Passwordstate

Enterprise Password Management

Installation Instructions

Table of Contents

1 System Requirements - General

Passwordstate has the following system requirements:

Web Server

Your web server which will host the Passwordstate web site can be any of the following Operating System versions, with required components:

- Microsoft Windows Server 2016, 2019, 2022
- Windows 11
- Internet Information Services (installed as a role in the Windows Operating System)
- Microsoft SQL Server 2012 Native Client (installed as part of Passwordstate installation)
- .NET Framework 4.7.2. or Higher
- PowerShell 5.0 or above
- Open JDK 21 or above (if using Browser based Remote Session Launcher)

Note: It is not recommended or supported to install Passwordstate on a Domain Controller WSUS Server, or SharePoint Server.

Database Server
You will need to have one of the following supported SQL Server versions installed prior to installing Passwordstate, so Passwordstate can connect to SQL Server and create a database. SQL Server can be installed either on the same web server as Passwordstate, or on any other Windows Server in your environment.

Any versions of SQL Server can be used i.e. Express, Standard, Enterprise.

- Microsoft SQL Server 2016
- Microsoft SQL Server 2017
- Microsoft SQL Server 2019
- Microsoft SQL Server 2022

Note: If you would like to use the High Availability module of Passwordstate, your distribution and publication databases must reside on SQL Server Standard or above – SQL Express can only act as a subscriber to SQL Server replication.

Important: SQL Server must be configured for mixed-mode authentication, so the Passwordstate web site can connect to SQL Server using an SQL Account. **Active Directory Accounts cannot be used to authenticate against the database.**

If you are unsure of how to install SQL Server, the Passwordstate.zip file contains some instructions for installing SQL Server 2019 Express edition.

Email Server
If you would like to receive emails generated from Passwordstate, you must also have an email server which is capable of sending anonymous SMTP emails, or emails from an authenticated mailbox

2 What Information is required for the Initial Setup

Prior to installing Passwordstate and running through the initial Setup Wizard, you will require the following information:

Let Passwordstate Create its Own Database

- A local SQL Account (not an Active Directory account) with sufficient permissions to create the database – at a minimum the '**dbcreator**' and '**securityadmin**' SQL Server roles are required (The '**sa**' account has these privileges, although some DBA's do not like to use this account due to its elevated privileges. Please note these account credentials are never used again, and not stored anywhere post installation of Passwordstate).

  During the initial setup, the following will occur:

  a. The Passwordstate database will be created and populated with some base data
  b. A SQL Account called '**passwordstate_user**' will be created, and will be given **db_owner** rights to the Passwordstate database only

Create Your Own Database, and Let Passwordstate Connect to it

- You will need to have created the empty database, and an SQL Account for Passwordstate to connect to this empty database. The SQL Account requires **db_owner** rights to the Passwordstate database only. It's recommended to call this database "**passwordstate**" and the SQL account should be called "**passwordstate_user**" to ensure easier troubleshooting at a later date, if required)

Additional Setup Information (these options below can be configured after you have installed Passwordstate, or during the initial set up wizard)

- Your **Registration Key** details for Passwordstate
- **Host Name** and **Port Number** of an **email server** capable of sending anonymous SMTP mail, or from an authenticated mailbox
- **SMTP Address** from which Passwordstate will send the emails from
- **Proxy Server Details** – Passwordstate can periodically check for the updates, and if your organization requires all internet access to go through a proxy server, you will need to specify the proxy host name and port number during the installation (this feature can also be disabled once you're using Passwordstate if required).

3 SQL Server Express, and SQL Port Number Considerations

If you intend to use SQL Server Express to host your Passwordstate database, please consider the following before installing Passwordstate:

1. If you're using SQL Server Express on a different server to where you installed Passwordstate, you may need to check if the TCP/IP Protocol is enabled (use **SQL Server Configuration Manager** -> **SQL Server Network Configuration**), and also the Windows Service '**SQL Server Browser**' is set to '**Automatic**' Startup Type, and has been started.
2. By default, SQL Server Express installs with an 'instance' name of **SQLExpress**. When you're configuring Passwordstate for first time use, specifically the '**Database Settings**' page, please ensure you have specified the name of the instance correctly i.e. fill out both the Database Server Name and Instance Name fields
3. If you intend to also install the High Availability instance of Passwordstate, SQL Server Express can only be used as the Subscriber for data replication, not the Publisher or Distribution database, which means you cannot use it for your primary Passwordstate server.

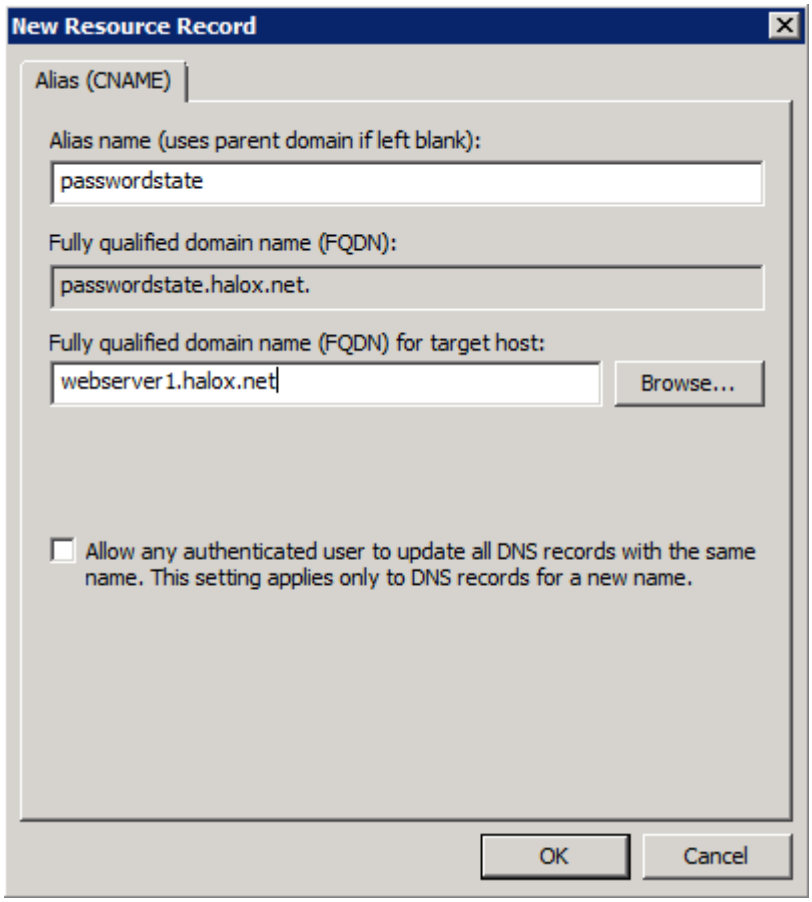
If you are running SQL Server on a non-standard port number, you will need to append the port number to the end of the Database Server Name during '**9. Configuring Passwordstate for First Time Use**' in the following way: ServerHostName,PortNumber i.e. **sqlserver1,8484**

4 Creating an Appropriate DNS Record

During the installation of Passwordstate, you have the option of using a URL which has the host name of the web server in it, or you can specify your own custom URL e.g. https://passwordstate

If you want to use your own custom URL, you will need to create a CNAME DNS entry as per the following instructions (please do not use host files for name resolution, as they do not work with Windows Authentication in IIS):

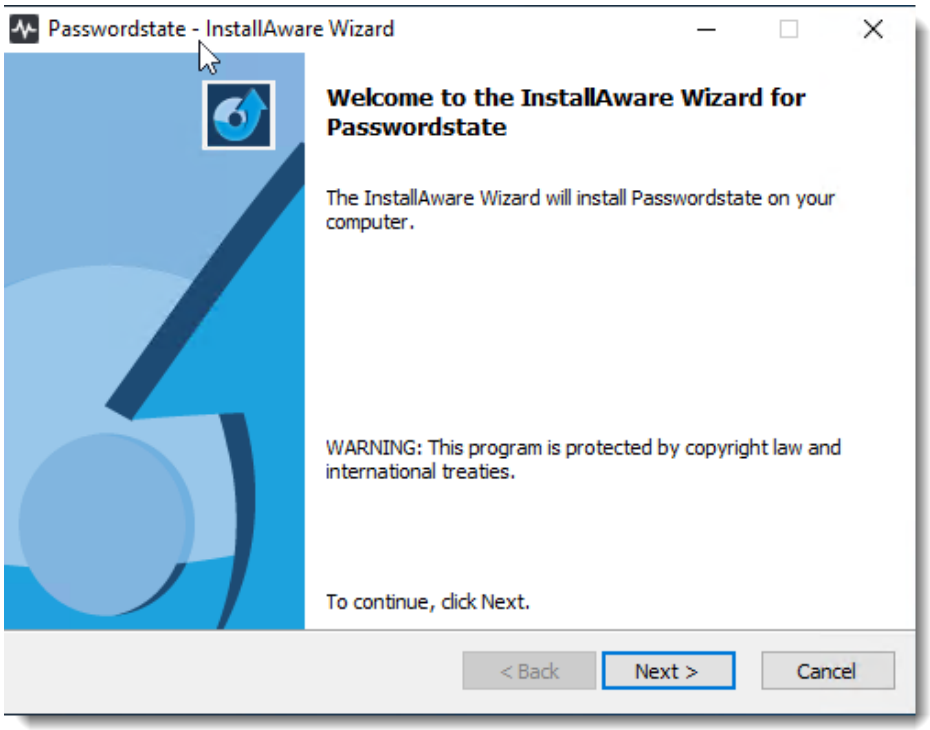
1. On your server hosting DNS, start '**DNS Manager**'
2. Right click on the appropriate domain, and select '**New Alias (CNAME)**'
3. As per the following screenshot, specify the name of your web server host name in the '**Fully qualified domain name (FQDN) for target host**' text box, then click on the '**OK**' button



5 Installing Passwordstate

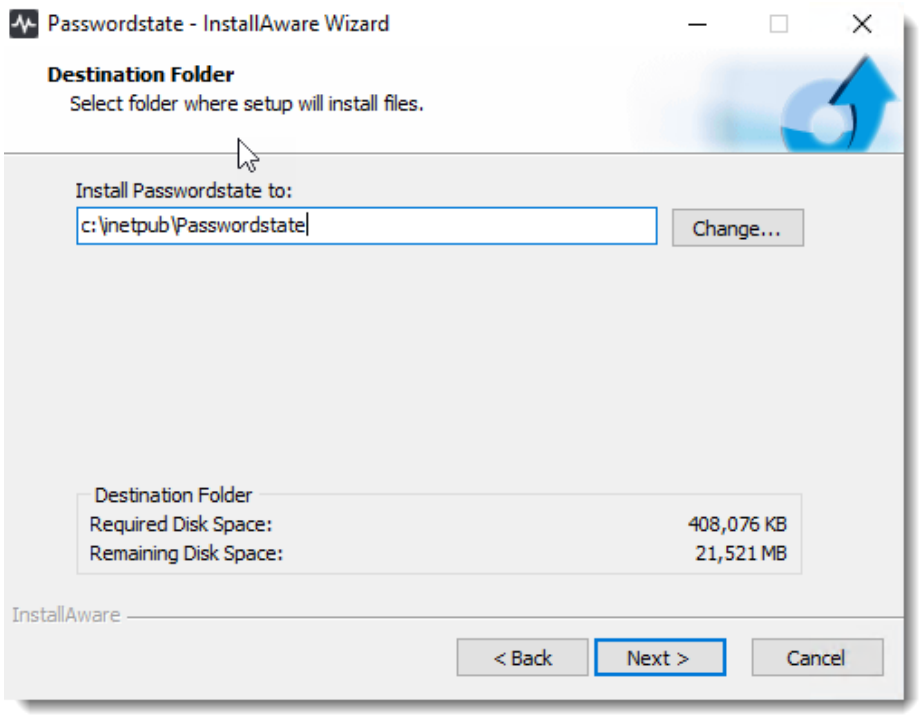
To install Passwordstate, run '**Passwordstate.exe**' and follow these instructions:

