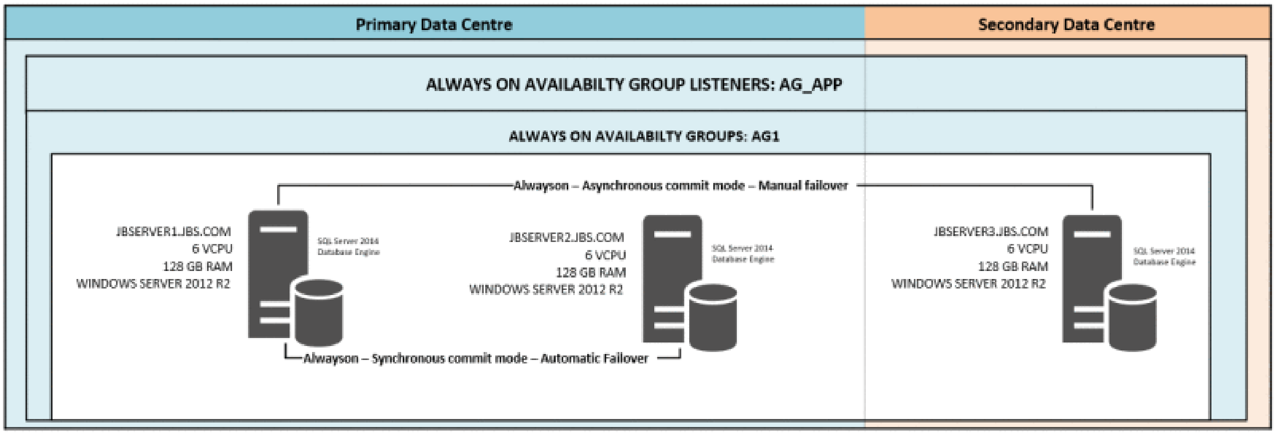
ALWAYSON – LOGIN SYNCHRONIZATION ACROSS REPLICAS

https://jbswiki.com/2017/09/18/alwayson-login-synchronization-across-replicas/

Author - Vivek Janakiraman

Environment



-> Create a Job called “**Login Synchronization**” on all the Database Servers as part of your Alwayson Availability group. In my Environment, the Job will be created on Database Server **JBSERVER1**, **JBSERVER2** and **JBSERVER3**.

The Job “**Login Synchronization**” will have the below script executed as part of it.

Logins_Transfer_Script_AlwaysOn.txt

```
set nocount on
create table #Sync_Logins (Script varchar(max))
Declare @sql nvarchar(max)
Declare @Primary_Replica varchar(20)

SELECT @Primary_Replica = primary_replica
  FROM sys.dm_hadr_availability_group_states a INNER JOIN sys.availability_group_listeners b
    ON a.group_id=b.group_id where b.dns_name='PPN-VDBSL'

IF (@Primary_Replica= @@servername) BEGIN;
    Print N'Script cannot run on primary Replica';
    drop table #Sync_Logins
    RETURN;
END;

SET @sql=N'';
set @sql = 'SELECT ''If not Exists (select loginname from master.dbo.syslogins where name =
'''''' +name +'''''''+') BEGIN CREATE LOGIN '' + QUOTENAME(name) + '' WITH PASSWORD=''
+ sys.fn_varbintohexstr(password_hash) + '' HASHED, SID=''
+ sys.fn_varbintohexstr(sid) + '', ''
+ ''DEFAULT_DATABASE=''+ QUOTENAME(COALESCE(default_database_name, ''master''))
+ '', DEFAULT_LANGUAGE='' + QUOTENAME(COALESCE(default_language_name,
''us_english''))
+ '', CHECK_EXPIRATION='' + CASE is_expiration_checked WHEN 1 THEN ''ON'' ELSE
''OFF'' END
+ '', CHECK_POLICY='' + CASE is_policy_checked WHEN 1 THEN ''ON'' ELSE ''OFF'' END + ''
END''
FROM ['+@Primary_Replica+'].master.sys.sql_logins
```

```
WHERE name !=''sa''
UNION ALL
--Windows logins:
SELECT ''If not Exists (select loginname from master.dbo.syslogins where name = ''''''+ name
+'''''''+') BEGIN CREATE LOGIN '' + QUOTENAME(name) + '' FROM WINDOWS WITH ''
+ ''DEFAULT_DATABASE=''+ QUOTENAME(COALESCE(default_database_name, ''master''))
+ '', DEFAULT_LANGUAGE='' + QUOTENAME(COALESCE(default_language_name,
''us_english''))+ '' END''
FROM ['+@Primary_Replica+'].master.sys.server_principals
WHERE type IN (''U'',''G'')
AND name NOT LIKE ''%\SQLServer2005MSSQLUser$%$%''
AND name NOT LIKE ''%\SQLServer2005SQLAgentUser$%$%''
AND name NOT LIKE ''%\SQLServer2005MSFTEUser$%$%''
AND name NOT IN (''BUILTIN\Administrators'', ''NT AUTHORITY\SYSTEM'');'
insert into #Sync_Logins
execute sp_executesql @sql
SET @sql=N'';
SET @sql = 'SELECT ''EXEC sp_addsrvrolemember '' + QUOTENAME(L.name) + '', '' +
QUOTENAME(R.name)
FROM  ['+@Primary_Replica+'].master.sys.server_principals L JOIN
['+@Primary_Replica+'].master.sys.server_role_members RM
ON L.principal_id=RM.member_principal_id
JOIN  ['+@Primary_Replica+'].master.sys.server_principals R
ON RM.role_principal_id=R.principal_id
WHERE L.type IN (''U'',''G'',''S'')
AND L.name NOT LIKE ''%\SQLServer2005MSSQLUser$%$%''
AND L.name NOT LIKE ''%\SQLServer2005SQLAgentUser$%$%''
AND L.name NOT LIKE ''%\SQLServer2005MSFTEUser$%$%''
AND L.name NOT IN (''BUILTIN\Administrators'', ''NT AUTHORITY\SYSTEM'', ''sa'');'
insert into #Sync_Logins
execute sp_executesql @sql
SET @sql=N'';
SELECT @sql=@sql+'     '+[Script] FROM #Sync_Logins;
EXECUTE master.sys.sp_executesql @sql;
drop table #Sync_Logins
```

-> Create a Linked Server to query the primary Replica. In my Environment, Linked servers JBSERVER2 and JBSERVER3 will be created on JBSERVER1.

Linked servers JBSERVER1 and JBSERVER3 will be created on JBSERVER2.

Linked servers JBSERVER1 and JBSERVER2 will be created on JBSERVER3.

-> The job will gracefully exit with a message “Script cannot run on primary Replica” if the job executes on Primary Replica. If the Job executes on the Secondary replica, It queries the list of Logins on the primary replica and will create the logins that are missing on the Secondary Replicas.

-> This solution just adds the missing Logins on the Secondary Replicas but will not Drop logins on the Secondary Replica that are not present on the Primary.