Integrating example with your existing skin (Step by Step)

Instructions for DIRECT Host Version: 3.6.+

These instructions will show you how to implement the Steam Auth Interop. A test harness javascript file (*interoptestexample.js*) is included. This file will simply alert out the variables that are produced by the interop. Using this example, you should be able to integrate these variables into your authentication service calls as needed.

1. Copy steaminterop.dll and steam\_api.dll from /bin subdirectory to host exe location
2. Copy steamauth.js and interoptestexample.js from the /js directory to the your skin /js directory
3. Open your mainwindow.html in your skin
4. **Following**   
   *<script src="mainwindow.js" type="text/javascript"></script>*   
   **add**   
   *<script src="/js/steamauth.js" type="text/javascript"></script>  
   <script src="/js/interoptestexample.js" type="text/javascript"></script>*
5. Save mainwindow.html
6. Open your workflow file - workflow.json
7. Add the following under the *loadDownloader* item  
    "loadSteam": {

"type": "loadInterop",

"name": "steam",

"filename": "{ModulePath}{LibraryPrefix}steaminterop.{LibraryExtension}"

},

1. Add "*loadSteam*" to the actions item in the load queue
2. Add the following under the *unloadDownloader* item  
    "unloadSteam": {

"type": "unloadInterop",

"name": "steam",

"filename": "{ModulePath}{LibraryPrefix}steaminterop.{LibraryExtension}"

},

1. Add "*unloadSteam*" to the actions item in the unload queue
2. Rebuild the skin using Patch Assistant
3. Make sure the steaminterop.dll and steam\_api.dll are still in the directory with the host executable
4. Run the exe with /DisableSecurity as the first argument (During production if you sign the dll you won't need this).