1. At the '**Passwordstate Installation Wizard**' screen, click on the '**Next**' button

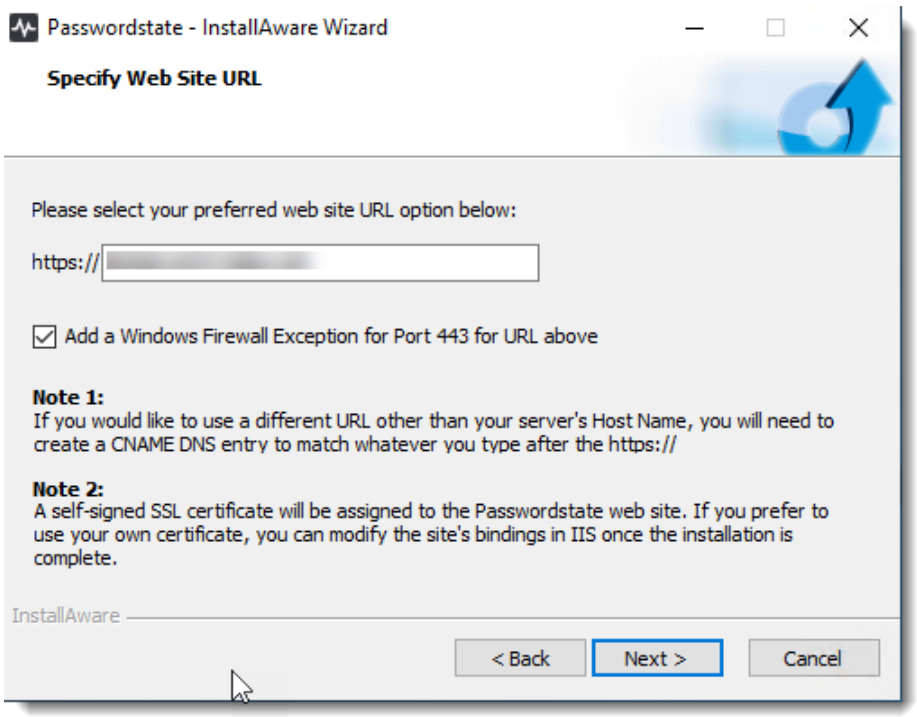


2. At the '**License Agreement**' screen, tick the option '**I accept the terms in the License Agreement**', then click on the '**Next**' button

3. At the '**Destination Folder**' screen, you can either accept the default path or change to a different location, then click on the '**Next**' button



4. At the '**Specify Web Site URL**' screen, specify the URL you would like to use, then click on the '**Next**' button

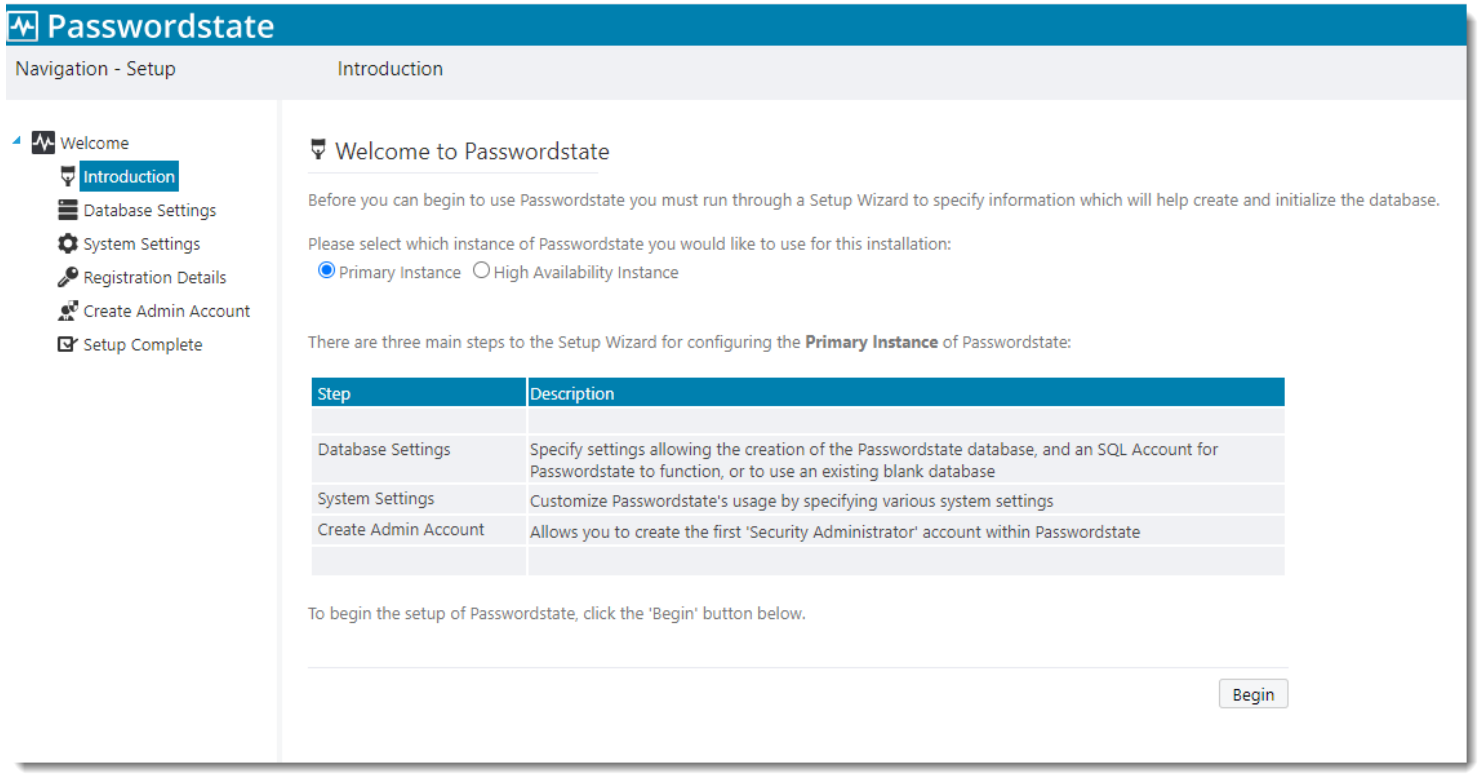


5. At the '**Completing the InstallAware Wizard for Passwordstate**' screen, click on the '**Next**' button

6. Once installed, click on the '**Finish**' button
7. If you have a Firewall enabled on your web server, you may need to open up the port number you specified during the install (default is 443), so that users are able to access the web site

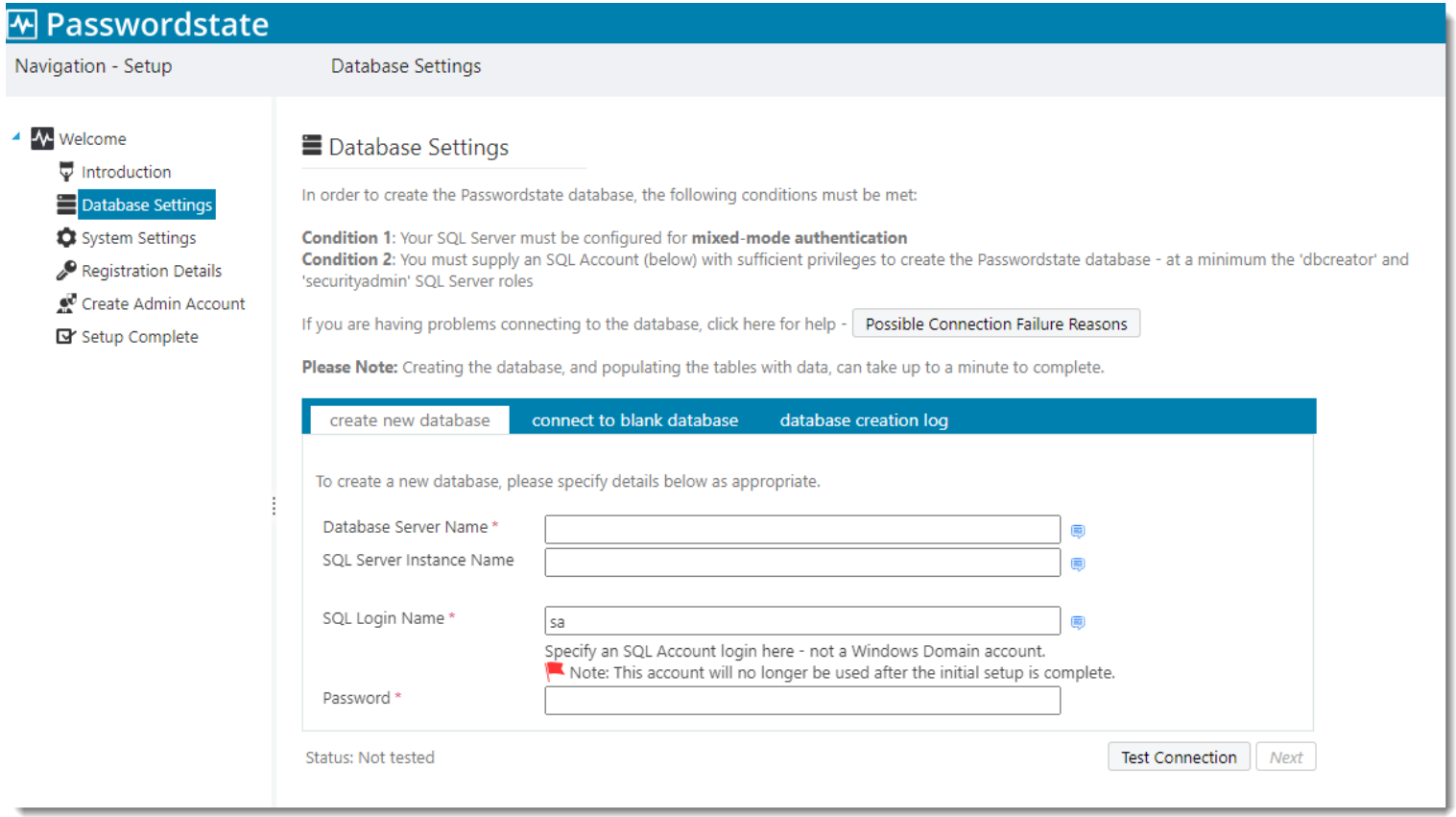
6 Configuring Passwordstate for First Time Use

Introduction - Now that Passwordstate is installed, you can direct your browser to the URL you specified during the initial install, and follow the initial Setup Wizard – this wizard will guide you through a series of questions for configuring Passwordstate for use.



