



# Click Studios

## SQL Server Transactional Replication for Passwordstate High Availability

# Table of Contents

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>OVERVIEW .....</b>  | <b>3</b>  |
| <b>2</b> | <b>PREREQUISITES .....</b>                                     | <b>4</b>  |
| <b>3</b> | <b>CONFIGURING THE DISTRIBUTION DATABASE .....</b>             | <b>9</b>  |
| <b>4</b> | <b>CREATING THE PUBLISHER.....</b>                             | <b>17</b> |
| <b>5</b> | <b>CREATING THE SUBSCRIBER.....</b>                            | <b>24</b> |
| <b>6</b> | <b>CONNECT PASSWORDSTATE TO YOUR REPLICATED DATABASE .....</b> | <b>34</b> |
| <b>7</b> | <b>TRANSACTIONAL REPLICATION CONSIDERATIONS.....</b>           | <b>36</b> |
| <b>8</b> | <b>HOW TO TROUBLESHOOT TRANSACTION REPLICATION.....</b>        | <b>37</b> |

# 1 Overview

This document will provide instructions for configuring SQL Server Transactional Replication for High Availability of the Passwordstate Database.

These instructions are created using **SQL Server 2019 Standard** edition and are intended as a guide only. If you have any technical issues with SQL Server, please contact Microsoft for support

**Note:** When using the High Availability module of Passwordstate, your distribution and publication databases must reside on SQL Server 2012, 2014, 2016, 2017 or 2019 **Standard** or above – SQL **Express** can only act as a subscriber to SQL Server replication. You must also be using the same version of SQL Server for both database servers.

Below is some information about a test environment used to document this process, to help you understand our instructions easier:

## tranweb01.halox.net

- Role – Passwordstate primary web server
- Microsoft Windows 2019 Server OS
- Primary Passwordstate website installed and is connected SQL on transql.halox.net

## transql01.halox.net

- Role – Passwordstate primary database Server
- Microsoft Windows 2019 Server OS
- Microsoft SQL Standard 2019
- Hosts a Passwordstate database with data stored in this database from several months of using Passwordstate in a non-high availability configuration.

## tranweb02.halox.net

- Role - Passwordstate secondary web server
- Microsoft Windows 2019 Server OS
- Has HA instance of Passwordstate installed, but not yet connected to a database. Instructions for HA Install located here:  
[https://www.clickstudios.com.au/downloads/version8/High\\_Availability\\_Installation\\_Instructions.pdf](https://www.clickstudios.com.au/downloads/version8/High_Availability_Installation_Instructions.pdf)

## transql02.halox.net

- Role – Passwordstate secondary database Server
- Microsoft Windows 2019 Server OS
- Microsoft SQL Server Express 2019

## Domain account:

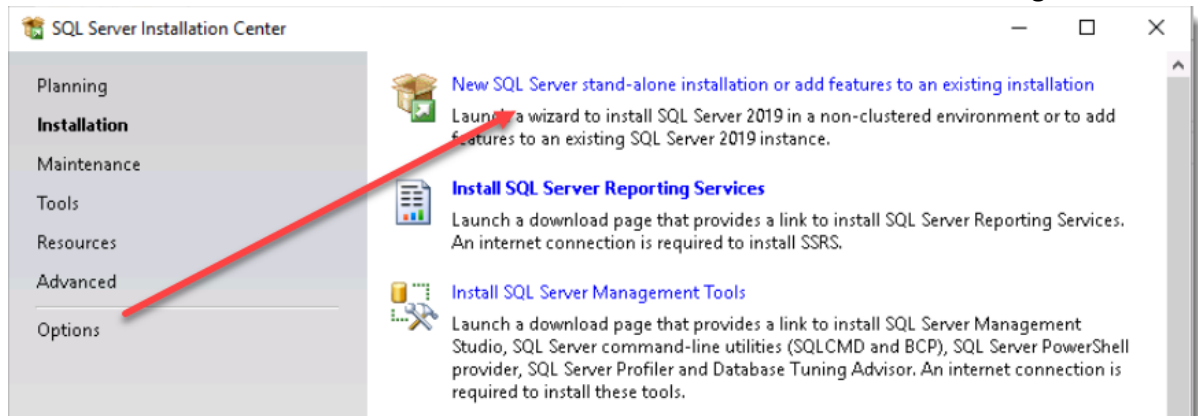
- halox\sqlha
- Is a member of the Domain Users group only

## Local SQL account used to authenticate Passwordstate websites to SQL

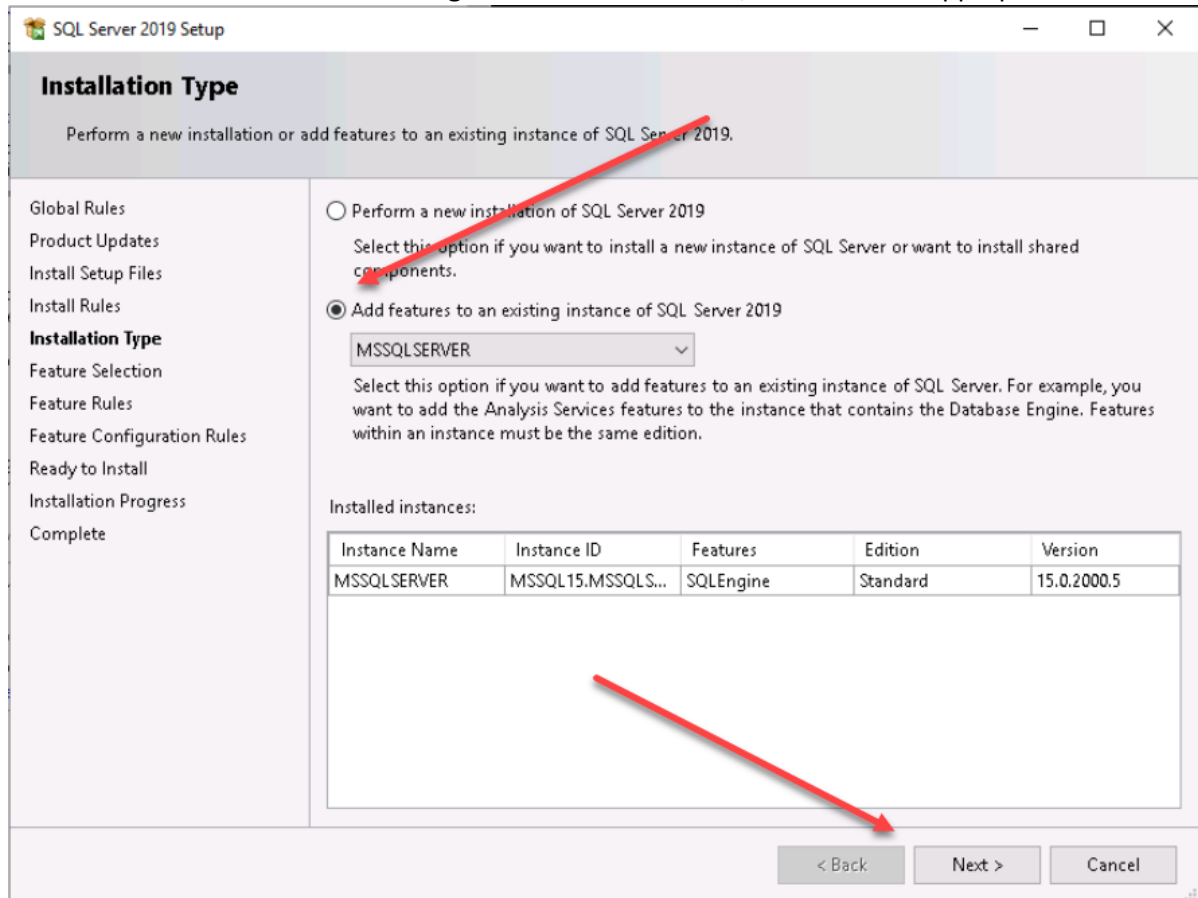
- passwordstate\_user

## 2 Prerequisites

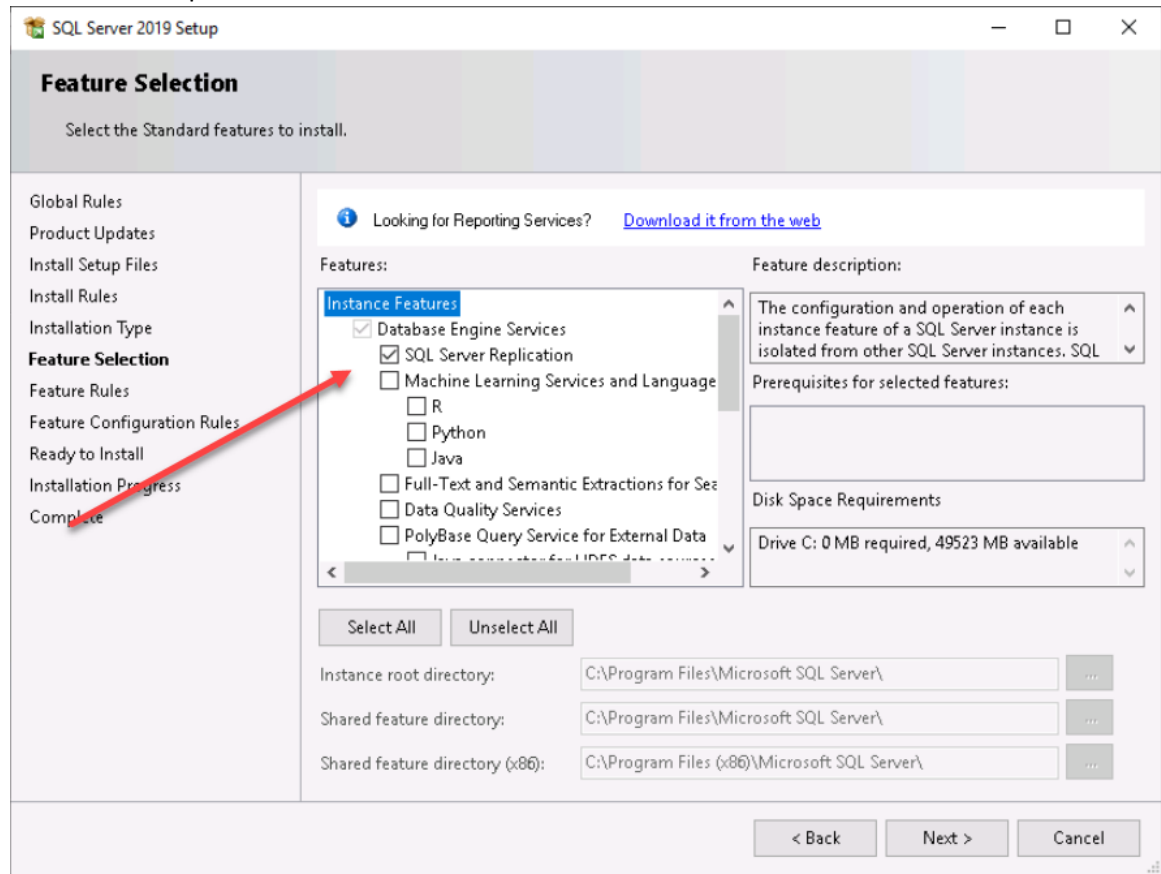
1. This process assumes you have a working install of Passwordstate, connected to a database called **Passwordstate**, running on an instance of SQL Server Standard or above.
2. Prior to starting this process, you should ensure that you have the **SQL Server Replication** feature installed on both your primary, and secondary database servers. You can install this by running your SQL Server installation, and selecting the following options:
  - Select **New SQL Server stand-alone installation or add new features to an existing installation**



- Select **Add features to an existing instance of SQL Server**, and select the appropriate instance

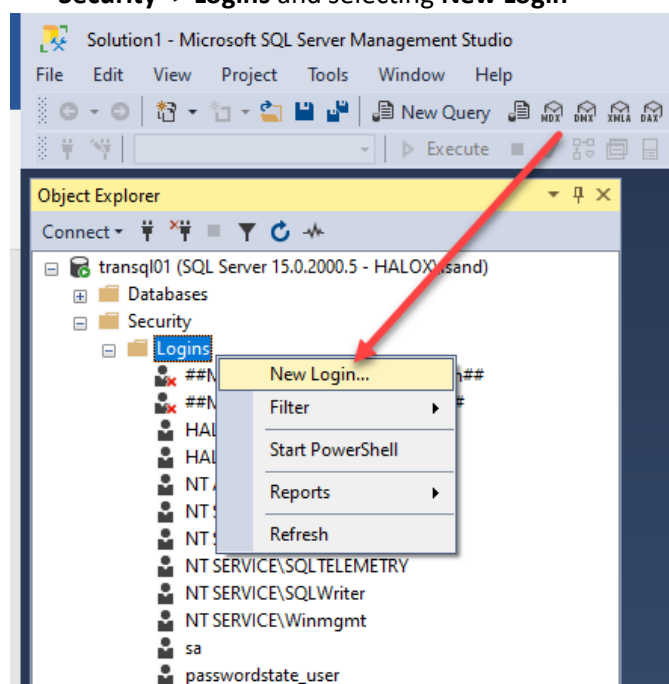


- Select the **SQL Server Replication** feature if isn't already selected, and then run through the rest of the process to finish the installation

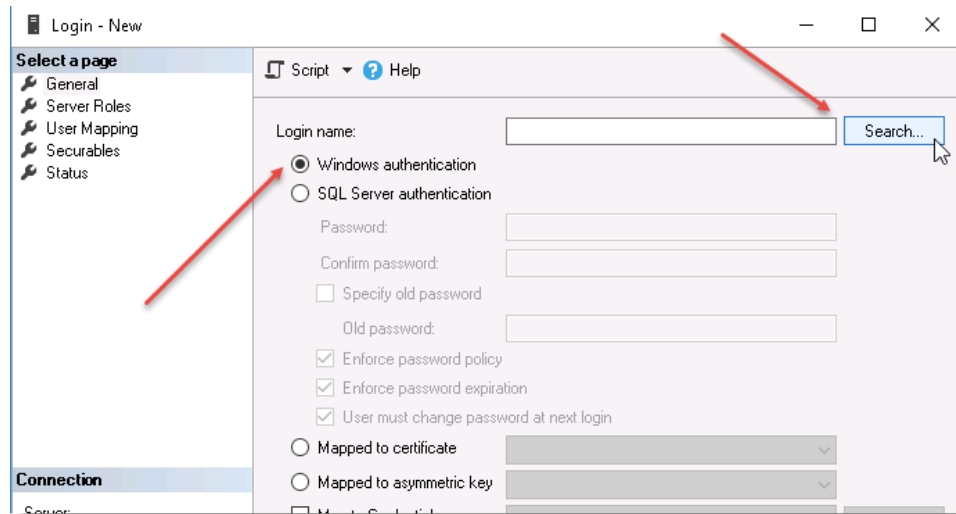


3. You will need a domain account with Domain User rights. For this process we have created an account called **halox\sqlha** in Active directory.

