'Orphans'

TO N'C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Orphans.mdf',

MOVE N'Orphans\_log'

TO N'C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Orphans\_log.ldf',

NOUNLOAD,

REPLACE,

STATS = 5

GO

--STEP 7.

--RUN THIS ON SERVER2

--RUN THIS ON SERVER2

--FIND SPECIFICALLY SANDY'S SID

USE ORPHANS

GO

Select LOGINNAME, DBNAME, SID--<< FIND LOGINS SID FOR (SANDY)

from sys.syslogins WHERE loginname = 'SANDY'

ORDER BY 1 DESC

Select NAME,sid --<< FIND SID FOR SANDY

from sys.sysusers WHERE NAME = 'SANDY'

ORDER BY 1 DESC

--To detect orphaned users in a given database, simply run the following.

USE Orphans

GO

EXEC sp\_change\_users\_login 'report'

--DESTINATION DATABASE

USE MASTER

GO

SELECT name as SQL\_LogIn,SID as SQL\_SID FROM sys.syslogins

WHERE [name] = 'sandy'

GO

USE Orphans

GO

SELECT name DataBase\_User,SID as Database\_SID FROM sysusers

WHERE [name] = 'sandy'

GO

--STEP 7. EXECUTE RESTORE

USE [master]

RESTORE DATABASE [Orphans]

FROM DISK = N'\\SERVER2\D\ORPHANS.BAK' WITH FILE = 1,

MOVE N'Orphans'

TO N'C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Orphans.mdf',

MOVE N'Orphans\_log'

TO N'C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Orphans\_log.ldf',

NOUNLOAD,

REPLACE,

STATS = 5

GO

--STEP 8 RERUN THE SCRIPTSS

--NOTICE THAT THE DATABASE USER SID IS MISSING EVEN THOUGH THE DATABASE HAS BEEN MOVED AND SHE WAS PART OF THE SOURCE DATABASE (ORPHANS)

--WE HAVE ORPHANED THIS SQL LOGIN

USE ORPHANS

GO

Select LOGINNAME, DBNAME, SID--<< FIND LOGINS SID FOR (SANDY)

from sys.syslogins WHERE loginname = 'SANDY'

ORDER BY 1 DESC

Select NAME,sid --<< FIND SID FOR SANDY

from sys.sysusers WHERE NAME = 'SANDY'

ORDER BY 1 DESC

--STEP 9.

--CREATE AND CORRECT THE MISSING SID FOR LOGIN SANDY (ON SERVER2)

USE [master]

GO

CREATE LOGIN [SANDY]

WITH PASSWORD=N'password123',

DEFAULT\_DATABASE=[master],

CHECK\_EXPIRATION=OFF,

CHECK\_POLICY=OFF

GO

--CHECK

USE ORPHANS

GO

Select LOGINNAME, DBNAME, SID--<< FIND LOGINS SID FOR (SANDY)

from sys.syslogins WHERE loginname = 'SANDY'

ORDER BY 1 DESC

Select NAME,sid --<< FIND SID FOR SANDY

from sys.sysusers WHERE NAME = 'SANDY'

ORDER BY 1 DESC

--STEP 10

--To resolve an orphaned user, resync the SID of the user to map to the login

USE Orphans

GO

EXEC sp\_change\_users\_login 'update\_one', 'SANDY', 'SANDY'

--STEP 11. VERIFY

USE ORPHANS

GO

Select LOGINNAME, DBNAME, SID--<< FIND LOGINS SID FOR (SANDY)

from sys.syslogins WHERE loginname = 'SANDY'

ORDER BY 1 DESC

Select NAME,sid --<< FIND SID FOR SANDY

from sys.sysusers WHERE NAME = 'SANDY'

ORDER BY 1 DESC