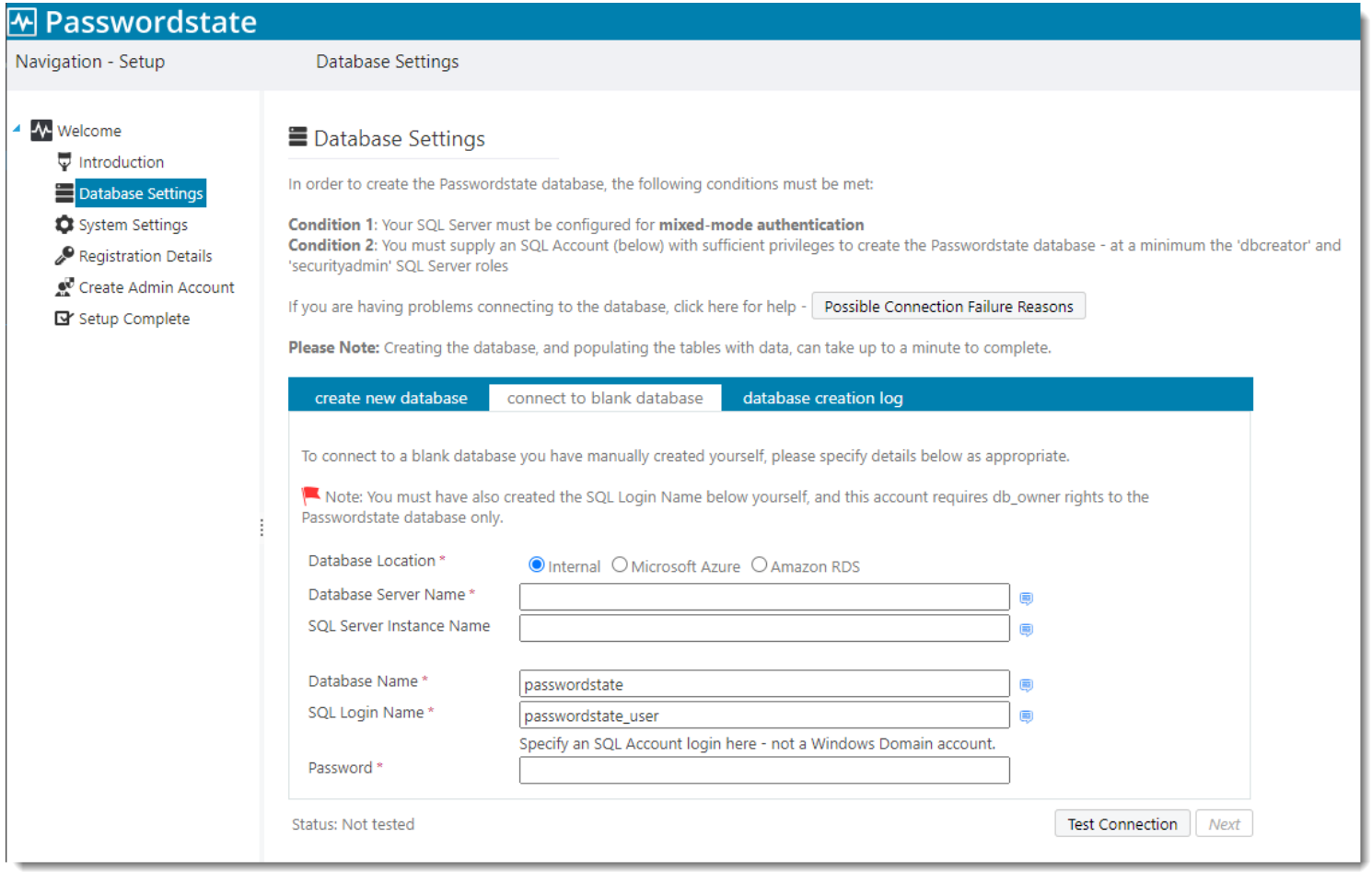
Database Settings – Create New Database - On this screen you will need to specify database server settings to allow Passwordstate to create the database for you. Please check the '**Database Creation Log**' if you have any issues connecting to the database.

Please Note: Creating the database, and populating the tables with data, could take up to a minute to complete. The account you use on this screen is not stored anywhere and is never used again by Passwordstate.

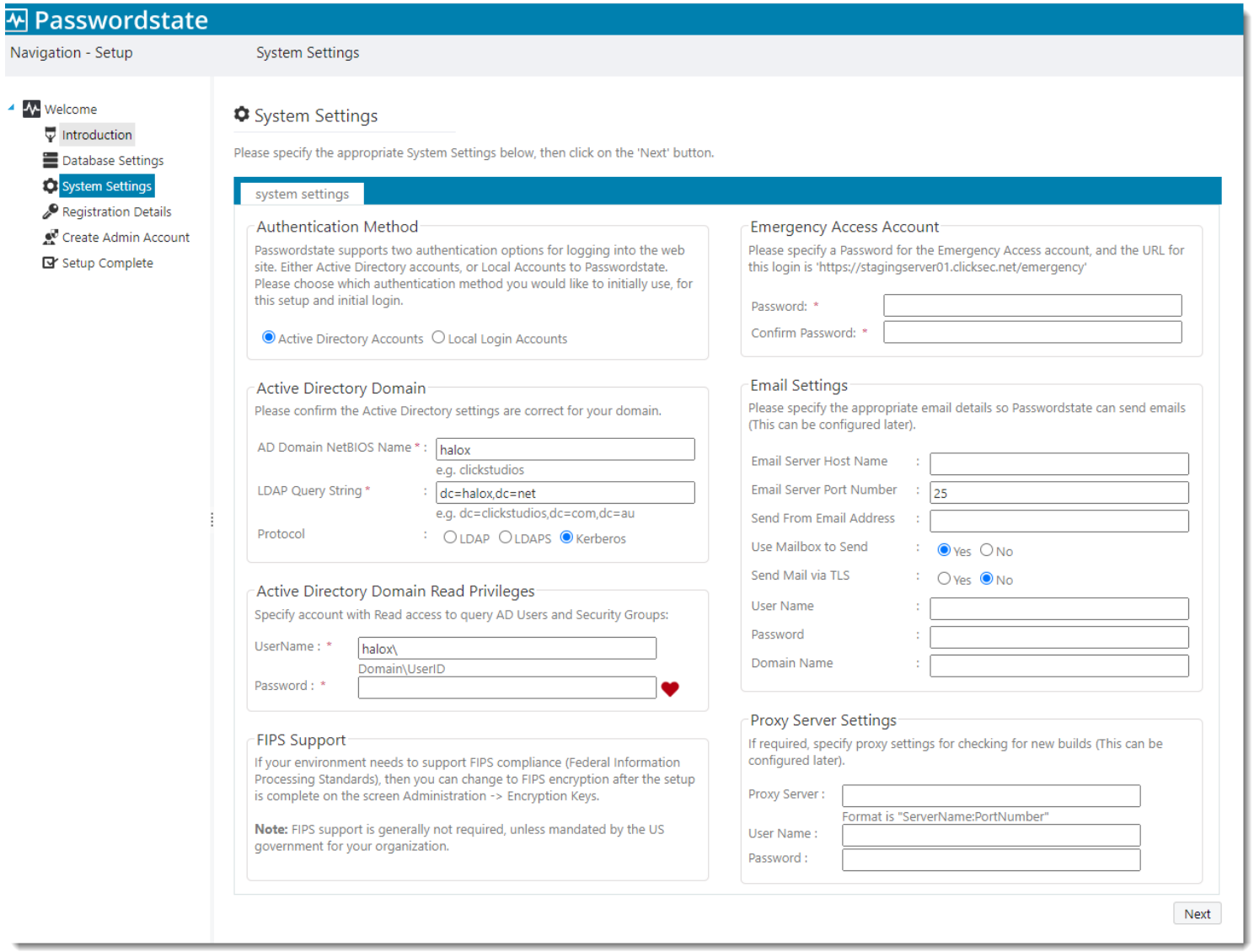


Database Settings – Connect to Blank Database – If you prefer to create the blank Passwordstate database yourself prior to tables being created and populated with data, you can do so by clicking on the '**Connect to Blank Database**' tab first.

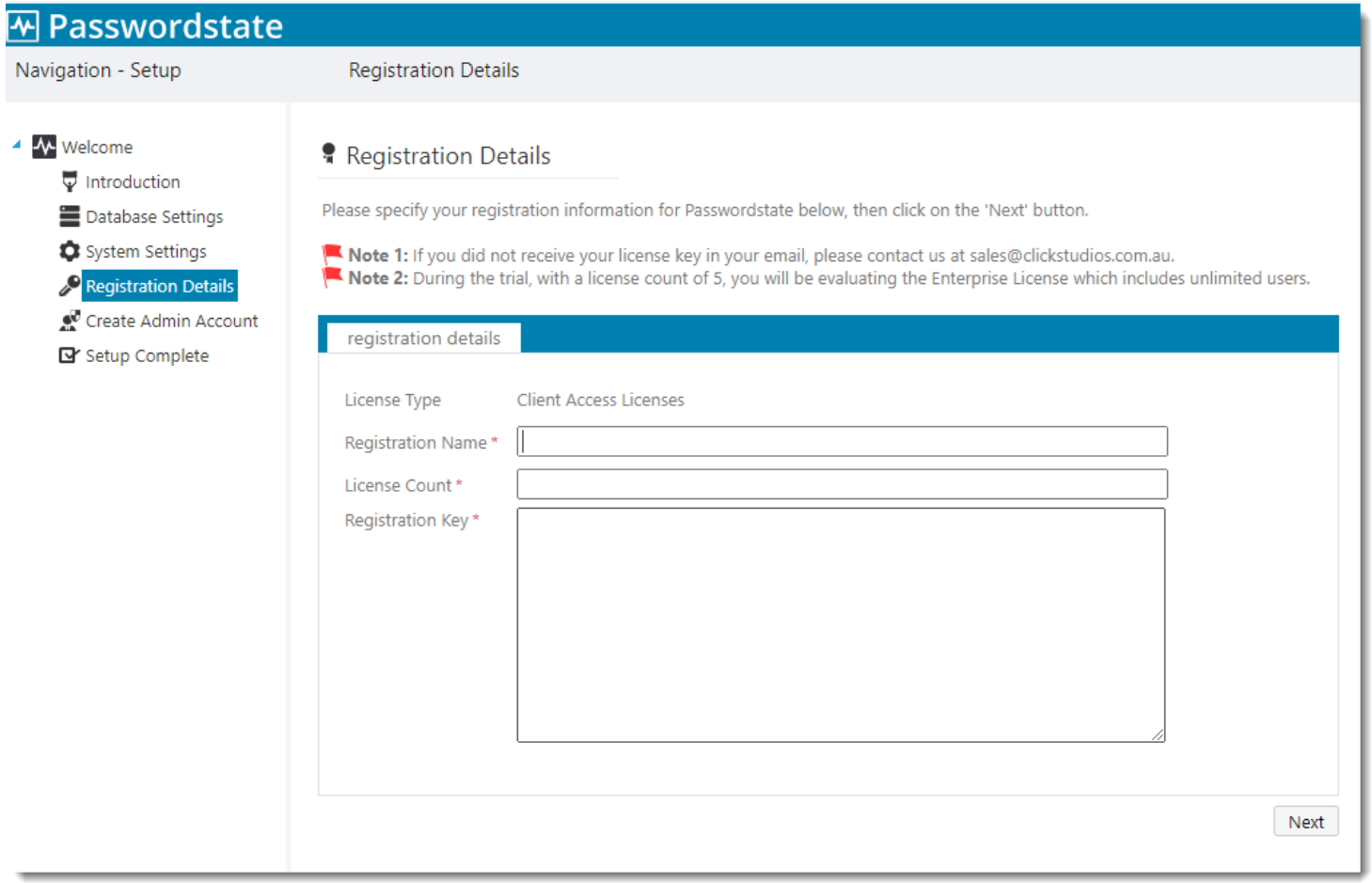
Please Note: You must first create a blank database to connect to, and an appropriate SQL Account which has **db_owner** rights to this database. If connecting to a Microsoft Azure or Amazon AWS database, please refer to their documentation for how to create the database and SQL Account.