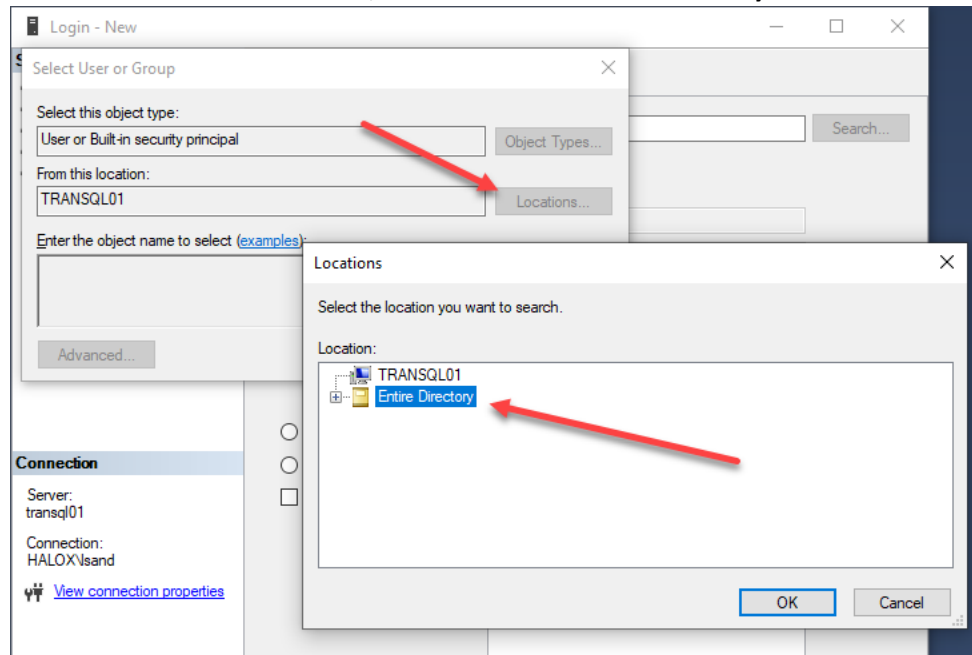
- Add this account into both SQL Servers using SQL Management Studios Tools by expanding **Security -> Logins** and selecting **New Login**



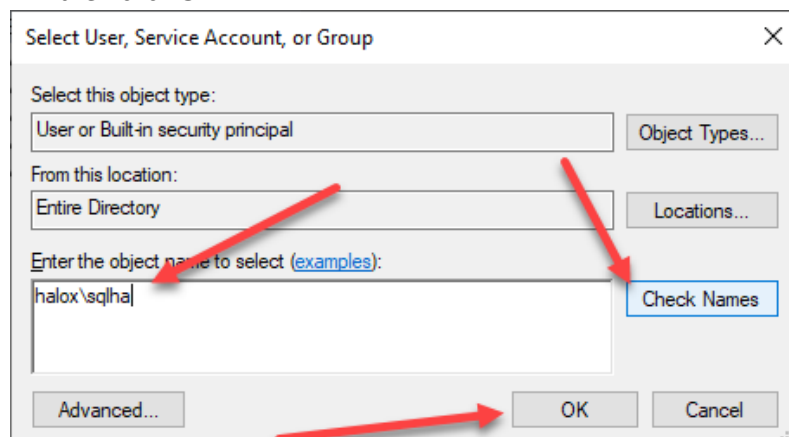
- Select **Windows Authentication** and then click **Search**



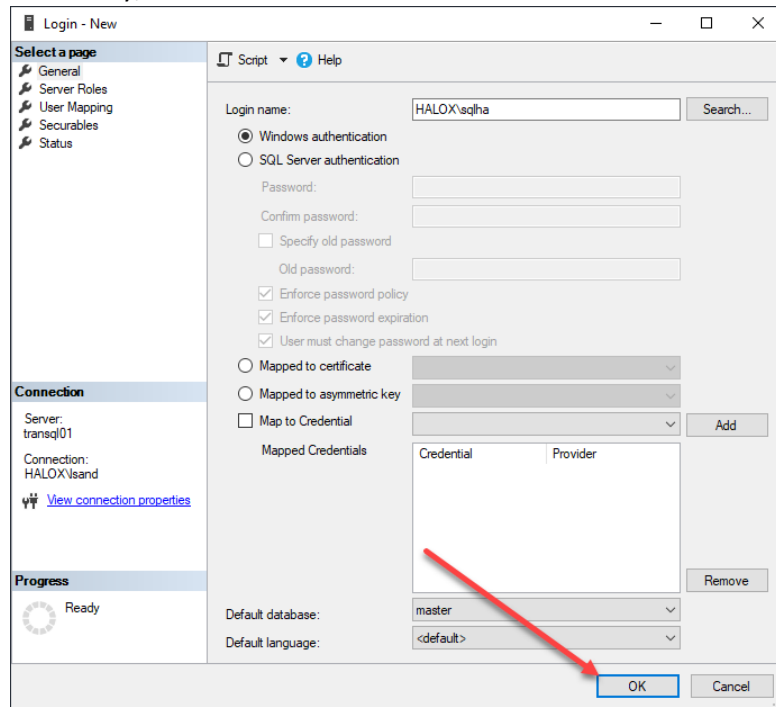
- Click the **Locations** button, and then select **Entire Directory**:



- Enter your domain account in the format of **domain\username** and click **Check Names**, and then click **OK**



- Finally, click **OK**

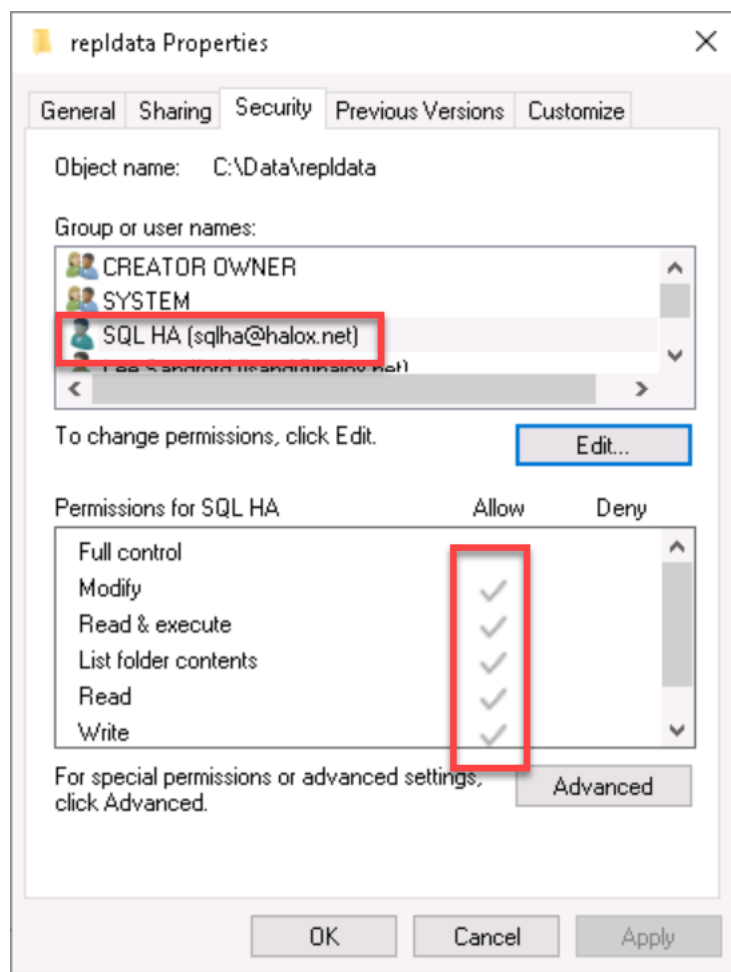


Ensure you repeat this process for your second database server, and then proceed to the next step.

- Later in this process you will need to specify a folder for the Transaction Replication snapshots to be stored in. Microsoft require this folder to be local to the Database server, and not a shared UNC path, even if it's a mapped drive to a shared UNC path.

If you have a second disk in your server, like a D drive for example, then it is preferred that you use this drive. For this documentation, we are using the C drive.

You will need to grant your domain account (halox\sqlha) **Modify** rights to this folder on your primary SQL Server. In this case the default folder in the wizard and the one we'll use is **C:\Data\repldata**



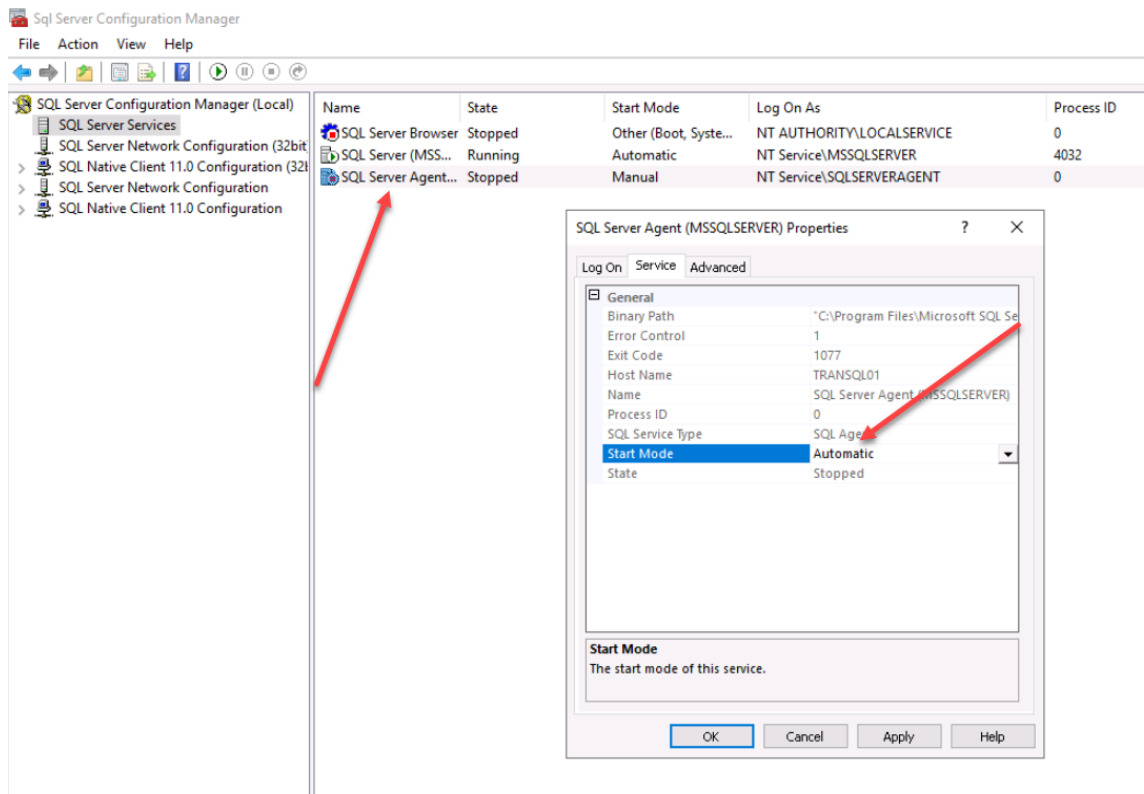


### 3 Configuring the Distribution Database

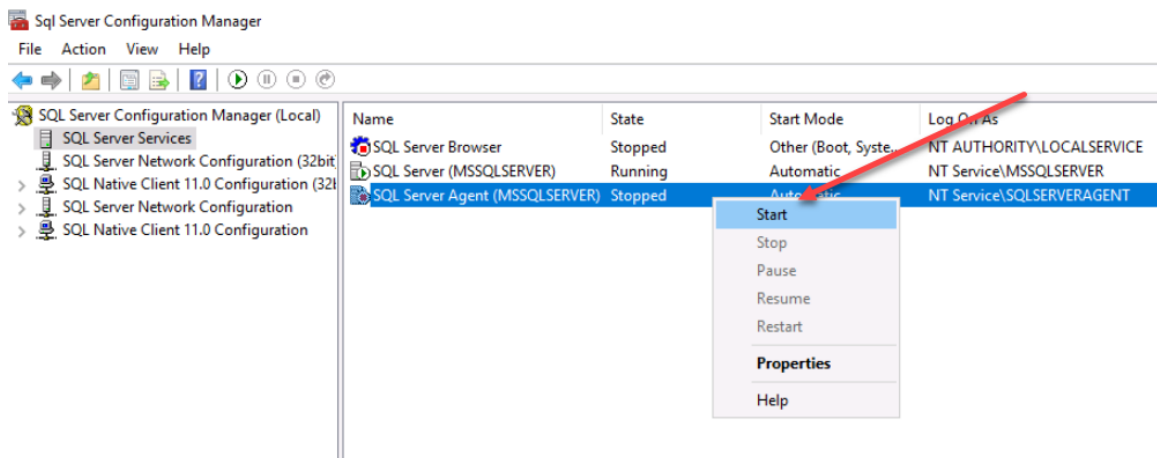
This section of the document outlines the process required to set up your primary database server as a Distribution Database. This is a once off process and may already be configured if you are using Transaction Replication for other applications on this server.

First, we need to configure the SQL Server Agent to start automatically. This service is what sends replication data to the remote SQL server. Follow these steps below to set the service correctly:

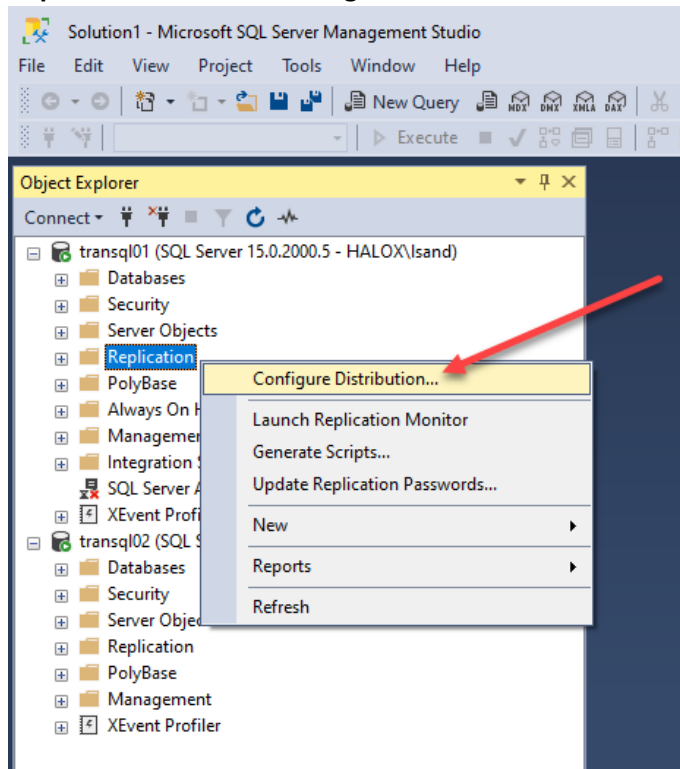
- Open **SQL Server Configuration Manager** on your primary database server, and go to the properties of the **SQL Server Agent** service and choose **Automatic** for the **Start Mode** property:



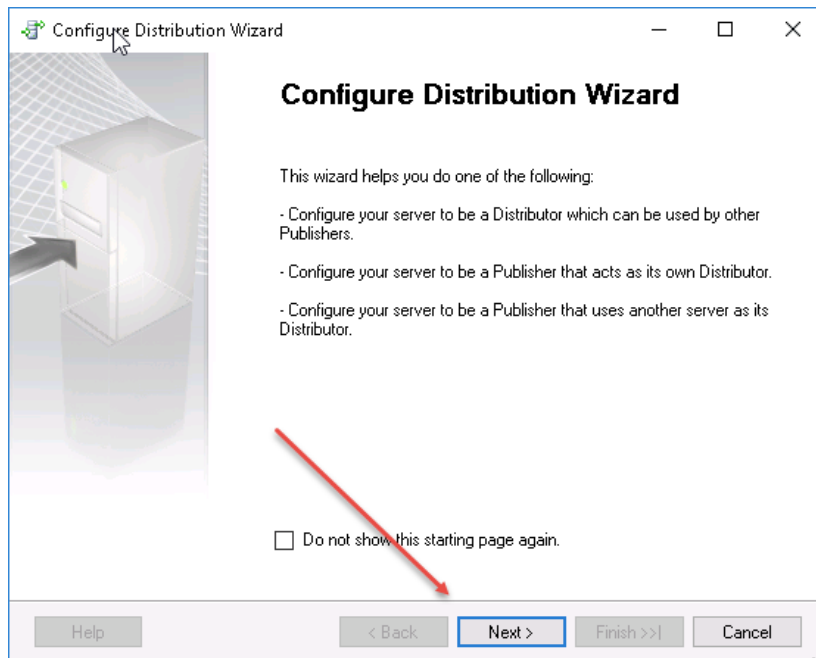
- Now Start the service:



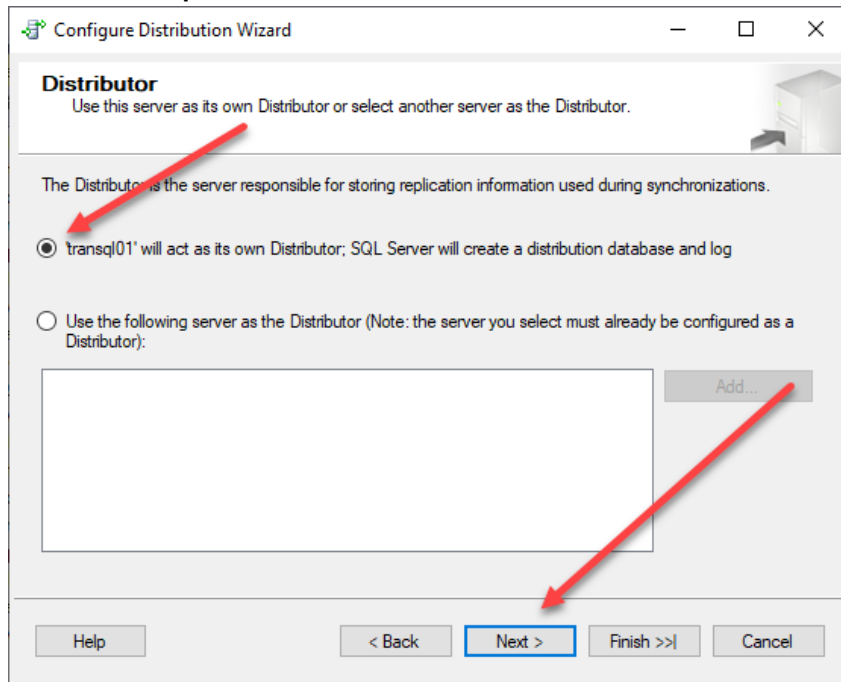
- Now we'll configure the Distribution Database. Using **SQL Management Studio Tools**, right click **Replication** and select **Configure Distribution**



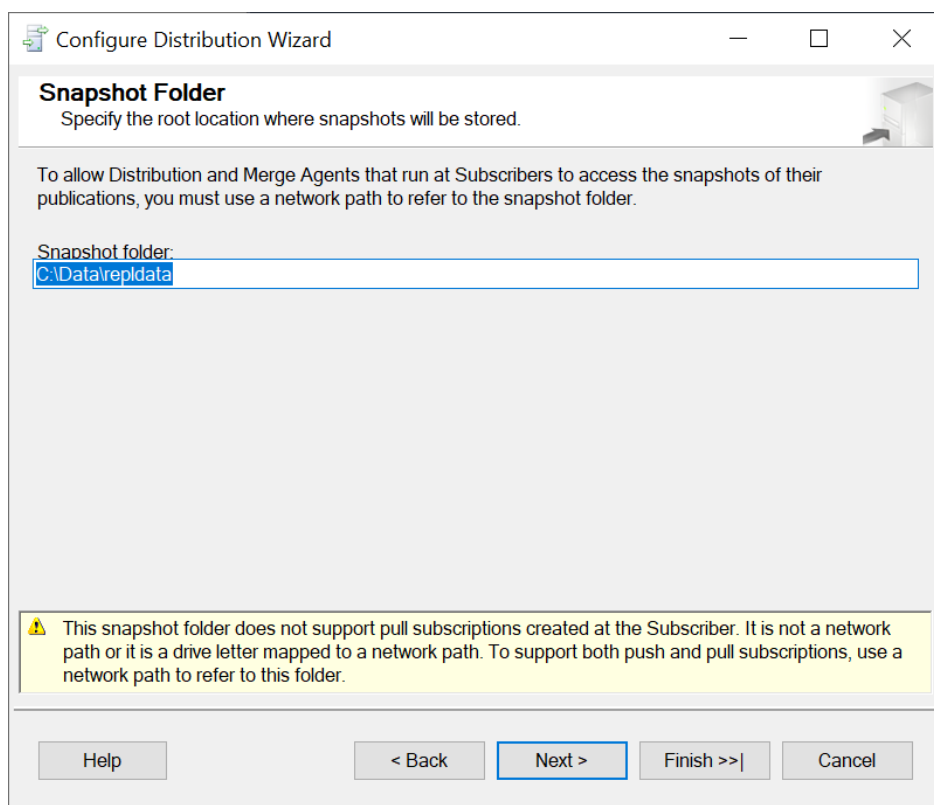
- Select **Next**



- Confirm **transql01** is selected and click **Next**



- On the **Snapshot Folder** screen, it is preferable to enter another disk on the server to store these snapshots. Don't use a UNC path to a network share, or map a drive letter to a UNC path for this screen. Change the path to the folder you created earlier in this process, (**C:\Data\repldata**), and click **Next**



- On the **Distribution Database** page, change the paths for the database file and the log file to your snapshot folder, and click **Next**

**Configure Distribution Wizard**

**Distribution Database**  
Select the name and location of the distribution database and log files.

The distribution database stores changes to transactional publications until Subscribers can be updated. It also stores historical information for snapshot and merge publications.

