



Migrating an On-Premise installation of Passwordstate to Azure Virtual Machine with an Azure SQL Database

Table of Contents

1	OVERVIEW.....	3
2	PREPARATION REQUIRED	4
3	INSTALL PASSWORDSTATE ON THE NEW WEBSERVER	5
4	MIGRATING THE DATABASE	6

1 Overview

Please use the following instructions as a guide for moving an existing **Passwordstate Website and Database** to **Azure**.

This guide assumes you are currently running either Passwordstate with major build **7, 8 or 9**.

When this process is finished, you will be running the latest version of Passwordstate 9, so an upgrade of your database will occur unless you are already running the latest version.

This guide also assumes you are intending on using an **Azure SQL database** - <https://azure.microsoft.com/en-au/products/azure-sql/database> instead of a self-hosted Microsoft SQL install on an Azure Virtual Machine.

****Note**** You can leave your currently production instance of Passwordstate running until you can confirm that the migration has worked. Once you can confirm the new website work then you should decommission the old server.

2 Preparation Required

1. Ensure your new **Virtual Machine** in Azure is built with a Microsoft Operating System that meets the Passwordstate system requirements: <https://www.clickstudios.com.au/passwordstate-system-requirements.aspx>
2. In your Azure portal, you'll need to create a **SQL Server** if you do not have one already, with an Admin account that you know the credentials for.
3. Consider preparing **DNS** and your **SSL certificate** for the new web server, and if you will need any open ports for incoming HTTPS traffic.
4. Log in to your current on-premise Passwordstate instance and add the new web server name to the list of authorized web servers. The **Server Role** you specify for this when adding it in should be "**Primary Server**". You can do this from the screen **Administration** -> **Authorized Web Servers**.
5. Take note of your **Emergency** password, just in case you need to use this later in this migration process. This password can be found under **Administration** -> **Emergency Access**
6. Take a backup of your existing Passwordstate instance, which includes the web installation files, and the database. Passwordstate has its own backup feature you may have already configured you can use under **Administration** -> **Backups**, or you can use your own 3rd party backup solution if you have one. Another option is to back up the data manually using this guide: <https://forums.clickstudios.com.au/topic/13911-manual-backup-using-sql-management-studio-tools/>

3 Install Passwordstate on the New Webserver

1. On your new server, you will now need to download the appropriate Passwordstate installation files from Click Studios check sums page: <https://www.clickstudios.com.au/passwordstate-checksums.aspx>
 - a. If your current version of Passwordstate is less than build 8995, then you will need to download the build **8995** source files
 - b. If your current version is **8995** or above, you should instead download the latest version of the **Version 9** source files.
2. Now you can follow the [General Installation Instructions](#) to install Passwordstate, stopping when you complete **Section 5**. You should **not** proceed to **Section 6 - 'Configuring Passwordstate for First Time Use'**
3. If you have just installed build **8995**, please grant the built in IIS account called '**Network Service**' **Modify** access to the '**c:\inetpub\passwordstate**'. Ensure these new permissions also apply to all nested files and folders. If you have installed a version later than **8995**, you can disregard this step completely.
4. Next, configure your **HTTPS IIS Binding** to match your existing web server if you intend on keeping the same URL. Otherwise set your HTTPS binding to any new value you like.
5. Assign your **SSL Certificate** to your HTTPS binding that you previously prepared on your new server. If you intend on using a new certificate, more information about SSL Certificates can be found here: <https://forums.clickstudios.com.au/topic/2978-passwordstate-certificates-explained/>
7. You will most likely be using an existing **CNAME DNS** entry to resolve the URL to your old web server. You will now need to edit the DNS entry and update the host name of your new web server, or you can create a new CNAME entry pointing to the new server.
8. Next you need to copy the entire **ConnectionStrings** and **AppSettings** sections from your existing installation's web.config file into the new website. (The default location for the web.config file is C:\inetpub\Passwordstate\web.config)

Example of these lines that need copying can be seen in screenshot below:

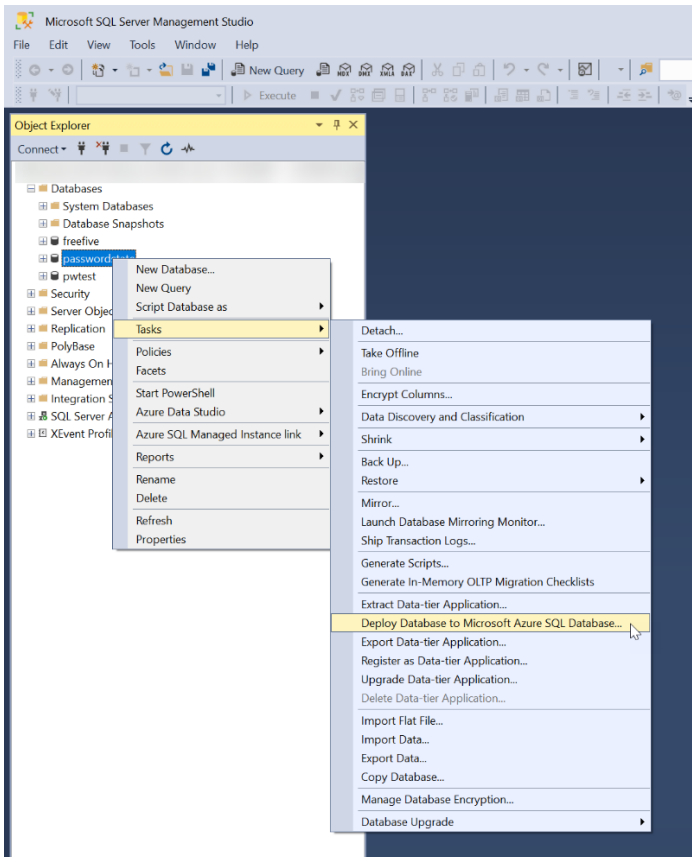
```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <configSections>
    <sectionGroup name="telerik.web.ui">
      <section name="radScheduler" type="Telerik.Web.UI.RadSchedulerConfigurationSection" allowDefinition="MachineToApplication" requirePermission="false" />
      <section name="radCompression" type="Telerik.Web.UI.RadCompressionConfigurationSection" allowDefinition="MachineToApplication" requirePermission="false" />
    </sectionGroup>
  </configSections>
  <connectionStrings>
    <add name="PasswordstateConnectionString" connectionString="Data Source=mydbserver.domain.net;Initial Catalog=passwordstate;User ID=passwordstate_user;Password=6ce7Wk25#tG" providerName="System.Data.SqlClient" />
  </connectionStrings>
  <appSettings>
    <add key="SetupStage" value="Setup Complete" />
    <add key="Secret1" value="e01-1-69f219caa408ca456012cf64fbbca85ab5ee4faa03fe670c62e19a4f503619cf6c3e14bbc74def512fe855de3c456f34306dbef0ebc7136234b6636663eff7" />
    <add key="Secret2" value="1c1-1-6f293f35f06180951f398db96207df5ba80640fc93ef6a97f955ef3a9038845cb4d4ac574899f2460fc7a1f7f1e17a77561c39295b6759df086e744ee4269c17195" />
  </appSettings>
  <system.web>
    <customErrors mode="Off" defaultRedirect="/error/generalerror.aspx" />
    <xhtmlConformance mode="Strict" />
    <webServices>
      <protocols>
        <add name="HttpPost" />
      </protocols>
    </webServices>
  </system.web>
</configuration>
```

****Note**** If you have encrypted your database Connection string, or AppSettings section of the web.config file, you will need to decrypt these prior to moving this file. Information about what an encrypted web.config file looks like, and how to decrypt it can be found in this forum post: <https://forums.clickstudios.com.au/topic/2699-encrypting-and-decrypting-the-webconfig-file/>

