Winlogon Notification Library for Windows 7

Introduction:

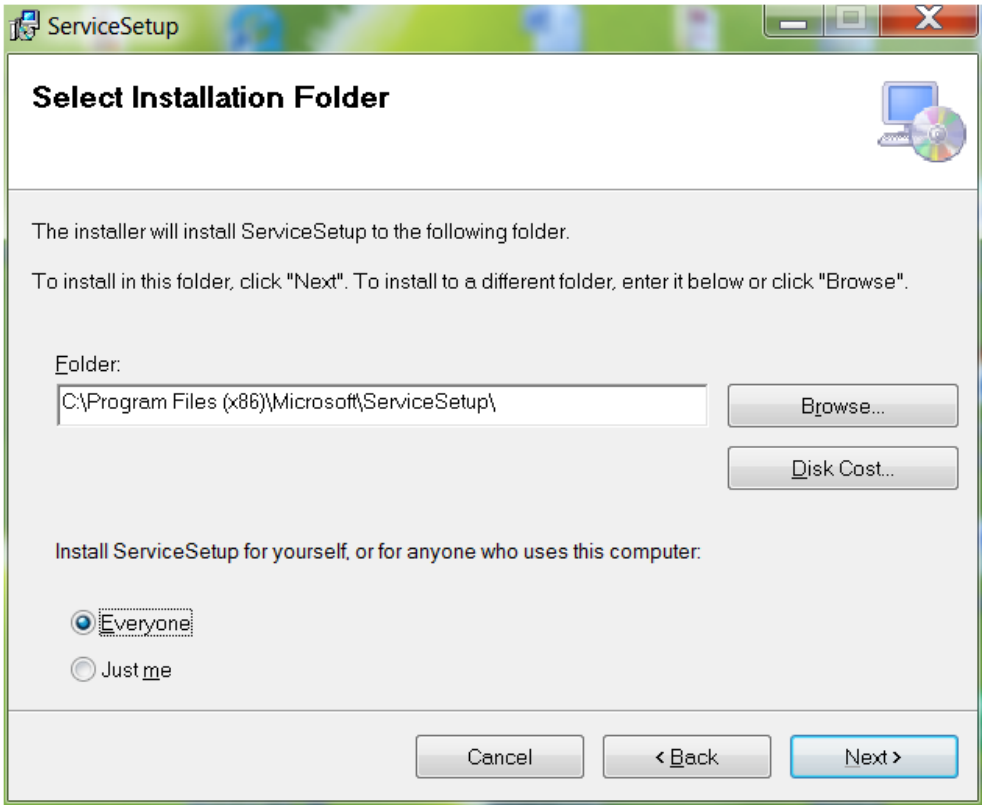
Date: 02/23/2013

This events handler library is designed as a windows service which starts automatically as a Local System service during the system start up. An XML file is used as a database for storing custom application paths, target window settings, etc. Service is named as '**A Win 7 Events Library**'. This document is produced to help the client in installing the supplied installation file, modifying XML for application paths and testing the events.

Installation Process:

Step 1: Run the Setup.exe file

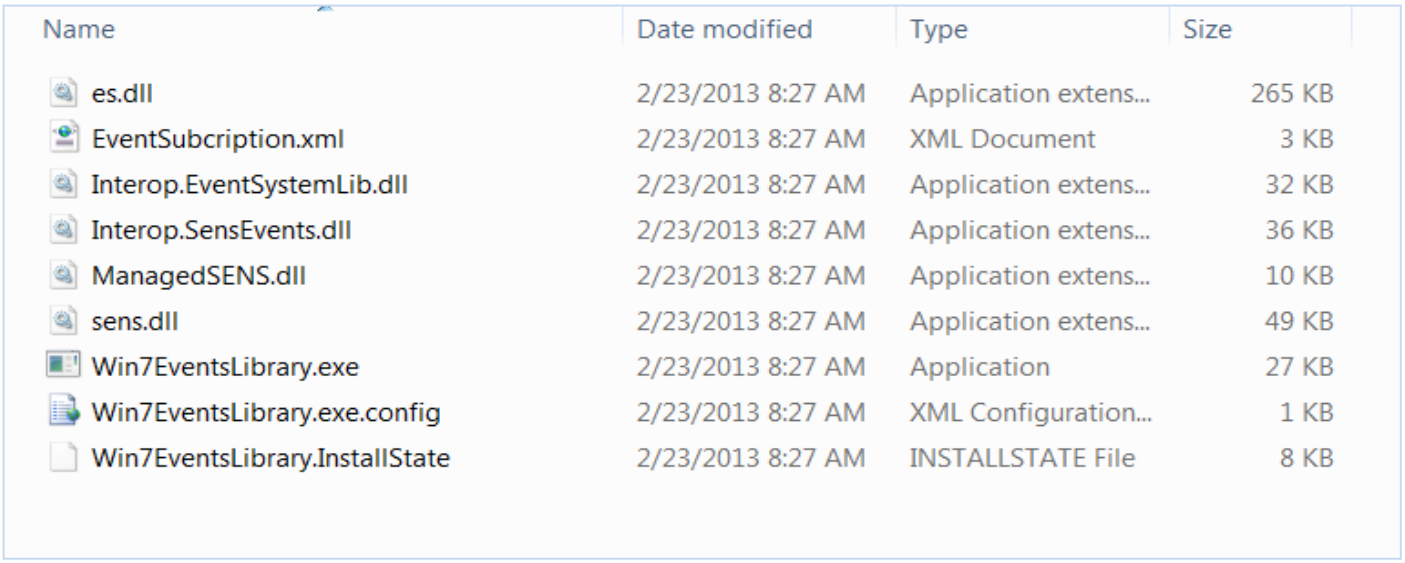
Step 2: Select the folder to install the setup file and select **Everyone** For "who uses this Computer" option.