Distribution database name:  
distribution

Folder for the distribution database file:  
C:\Data\repldata

Folder for the distribution database log file:  
C:\Data\repldata

The paths must refer to disks that are local to the Distributor and begin with a local drive letter and colon (for example, C:). Mapped drive letters and network paths are invalid.

Help < Back **Next >** Finish >>| Cancel

- Ensure your Primary database server is selected and click **Next**

**Configure Distribution Wizard**

**Publishers**  
Enable servers to use this Distributor when they become Publishers.

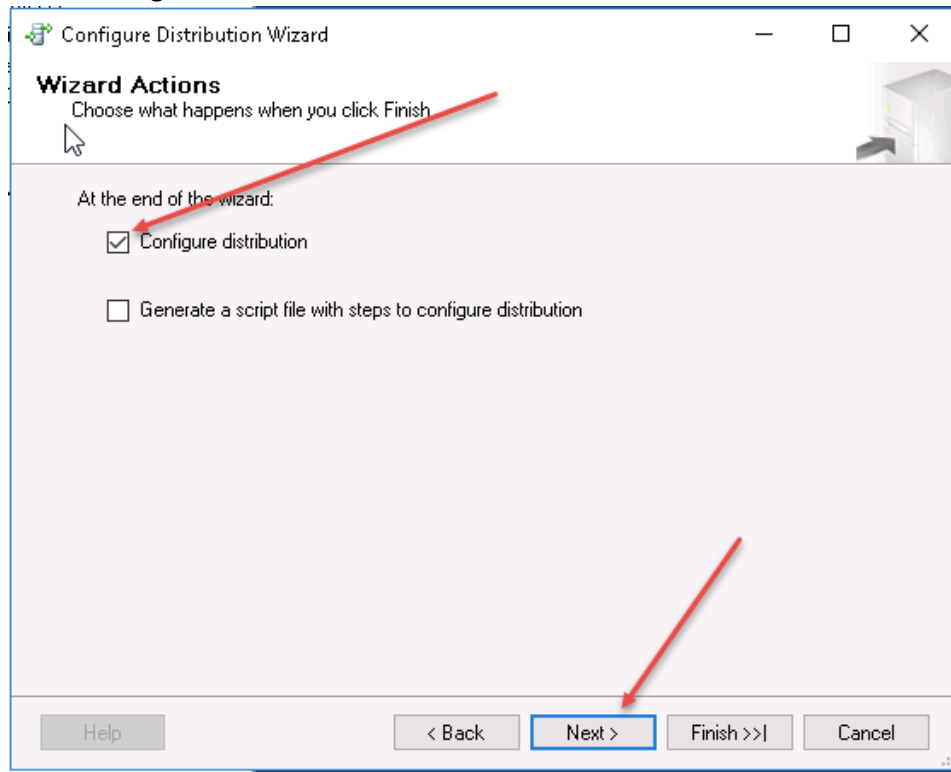
Publishers:

|                                     | Publisher ▲ | Distribution Database |
|-------------------------------------|-------------|-----------------------|
| <input checked="" type="checkbox"/> | transql01   | distribution          |

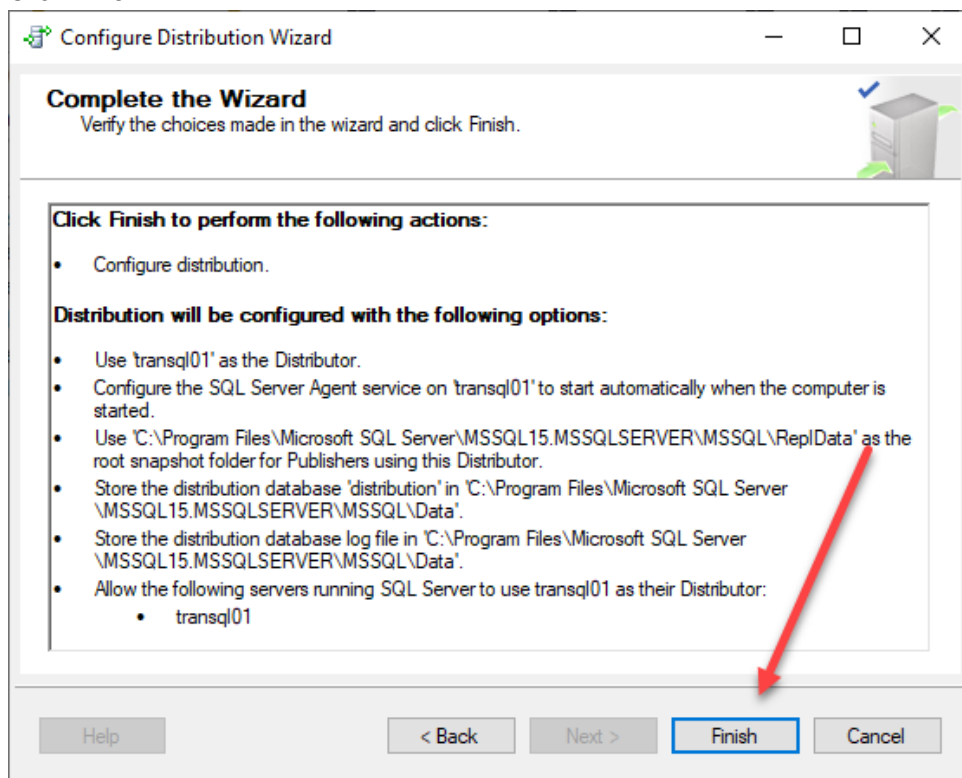
Add ▼

Help < Back **Next >** Finish >>| Cancel

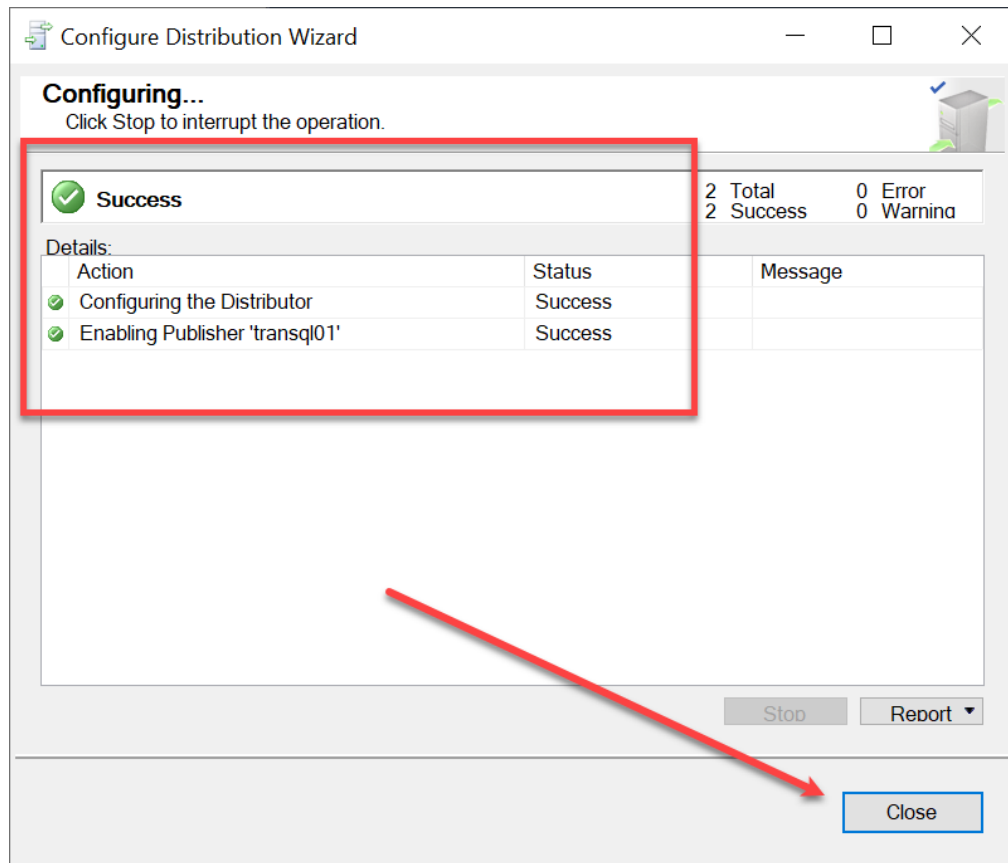
- Select **Configure Distribution** and click **Next**



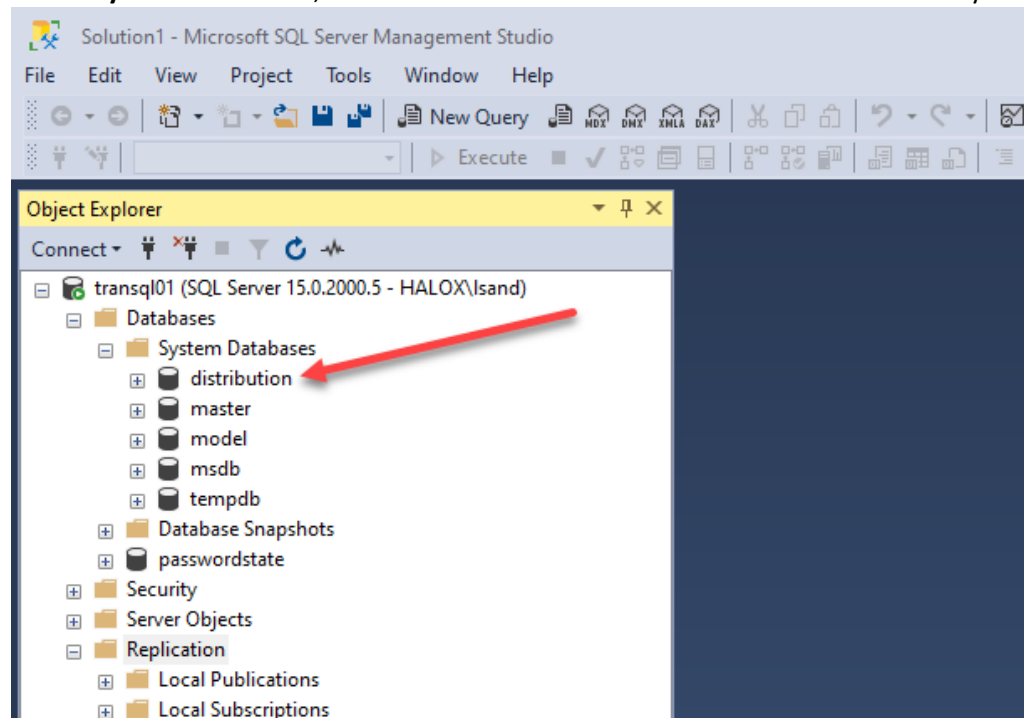
- Click **Finish**



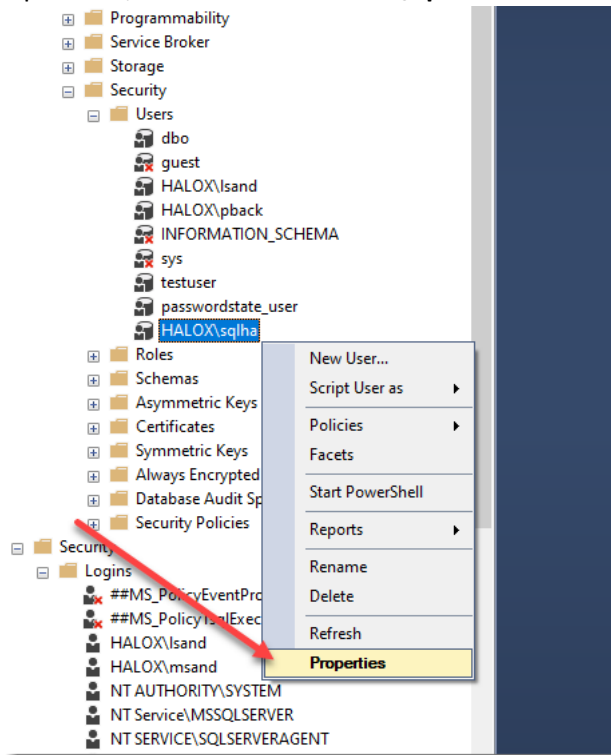
- The Distributor will now be configured, and when finished it will show “Success”. You can now click Close:



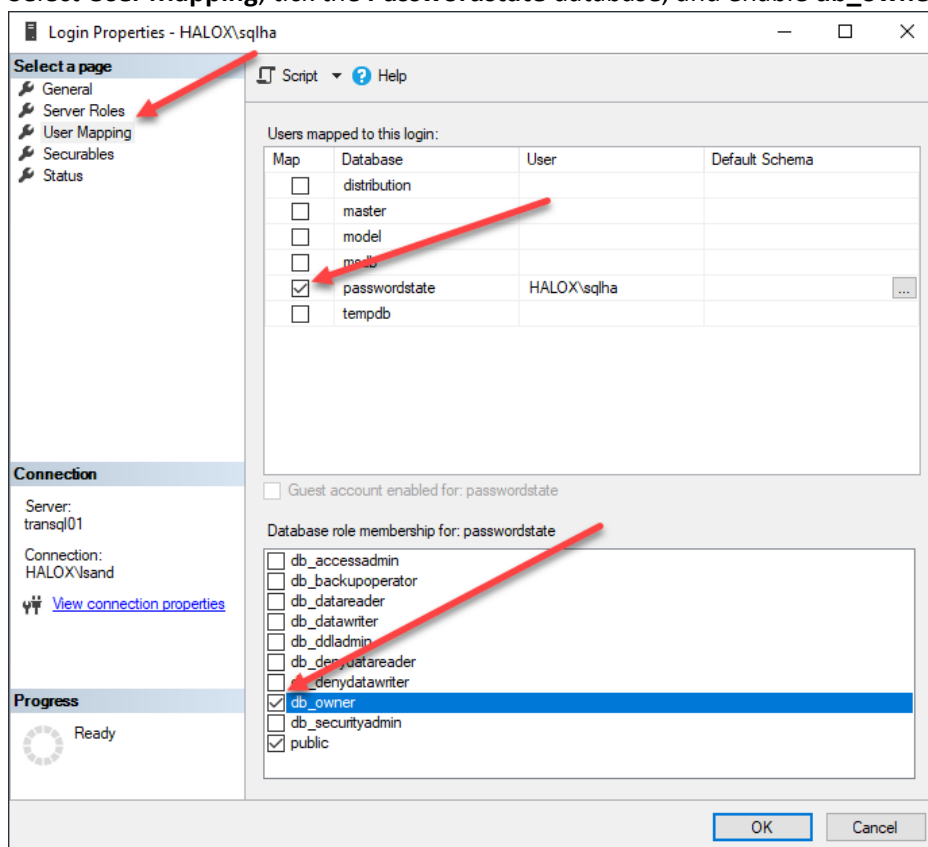
- Under **System Databases**, ensure the Distribution Database has been successfully created