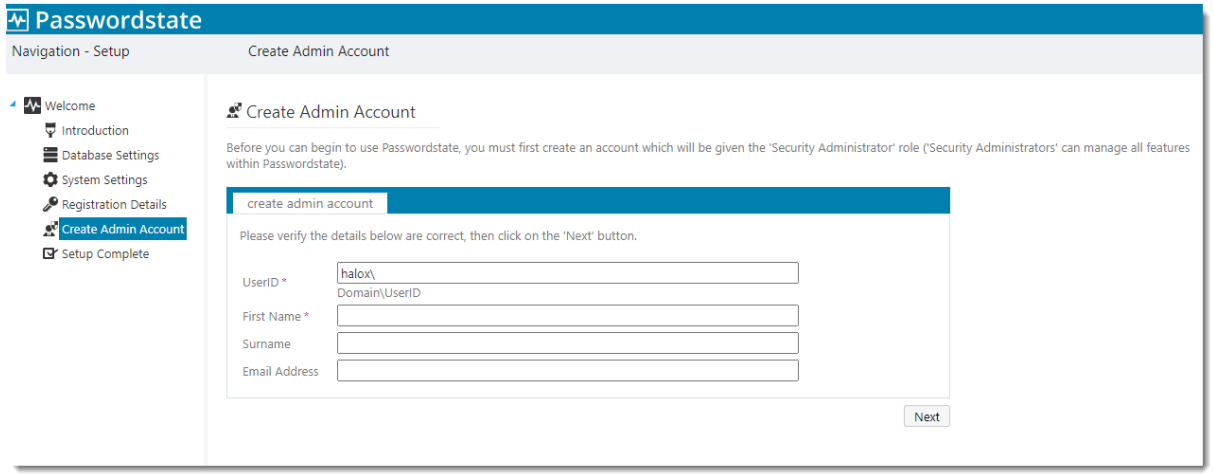
System Settings – On this screen you specify various system wide settings for Passwordstate usage. Please take note of fields with a red * as these are required fields. Please take note of the **Emergency password**, as this can be used to access your system in the event you've accidently locked yourself out. This can be changed at any stage after you have access to your Passwordstate website. Click Studios support can help you recover this as well if required.



Registration Details – On this screen you specify the license key details, which were emailed to you when registering for the download on Click Studios' web site.

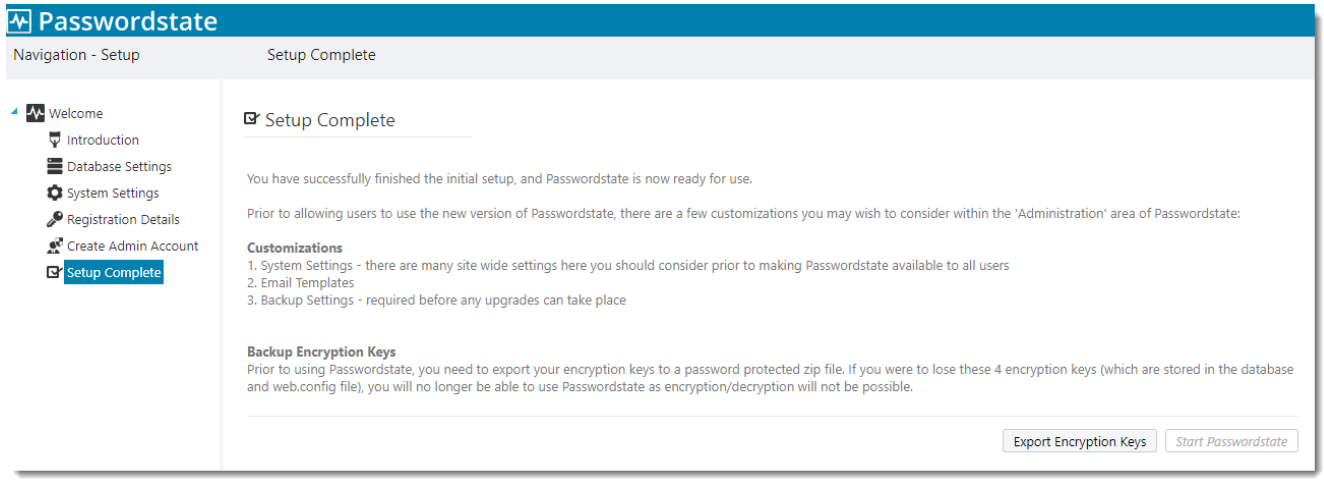


Create Admin Account – On this screen you specify details for the first user account to be created in Passwordstate. This account will be granted Security Administrator privileges, and assign all of the Security Administrator roles which means it has full control over the system.



Setup Complete – The installation is now complete and you can begin using Passwordstate. Prior to granting access, or informing users of the new version, you may wish to review some of the system wide settings found under the '**Administration**' area of Passwordstate.

Export Encryption Keys – It is very important you export your encryption keys for safe storage outside of Passwordstate. If you were to lose your web.config file in a disaster, Click Studios would not be able to help you rebuild your Passwordstate environment. The split encryptions keys are stored in the web.config file, and within the database.



7 Encrypting the Database Connection String in the Web.config file

It is recommended you encrypt the database connection string within the web.config file, so the SQL Account credentials used to access the Passwordstate database is unreadable from anyone who can read the file system on your web server.

To encrypt the database connections string, please follow these instructions:

Encrypt Connection String

- Open a command prompt (as Administrator) and type **CD C:\Windows\Microsoft.NET\Framework64\v4.0.30319**
- Type the following:

  aspnet_regiis.exe -pef "connectionStrings" "c:\inetpub\passwordstate" (change the path if you've installed Passwordstate to a different location)
- Then restart the Passwordstate Windows Service

Decrypt Connection String

- Open a command prompt (as Administrator) and type **CD C:\Windows\Microsoft.NET\Framework64\v4.0.30319**
- Type the following:

  aspnet_regiis.exe -pdf "connectionStrings" "c:\inetpub\passwordstate" (change the path if you've installed Passwordstate to a different location)
- Then restart the Passwordstate Windows Service

Note 1: If you intend to rename your server host name, or move your Passwordstate install to a different server, you should decrypt these settings first.

Note 2: If you do not wish to use an SQL Account to connect to your database server, please refer to the section below in this document titled '**Configure Passwordstate to use a Managed Service Account (MSA) to connect to the database**'.

8 Encrypting the appSettings Section within the Web.config file

It is recommended you encrypt the appSettings section within the web.config file, as this section of the file stores half of your split encryption keys.

To encrypt the appSettings section, please follow these instructions:

Encrypt appSettings Section

- Open a command prompt (as Administrator) and type **CD C:\Windows\Microsoft.NET\Framework64\v4.0.30319**
- Type the following:

  aspnet_regiis.exe -pef "appSettings" "c:\inetpub\passwordstate" (change the path if you've installed Passwordstate to a different location)
- Then restart the Passwordstate Windows Service

Decrypt appSettings Section

- Open a command prompt (as Administrator) and type **CD C:\Windows\Microsoft.NET\Framework64\v4.0.30319**
- Type the following:

  aspnet_regiis.exe -pdf "appSettings" "c:\inetpub\passwordstate" (change the path if you've installed Passwordstate to a different location)
- Then restart the Passwordstate Windows Service

Note 1: If you encrypt the AppSettings section of your web.config file, it is imperative you keep an exported copy of your encryption keys in a safe place, as they may be required in the event of a server rebuild, or server move. You can export your encryption keys to a password protected zip file under **Administration** -> **Encryption Keys** once you have access to your website.

Note 2: If you intend to rename your server host name, or move your Passwordstate install to a different server, you should decrypt these settings first.

9 SSL Certificate Considerations

By default, Passwordstate comes installed with a Self-Signed Certificate as this is the only type of certificate **Click Studios** can supply during the install.

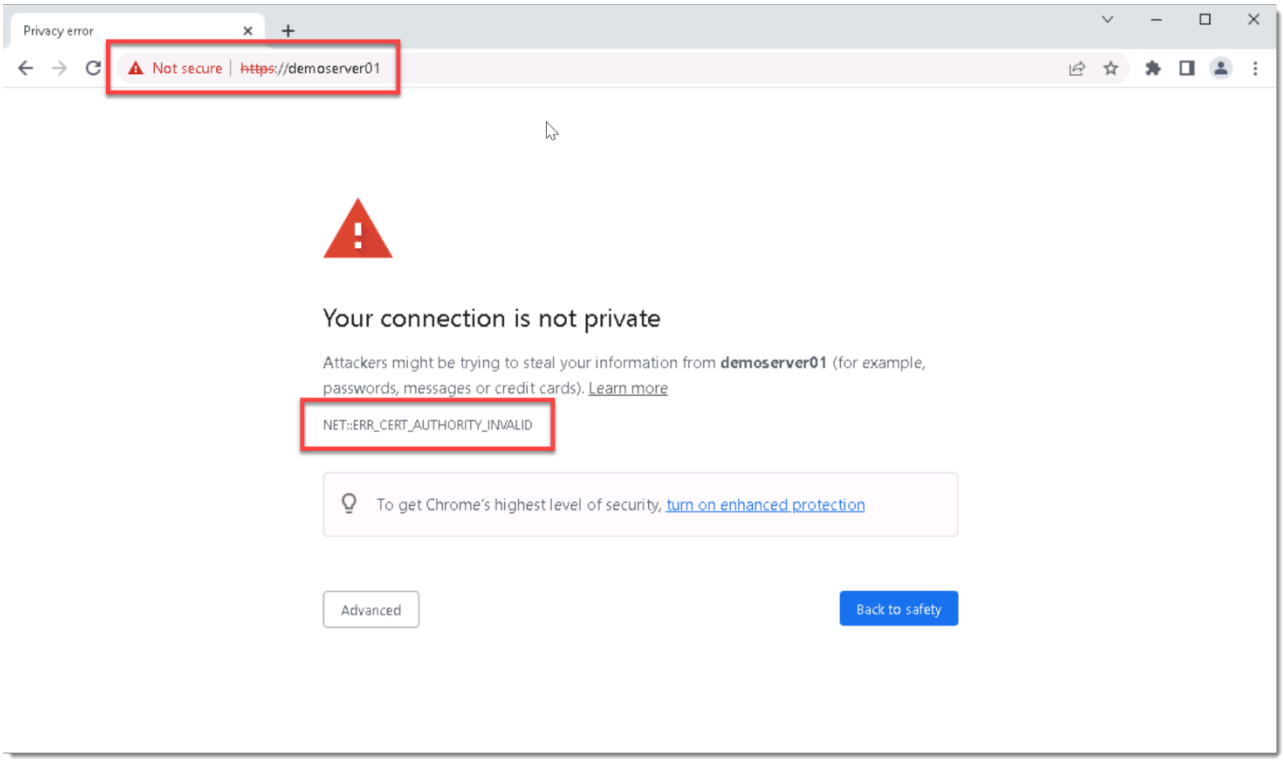
Typically, you would change your certificate to something more secure such as one issued from your **Internal Certificate Authority**, or a purchased one from an **online provider**.