Step 3: Select next till the installation process is finished.

Step 4: Once the installation process is successfully completed, the service will start automatically.

Step 5: The project installation folder will contain these files



Name	Date modified	Type	Size
es.dll	2/23/2013 8:27 AM	Application extens...	265 KB
EventSubcription.xml	2/23/2013 8:27 AM	XML Document	3 KB
Interop.EventSystemLib.dll	2/23/2013 8:27 AM	Application extens...	32 KB
Interop.SensEvents.dll	2/23/2013 8:27 AM	Application extens...	36 KB
ManagedSENS.dll	2/23/2013 8:27 AM	Application extens...	10 KB
sens.dll	2/23/2013 8:27 AM	Application extens...	49 KB
Win7EventsLibrary.exe	2/23/2013 8:27 AM	Application	27 KB
Win7EventsLibrary.exe.config	2/23/2013 8:27 AM	XML Configuration...	1 KB
Win7EventsLibrary.InstallState	2/23/2013 8:27 AM	INSTALLSTATE File	8 KB

EventSubcription.xml file contains the records for each handled event.
Each event record has four properties

1. **Event ID** – Any unique ID
2. **Event Name** – Name of the event handled. This field acts as a record matching property in order to get the application path and target window dynamically. Changing the Event names requires rebuilding the service.
3. **Execute Path** – Full path of the application/program that needs to be run every time the event is handled.
4. **Target Window** – This property supplies information to the event handler about where the application has to be run, Winlogon screen or Default windows desktop. Values supplied in this field needs to be as exactly below:
   i. **WinSta0\Winlogon – Display application in Log On Screen**
   ii. **WinSta0\Default – Display application in default Windows desktop.**

Once the Settings UI is designed, filling the Execute Path and Target Window properties for events will be taken care in a user friendly manner.

Step 6: After installation is complete, edit the **EventSubcription.xml** file to change the applications to run for each event. The user may selects or change the ExecutePath and the TargetWindow they required for their application.