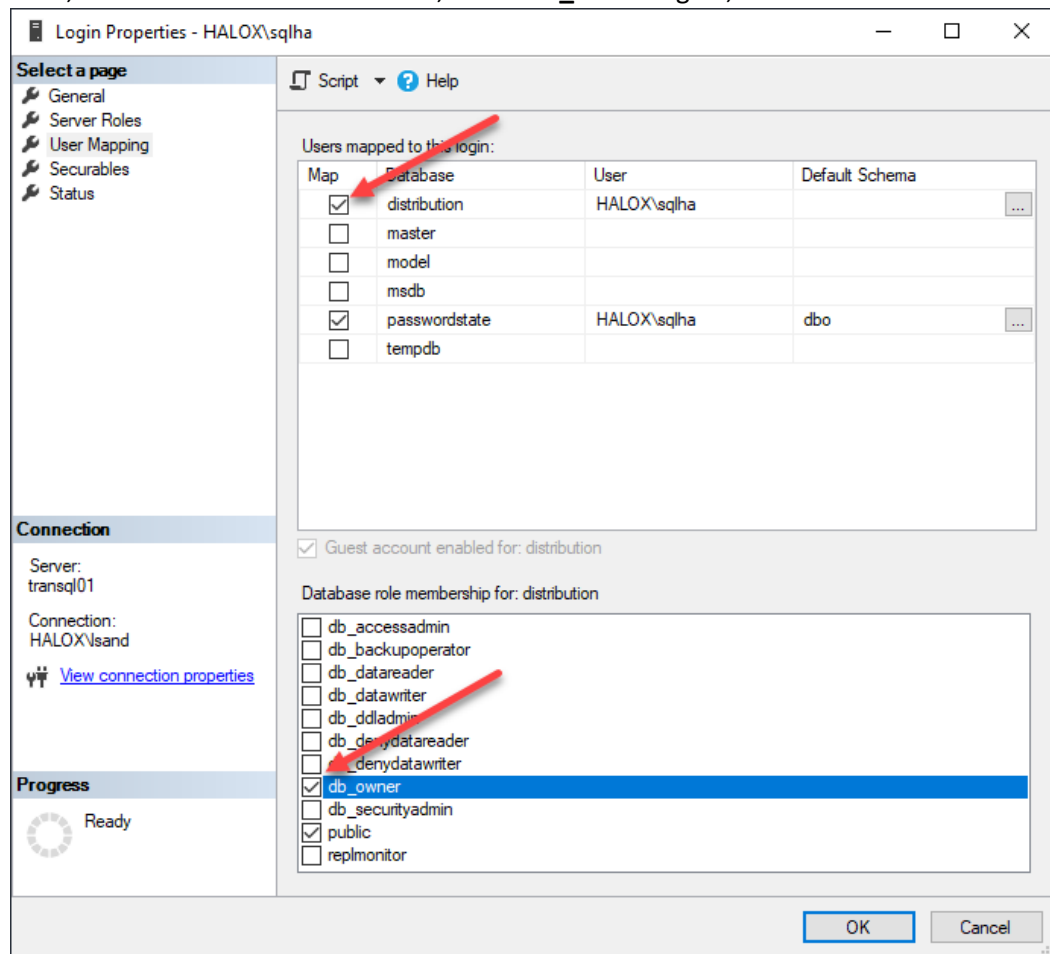
- Expand **Security** -> **Logins** and select **Properties** for the domain account which will be used for replication, in this case it is **halox\sqlha**



- Select **User Mapping**, tick the **Passwordstate** database, and enable **db\_owner** rights



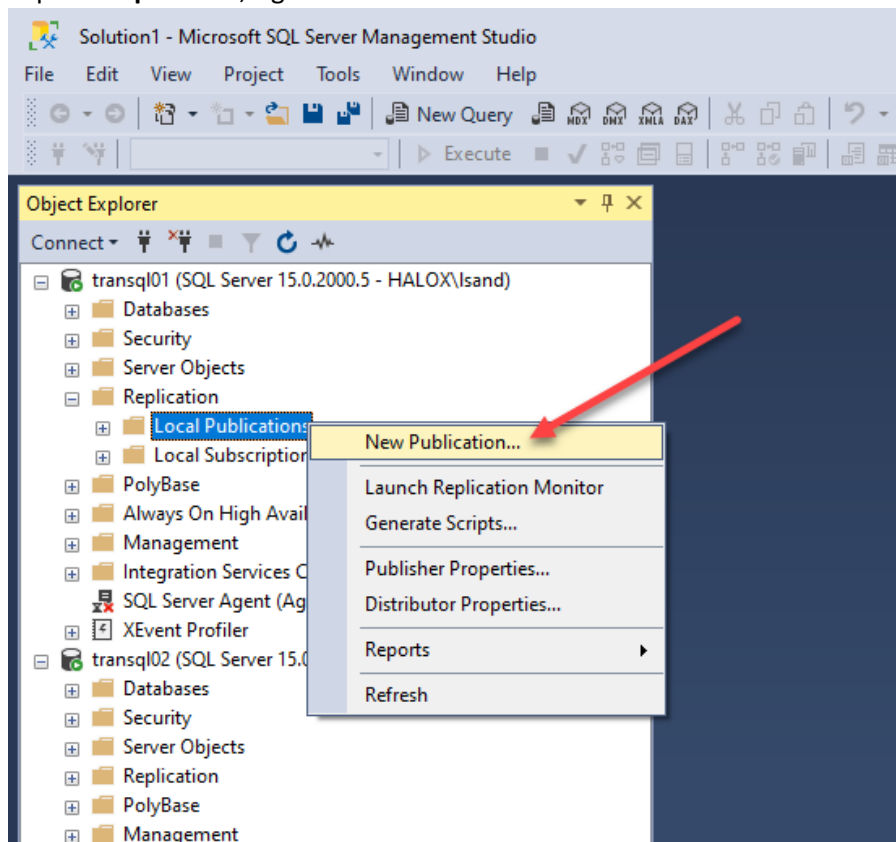
- Next, tick the **Distribution** database, select **db\_owner** rights, and click **OK**



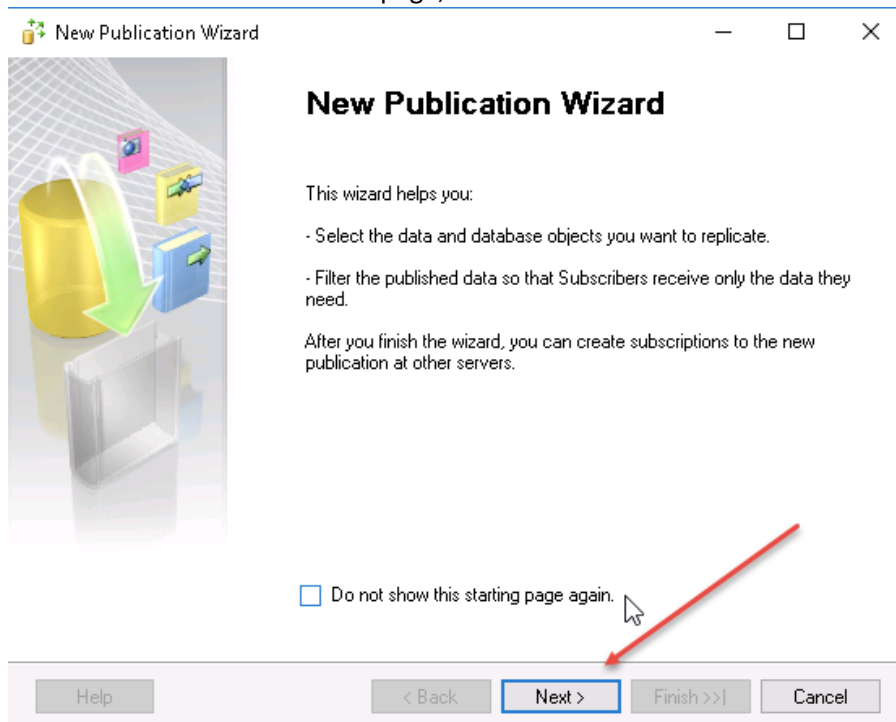


## 4 Creating the Publisher

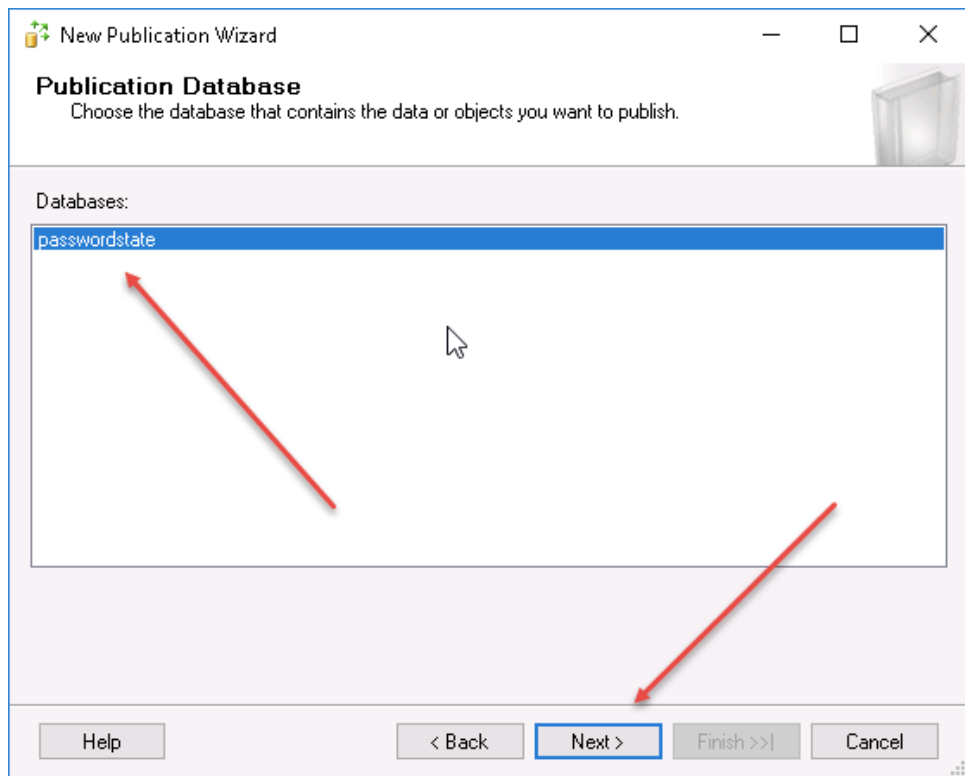
- Open **SQL Management Studio Tools** and connect to **transql01.halox.net**
- Expand **Replication**, right click **Local Publication** and select **New Publication**



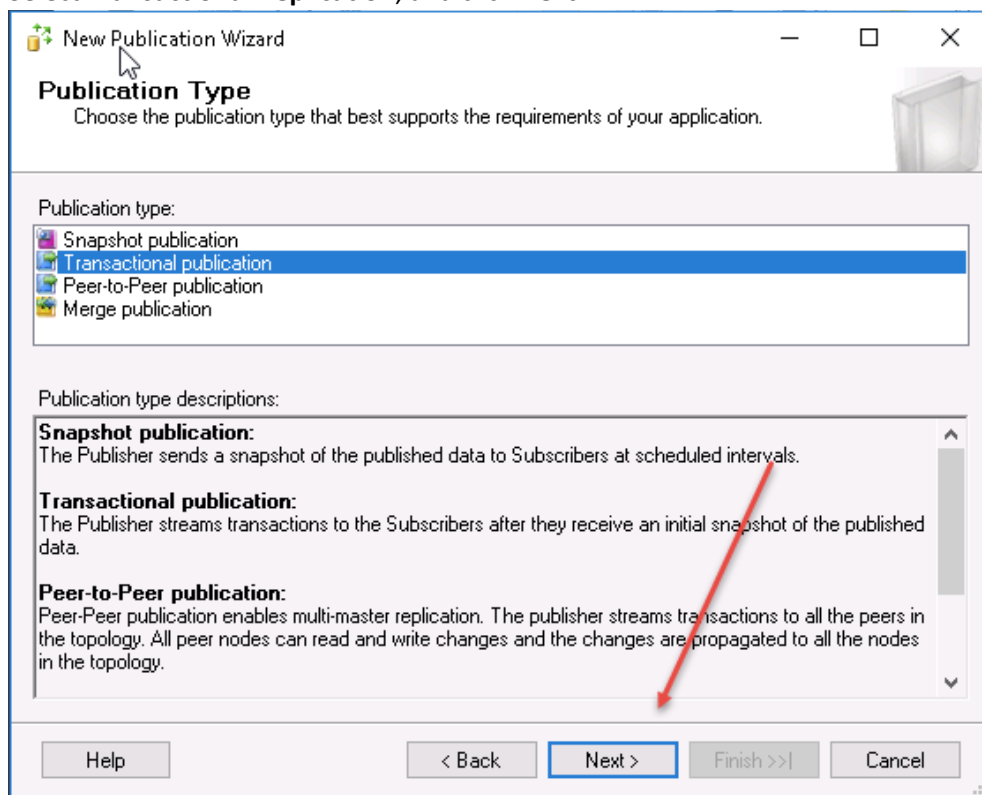
- At the New Publication Wizard page, click **Next**



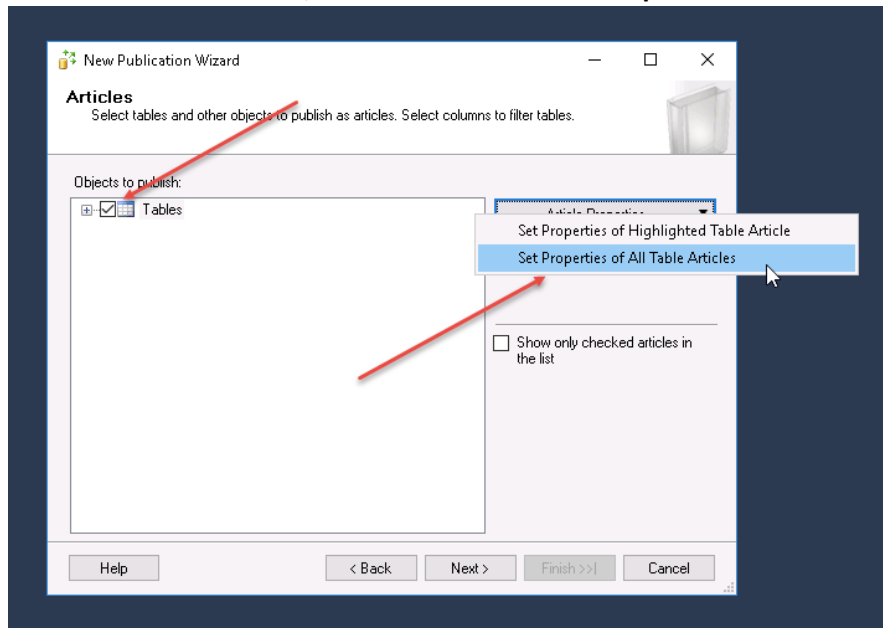
- Select the **Passwordstate** database and click **Next**



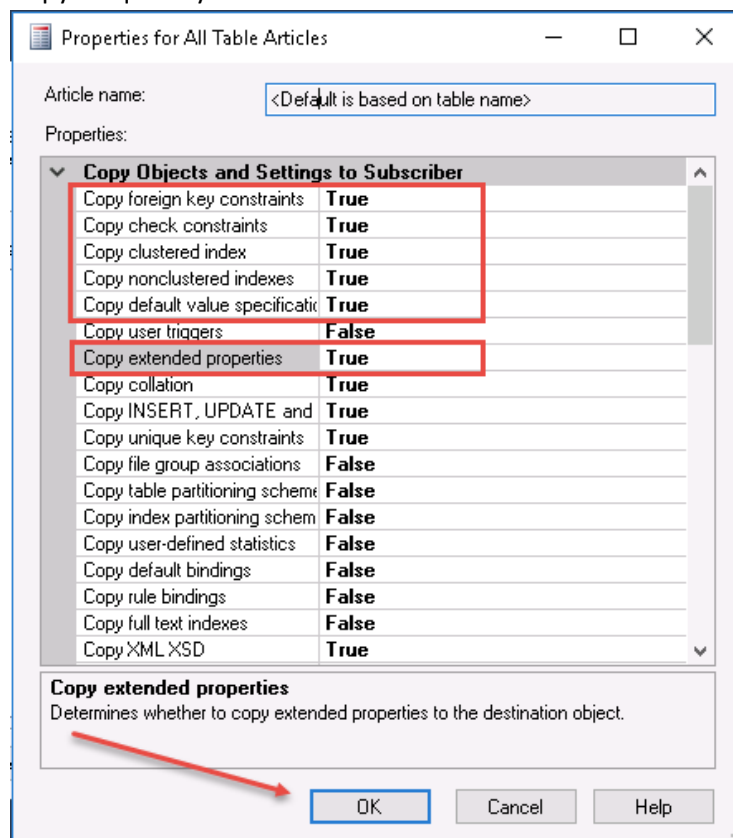
- Select **Transactional Replication**, and click **Next**