If you have your own SSL certificate installed on the web server you'd prefer to use, you can modify the bindings for the site in IIS, and select the appropriate certificate.

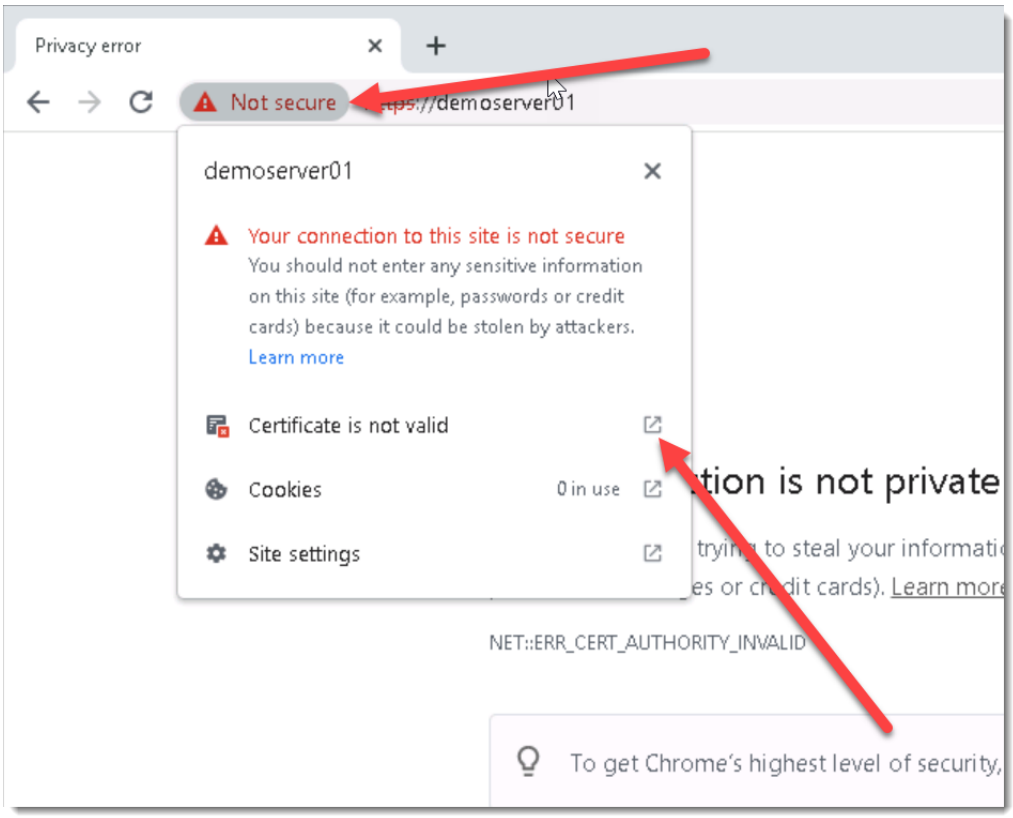
If you wish to continue using the self-signed SSL certificate, then you may want to instruct your users to "Install" the certificate on their computer, so the various Internet browsers don't complain about the certificate not being issued by a trusted authority.

To install the certificate, you can follow these steps below. You will effectively be exporting the certificate and reimporting it into your Trusted Root store on your machine. The example below uses Google Chrome as the browser, but you can achieve the same thing in other browsers.

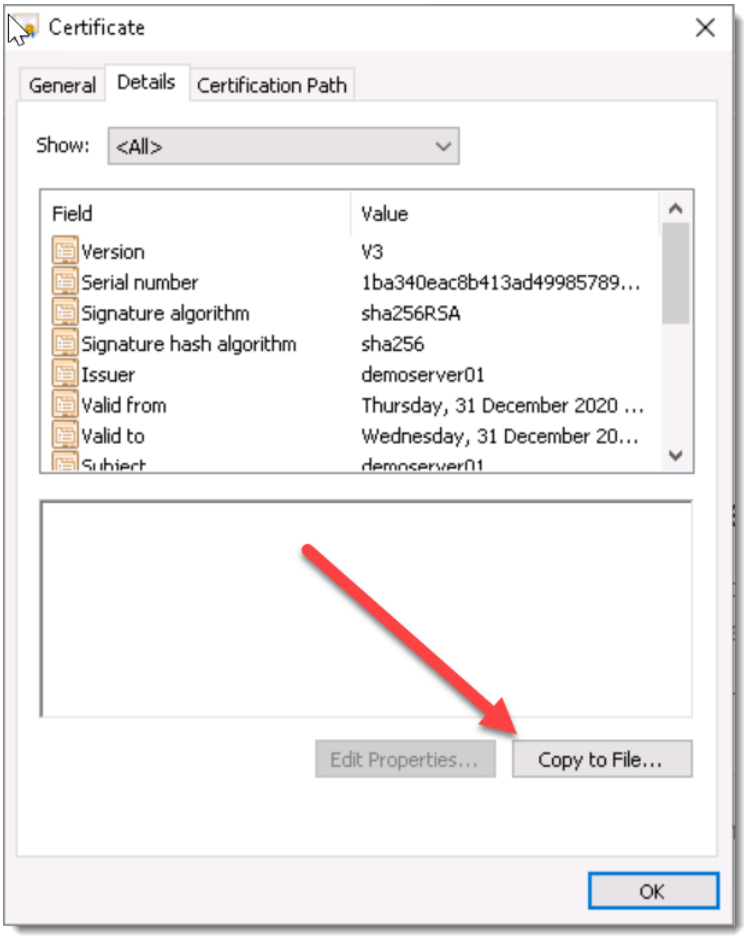
Using **Chrome**, browse to your Passwordstate web site and you should see a screenshot with an error saying **NET:ERR_CERT_AUTHORITY_INVALID**



Click on the **Not Secure** button, and then click on the **Show Certificate** button:



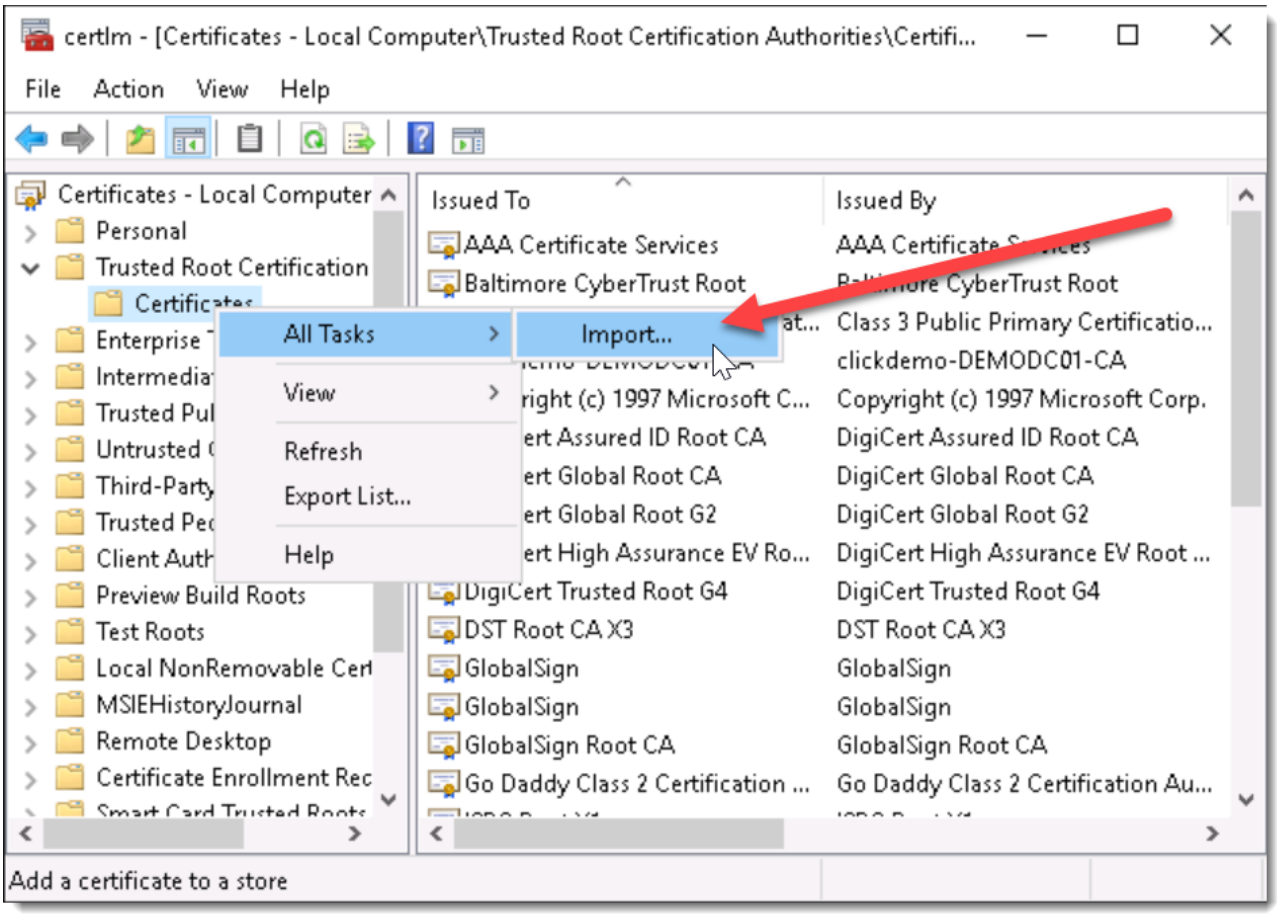
Under the **Details** tab, click **Copy to File…**



Run through the **Certificate Export Wizard**, leaving all default options. During the process, set a name for your certificate (which can be anything), and save it to disk.

Now go to **Start** -> **Run** and type in **certlm.msc** and hit enter. This opens up the **Local Computer Certificate Store** on your computer.

Expand out **Trusted Root Certificate Authorities** and right click **Certificates**, and choose **All Tasks** -> **Import**:



Now run through the import process, using all default options, and browse to the certificate you saved disk in the step above. Once this completes, you will see a **Successful Import** message.

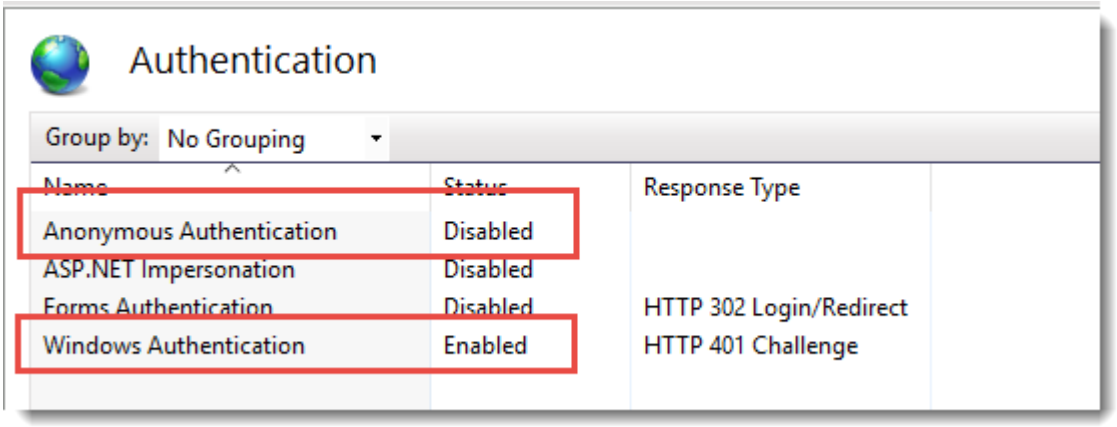
You can now restart your browser and try browsing to your Passwordstate URL again, and you will no longer see the Browser Warning about the certificate.