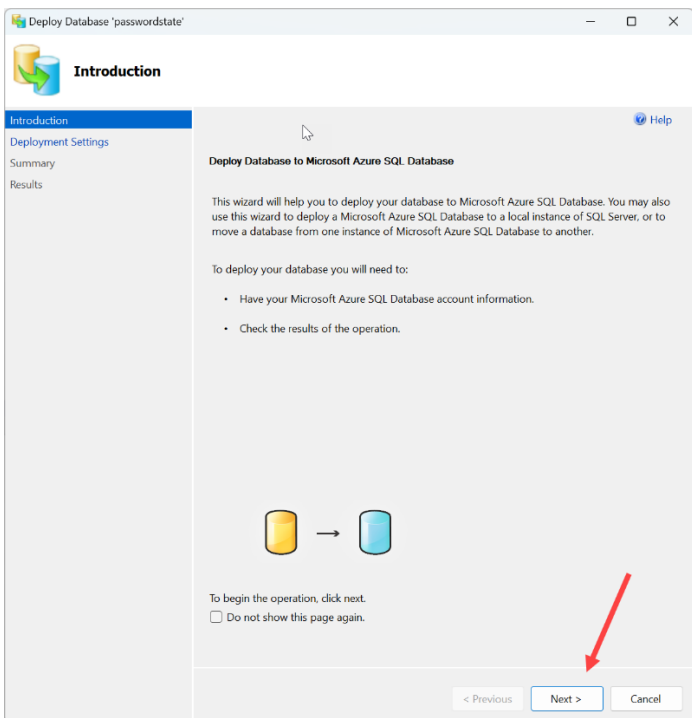
9. At this point, **do not log** into this new website, instead follow the next section of this guide which will advise how to migrate your database.

4 Migrating the Database

1. Open **SSMS** (SQL Server Management Studio) and connect to your existing on-premise database server
2. Right click your Passwordstate database and select **Deploy Database to Microsoft Azure SQL Database...**



3. Click Next on the **Introduction** page



4. Click **Connect**, and enter the details of your **Azure SQL Server** and the **Admin** credentials of the server. Now choose the **Edition** of database you wish to use and click **Next**

Deploy Database 'passwordstate'

Deployment Settings

Specify Target Connection

Specify the name of the instance of SQL Server or the Microsoft Azure SQL Database server that will host the deployed database, name the new database, and then click Connect to login to the target server.

Server connection: cspws [Connect...](#)

New database name: passwordstate

Microsoft Azure SQL Database settings

Edition of Microsoft Azure SQL Database: Standard

Maximum database size (GB): 250

Service Objective: S0

Other settings

Temporary file name: \\AppData\\Local\\Temp\\passwordstate-20230609124631.bacpac [Browse...](#)

< Previous Next > Cancel

5. When finished, click **Close**

Deploy Database 'passwordstate'

Results

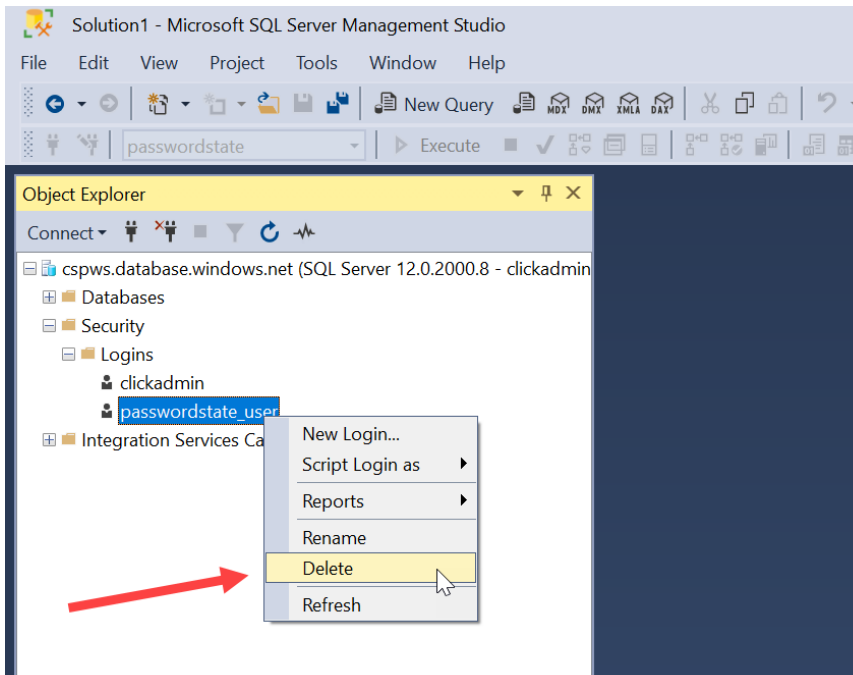
Operation Complete

Summary:

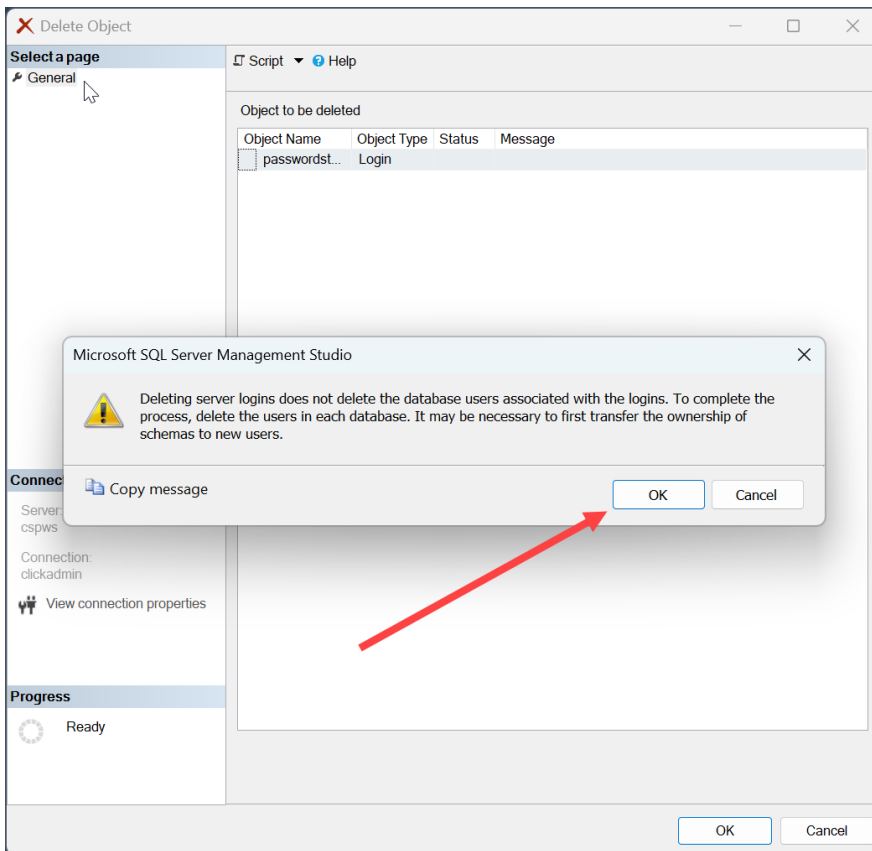
Name	Result
Exporting database	Success
Extracting schema	Success
Extracting schema from database	Success
Resolving references in schema model	Success
Validating schema model	Success
Validating schema model for data package	Success
Validating schema	Success
Exporting data from database	Success
Exporting data	Success
Processing Export	Success
Processing Table '[dbo].[Hosts]'	Success
Processing Table '[dbo].[PasswordStrengthPolicies]'	Success
Processing Table '[dbo].[PrivilegedAccounts]'	Success
Processing Table '[dbo].[PrivilegedAccountsACL]'	Success
Processing Table '[dbo].[PR_ADDDomains]'	Success
Processing Table '[dbo].[PR_PrivilegedAccounts]'	Success
Processing Table '[dbo].[HostsTreeViewState]'	Success
Processing Table '[dbo].[PR_UserAccounts]'	Success
Processing Table '[dbo].[HostsTreeViewExpandStatus]'	Success
Processing Table '[dbo].[HostsACL]'	Success
Processing Table '[dbo].[HostDocuments]'	Success
Processing Table '[dbo].[HostFolderDocuments]'	Success
Processing Table '[dbo].[HostTypes]'	Success
Processing Table '[dbo].[ImportErrors]'	Success
Processing Table '[dbo].[KeyRotationStatus]'	Success
Processing Table '[dbo].[Scripts]'	Success