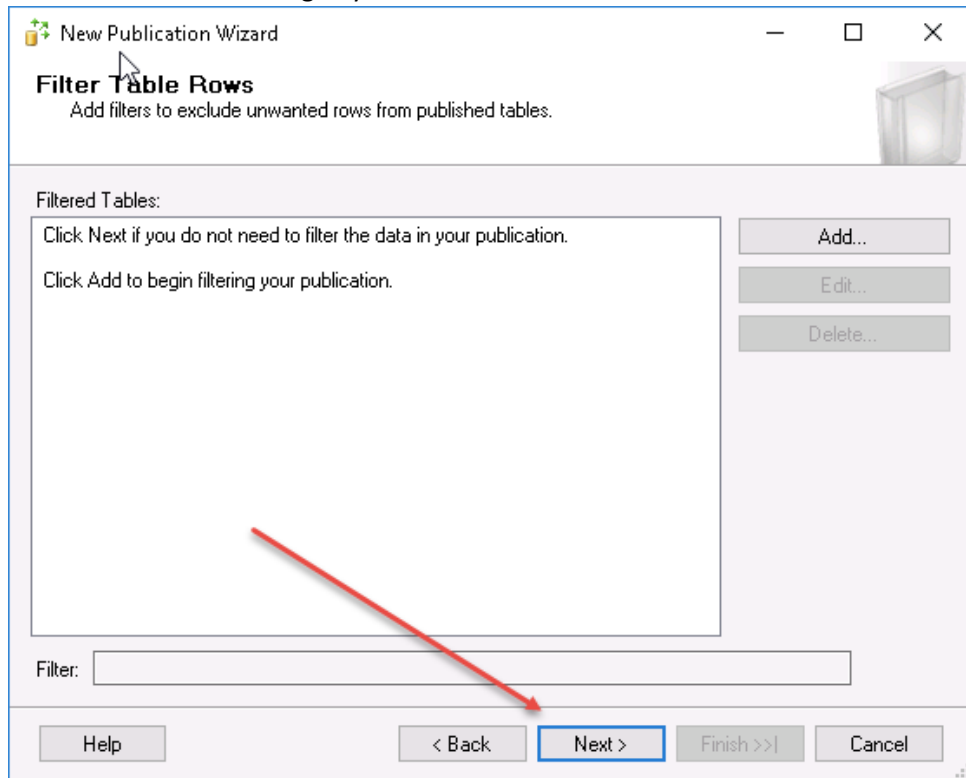
- Tick the **Tables** checkbox, and then select the **Set Properties of All Table Articles**



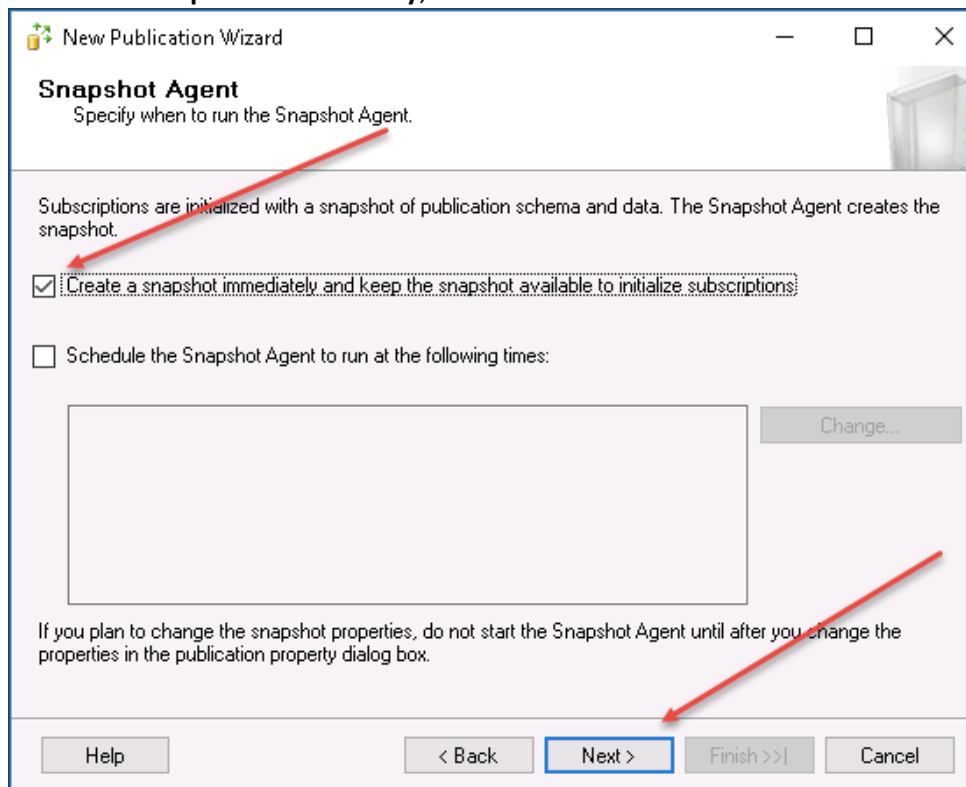
- Set the following options to **True** if not already set, and click **OK**, and then click **Next**
  - Copy foreign key constraints
  - Copy check constraints
  - Copy clustered index
  - Copy nonclustered indexes
  - Copy default value specifications
  - Copy extended properties
  - Copy unique key constraints



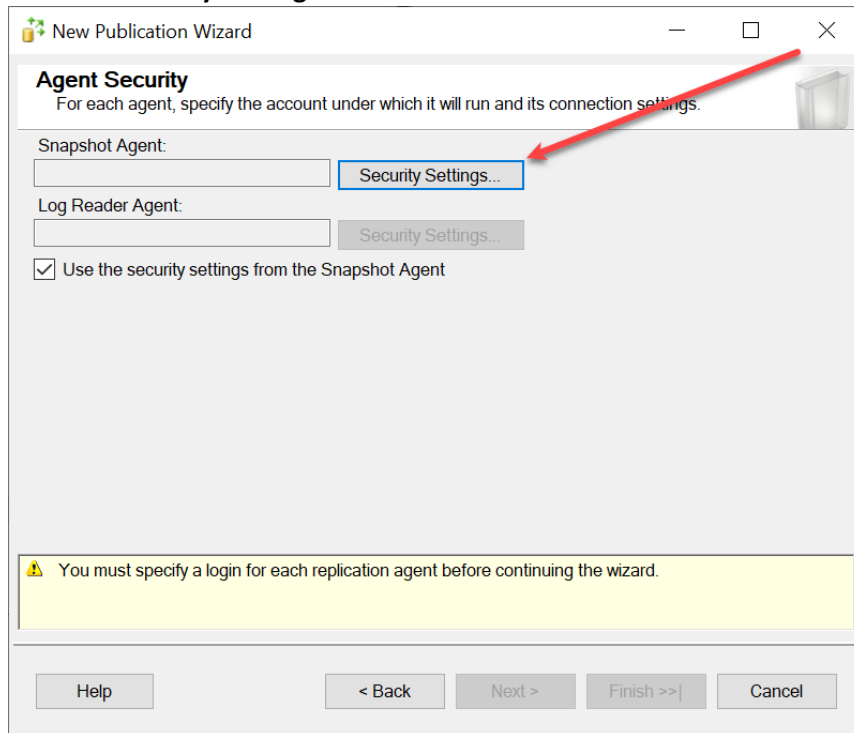
- Click **Next** without adding any filters



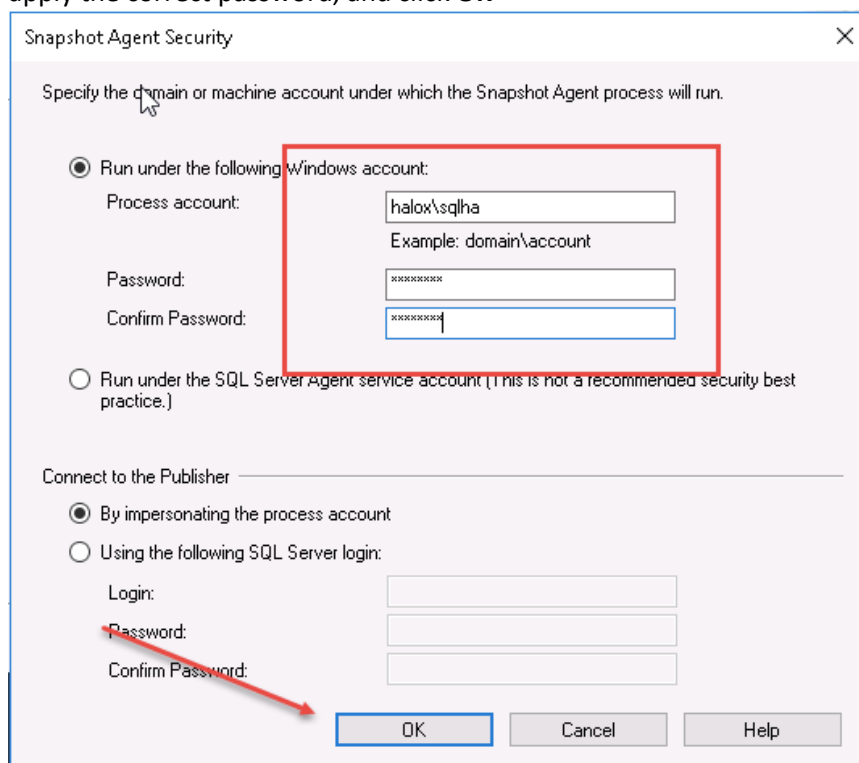
- Tick **Create snapshot immediately**, and click **Next**



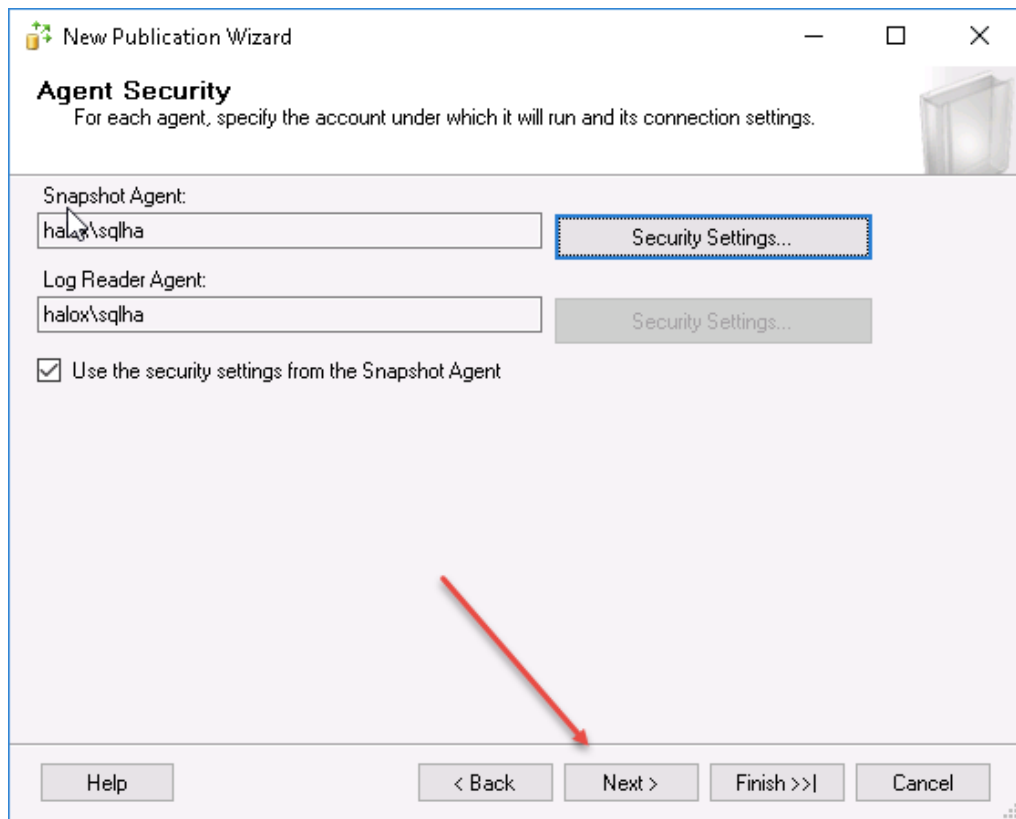
- Click the **Security Settings** button



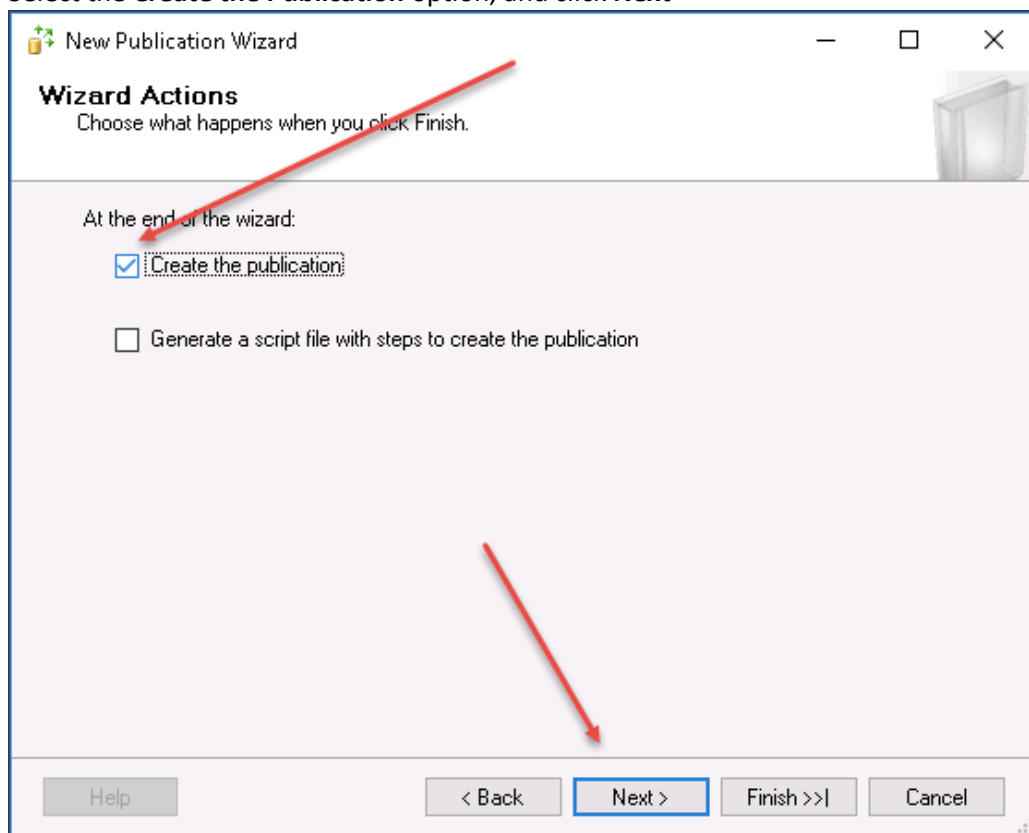
- Click **Security Settings** button and set you domain account in the format of **domain\username**, and apply the correct password, and click **OK**



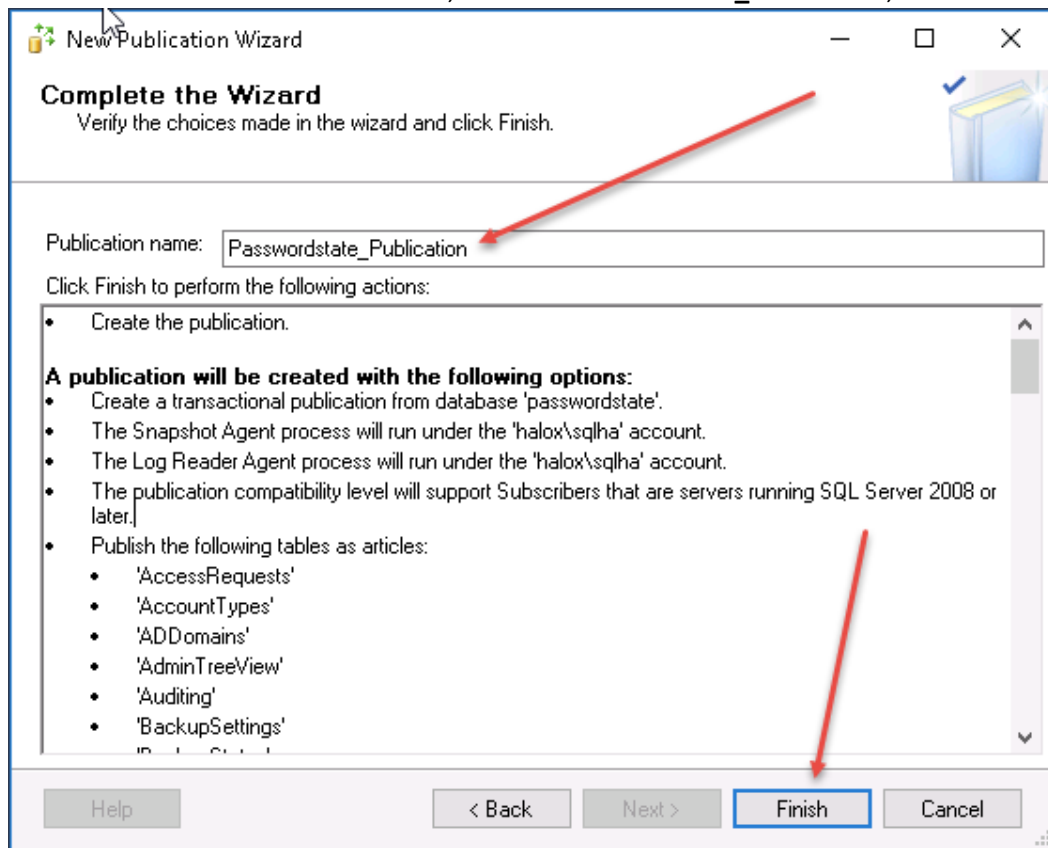
- Click **Next**



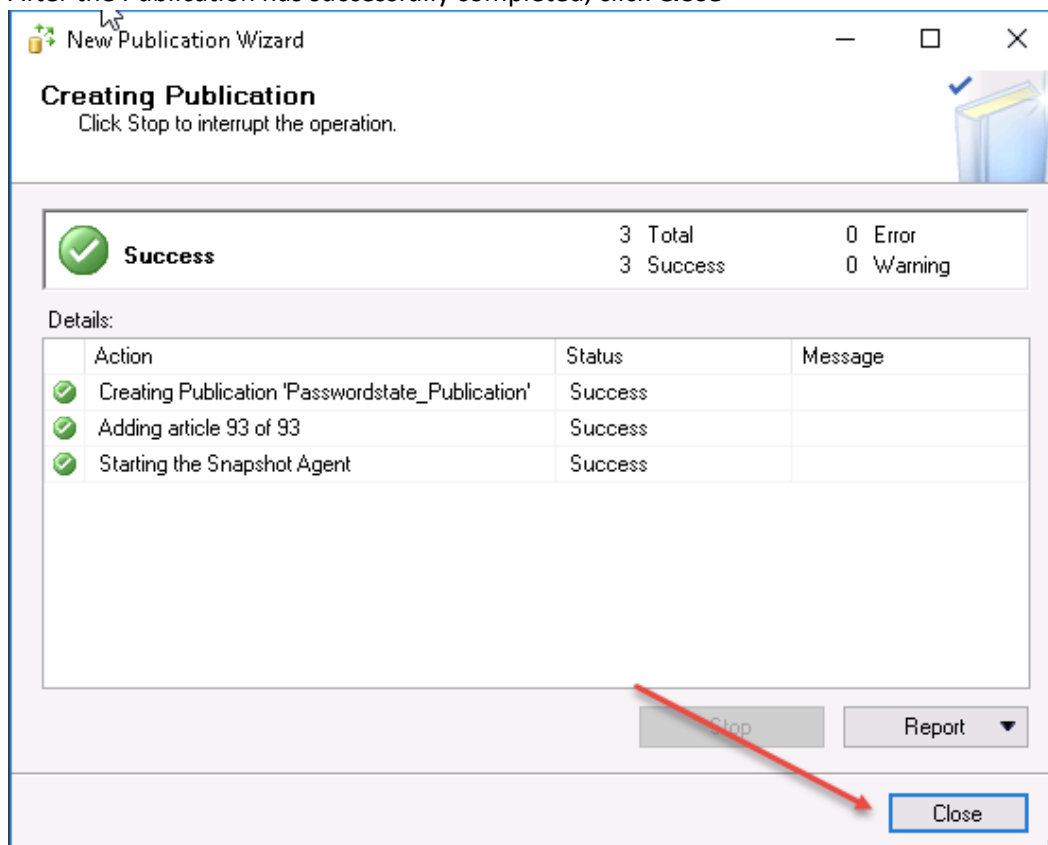
- Select the **Create the Publication** option, and click **Next**