Note 1: For an in-depth explanation of the different types of certificates you can use on your Passwordstate website, please see this forum post: https://www.clickstudios.com.au/community/index.php?/topic/2978-passwordstate-certificates-explained/

10 Single Sign-On with Active Directory Accounts

If you choose the '**Active Directory Integrated**' version of Passwordstate, it is also possible to configure Internet Information Services to allow single sign on for authentication.

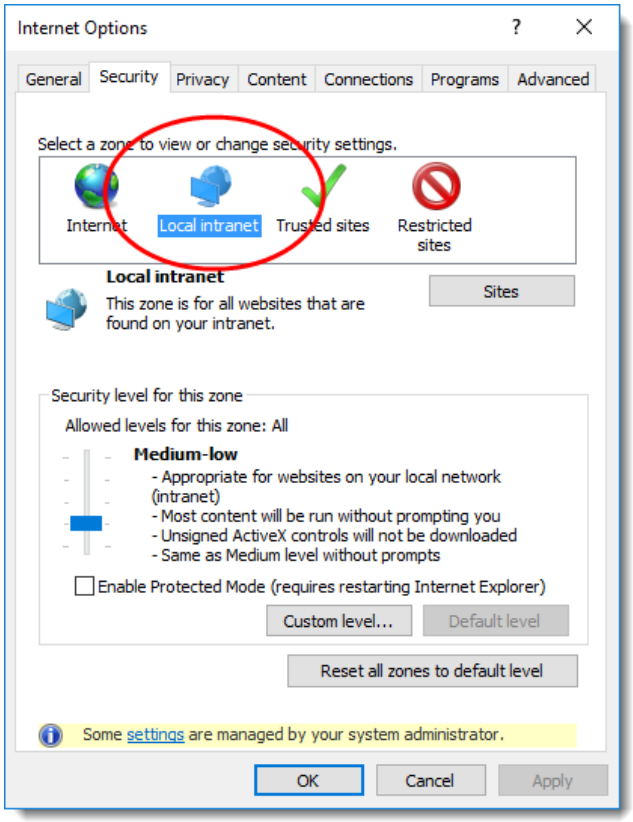
To do this, simply disable '**Anonymous Authentication**' for the site in IIS, like in the screenshot below. Please note that when doing this, it may cause issues authentication from Mac and Linux desktops.



If after enabling single-sign on your browser prompts you for your domain credentials, then below is the most common cause of this.

If using Internet Explorer or Chrome, check the Passwordstate web site is being detected in the **'Local Intranet'** security zone in Internet Explorer, and the option for '**User Authentication**' is set to **'Automatic logon only in Intranet zone**'.

You may need to add the URL of the site to a group policy which forces Internet Explorer to detect the site is in the intranet zone. Alternatively, each user can add this manually in Internet Explorer via the **Internet Options -> Security Tab.** Below is a screenshot of this setting.



- If using Firefox, it does this by design. To fix this, you can install a Firefox extension called **˜Integrated Authentication for Firefox™.** Now in your Firefox browser click **Tools** -> **Integrated Authentication Sites** -> Enter in your passwordstate URL - ensure there is no backslash on the end - eg https://passwordstate.yourdomain.com

11 Configure Passwordstate to use a Managed Service Account (MSA) to connect to the database

As of Build 7301, it is possible to configure Passwordstate to use a Managed Service Account to communicate with the database server, instead of a SQL Login Account. Below are the following steps required in order to configure support for this.

Note 1: If you are wanting to do Password Resets and Account Heartbeat validations across non-trusted domains, then you cannot use a MSA account for database connectivity – the Application Pools in IIS and the Passwordstate Windows Service executes the PowerShell scripts for these features, initiating connections to the remote hosts. If there are no domain trusts in place, this will cause issues as the MSA account makes the connection to the remote host.

Create a Managed Service Account (MSA)

- On your domain controller, open PowerShell console as an Admin, and execute the following commands – Please note you will be prompted for a password, to which you can enter any password of your choice:

  New-ADServiceAccount -Name <MSAAccountName> -RestrictToSingleComputer -AccountPassword (Read-Host -AsSecureString) -Path "cn=<MyCN>,dc=<MyDC>,dc=<MyDC>" (replace the variables in <> as appropriate)

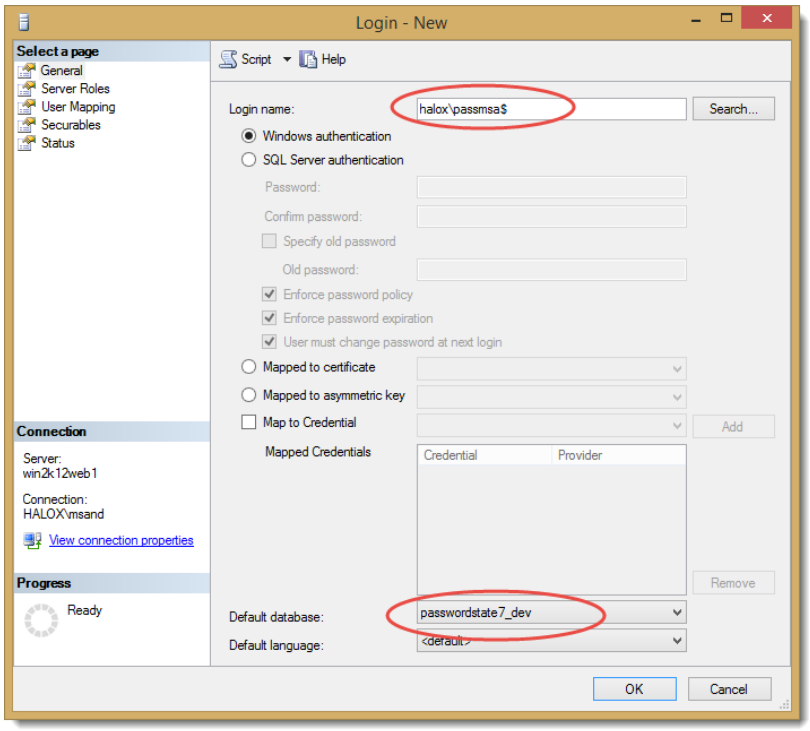
  Add-ADComputerServiceAccount -Identity "<MyWebServerName>" -ServiceAccount "<MSAAccountName> " (The Web Server Name is where the MSA Account will be used)

- On your Passwordstate Web Server, open PowerShell console as Admin, and execute the following commands:
  - **Add-WindowsFeature RSAT-AD-PowerShell** (this role may already be installed)
  - **Import-Module ActiveDirectory**
  - **Install-ADServiceAccount -Identity <MSAAccountName>**

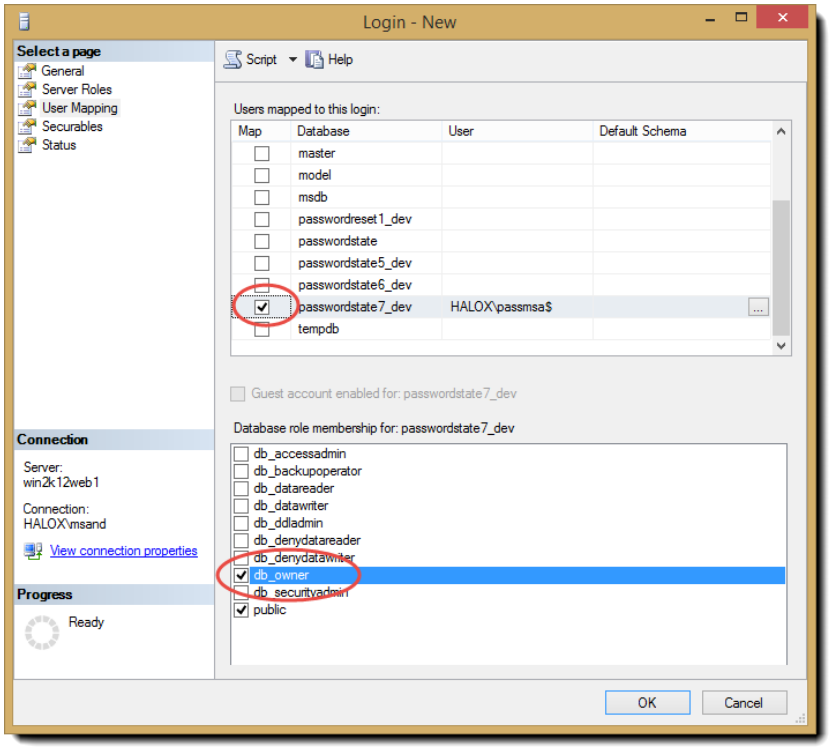
MSA Account and SQL Server

You now need to add a new Windows login within your SQL Server using SQL Management Studio Tools, and you can use the screenshots below as a guide – in our example, the MSA account is called **passmsa**, and whenever referencing an MSA account you must append the **$** symbol to the end.

1. Create the MSA Login Account

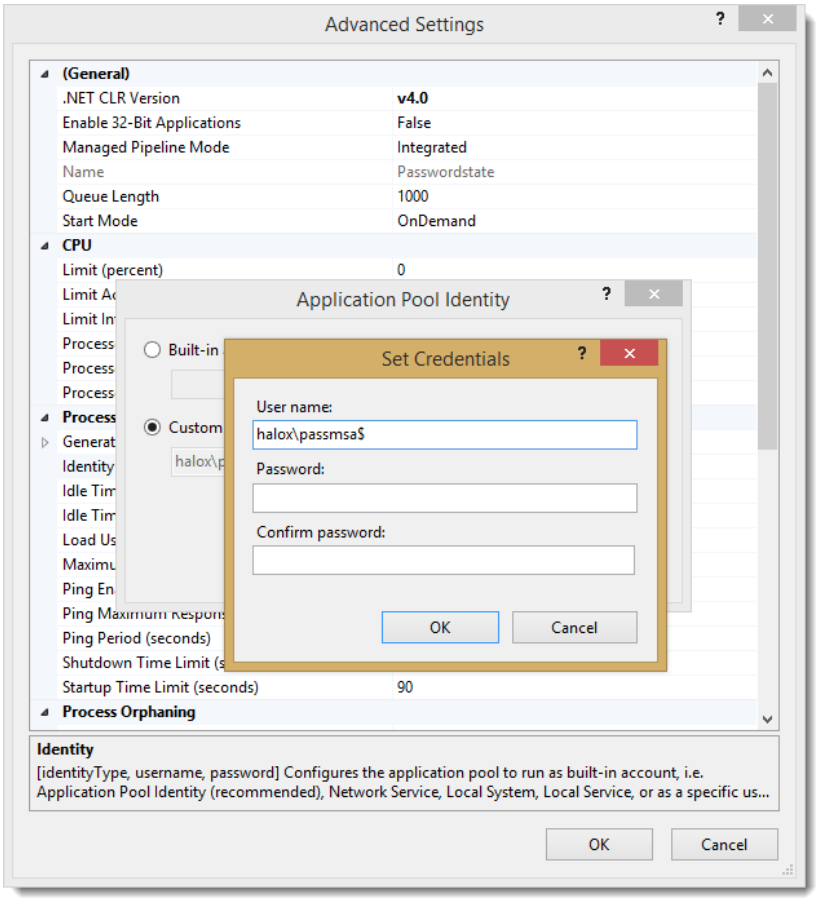


2. Grant the MSA Account db_owner rights to the Passwordstate database



Configure Passwordstate IIS Application Pools

You need to open Internet Information Services Manager, and modify the "**Identity**" for all of the Passwordstate Application Pools so it uses the MSA Account. When specifying the MSA Account to use, you leave the password fields blank, as per the screenshot below.



