Passwordstate

Enterprise Password Management

App Server Install and Administration Guide

Table of Contents

1 App Server Overview

This document will detail instructions for installing the Passwordstate App Server, and configuring System wide settings related to this feature. This feature will allow you to use the native Android and iOS Mobile App, but can also be used as a portal for the Self Destruct Messages feature.

Typically, you would install this App Server in your DMZ, so you can allow access to it from outside your network. The App Server communicates securely back to your core Passwordstate website, on the port you use to access your Passwordstate site.

Alternatively, you can install the App Server on your Passwordstate webserver, and you may want to do this is you have limited servers in your environment.

For more information on how to use the Mobile App once this App Server has been installed, please refer to the **Mobile App User Manual** found under the **Help Menu** in Passwordstate.

For more information about using the App Server with the Self Destruct Portal, please see **Administration** -> **System Settings** -> **Self Destruct Messages**.

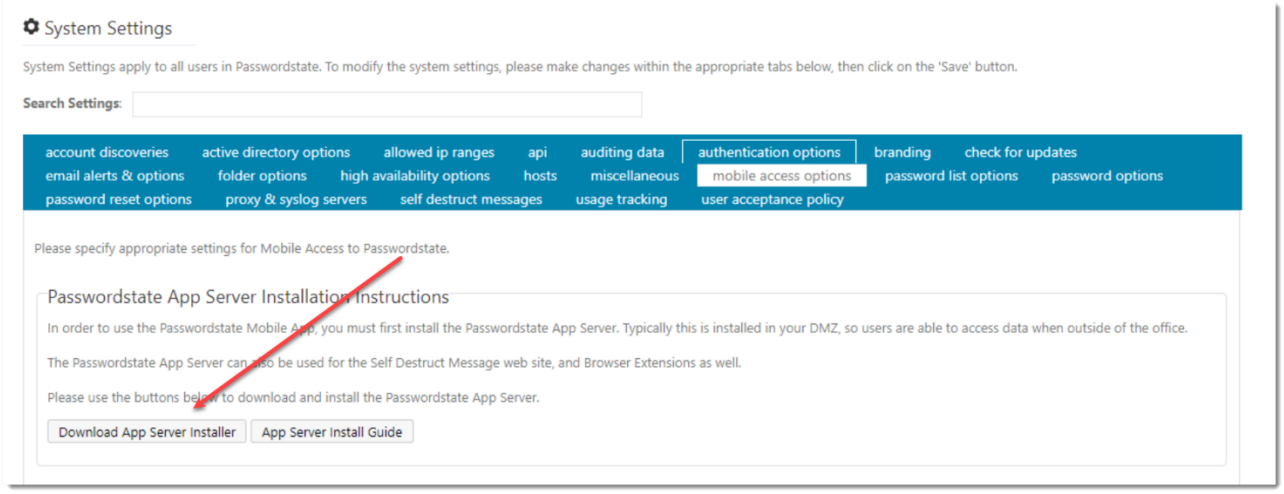
2 Prerequisites

In order for the App Server to function correctly, you will need to following requirements:

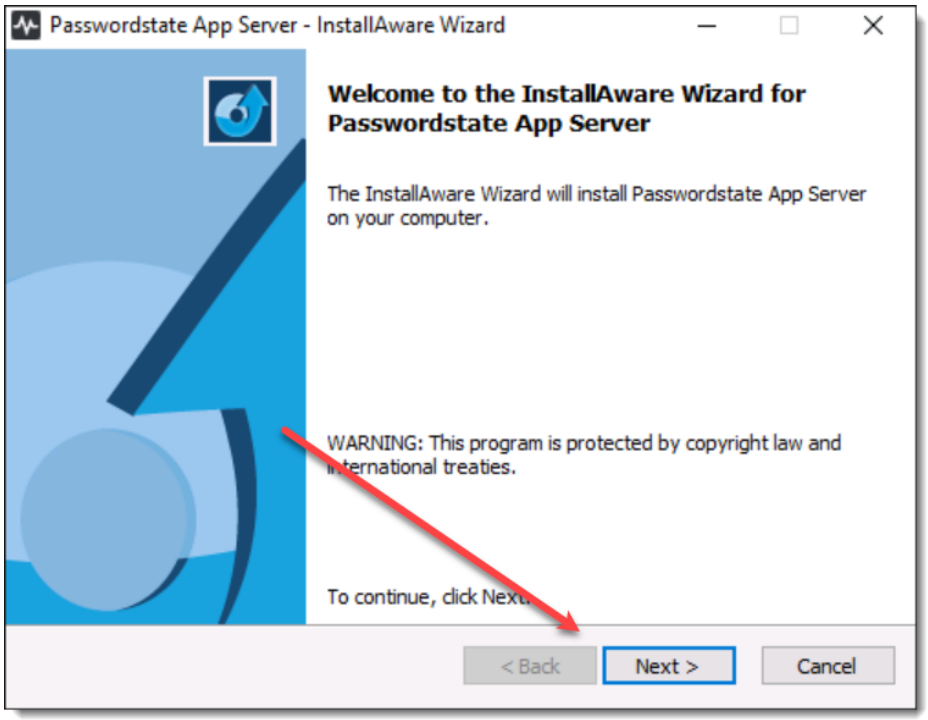
- An existing install of **Passwordstate 9** or above
- A Windows server running either **Windows 11**, or **Windows Server 2016** or above (*This can be your Passwordstate web server, or another server of your choice*)
- **Microsoft .NET Framework 4.7.2**
- A trusted SSL certificate (*More information on this in Section 4 of this guide*)
- From where your App Server is installed, you must have an open port for it to connect back to your SQL Server – this port is 1433 by default. If you are using a SQL Listener for database connectivity, you will need to allow port 1434 UDP – unless you have modified the default port.

3 Installing the App Server

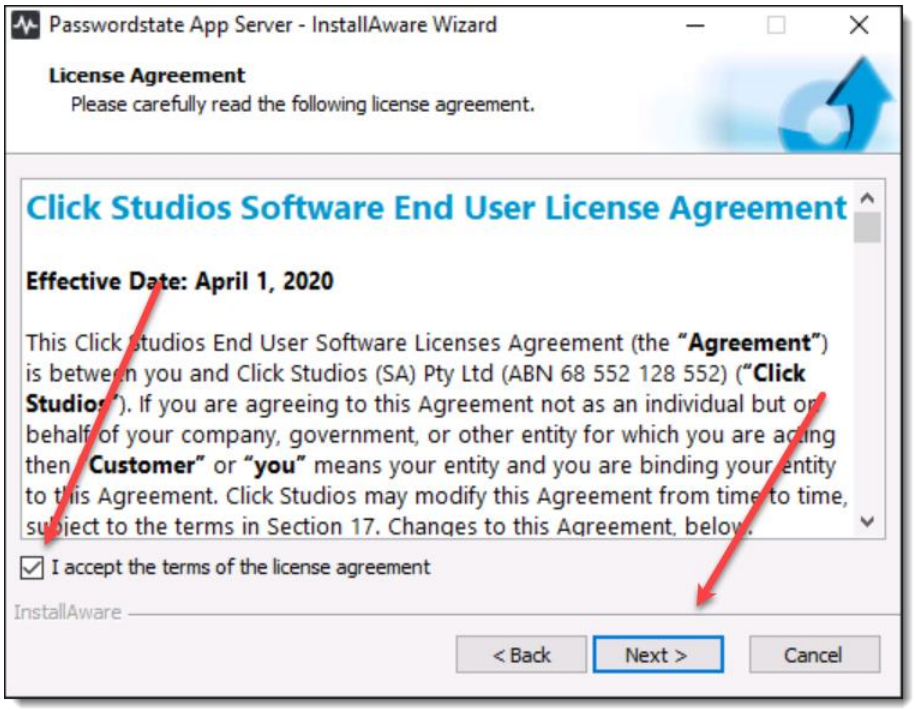
1. In Passwordstate, go to **Administration** -> **System Settings** -> **Mobile Access Options** and click the **Download App Server Installer** button:



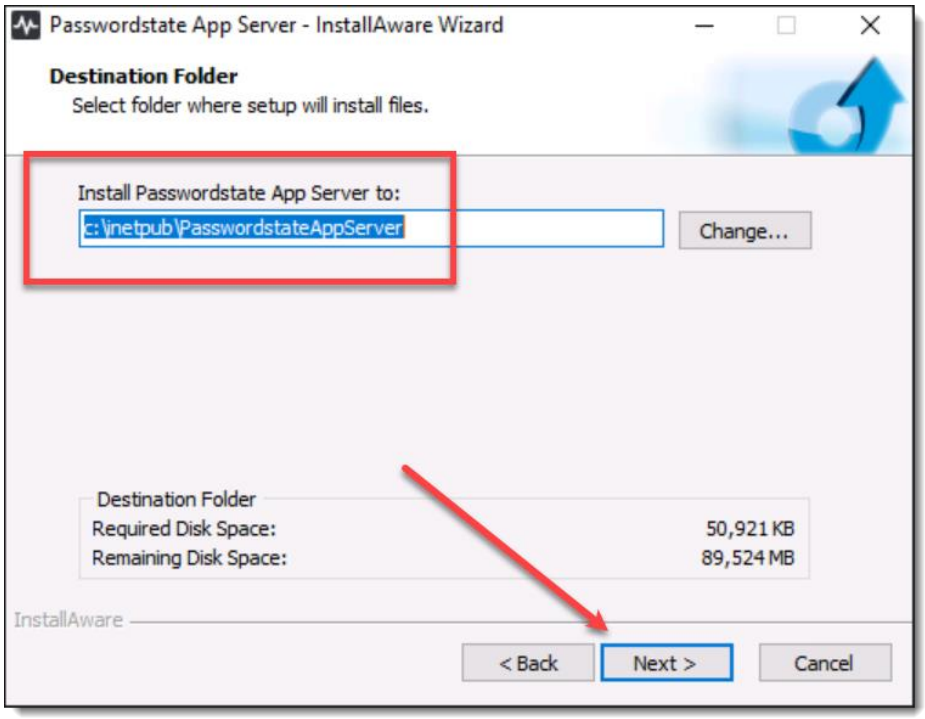
2. A file will download in your browser, and you'll need to transfer this to the server you are installing the App Server on
3. As an Administrator, run the **PasswordstateAppServer.exe** file
4. Click **Next**:



5. Accept the terms of the license agreement, and click **Next**:



6. Accept the default install path, and click **Next**:

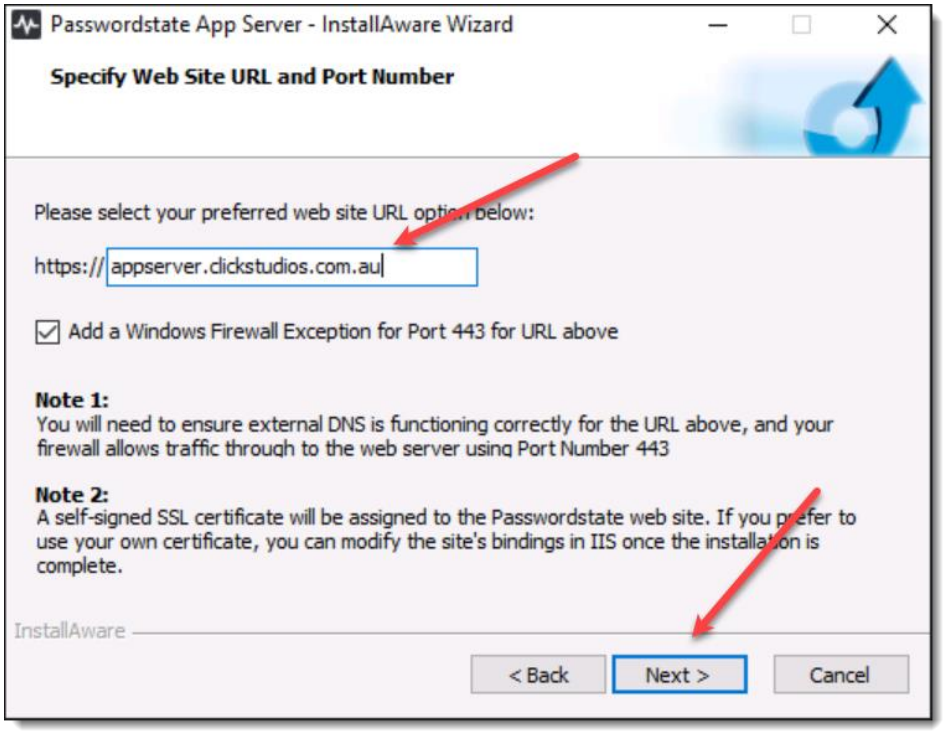


7. Set a URL of your choice, and click **Next**

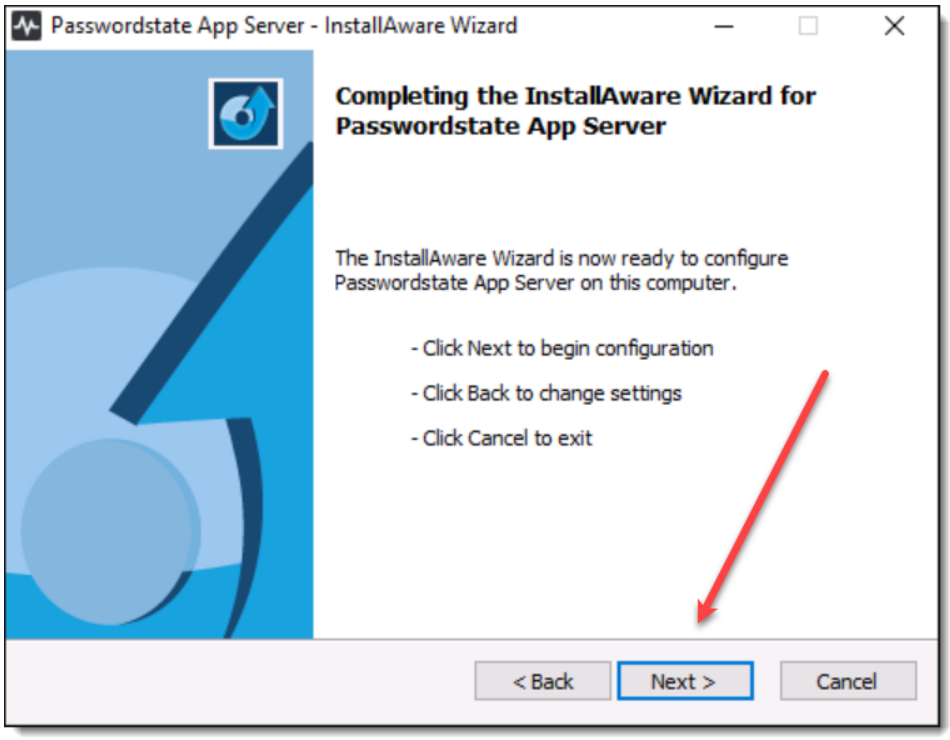
Note 1: This URL will require a functioning DNS entry, whether it be internal, or external from your network

Note 2: The default port set is 443, but this can be changed in IIS later if you require a custom port for any reason

Note 3: This installer will set up a website in IIS with a Self Signed Certificate. You will need to change this certificate to a trusted one post install, to ensure for seamless access to your Mobile App.