- Give the Publication a relevant name, such as **Passwordstate\_Publication**, and click **Finish**



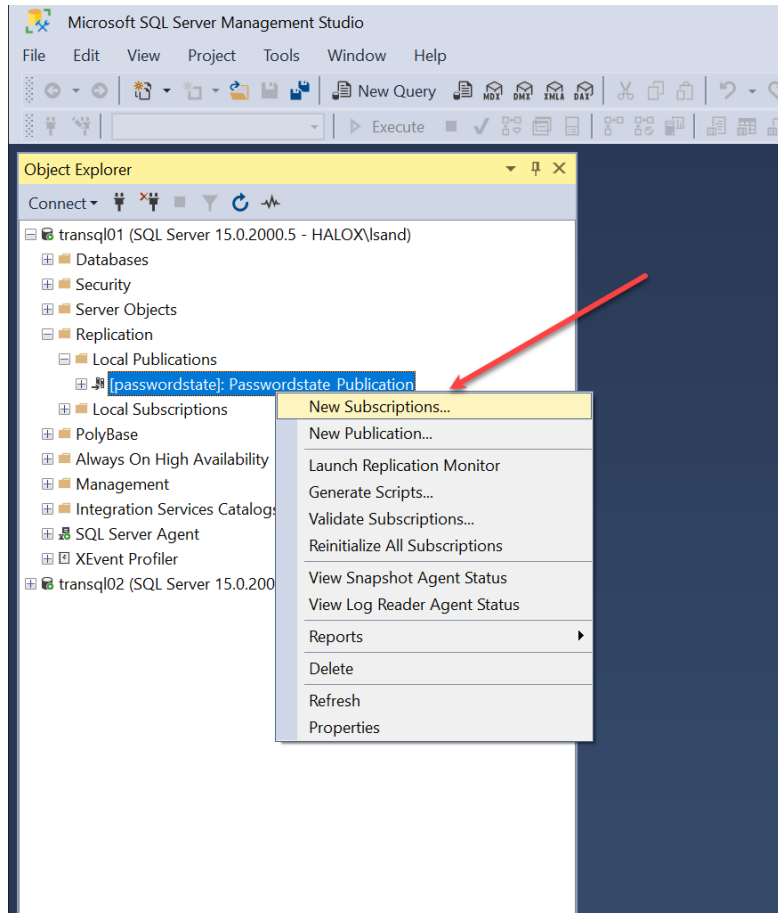
- After the Publication has successfully completed, click **Close**



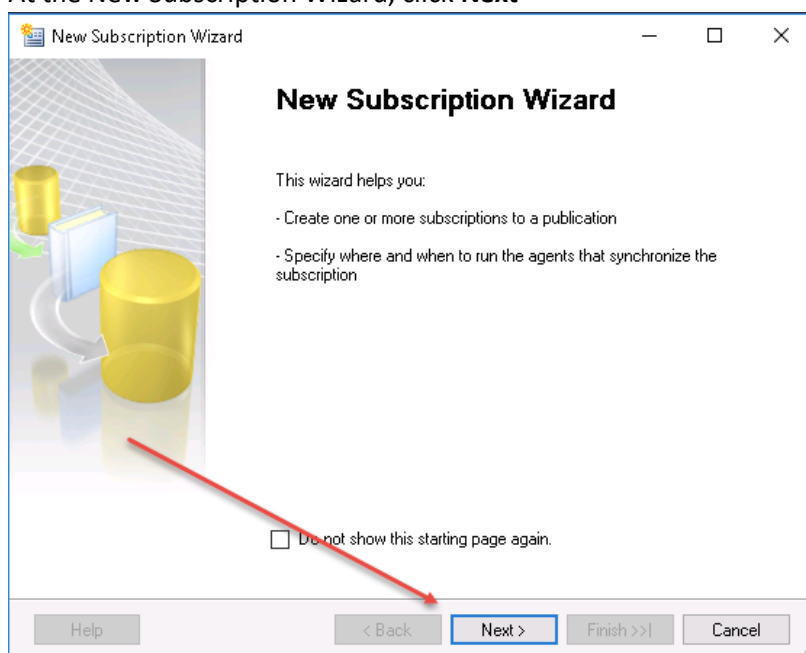
## 5 Creating the Subscriber

Once the publisher is created the next step is to create the subscriber for it.

- In **SQL Server Management Studio**, ensure you are connected to **transql01**, right click **Passwordstate\_Publication** and select **New Subscriptions**

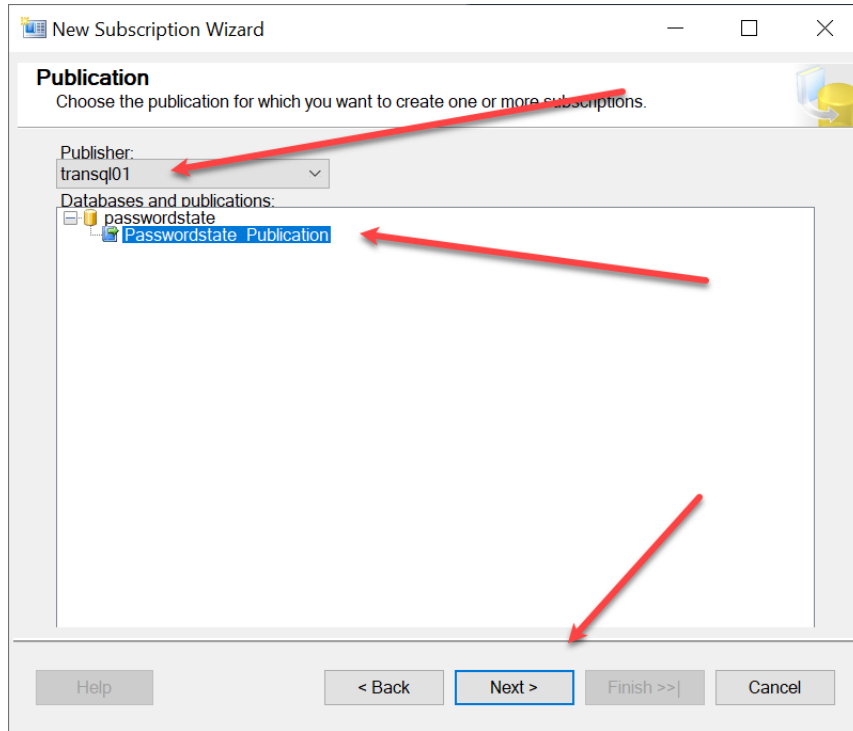


- At the New Subscription Wizard, click **Next**

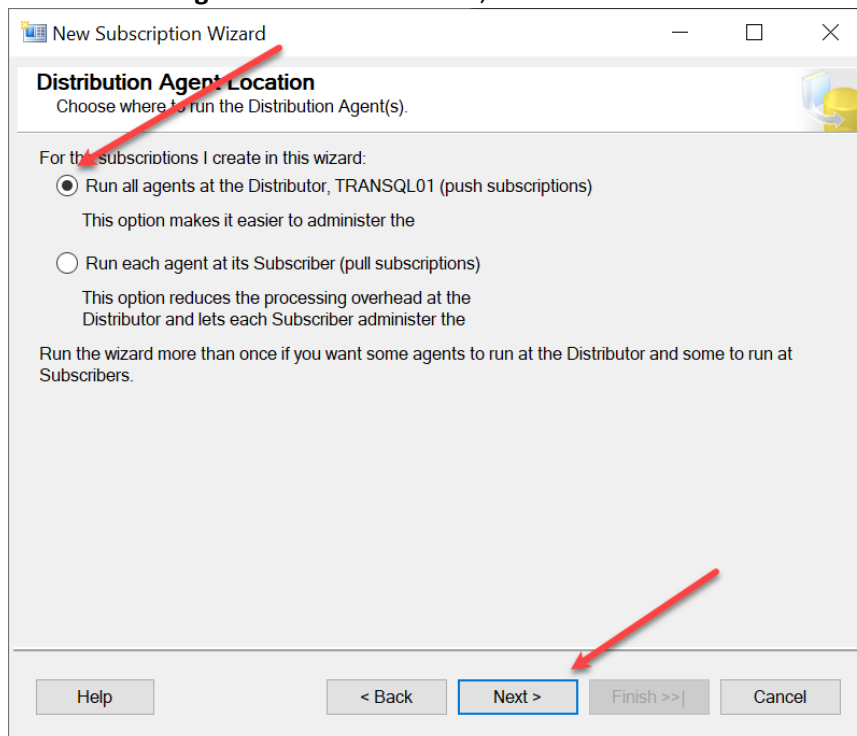




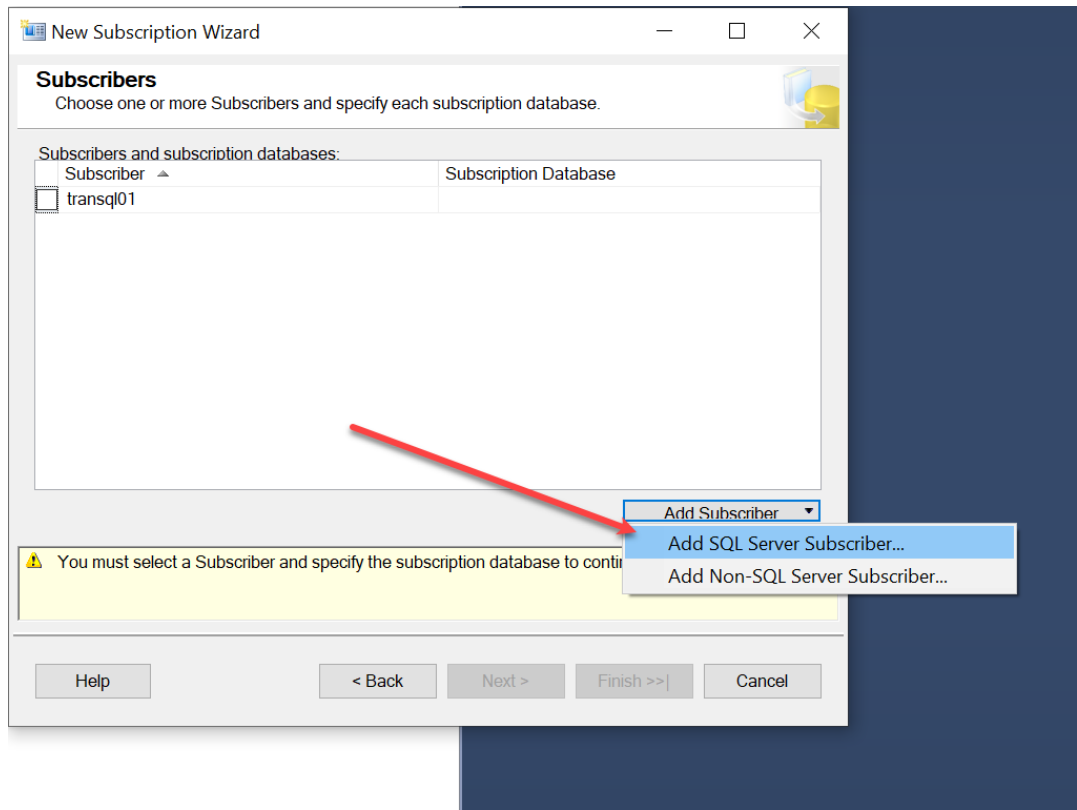
- Ensure you have **transql01** selected as the Publisher, and also the **Passwordstate\_Publication**, and click **Next**



- Select **Run all Agents at the Distributor**, and click **Next**

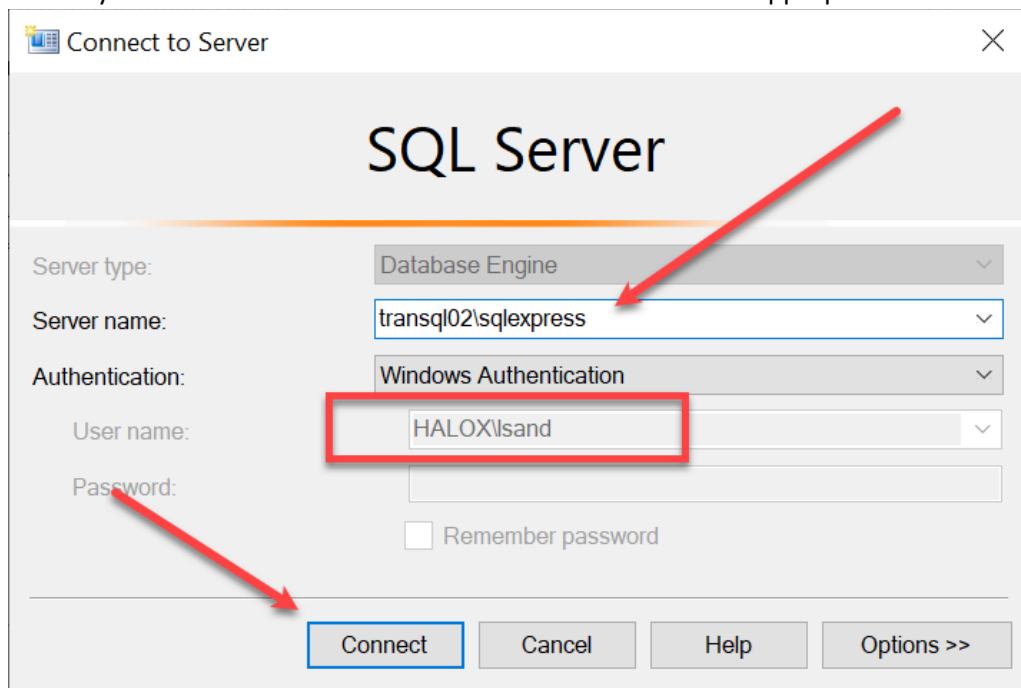


- Click **Add SQL Server Subscriber**

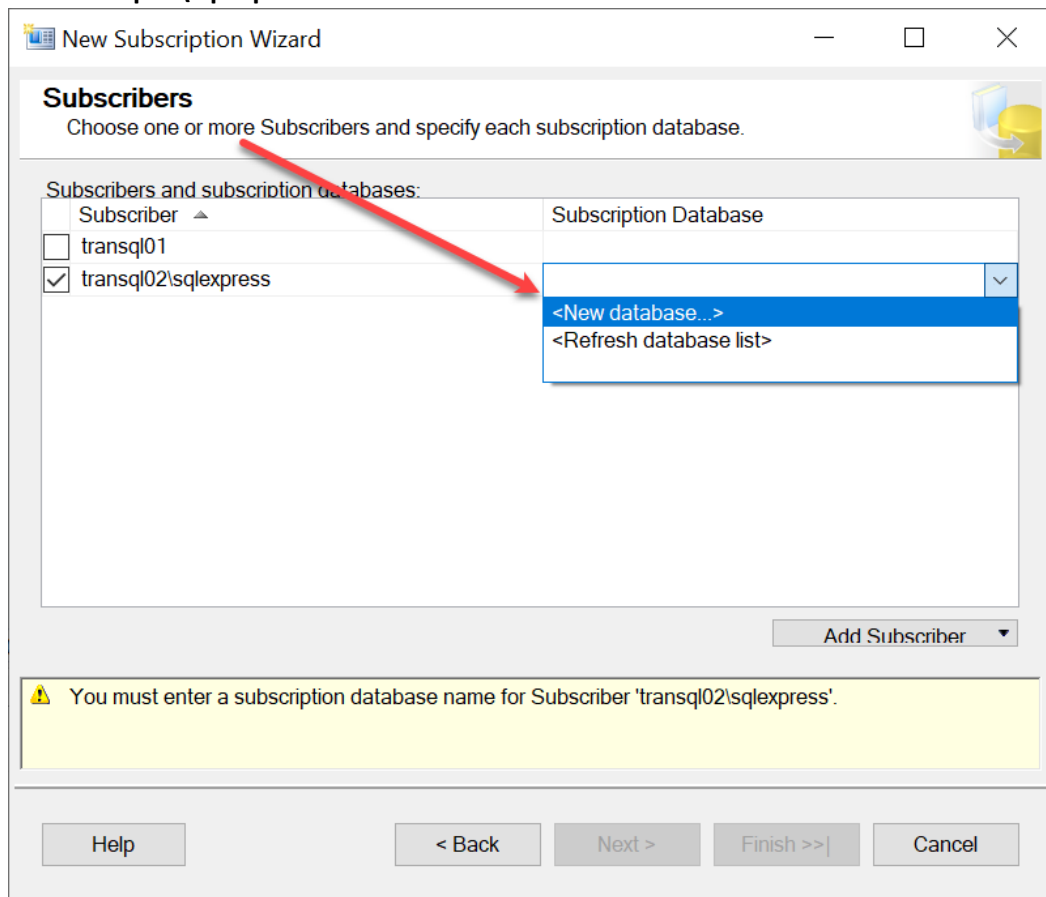


- Type in the name of your second database server, (**transql02\sqlexpress**), that you'll be replicating data to. By default, it will try to use the currently logged in user, (**halox\lsand**), to establish the connection. If this account does not have permissions in SQL on transql02, you'll either need to grant it permissions or use another account that does, such as the local SA account for transql02 server.