Modify the Passwordstate web.config file

- Open the web.config file in the root of the Passwordstate folder (open as Admin with notepad or equivalent)
- Change the line:

<add name="PasswordstateConnectionString" connectionString="Data Source=<ServerName>;Initial Catalog=passwordstate;User ID=passwordstate_user;Password=<MyPassword>" providerName="System.Data.SqlClient" />

to read like:

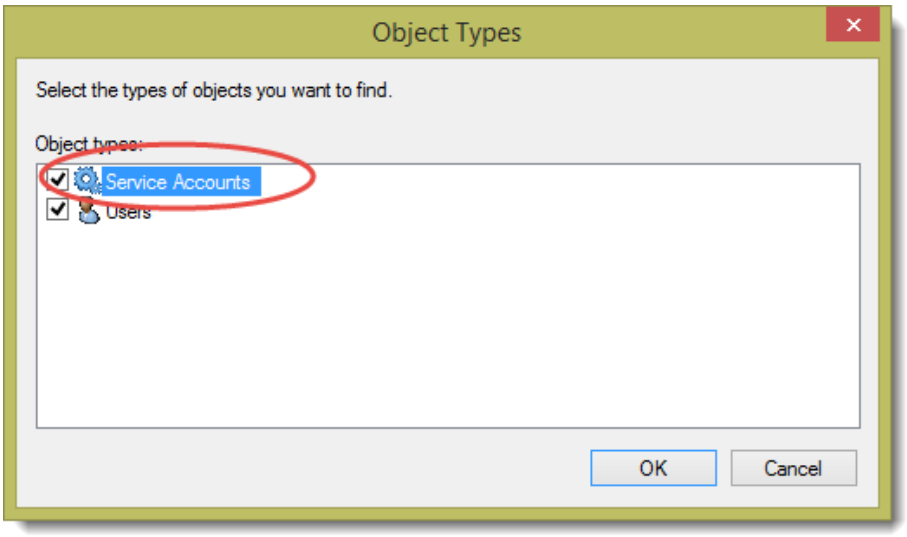
<add name="PasswordstateConnectionString" connectionString="Data Source=<ServerName>;Initial Catalog=passwordstate**;Integrated Security=SSPI;" providerName="System.Data.SqlClient"** />

Now save the file and exit notepad.

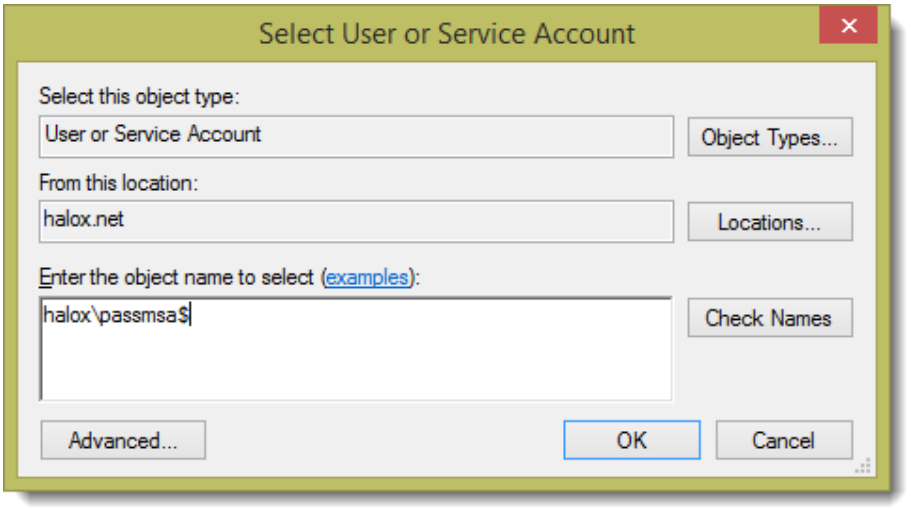
Configure Passwordstate Windows Service

We now need to change the '**Log On As**' property for the Passwordstate Windows Service to use the MSA Account.

When doing so, you may need to select the '**Service Accounts**' Object Type in order to find the account in Active Directory, as per the screenshot below:



And also leave the password for the account blank, just like the Application Pools.



Now restart the Passwordstate Windows Service.

File System NTFS Permissions

There are certain features where images and logos need to be written and read from the file system, requiring your MSA account to have access to do so. Please also apply **Modify** NTFS Permissions for the MSA Account to the Passwordstate folder, and all nested files/folders. (default folder path is **c:\inetpub\passwordstate**)

Encrypted Web.config Settings and MSA Account

If you have encrypted either of the database connection string or AppSettings section in the web.config file, and you are using an MSA account, you also need to apply permissions to the RSA Key Container for the MSA Account, so the account can decrypt these settings. Below are instructions for how to do this:

- Open a command prompt as Admin and type **CD C:\Windows\Microsoft.NET\Framework64\v4.0.30319**
- Type the following:
  - **aspnet_regiis.exe -pa "NetFrameworkConfigurationKey" "Domain\MSA-AccountName$"** (change the path if you've installed Passwordstate to a different location)
- The restart the Passwordstate Windows Service

Using the WinAPI with an MSA Account

Passwordstate has an API which uses identity of the script runner to authenticate. If you are using an MSA account on your Passwordstate website, the WinAPI will not work by default, and you'll need to use a SQL account to establish connections for the WinAPI only.

Steps to Configure the WINAPI to work with an MSA Account:

1. Ensure you have a SQL account that has **db_owner** permissions to your Passwordstate database.
2. Open the c:\inetpub\passwordstate\WinAPI**web.config** file as an Administrator and insert the following code. This code assumes your SQL account is called "**passwordstate_user**" and the password is "**Welcome01**"

```
<connectionStrings>
   <remove name="PasswordstateConnectionString" />
   <add name="PasswordstateConnectionString" connectionString="Data Source=webserver01;Initial Catalog=passwordstate;User ID=passwordstate_user;Password=Welcome01" providerName="System.Data.SqlClient" />
  </connectionStrings>
```

12 X-Forwarded-For Support

When Passwordstate adds auditing data to the database, it records the IP Address of the client who initiated an action which triggered the audit event.

As Passwordstate supports the “X-Forwarded-For (XFF) HTTP header field” for identifying the originating IP address of a client, and if you use any form of Load Balancing or Proxy Server caching, you may need to make configuration changes to your device/appliance. This will ensure the correct IP Address of the client is reported, instead of the load balancer or proxy server.

You can find the configuration screen for this in Passwordstate at **Administration** -> **System Settings** -> **Proxy & Syslog Servers**.

13 Troubleshooting Connectivity Issues

If when you first try and browse to the Passwordstate web site you get a blank page, or an error saying '**The page cannot be displayed because an internal server error has occurred**.', this may be caused by the order in which you installed Internet Information Services and the .Net Framework 4.7.2 – if you install the .NET Framework first, this error may occur.

Note: These instructions only apply to Microsoft Windows Server 2008 R2 and Windows 7

To resolve this, follow these instructions:

- Open a Command Prompt as an Administrator
- Type **CD C:\Windows\Microsoft.NET\Framework\v4.0.30319** or **C:\Windows\Microsoft.NET\Framework64\v4.0.30319** depending on our operating system version
- Now type **aspnet_regiis -i**
- After ASP.NET has been re-registered, ensure the Passwordstate Application Pool in IIS is set to 'Integrated Managed Pipeline Mode', and then restart IIS (you need to open the Internet Information Services (IIS) Manager tool to do this)
- Now open your browser and point it back to the Passwordstate web site

You may need to do this for both the 32bit and 64bit versions on the Framework directories above if you still experience issues.

Troubleshooting 'internal server error has occurred'

14 McAfee and other AV and Constant Logout Issues

McAfee's Anti-Virus On-Demand scan, and some other AV vendors, can cause issues with logging users out of Passwordstate prematurely, before the default IIS session time of 10 minutes.

The On-Demand scan process isn't blocking the accessing of any files, but when it scans either the web.config file, or any files in the /bin folder, it can cause sessions in IIS to end. We recommend temporarily excluding the Passwordstate folder from On-Demand scanning, as this has helped a lot of customers.

If you are seeing the same symptoms, but are using a different Anti-Virus suite, please also temporarily exclude the Passwordstate folder from real-time scanning, as well as the w3wp.exe process, which is Internet Information Services (IIS).

Please note this isn't a long-term fix and the responsibility for a resolution is with the AV Vendor.

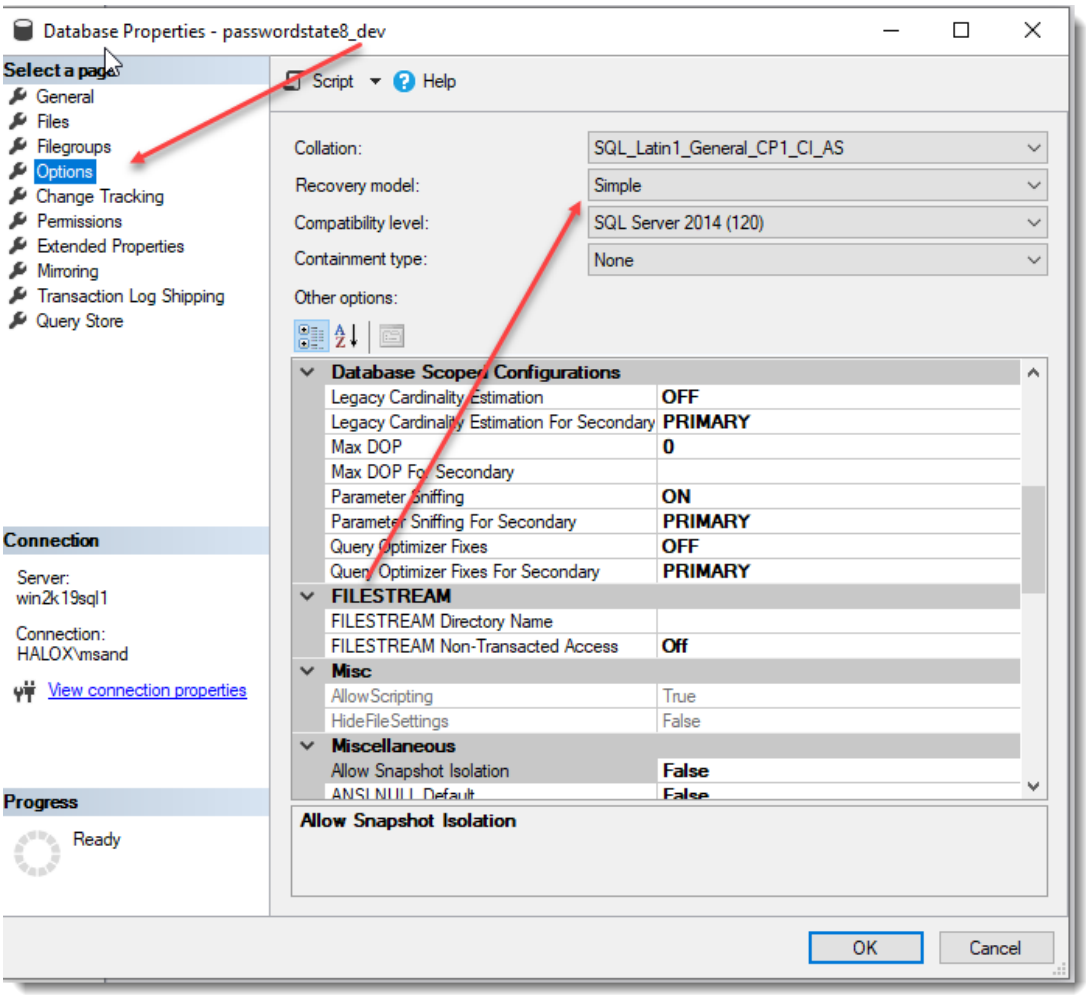
15 SQL Server Database Size Management

In version 9 of Passwordstate, when creating databases in SQL Server, the 'Recovery Model' for databases is set to **SIMPLE**. If you have installed a prior build, then the recovery mode will be **FULL**.

