If you are working with sensitive data, you will want to make sure to **secure the offline data** in case the client computer is compromised.

Encrypt the cache

You should use a CryptoStream to encrypt the cache before saving it to the file system. To prevent discovery of the encryption key, use a secure hash of the user's credentials (e.g. username and password) to derive the encryption key and **never** store the key or password on the file system where it can be discovered. This way, the serialized cache can only be decrypted by someone with the correct credentials.

To learn more about encryption, see the SymmetricAlgorithm class.

Authenticate while offline

When offline, there is no host to authenticate the user, and you cannot call the *Login* or *LoginAsync* method. This is ok, since there is no access to the database or other non-local datasource in this circumstance. To authenticate the user, you can:

1. Check for the successful decryption of the cache which will indicate that they have entered the correct credentials, or
2. If you are not encrypting the cache, compare the hash of the credentials to the locally stored hash that was computed from a prior successful login.