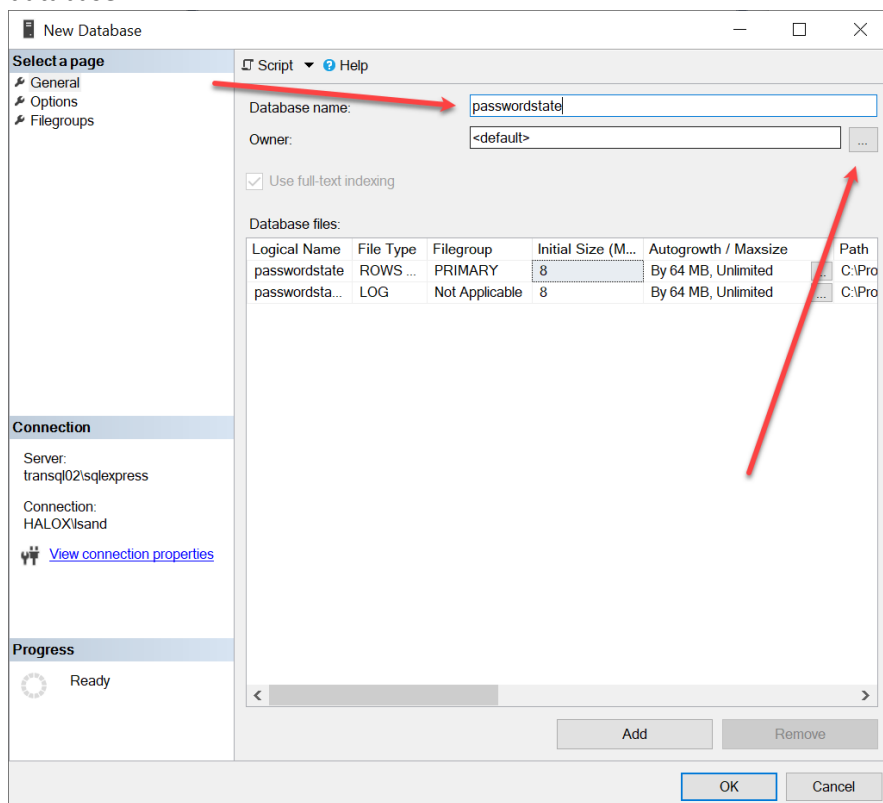
Ensure you include the instance name on this remote server if appropriate:



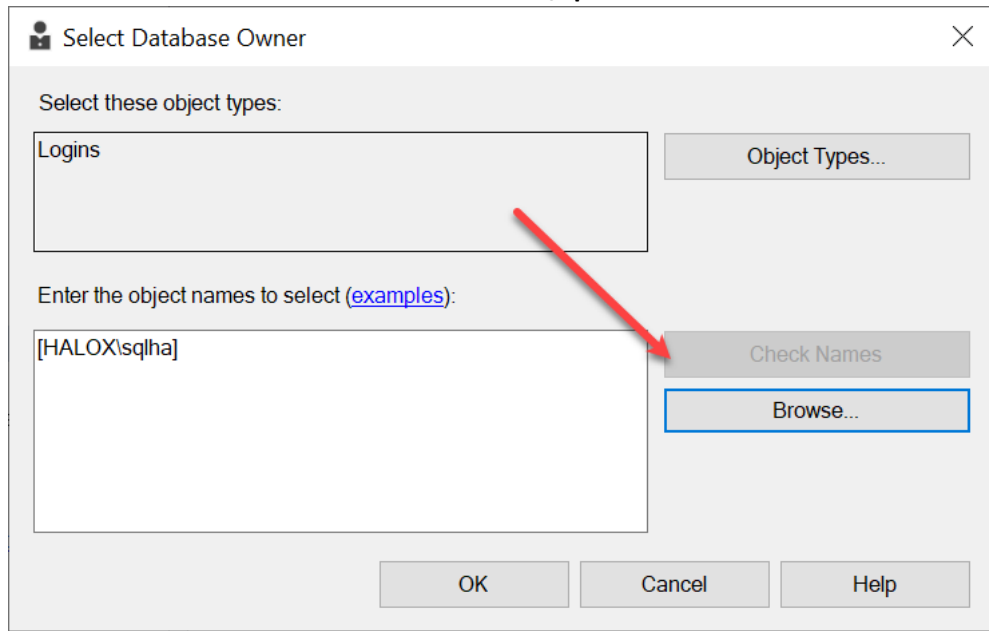
- Tick **transql02\sqlexpress** and select **New Database**



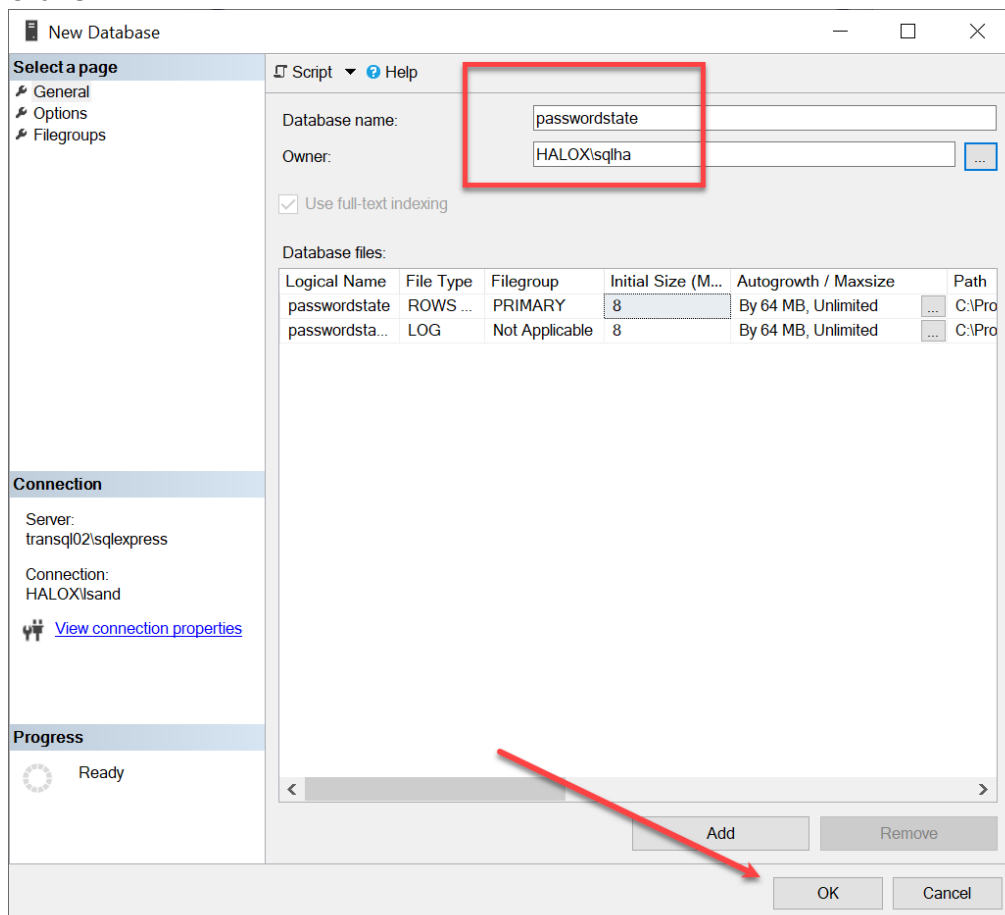
- Set the database name to be **passwordstate**, and then click the **button** to set an Owner of the database



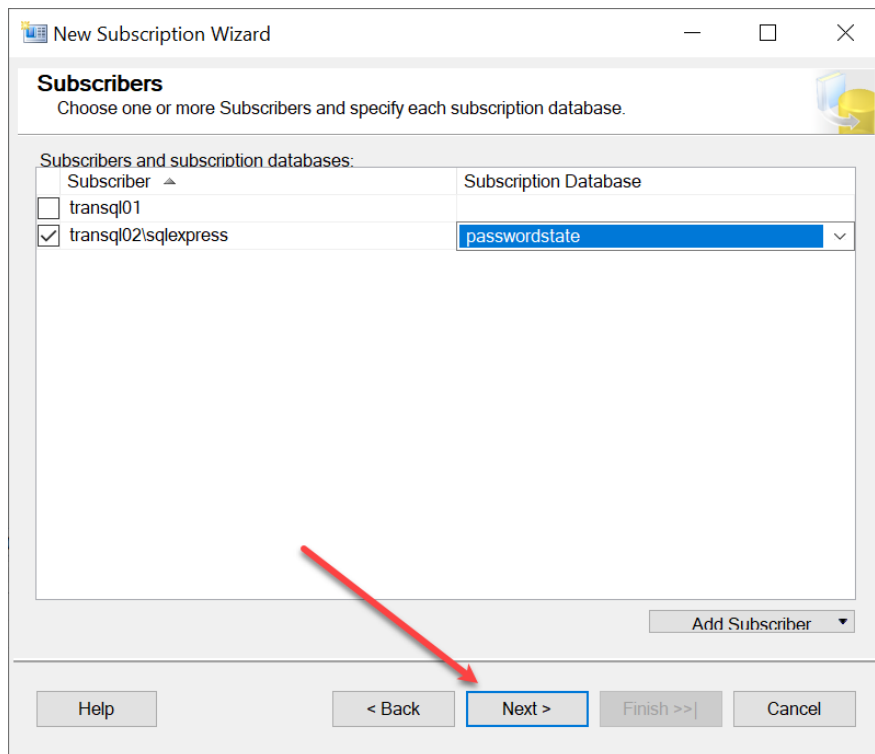
- Click the Browse button and select the **halox\sqlha** account and click **OK**



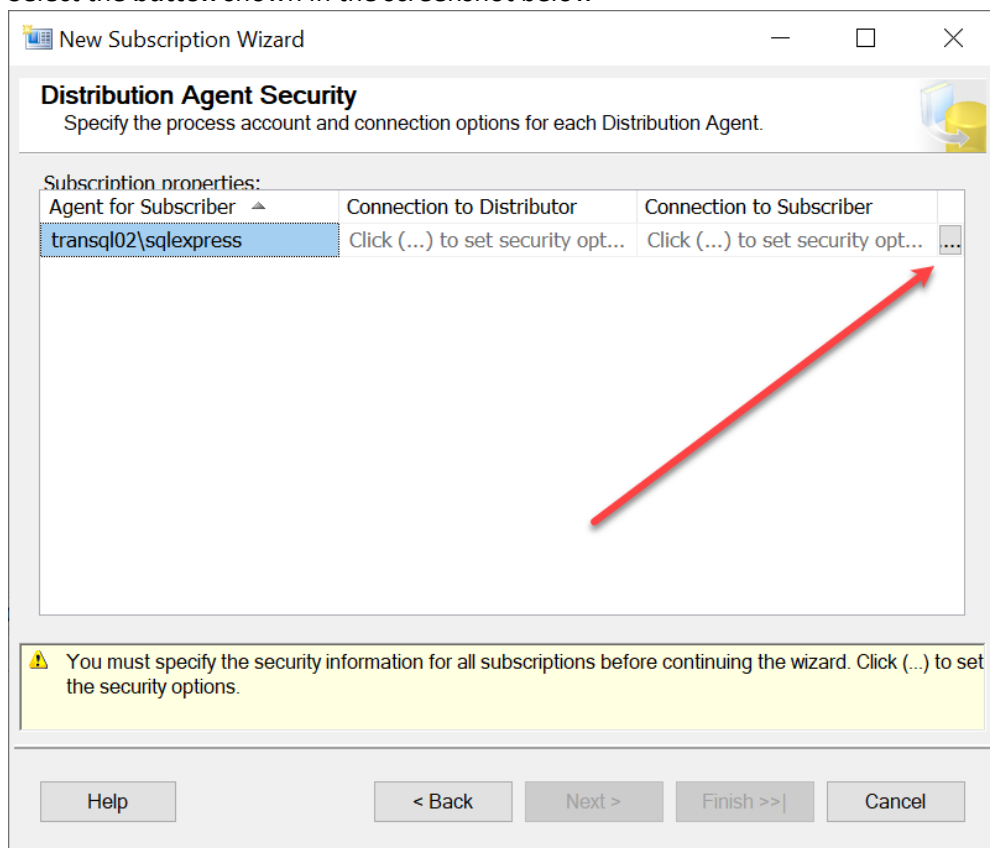
- Click **OK**



- Click **Next**



- Select the **button** shown in the screenshot below



- Set your domain account in the format of **domain\username**, and apply the correct password, and click **OK**

**Distribution Agent Security**

Specify the domain or machine account under which the Distribution Agent process will run when synchronizing this subscription.

☒ Run under the following Windows account:

Process account:   
Example: domain\account

Password:   
Confirm Password:

☐ Run under the SQL Server Agent service account (This is not a recommended security best practice.)

Connect to the Distributor

☒ By impersonating the process account  
☐ Using a SQL Server login

The connection to the server on which the agent runs must impersonate the process account. The process account must be a member of the Publication Access List.

Connect to the Subscriber

☒ By impersonating the process account  
☐ Using the following SQL Server login:

Login:   
Password:   
Confirm password:

The login used to connect to the Subscriber must be a database owner of the subscription database.

**OK** **Cancel** **Help**

- Click **Next**

**New Subscription Wizard**

**Distribution Agent Security**  
Specify the process account and connection options for each Distribution Agent.

Subscription properties:

| Agent for Subscriber | Connection to Distributor | Connection to Subscriber  |
|----------------------|---------------------------|---------------------------|
| transql02\sqlexpress | Impersonate 'halox\sqlha' | Impersonate 'halox\sqlha' |

**Help** **< Back** **Next >** **Finish >>|** **Cancel**

- Select **Run continuously**, and click **Next**

The screenshot shows the 'New Subscription Wizard' dialog box, specifically the 'Synchronization Schedule' step. The title bar reads 'New Subscription Wizard'. The main heading is 'Synchronization Schedule' with the instruction 'Specify the synchronization schedule for each agent.' Below this is a table with the following data:

| Subscriber           | Agent Location | Agent Schedule   |
|----------------------|----------------|------------------|
| transql02\sqlexpress | Distributor    | Run continuously |

Two red arrows are present: one points from the 'Run continuously' text in the 'Agent Schedule' column to the 'Next >' button at the bottom, and another points from the 'Next >' button to the 'Next >' button itself. The bottom of the dialog contains buttons for 'Help', '< Back', 'Next >', 'Finish >>|', and 'Cancel'.