8. Click **Next**



9. Click **Finish**

4 SSL Certificate Considerations

Typically, this Passwordstate App Server will be installed in your DMZ, and you will then permit access to this website from outside your network, by opening your firewall on Port 443 by default, and having an external functioning DNS entry. There are two reasons for this:

1. Users on mobile phones or tablets that aren't joined to your domain can connect to the AppServer, and use the Mobile App
2. Users can access the Self Destruct Messages portal from any machine outside your network, from any internet connection, and read Self Destruct Messages.

When installing the App Server, it will create a new website in IIS called **passwordstateappserver**. If IIS has not been installed already, the installer will automatically set up and configure IIS for you.

The installer will create a Self-Signed Certificate using the name of the URL that you have entered during the install process.

It is highly recommended you change this certificate and use a purchased SSL certificate and assign it to your **passwordstateappserver** website, to ensure functional use of the Mobile App. Also, if you intend on accessing your Self Destruct Portal using the Passwordstate App Server, instead of the embedded Self Destruct Portal, then you should also assign a purchased trusted certificate from an online authority.

The Self Signed certificate that comes with the installer or even a certificate issued by your internal Certificate Authority will present issues for your end users, as the devices from outside the network will not trust these certificates.

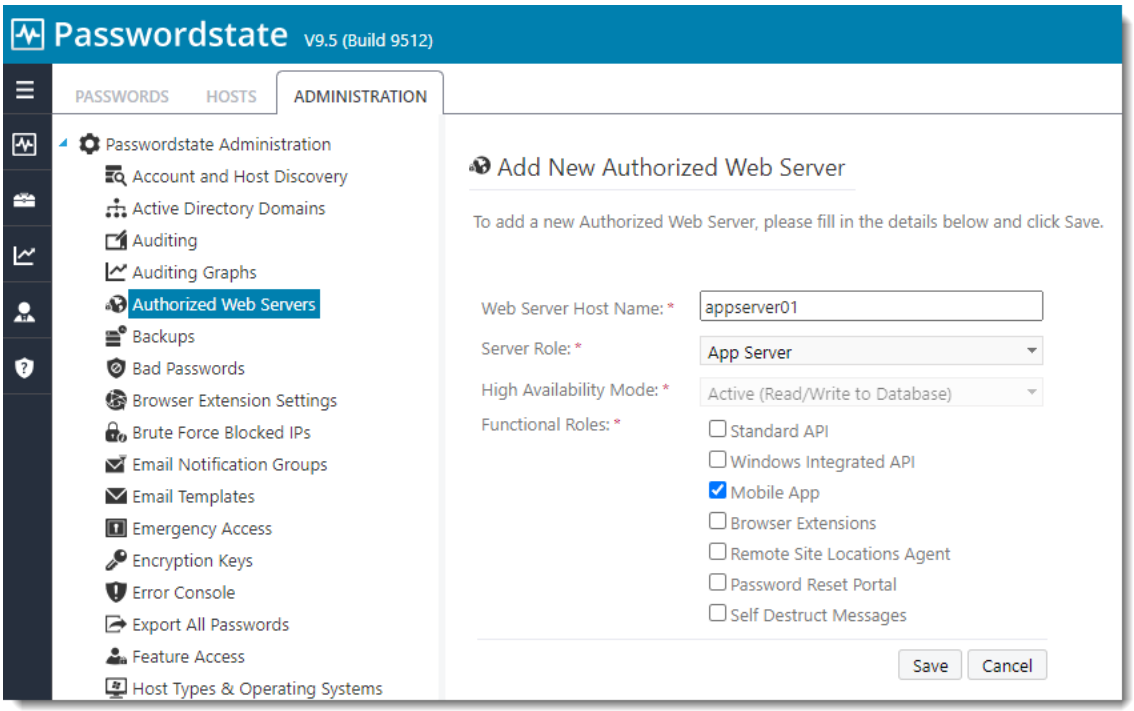
If your URL for the App Server is https://appserver.contoso.com, then you should use a certificate with a matching name of appserver.contoso.com, or even a wildcard certificate of *.contoso.com.

5 Initial App Sever Configuration

Please follow these instructions for the initial configuration of the App Server, before it can be used.

5.1 Authorized Web Server

In Passwordstate, under **Administration** -> **Authorized Web Servers**, click the **Add** button to add in the App Server. Enter in the netbios name of the server where you installed the App Server, and also assign the App Server role. Once this is done, click Save.



5.2 Web.config File Changes

As a once off process, you will need to copy the following 3 lines from the web.config file from your main Passwordstate web site folder, to the web.config file in this App Server folder.

- Core Passwordstate vault web.config file default location: c:\inetpub**Passwordstate**\web.config
- App Server web.config file default location: C:\inetpub**PasswordstateAppServer**\web.config

Note: Only copy **PasswordstateConnectionString** and **Secret1** and **Secret2** lines

When looking at the web.config file for your main Passwordstate installation, if you do not see your database connection string, or appSetting section, then most likely you have encrypted these sections of the file, and you will need to decrypt them in order to copy across the contents. To do this, follow the instructions below – and only do this when users are not using Passwordstate:

- Open command prompt as Administrator, and run the following commands
    - CD C:\Windows\Microsoft.NET\Framework64\v4.0.30319
    - aspnet_regiis.exe -pdf "connectionStrings" "c:\inetpub\passwordstate"
    - aspnet_regiis.exe -pdf "appSettings" "c:\inetpub\passwordstate"

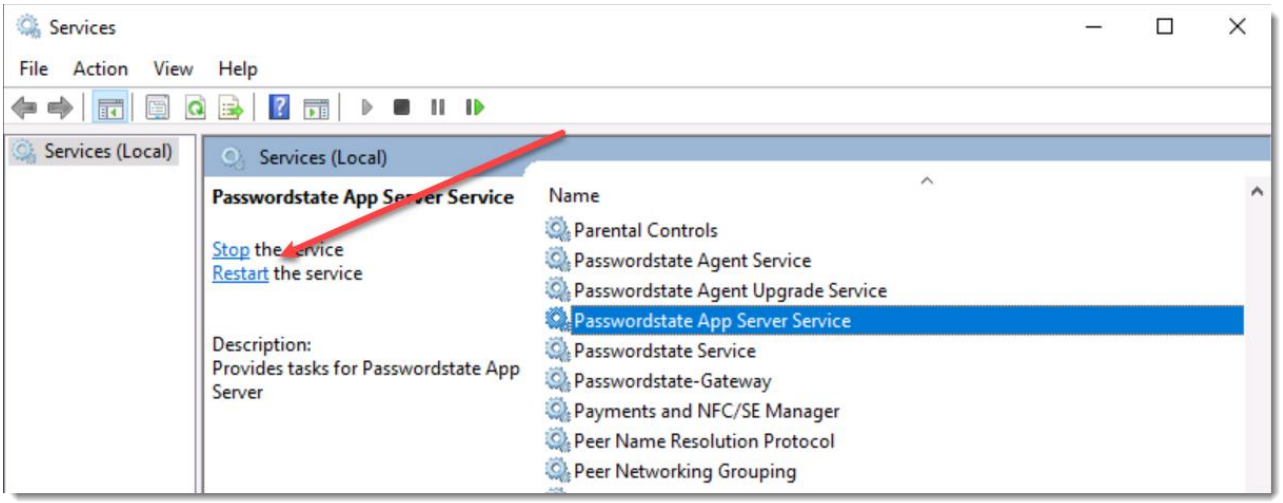
To re-encrypt them again afterwards, use the commands:

- aspnet_regiis.exe -pef "connectionStrings" "c:\inetpub\passwordstate"
- aspnet_regiis.exe -pef "appSettings" "c:\inetpub\passwordstate"

Example of App Server web.config file after inserting connection string and secret keys

Note: From where your App Server is installed, you must have an open port for it to connect back to your SQL Server – this port is 1433 by default. If you are using a SQL Listener for database connectivity, you will need to allow port 1434 UDP – unless you have modified the default port.