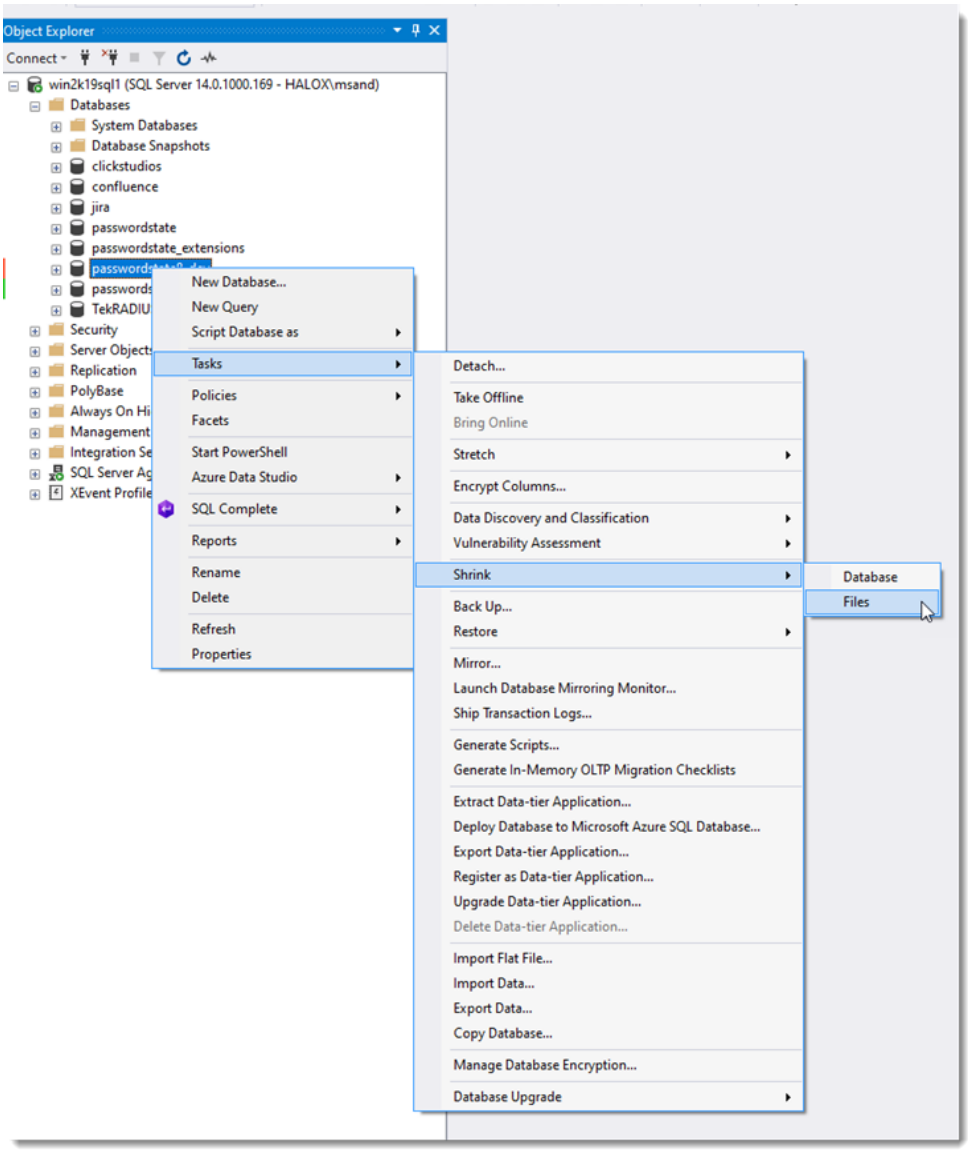
When the recovery mode is set to **FULL**, from a database management perspective, this means your Database Administrators must be managing the growth of the database transaction log, so it does not grow too large in size i.e. truncating the transaction log on a regular basis.

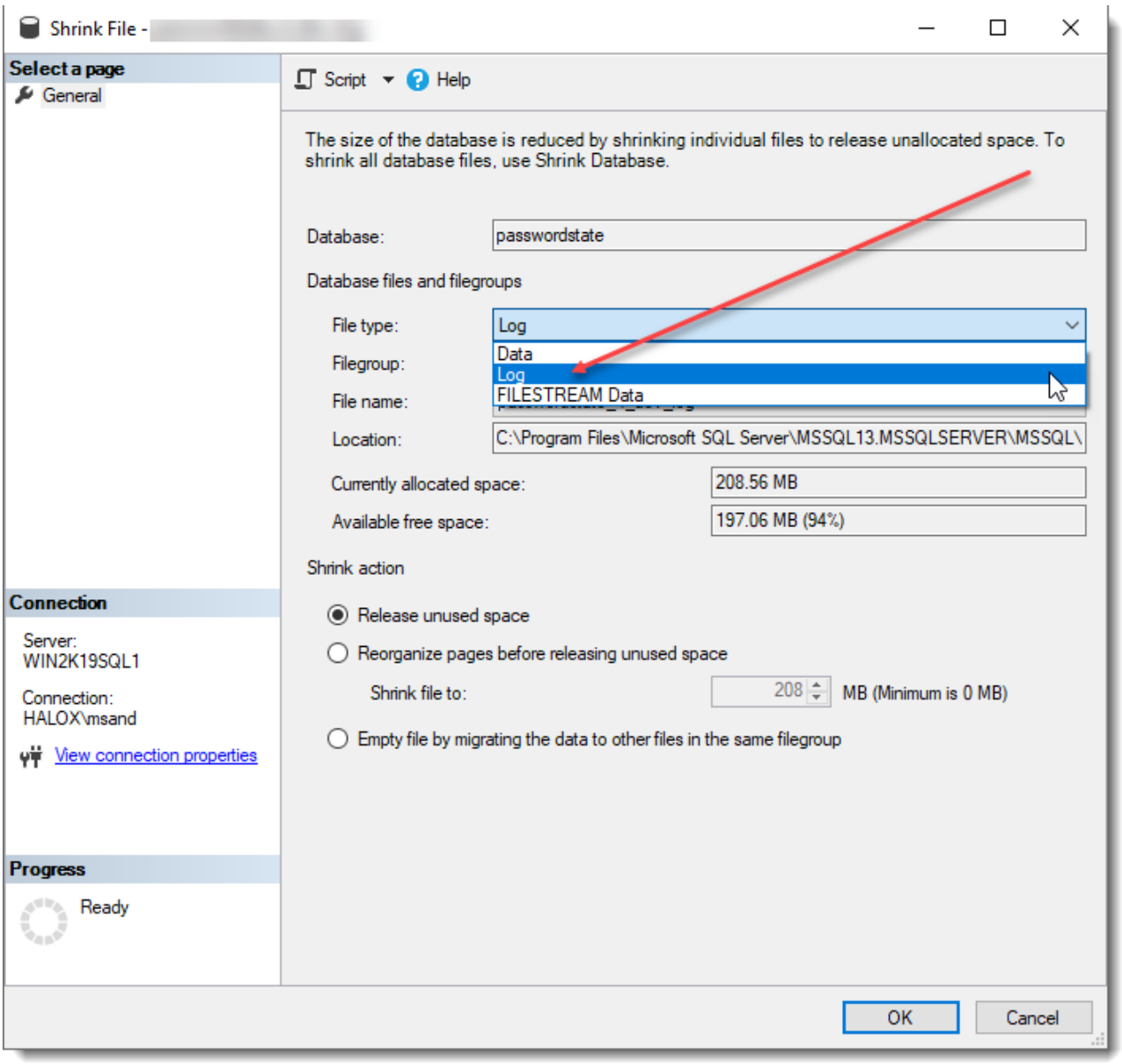
If a daily backup of your database is an option for you, in terms of recovery, then we can simplify the database management required, by setting the 'Recovery Model' to simple. Please follow the instructions below to do this, and also you can shrink your files if required, with the instructions below as well.

Right click on your database, select 'Properties', and set Recovery Model to 'Simple'.



Then right click again on your database, and choose to shrink your files – see screenshots below. Try shrinking both the Data and Log files – more than likely it will be your Log file which is causing your issue, and now that you have set Recovery Model to Simple, you should not see this issue again).


Object Explorer
Connect
win2k19sql1 (SQL Server 14.0.1000.169 - HALOX\msand)
Databases
System Databases
Database Snapshots
clickstudios
confluence
jira
passwordstate
passwordstate_extensions
Security
Replication
PolyBase
Management
XEvent Profile
New Database...
New Query
Script Database as
Tasks
Policies
Facets
Start PowerShell
Azure Data Studio
SQL Complete
Reports
Rename
Delete
Refresh
Properties
Detach...
Take Offline
Bring Online
Stretch
Encrypt Columns...
Data Discovery and Classification
Vulnerability Assessment
Shrink
Back Up...
Restore
Mirror...
Launch Database Mirroring Monitor...
Ship Transaction Logs...
Generate Scripts...
Generate In-Memory OLTP Migration Checklists
Extract Data-tier Application...
Deploy Database to Microsoft Azure SQL Database...
Export Data-tier Application...
Register as Data-tier Application...
Upgrade Data-tier Application...
Delete Data-tier Application...
Import Flat File...
Import Data...
Export Data...
Copy Database...
Manage Database Encryption...
Database Upgrade
Database
Files


Shrink File -
Select a page
General
Script
Help
The size of the database is reduced by shrinking individual files to release unallocated space. To shrink all database files, use Shrink Database.
Database:
passwordstate
Database files and filegroups
File type:
Log
Filegroup:
Data
Log
File name:
FILESTREAM Data
Location:
C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\
Currently allocated space:
208.56 MB
Available free space:
197.06 MB (94%)
Shrink action
Release unused space
Reorganize pages before releasing unused space
Shrink file to:
208
MB (Minimum is 0 MB)
Empty file by migrating the data to other files in the same filegroup
Connection
Server:
WIN2K19SQL1
Connection:
HALOX\msand
View connection properties
Progress
Ready
OK
Cancel