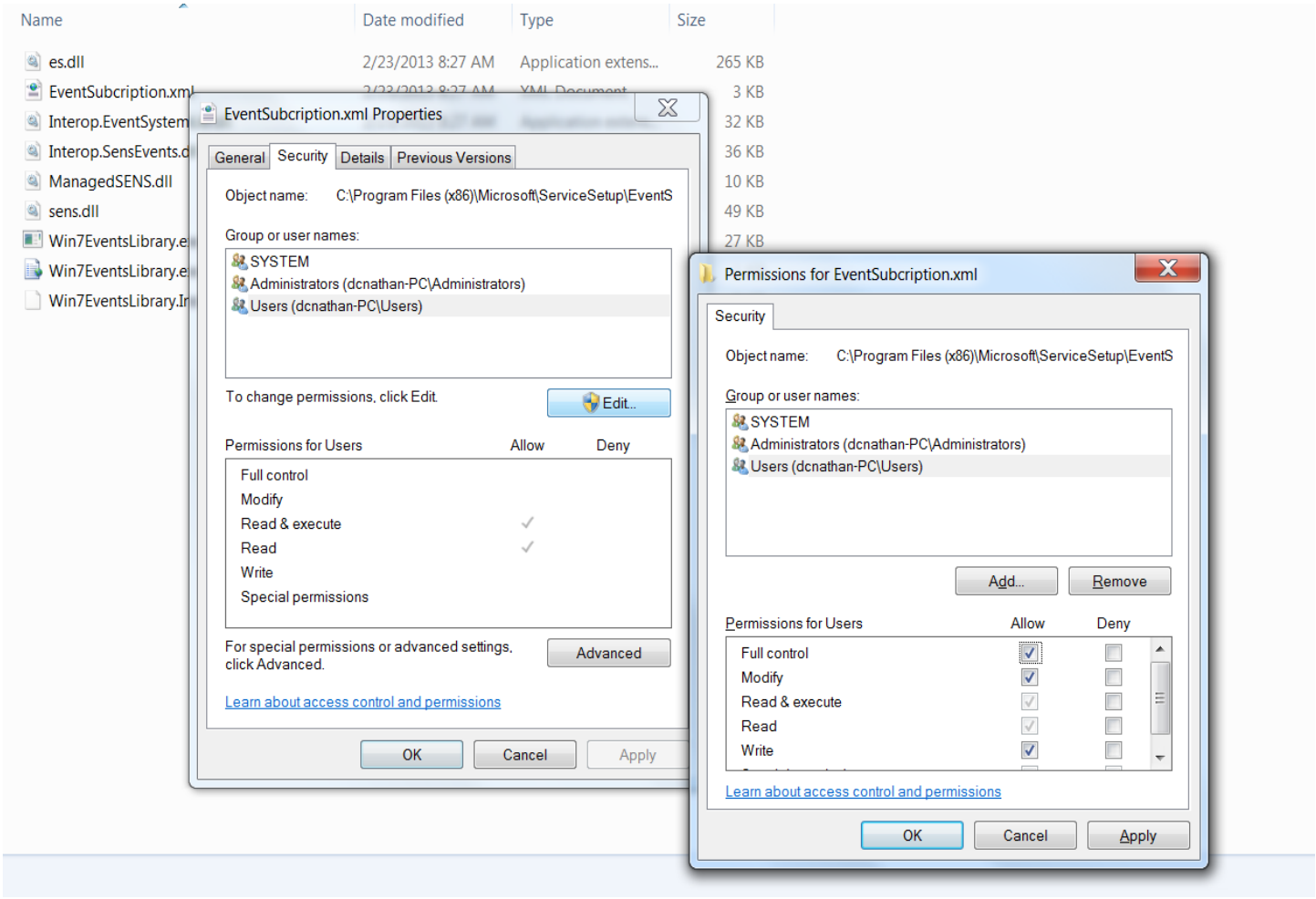
Example: Screen shows before changing the path of logoff event.

```
            </xs:element>
          </xs:choice>
        </xs:complexType>
    </xs:element>
</xs:schema>
<Table>
    <EventID>E001</EventID>
    <EventName>logoff</EventName>
    <ExecutePath>C:\Users\dcnathan\Documents\Nathan\ELANCE\Source code and articles\logon\Sample Application\bin\Debug\SampleApplication.exe</ExecutePath>
    <TargetWindow>WinSta0\Winlogon</TargetWindow>
</Table>
<Table>
    <EventID>E002</EventID>
    <EventName>logon</EventName>
    <ExecutePath>C:\Windows\system32\mspaint.exe</ExecutePath>
    <TargetWindow>WinSta0\Default</TargetWindow>
</Table>
<Table>
    <EventID>E003</EventID>
    <EventName>displaylock</EventName>
    <ExecutePath>C:\Projects\Win7EventsLibrary\ServiceApp\bin\Debug\ServiceApp.exe</ExecutePath>
    <TargetWindow>WinSta0\Winlogon</TargetWindow>
```

After changing the path of logoff event

```
</xs:schema>
<Table>
    <EventID>E001</EventID>
    <EventName>logoff</EventName>
    <ExecutePath>C:\Windows\system32\calc.exe</ExecutePath>
    <TargetWindow>WinSta0\Winlogon</TargetWindow>
</Table>
<Table>
    <EventID>E002</EventID>
    <EventName>logon</EventName>
    <ExecutePath>C:\Windows\system32\mspaint.exe</ExecutePath>
    <TargetWindow>WinSta0\Default</TargetWindow>
</Table>
<Table>
    <EventID>E003</EventID>
    <EventName>displaylock</EventName>
    <ExecutePath>C:\Projects\Win7EventsLibrary\ServiceApp\bin\Debug\ServiceApp.exe</ExecutePath>
    <TargetWindow>WinSta0\Winlogon</TargetWindow>
</Table>
<Table>
    <EventID>E004</EventID>
    <EventName>displayunlock</EventName>
    <ExecutePath>C:\Windows\system32\calc.exe</ExecutePath>
    <TargetWindow>WinSta0\Default</TargetWindow>
```

Step 7: Once the updates are done save the xml file. If permission denied while saving the file, go to the properties of xml file and assign permission to all. Then save the file.



Step 8: Test the events.

Completed Events :

Event	Suggested Window to Display the application
DisplayLock	Windows Logon Screen
DisplayUnlock	Windows Default Desktop
Logon	Windows Default Desktop
Logoff	Windows Logon Screen
ServiceStart	Windows Logon Screen/ Windows Default Desktop
ServiceStop	Windows Logon Screen/ Windows Default Desktop Application will not be open if the service is stopped during system shutdown.
StartScreenServer	Event fired when the user is idle for some specified time and an application without UI can be run behind the screensaver screen.
StopScreenServer	The event fired when the screen saver service is stopped by the user and corresponding exe files can be displayed at the user's default window screen.