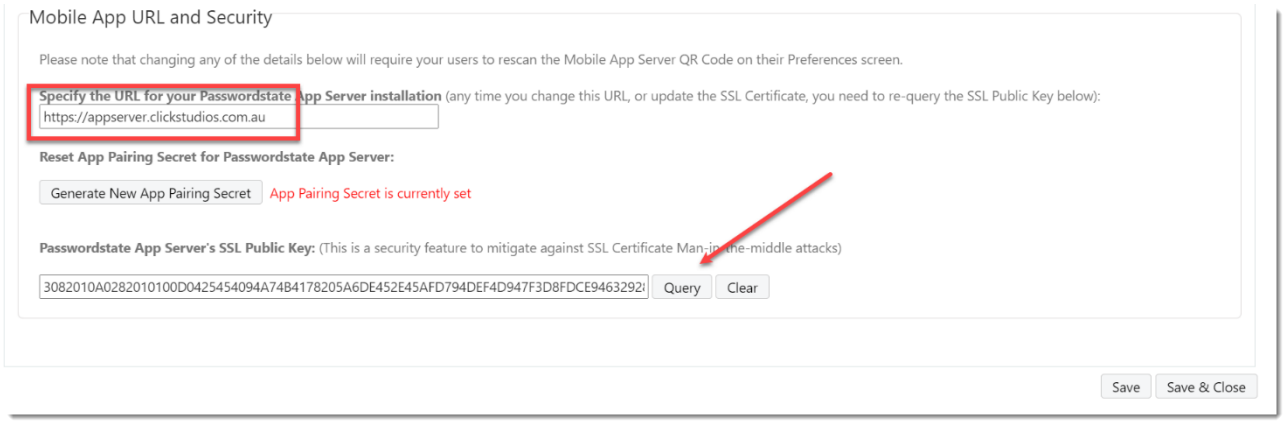
Now restart the Passwordstate App Server Windows Service:



5.3 URL and APP Server Public Key

In Passwordstate, under **Administration** -> **System Settings** -> **Mobile Access Options**, set your URL for your App Server, including port if it is different to 443. Also, for added security, you should generate a SSL Public Key by clicking the **Query** button.

Once these fields are populated, click the **Save** button:

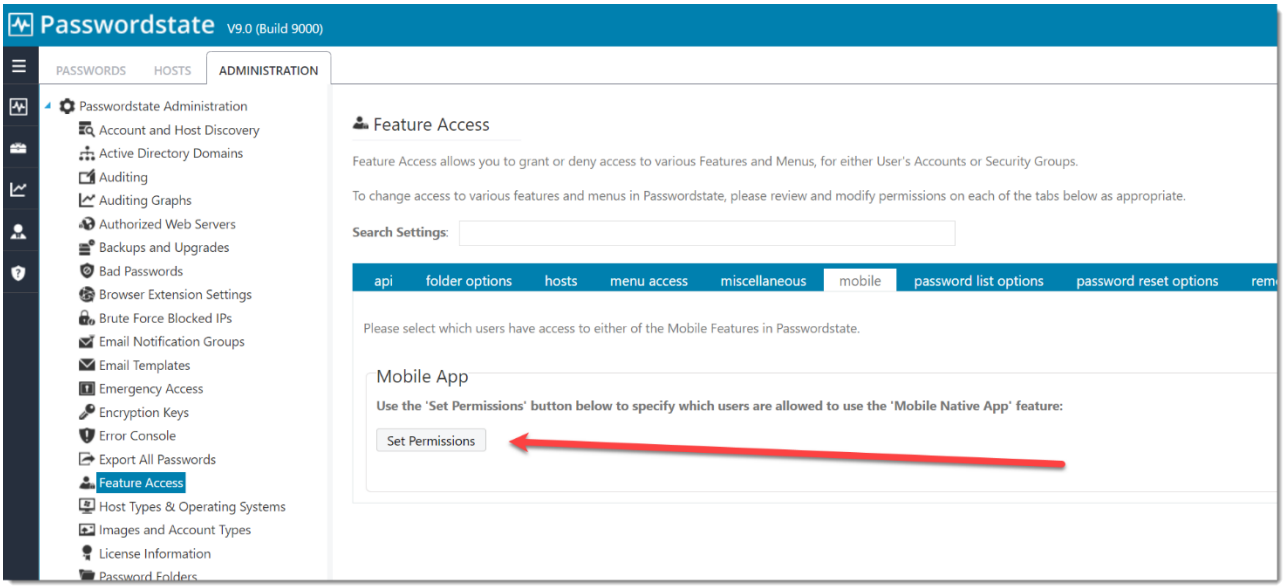


6 Mobile App Permissions

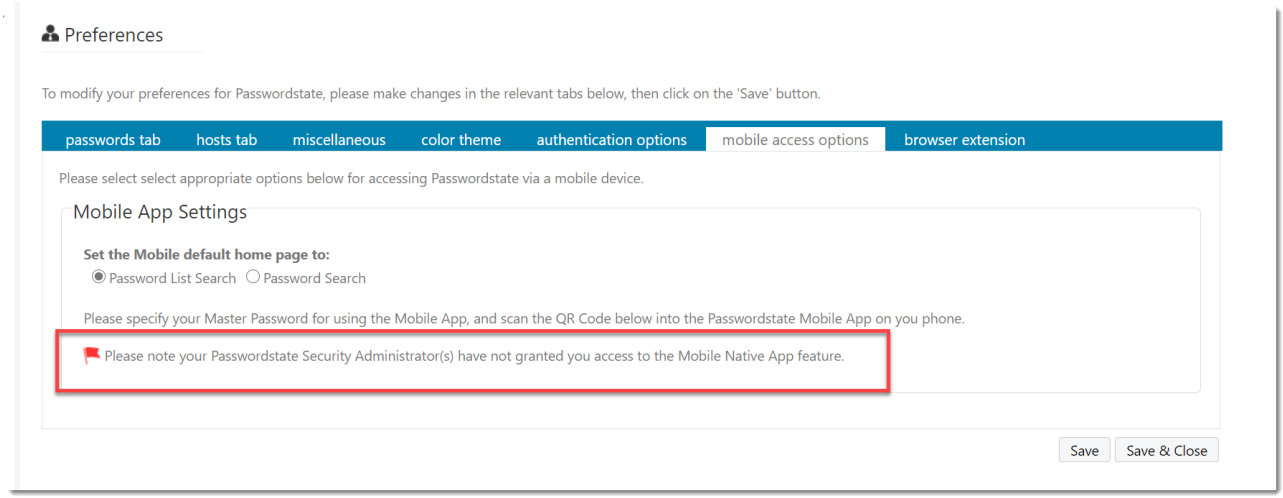
Passwordstate has the ability to enable or disable features for groups of users, and it's possible to prevent users from using the Mobile App. There are several places to consider when granting or denying these permissions.

