Lucifure Stash Version 1.X – Readme

1. *Package Folder Structure*

Lucifure.Stash.v1.X
- Docs
- Lib
  - 4.0
- Lib-Alt
  - 4.0
- Lucifure.Stash.Tutorial

2. *Version 1.2X Changes*

See CodePlex page. http://lucifurestash.codeplex.com/

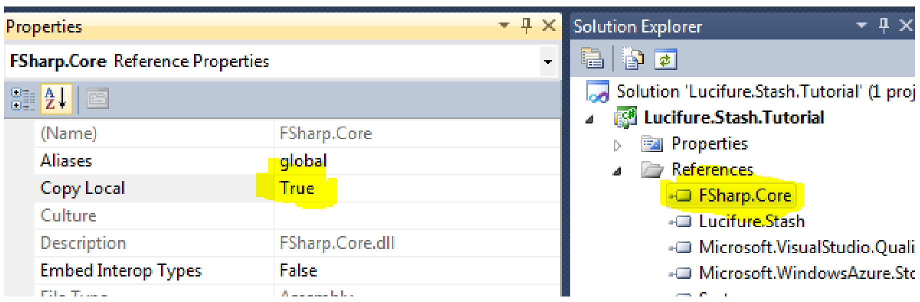
3. *FSharp.Core.dll dependency.*

Lucifure Stash is written in F#, targets .NET 4.0, and requires FSharp.Core.dll to run.

Two versions of Lucifure Stash are provided in the Lib folder.

- Lib contains a standalone version which includes FSharp.Core.dll
- Lib-Alt contains a version which requires the FSharp.Core.dll to be included in the project using Lucifure Stash.

If you do not wish to use the standalone version, you may download and install the F# runtime from http://www.microsoft.com/download/en/details.aspx?DisplayLang=en&id=15834 . Please note that you could choose not to use the standalone version and instead include the FSharp.Core.dll in your project and set local copy as true to deploy on Windows Azure. (Both options are provided such that you can configure your project effectively as per your needs.)


Properties
Solution Explorer
Solution 'Lucifure.Stash.Tutorial' (1 proj
Lucifure.Stash.Tutorial
Properties
References
FSharp.Core
Lucifure.Stash
Microsoft.VisualStudio.Quali
Microsoft.WindowsAzure.Stc

FSharp.Core Reference Properties

(Name)	FSharp.Core
Aliases	global
Copy Local	True
Culture	
Description	FSharp.Core.dll
Embed Interop Types	False

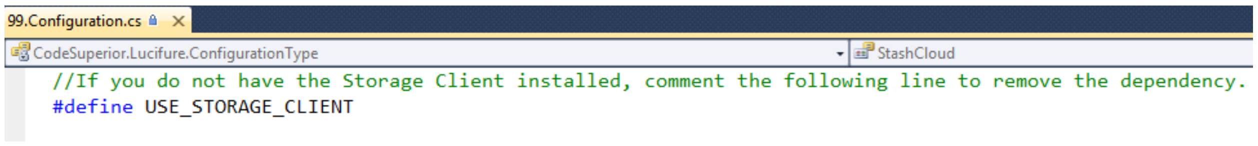
The tutorial targets the standalone version so there is no need to make any changes in the tutorial.

4. *Tutorial*

The tutorial is a the easiest way to get familiar with Lucifure Stash. It is written in C# and is in the form of unit tests. The source files in the tutorial are sequenced and each introduce one major concept and continue building for there.

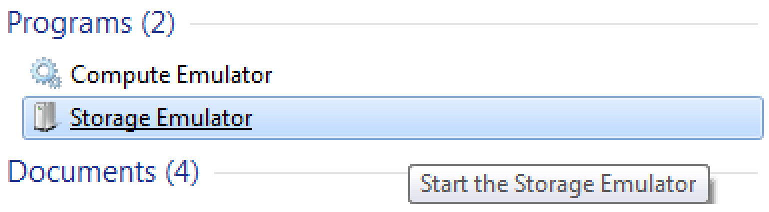
a. The tutorial reference the Microsoft.WindowsAzure.StorageClient dll. If this dll is not present on your machine, you can either download it (install Azure SDK), or simply remove it from the project and comment out the #define in "99.Configuration.cs".

Lucifure Stash does not use this dll but it is included in the tutorial to demonstrate credential sharing if you are already using the Microsoft Storage Client.



