Password Encryption

Components that are required to authenticate against a database or another repository need to have user credentials specified in configuration files. Such sensitive information must be encrypted so that the actual value of the password will be known only during runtime, protecting Aspire from potential malicious attacks.

On this page

Create the Main Key

All password encryption / decryption is based on a main key. Use the following steps to create a new random main key:

1. Go to ASPIRE_HOME
2. Run *bin/generateMasterKeyFile*. This script creates an encrypted main password file (in the config/security directory). This file will contain a random key used to decrypt passwords inside Aspire.

Notes:

- The main key file must be secured by the operating system. This means that administrators should grant "read" access only to the user running Aspire.
- The main key file path value is stored inside the settings file as a property called "masterKeyFilePath"

> **Important security concern**
>
> If you skip this step, **all of the encryption on Aspire will use the default main key**, which comes with all Aspire Distributions. All of your encrypted passwords will be exposed (to be decrypted in any other Aspire Distribution). **Using a main Key is *extremely important* to prevent this exposure.**

Password Encryption Aspire Administration UI

If you want to secure Aspire access to the Administration UI using the ConfigFile method, the best and most secure way is by encrypting the passwords. Complete the following steps.

1. Run *bin/encryptPassword.* This script will create passwords of the "administrator" and "developer" users to be encrypted inside the *settings.xml* file when you use the optional "username" parameter.
2. Enable the ConfigFile authentication on the *settings.xml* file:

   settings.xml

   ```
   <authentication>
     <type>ConfigFile</type>
   </authentication>
   ```

3. Start Aspire and use the "administrator" or "developer" users to log in using the recently added passwords.

> **Policy passwords**
>
> UI passwords must comply to policy rules as described in UI Password Policy.

Password Encryption for Maven Repositories and Custom Applications

You always need to encrypt your password to Aspire Maven repository. Also If you are creating a custom application which requires password encryption, you need to encrypt it. In both cases, use the "encryptPassword" script.

- The encrypted password will be stored as a property in the *settings.xml* file.
- In case of Maven repository you might need to copy the encrypted password from the property to the "remoteRepository/password" element in *settings.xml.*
- In case of custom application the property should be specified to the appropriate component's configuration in the *application.xml* file.

To use password encryption, complete the following steps.

1. Run *bin/encryptPassword*. This script will prompt for the password you need to encrypt, and an optional username. The password is as the property with the name passwordEncrypted written to the Aspire *settings.xml* file. When you specify the username then the name of the property will be usernamePassword.
2. Reference the created property from the component's configuration using ${propertyName} syntax.
3. Make sure your component allows password encryption.

Important Notes

By default, Aspire uses a secret pre-configured main key unless an administrator uses **generateMasterKeyFile** to create a new one.