6.1 Feature Access

As a Security Administrator, under **Administration** -> **Feature Access** -> **Mobile**, you can **allow** or **deny** users in Passwordstate from using the Mobile App:

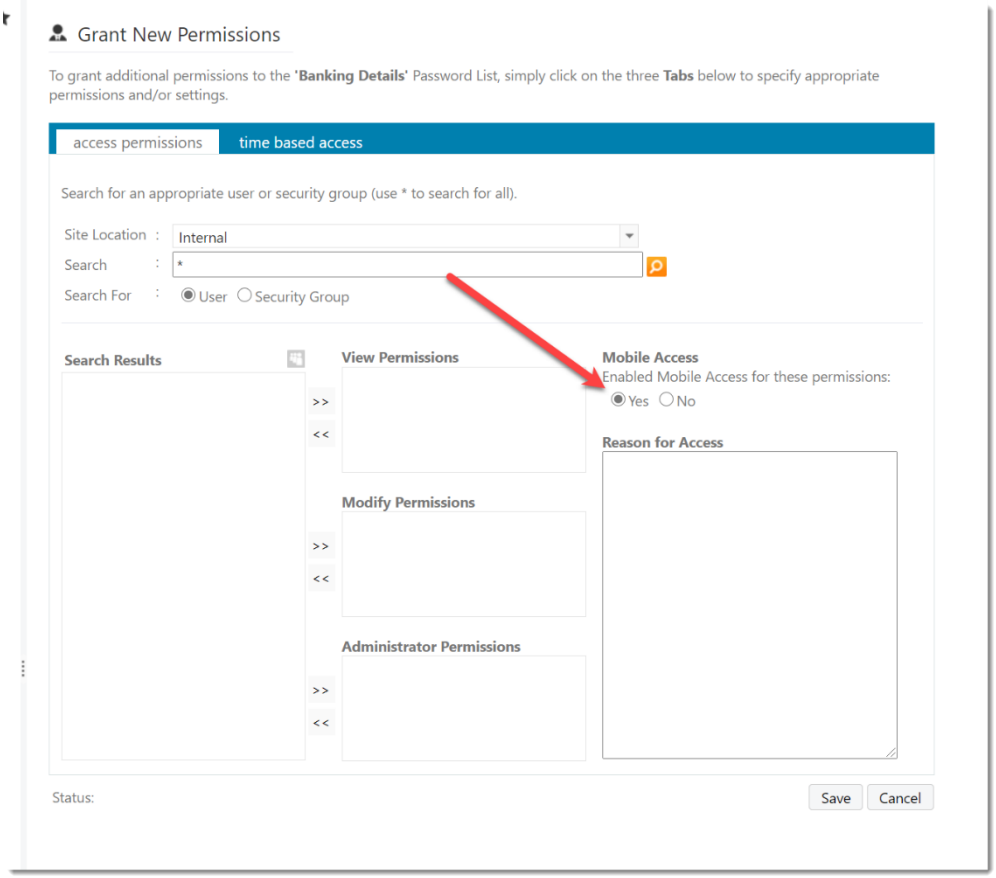


If the user, or a Security Group the user is in, has been granted access on the screen above, they will be able to scan the QR code under their own personal preferences. If you deny them access, then they will be presented with this warning under their own personal preferences, and will not be able to use the App:



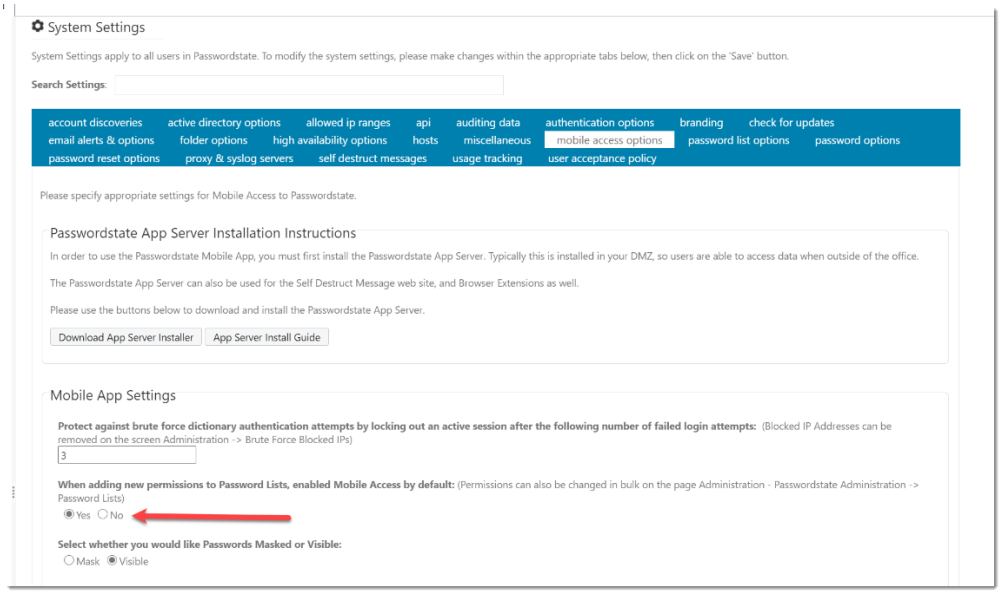
6.2 Mobile Access to Password Lists

For a greater level of control as to what passwords and Password Lists will show in the mobile App, you can turn off or on Mobile access at a Password List level. By default, when granting a user access to a Password List, it will also enable Mobile access too:



If you select the above option to **No** when granting a user access, that Password List will not show up in the Mobile App for that user.

There is also an option in the System Settings to set that above option to be **No** by default, instead of **Yes**:

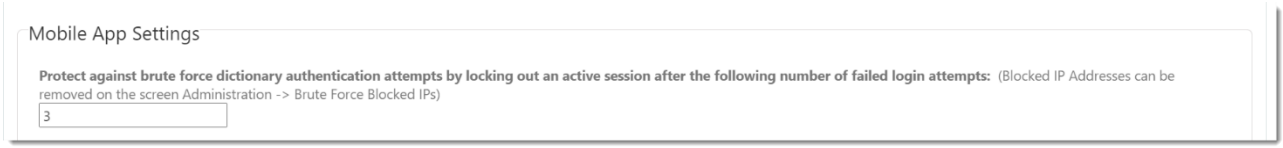


7 Mobile App System Settings

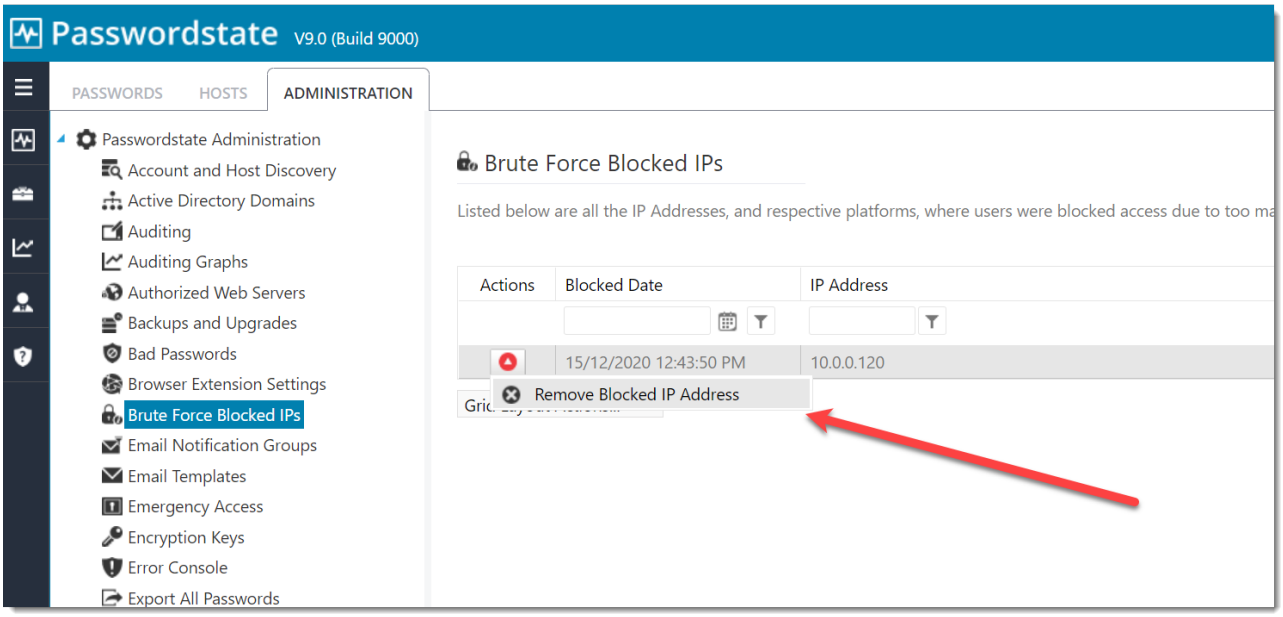
Under **Administration** -> **System Settings** -> **Mobile Access Options**, there are also a number of other settings you can configure:

7.1 Brute Force Protection

If a user tries to log into the Mobile App, there is a brute force setting you can configure to prevent them from logging in again if they failed entering their **Master** password. Three is the default setting but is configurable:

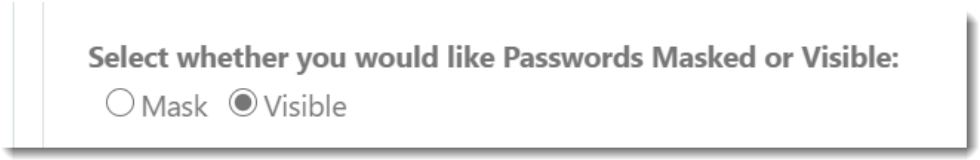


If a user fails to log in and the Brute Force feature locks them out of the Mobile App, the user will be presented with an error message on their Mobile App screen, and a Passwordstate Administrator must unblock them from this screen:



7.2 Mask Passwords or make them Visible

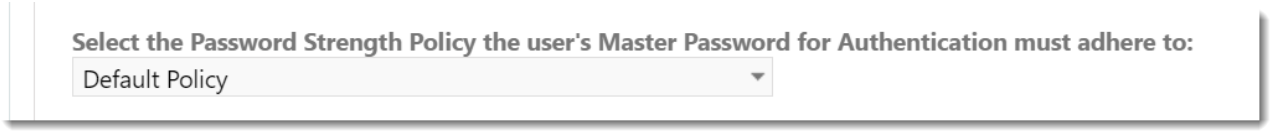
By default, your passwords in the Mobile App will be masked, and your users must press their finger on the masked password on the phone or tablet and choose **Toggle Mask** to see the password. You can make the password visible by default by changing this system setting option to **Visible**



7.3 Password Strength Policy

When a user attempts to use the Mobile App for the first time, they must create a Master Password under their own personal preferences. When the users need to authenticate into their Mobile App, this is the

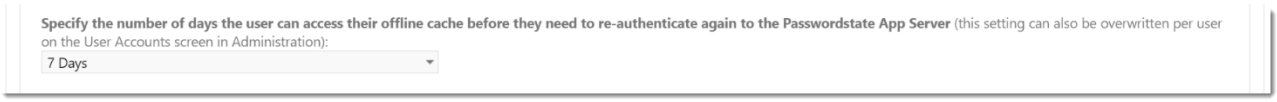
password that they need to enter. This system setting can force the user to adhere to a specific Password Strength Policy of your choice:



7.4 Offline Cache

The Mobile App will sync with Passwordstate and download an offline cache of all Password Lists and Passwords user has access to. This feature improves performance dramatically and also allow your users to have access to their passwords when their device does not have access to the internet.

This System Setting below can be configured to require a manual authentication into the App more frequently, or less frequently, depending on your company policy.



By default, this setting is set to 7 days, and this value can be overwritten on a per user basis under **Administration** -> **User Accounts**, on the **Mobile Access Options** tab on each user in the system.

8 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 and change to the folder C:\Windows\Microsoft.NET\<Framework or Framework64>\v4.0.30319
- Type the following:
    - aspnet_regiis.exe -pef "connectionStrings" "c:\inetpub\PasswordstateAppServer" (change the path if you've installed the App Server to a different location)

Decrypt Connection String

- Open a command prompt and change to the folder C:\Windows\Microsoft.NET\<Framework or Framework64>\v4.0.30319
- Type the following:
    - aspnet_regiis.exe -pdf "connectionStrings" "c:\inetpub\PasswordstateAppServer" (change the path if you've installed the App Server to a different location)

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

9 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 and type CD C:\Windows\Microsoft.NET\Framework64\v4.0.30319
- Type the following:
    - aspnet_regiis.exe -pef "appSettings" "c:\inetpub\PasswordstateAppServer" (change the path if you've installed the App Server to a different location)

Decrypt appSettings Section

- Open a command prompt and type CD C:\Windows\Microsoft.NET\Framework64\v4.0.30319
- Type the following:
    - aspnet_regiis.exe -pdf "appSettings" "c:\inetpub\PasswordstateAppServer" (change the path if you've installed the App Server to a different location)

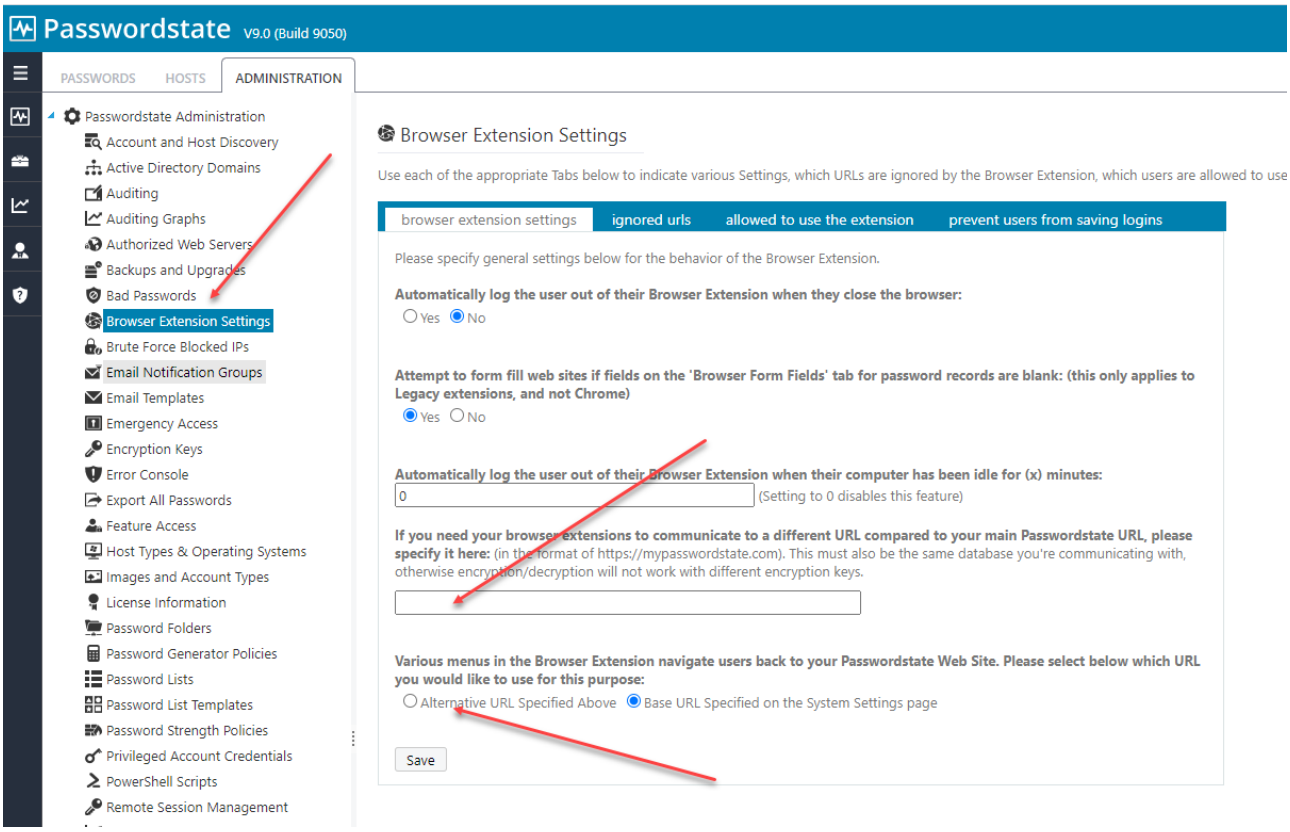
Note: If you intend to rename your server host name, or move your Passwordstate App Server install to a different server, you should decrypt these settings first.