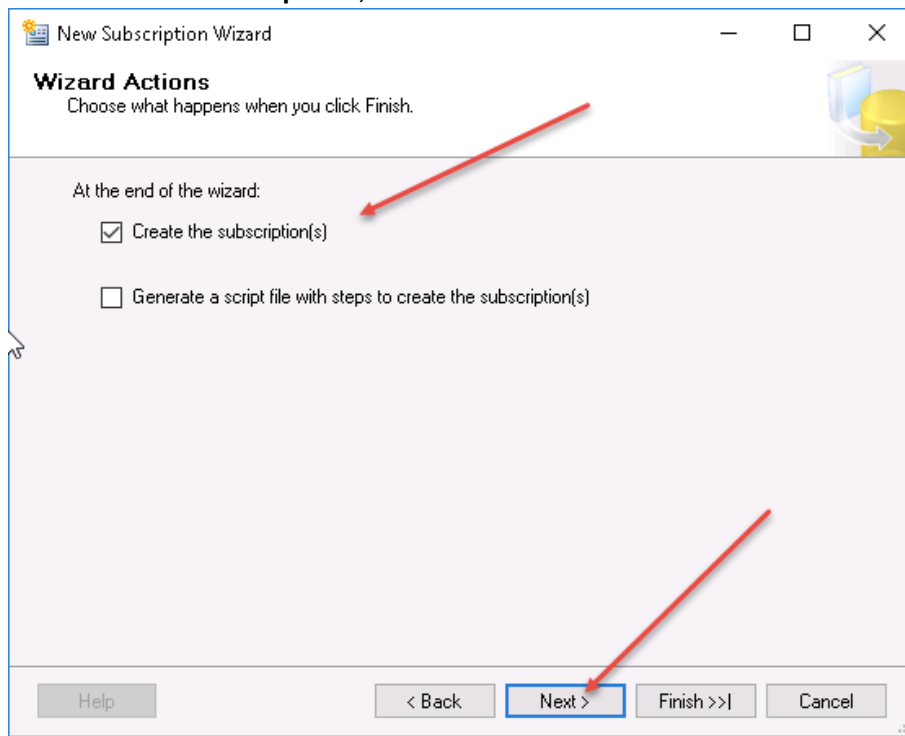
- Ensure you select to **Initialize Immediately**, and click **Next**

The screenshot shows the 'New Subscription Wizard' dialog box, specifically the 'Initialize Subscriptions' step. The title bar reads 'New Subscription Wizard'. The main heading is 'Initialize Subscriptions' with the instruction 'Specify whether to initialize each subscription with a snapshot of the publication data and'. Below this is a table with the following data:

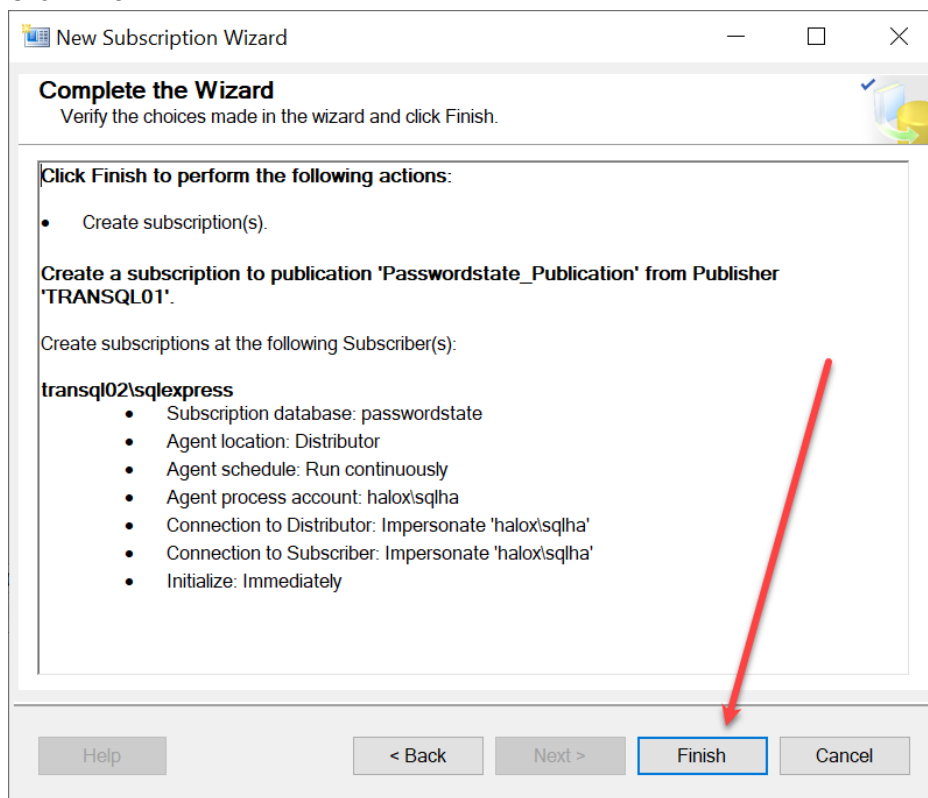
| Subscriber           | Memory Optimized         | Initialize                          | Initialize When |
|----------------------|--------------------------|-------------------------------------|-----------------|
| transql02\sqlexpr... | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Immediately     |

Two red arrows are present: one points from the 'Initialize' checkbox to the 'Initialize When' column, and another points from the 'Initialize When' column to the 'Next >' button at the bottom. Below the table is a scroll bar. At the bottom of the dialog, there is a yellow warning box with a triangle icon and the text: 'The Snapshot Agent must run and generate a snapshot of the publication before the subscriptions can be initialized.' The bottom of the dialog contains buttons for 'Help', '< Back', 'Next >', 'Finish >>|', and 'Cancel'.

- Tick **Create the Subscriptions**, and click **Next**

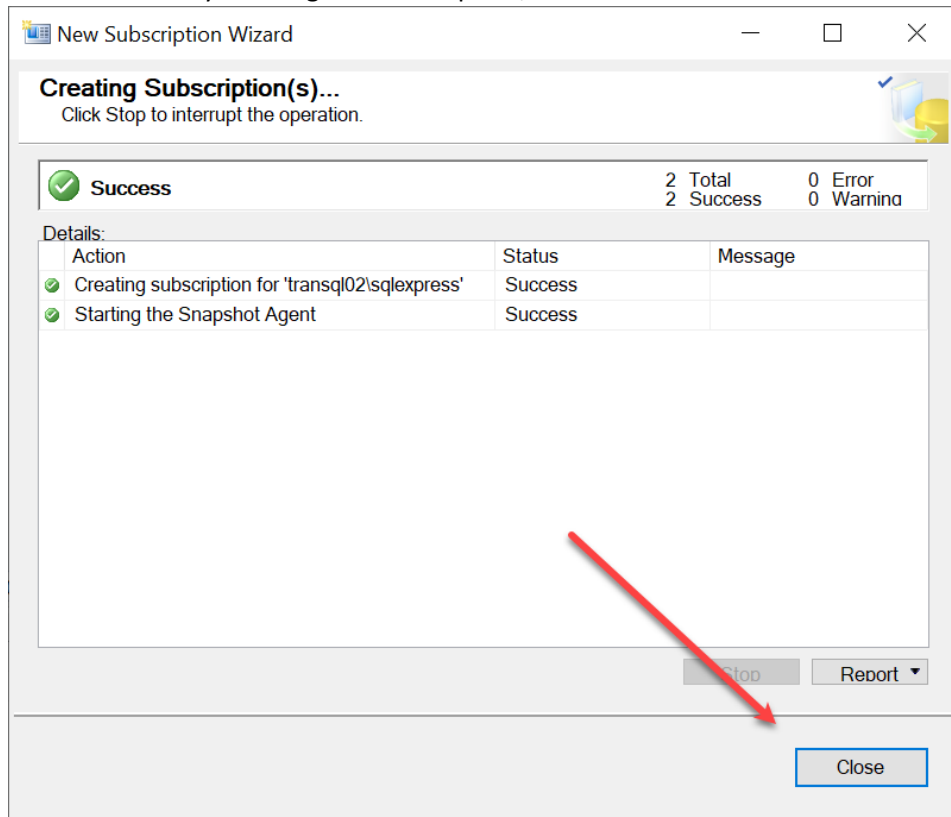


- Click **Finish**





- After successfully creating the Subscription, click **Close**



**Note:** If creating the subscriber fails at this point, and you are using SQL Server 2019, this may be caused by a bug in SQL. Please see **Section 7: How to troubleshoot Transaction Replication** for a fix for this issue.

## 6 Connect Passwordstate to your replicated Database

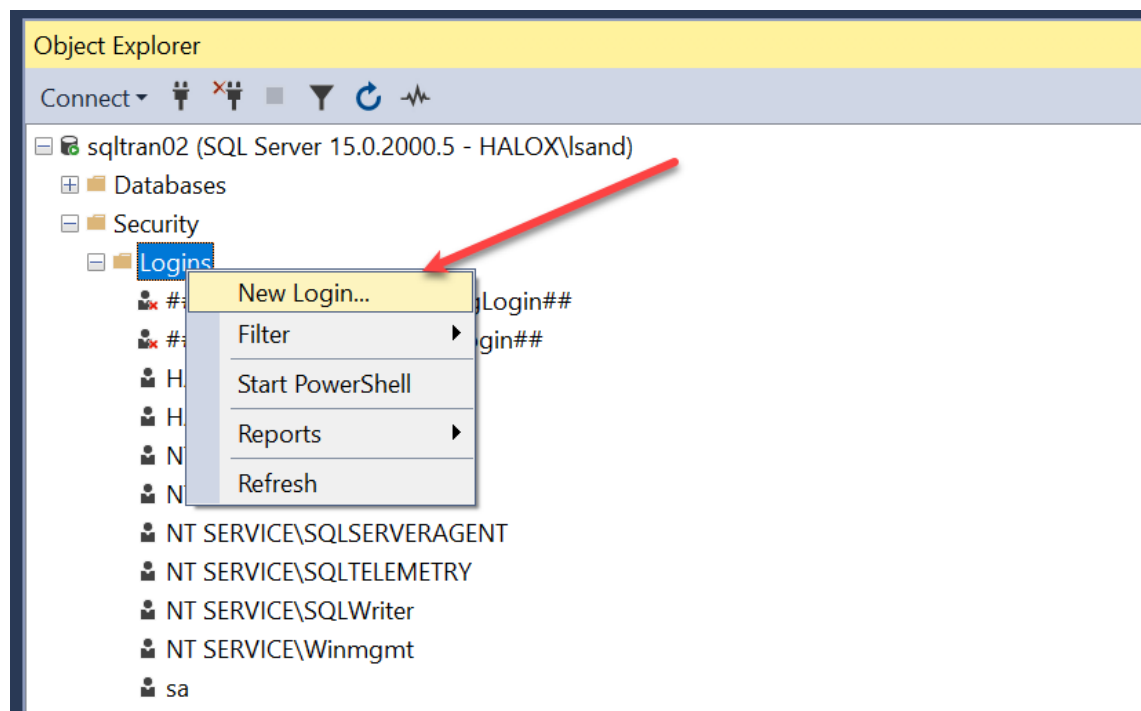
Now that you have your data replicating to a second database, you can set up your second Passwordstate website so you can have instant read only fail over access. Please note you will need to High Availability Module in Passwordstate for this, as this allows for two production instances of Passwordstate running in real time.

To set up your second Passwordstate website, please follow these instructions:

[https://www.clickstudios.com.au/downloads/version8/High\\_Availability\\_Installation\\_Instructions.pdf](https://www.clickstudios.com.au/downloads/version8/High_Availability_Installation_Instructions.pdf)

Prior to installing your second website using the instructions above, you'll need to add your **passwordstate\_user** SQL account into your second database instance.

- Open **SQL Management Studio Tools** and connect to your second database server, and under Security -> Logins, select **New Login**:



- Ensure you select SQL Server Authentication, and set your name for the account and a password. Note these details will be used during the initial install of your High Availability server:

Login - New

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Script Help

Login name: passwordstate\_user

Windows authentication

SQL Server authentication

Password: .....

Confirm password: .....

Specify old password

Old password: .....

Enforce password policy

Enforce password expiration

User must change password at next login

Mapped to certificate

Mapped to asymmetric key

Map to Credential

Mapped Credentials

| Credential | Provider |
|------------|----------|
|------------|----------|

Default database: master

Default language: <default>

OK Cancel

- Now you need to give this new account db\_owner rights to the Passwordstate database:

Login - New

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Script Help

Users mapped to this login:

| Map                                 | Database      | User               | Default Schema |
|-------------------------------------|---------------|--------------------|----------------|
| <input type="checkbox"/>            | master        |                    |                |
| <input type="checkbox"/>            | model         |                    |                |
| <input type="checkbox"/>            | msdb          |                    |                |
| <input checked="" type="checkbox"/> | passwordstate | passwordstate_user |                |
| <input type="checkbox"/>            | tempdb        |                    |                |

Guest account enabled for: passwordstate

Database role membership for: passwordstate

- ☐ db\_accessadmin
- ☐ db\_backupoperator
- ☐ db\_datareader
- ☐ db\_datawriter
- ☐ db\_dladm
- ☐ db\_denydatareader
- ☐ db\_denydatawriter
- ☒ db\_owner
- ☐ db\_securityadmin
- ☒ public

OK Cancel

## 7 Transactional Replication Considerations

### **Allow large blocks of data to replicate:**

If using transactional replication, you will need to run the following SQL Statement against your primary and your secondary database as a once off process. This will allow replication of documents to occur between the two databases.

The below statement configures SQL Server to allow an unlimited size document to replicate as by default SQL is configured with a maximum size limit.

```
GO
EXEC sp_configure 'show advanced options', 1 ;
RECONFIGURE ;
GO
EXEC sp_configure 'max text repl size', -1 ;
GO
RECONFIGURE;
GO
```

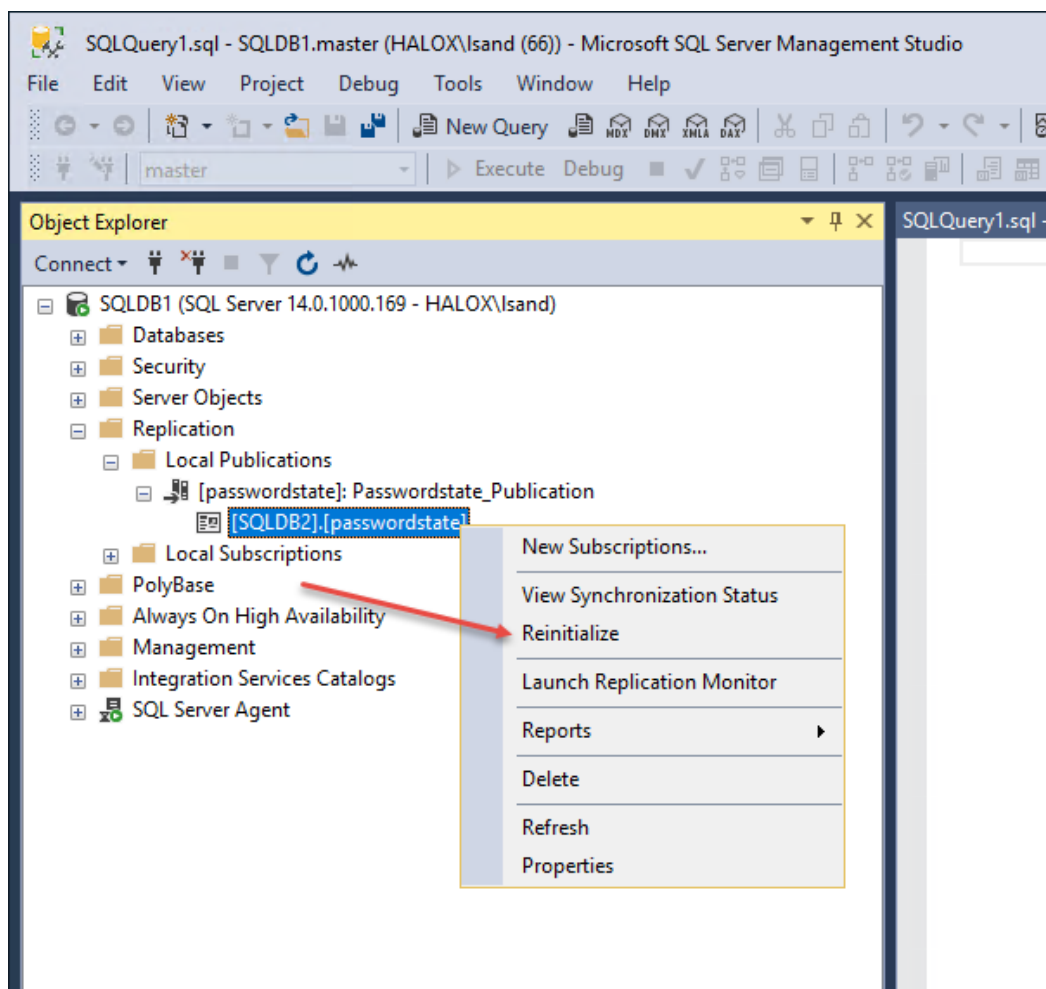
## 8 How to troubleshoot Transaction Replication

- Ensure your SQL Services are started on both servers and are running under the default accounts
- Ensure your domain account, which was **halox\sqlha** in this document, the following permissions:
  - It must have **db\_owner** rights to both Passwordstate databases on both servers
  - It must have **db\_owner** rights on the Distribution database on your Primary database server
  - It must have **Modify permissions** to the snapshot folder on your primary database server
- If you Launch the Replication Monitor from the Publication, it will give you any errors if you are experiencing them.
- A quick test to see if replication is working between the two Passwordstate databases, is to run the following SQL Query in **SQL Management Studio** when connected to each database server. This will return a count of the Auditing table, and you should get the exact same results from both queries if replication is working successfully

USE Passwordstate

SELECT COUNT(\*) FROM Auditing

- Check your snapshot folder for any files. If there are no files then replication is not working. Check all permissions are set correctly, and possibly reinitialize the replication



### SQL Server2019 Issue:

SQL Server 2019, at the time of writing this document, contained a bug which affected the creation of snapshots. If you find that your replication is failing try adding the following environment variables to your System Path on your primary database server:

`%SystemRoot%\SysWOW64\`

`%SystemRoot%\SysWOW64\1033`