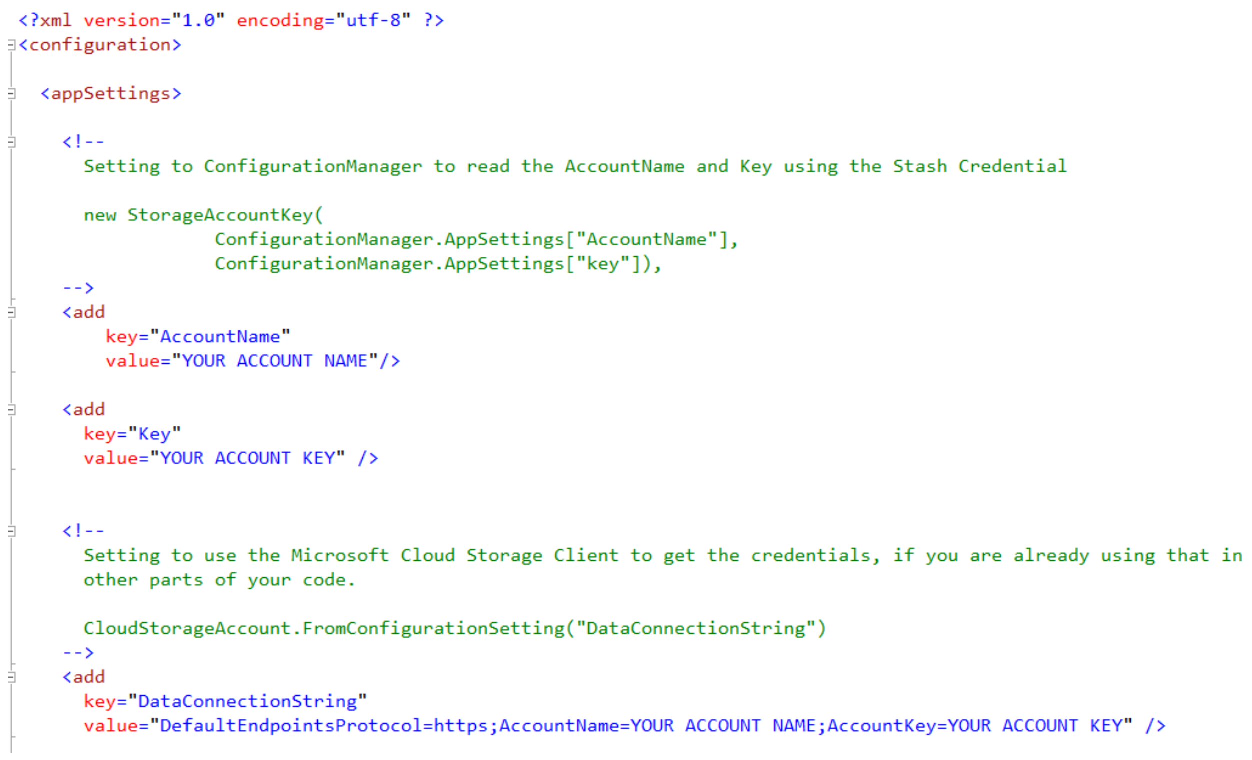
```
//If you do not have the Storage Client installed, comment the following line to remove the dependency.
#define USE_STORAGE_CLIENT
```

b. Out of the box, the tutorial use the Azure storage emulator, so it is imperative that it is started before running the tutorial.



c. To target cloud storage instead of the emulator

i. Add your storage account name and key to **App.Config**. You can fill in either or both of the sections.



```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>

  <appSettings>

    <!--
      Setting to ConfigurationManager to read the AccountName and Key using the Stash Credential

      new StorageAccountKey(
                  ConfigurationManager.AppSettings["AccountName"],
                  ConfigurationManager.AppSettings["key"]),
    -->
    <add
        key="AccountName"
        value="YOUR ACCOUNT NAME"/>

    <add
      key="Key"
      value="YOUR ACCOUNT KEY" />


    <!--
      Setting to use the Microsoft Cloud Storage Client to get the credentials, if you are already using that in
      other parts of your code.

      CloudStorageAccount.FromConfigurationSetting("DataConnectionString")
    -->
    <add
      key="DataConnectionString"
      value="DefaultEndpointsProtocol=https;AccountName=YOUR ACCOUNT NAME;AccountKey=YOUR ACCOUNT KEY" />
```

ii. Modify **99.Configuration.cs** by changing the enumeration in the line

```
// This method, demonstrates various ways of setting up the credentials.
public
static
StashClient<T>
GetClient<T>(
    StashClientOptions                    options)
{
    // Change the type here to target different storage accounts and different credential methods.
    ConfigurationType
    configType = ConfigurationType.StashEmulator;
```

to either of the following depending on your needs.

```
ConfigurationType
configType = ConfigurationType.StashCloud;
```

or

```
ConfigurationType
configType = ConfigurationType.StorageAccountCloud;
```