10 Browser Extension and Self Destruct Web Site Usage

The Passwordstate App Server can also be used for the Browser Extensions, and the Self Destruct Message web site if required.

Browser Extensions

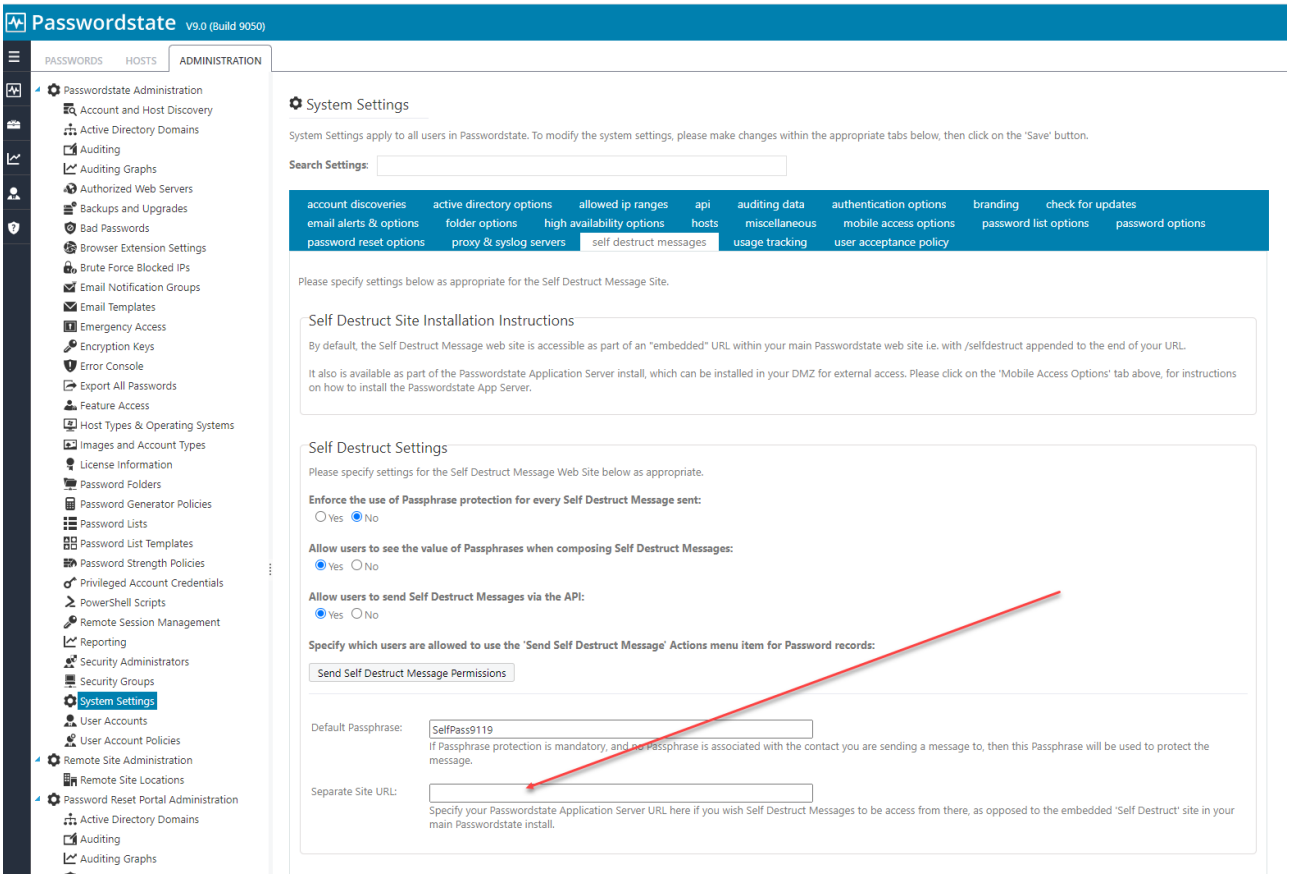
If you want your user's browser extensions to communicate via the URL of your Passwordstate App Server, instead of your normal Passwordstate instance, you can configure this below. Users will need to log out of their extension, and log back into Passwordstate to pick up this change.



Self Destruct Message Web Site

By default, the Self Destruct Message web site is embedded with your normal Passwordstate web site, but it can also be used with the App Server as well if required i.e. if you install the App Server is installed in the DMZ.

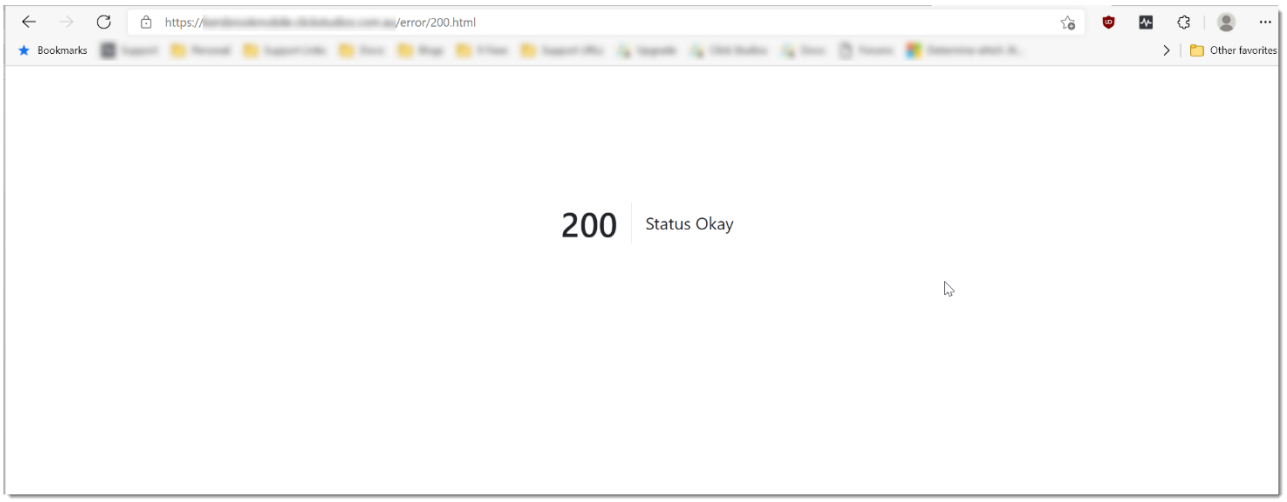
To use the Self Destruct web site with the App Server, simple specify your App Server URL below in the following screenshot.