< Previous Next > Close

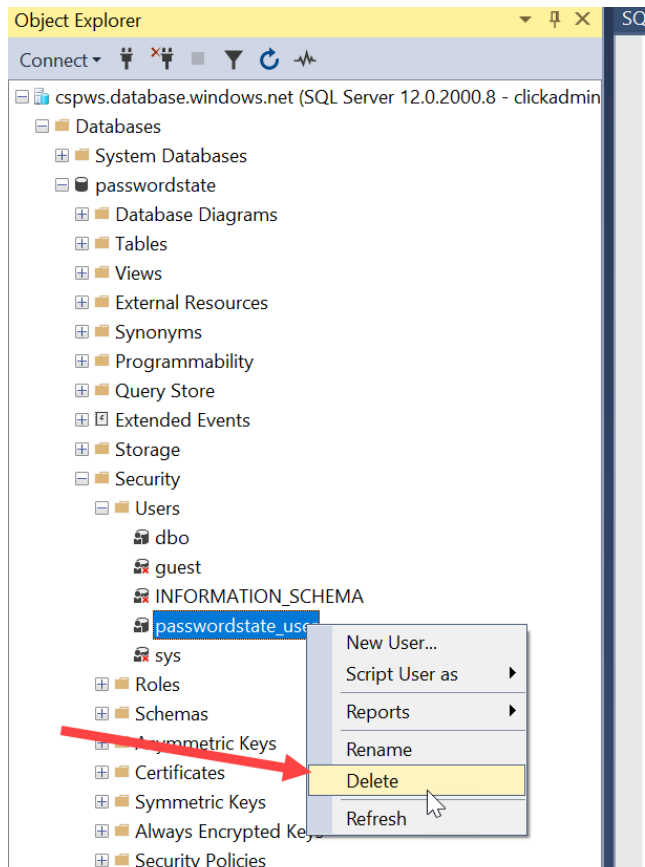
6. At this point, your database has been migrated up to Azure. The on-premise **passwordstate_user** SQL account has been migrated up to the Azure SQL database, so we will need to remove this and recreate it, and assign it the correct permissions. To do this, use **SSMS** and connect to your Azure SQL Server
7. Delete the **passwordstate_user** account from the **Security** → **Logins** folder



8. Click **OK** to the warning message:



9. Also delete the **passwordstate_user** from the Security → Users folder inside the Passwordstate database



10. Next, to recreate the user, run the following script and set a password of your choice:

```
USE master
CREATE LOGIN passwordstate_user WITH password='<choose a password>'
CREATE USER passwordstate_user FOR LOGIN passwordstate_user WITH DEFAULT_SCHEMA=[dbo]
```

11. Now you can assign the passwordstate user **db_owner** permissions by running this script:

```
USE passwordstate
CREATE USER passwordstate_user FOR LOGIN passwordstate_user WITH DEFAULT_SCHEMA=[dbo]
EXEC sp_addrolemember 'db_owner', 'passwordstate_user';
```

12. Now you need to make some final changes to your web.config file on your new webserver server. Open the **c:\inetpub\passwordstate\web.config** file and change the **Data Source** and **Password** in the connection string to be relevant to your new Azure SQL server. The password is what you set in **Step #10** above.



13. Restart the **Passwordstate Windows Service** on your new web server (after the Service has been restarted, it will recreate any custom Logos or images you have uploaded into Passwordstate - you may need to give it a minute or two to do this)
14. Now you can attempt to log into Passwordstate for the first time after this migration, and you will most likely be prompted to upgrade the database. This is normal, and you can continue at this point until the upgrade process finishes.
15. If you now log in you should see all your data as it was when on premises. Everything including passwords, users, settings and license keys. If you find the current build is **8995**, you will need to follow **Section 3** in this guide to upgrade to the latest build of **Version 9**:
https://www.clickstudios.com.au/downloads/version9/Upgrade_Instructions.pdf

At this point, your Passwordstate instance has now been moved to Azure and your migration is complete. We encourage you to do some user acceptance testing and when complete, you can safely decommission the old server.

This concludes the migration process, and if you have any queries about this, you are more than welcome to log a support call with Click Studio via our support page: <https://www.clickstudios.com.au/support.aspx>