Passwordstate V9.0 (Build 9050)
PASSWORDS
HOSTS
ADMINISTRATION
Passwordstate Administration
Account and Host Discovery
Active Directory Domains
Auditing
Auditing Graphs
Authorized Web Servers
Backups and Upgrades
Bad Passwords
Browser Extension Settings
Brute Force Blocked IPs
Email Notification Groups
Email Templates
Emergency Access
Encryption Keys
Error Console
Export All Passwords
Feature Access
Host Types & Operating Systems
Images and Account Types
License Information
Password Folders
Password Generator Policies
Password Lists
Password List Templates
Password Strength Policies
Privileged Account Credentials
PowerShell Scripts
Remote Session Management
Reporting
Security Administrators
Security Groups
System Settings
User Accounts
User Account Policies
Remote Site Administration
Remote Site Locations
Password Reset Portal Administration
Active Directory Domains
Auditing
Auditing Graphs
System Settings
System Settings apply to all users in Passwordstate. To modify the system settings, please make changes within the appropriate tabs below, then click on the 'Save' button.
Search Settings:
account discoveries
active directory options
allowed ip ranges
api
auditing data
authentication options
branding
check for updates
email alerts & options
folder options
high availability options
hosts
miscellaneous
mobile access options
password list options
password options
password reset options
proxy & syslog servers
self destruct messages
usage tracking
user acceptance policy
Please specify settings below as appropriate for the Self Destruct Message Site.
Self Destruct Site Installation Instructions
By default, the Self Destruct Message web site is accessible as part of an "embedded" URL within your main Passwordstate web site i.e. with /selfdestruct appended to the end of your URL.
It also is available as part of the Passwordstate Application Server install, which can be installed in your DMZ for external access. Please click on the 'Mobile Access Options' tab above, for instructions on how to install the Passwordstate App Server.
Self Destruct Settings
Please specify settings for the Self Destruct Message Web Site below as appropriate.
Enforce the use of Passphrase protection for every Self Destruct Message sent:
Yes
No
Allow users to see the value of Passphrases when composing Self Destruct Messages:
Yes
No
Allow users to send Self Destruct Messages via the API:
Yes
No
Specify which users are allowed to use the 'Send Self Destruct Message' Actions menu item for Password records:
Send Self Destruct Message Permissions
Default Passphrase:
SelfPass9119
If Passphrase protection is mandatory, and no Passphrase is associated with the contact you are sending a message to, then this Passphrase will be used to protect the message.
Separate Site URL:
Specify your Passwordstate Application Server URL here if you wish Self Destruct Messages to be access from there, as opposed to the embedded 'Self Destruct' site in your main Passwordstate install.

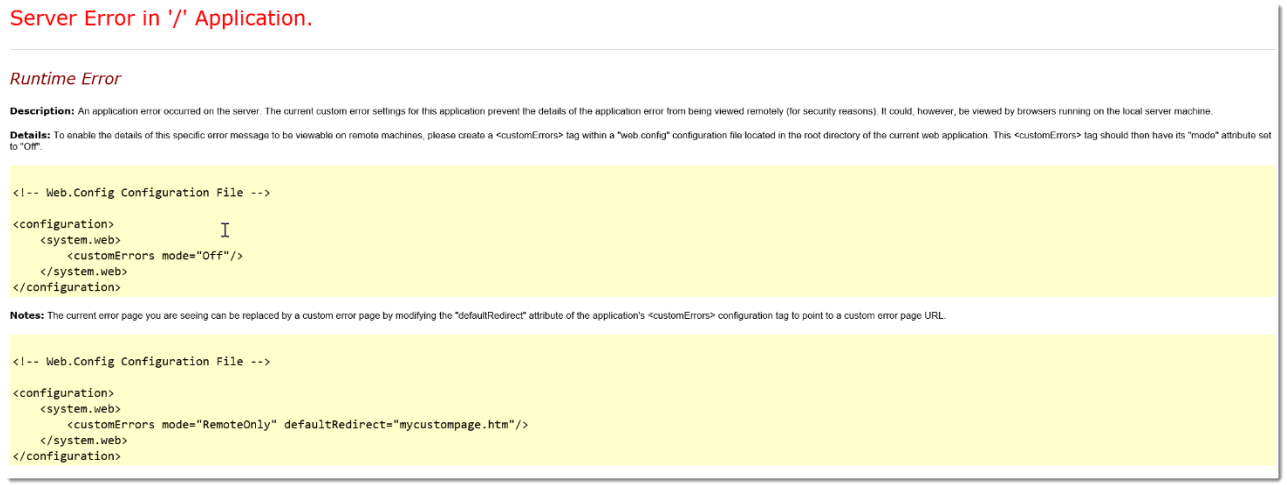
11 AppServer Troubleshooting Guide

In this section we'll troubleshoot the App Server not working

First test to do is try browsing to your App Server URL and if the website is working correctly, you will get a **200 | Status Okay** message as per the below screenshot. This is the URL you have set under **Administration** -> **System Settings** -> **Mobile Access Options**



If you see an Error like this screenshot below, log directly into your App Server and try browsing to your URL again. You should get a more verbose error message when logged directly into the server which can help troubleshoot the issue:



Error 1:

"A network-related or instance-specific error occurred while establishing a connection to SQL Server. The server was not found or was not accessible. Verify that the instance name is correct and that SQL Server is configured to allow remote connections"

What does this mean?

Your App Server needs to be able to communicate directly to your Passwordstate database in order for it to work successfully. There's a few possible reasons why your App Server cannot comminucate to your database server, and below are some possible fixes for this:

Fix 1:

First confirm that you have copied the Connection String out of your web.config file from your core Passwordstate webserver across to the App Server web.config file. The database server name, SQL user and password need to be the same on both servers.

Please note if your web.config file is encrypted, you cannot copy the encrypted text across from the primary web.config file to the App Server web.config. You will need to decrypt it first and copy across the clear text you see. A good tutorial on what an encrypted web.config file looks like can be found here: https://forums.clickstudios.com.au/topic/2699-encrypting-and-decrypting-the-webconfig-file/

An example of a clean Connection String to copy across can be seen in screenshot below:

Fix 2:

Remote Directly into the App Server and perform a database connectivity test to you Passwordstate database server. An example of this in Powershell using the connection string above is:

Test-NetConnection webserver01 -Port 1433

1433 is the default port used for SQL, but is configurable. If you have changed the port, it will be referenced in your Connection String as per below screenshot:

If this Open Port test returns a failure, then most likely a firewall is blocking access between your App Server and Database server.

If this test fails it could also possibly be DNS for the Database server name is not resolvable from your App Server. A quick test to check DNS working when logged into your app server is as follows, either in a command prompt, or in Powershell:

NSLookup webserver01

If this test fails, please investigate DNS to confirm it is set up correctly.

Fix 3:

If you are using SQL Express you may need to enable TCP/IP on your database server, and possibly set a static Port. SQL Express by default uses a dynamic port and setting a static port can ensure firewall rules are working correctly. An example of what a SQL Express instance looks like in you web.config file is as per below screenshot:

To learn how to fix this issue, please see Section 3 in this document:
https://www.clickstudios.com.au/downloads/version9/Installation_Instructions.pdf

Fix 4:

If you are using the High Availability Module of Passwordstate, it's possible you are also using SQL Replication with a "SQL Listener". If using one of these you may need to open port **1434 UDP** on your database server, unless you have specially have changed this port.