

SyAM Software

Using Management Utilities to Install a Desktop Shortcut for Site Manager Self Service Ticketing

To make it easy for end users to submit incident reports, use Management Utilities to deploy an icon on user desktops that points to the URL of the Site Manager's Self Service Ticketing module. This procedure is supported for Windows 7 and Windows XP machines with the System Client installed and reporting to the System Area Manager.

Here is a Visual Basic script that will create the desktop icon.

```
'Create a wshShell object
Set wshShell = wscript.CreateObject("wscript.Shell")

strOS = wshShell.ExpandEnvironmentStrings("%OS%")
If strOS = "Windows_NT" Then
    strVerKey = "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\"
    strVersion = wshShell.RegRead(strVerKey & "ProductName")
Else
    strVerKey = "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\"
    strVersion = wshShell.RegRead(strVerKey & "ProductName")
End If

retVal = InStr(strVersion, "XP")
If retVal = 0 Then
    strPath=wshShell.expandEnvironmentStrings("%Public%")
    strPath1=wshShell.expandEnvironmentStrings("%PROGRAMFILES%")
Else
    strPath=wshShell.expandEnvironmentStrings("%ALLUSERSPROFILE%")
    strPath1=wshShell.expandEnvironmentStrings("%PROGRAMFILES%")
End If

'Create a wshShortcut object
Set oShellLink = wshShell.CreateShortcut(StrPath& "\Desktop\SyAM Help Desk.lnk")

'Set the Target path for the shortcut
oShellLink.TargetPath = "http://192.168.100.156/selfservice.aspx"

'Set the icon
oShellLink.IconLocation = strPath1& "\Internet Explorer\iexplore.exe,1"

'Save the shortcut
oShellLink.Save
```

Items in **boldface** should be modified by the administrator so that the correct URL for the Site Manager is specified, and the shortcut is given a different name if desired. Save the script file; in this example, the script will be saved as shortcutscript.vbs.

Next, create a Windows batch file to run the VB script using cscript.

```
c:  
cd \  
cscript \\192.168.100.151\apps\shortcutscript.vbs  
exit
```

The administrator should modify the batch file to use the correct path for the network share. Normally, this would be the same shared directory specified in the Administration Settings area of Management Utilities as the default directory for deployment of client software and third party applications. In any case, the file deptsvc.exe must be present in this directory.

Save the batch file; in this example, the file will be saved as shortcutscript.bat. Copy the Visual Basic script file and the Windows batch file to the network share.

Browse to Management Utilities and create a Third Party Deployment Template. Enter a Template Name. Choose the Target Platform (Windows) and an authentication template with credentials to browse the network share and login to client machines with administrator privileges. Enter a path to the network share if necessary, then click the Find Files button to populate the drop down menu. Choose the Windows batch file. Click the Add button. Verify the correct batch file name and path, click Save Changes, then click OK to confirm.

Configure Template

Third Party Deployment Template

Define the paths to executables used to deploy third party software across your network.

Template Name:

Third Party Executable Details

Target Platform

Windows

Linux

Mac OS X

Select the authentication template with credentials to browse the network share entered below

Windows Admin ▾

Path to directory containing source executable file (depsvc.exe MUST be present in this directory):

Find Files... ▾

Timeout (minutes)

Parameters

Do not check for process completion (for BIOS updates)

Copy these files locally to each target machine before installing (required if the above is not a network share, cannot be used with local impersonation)

File Path	Parameters	Timeout	Ignore Process Completion	Platform
\\192.168.100.151\apps\shortcutscrip.bat		10	No	Windows Edit Delete

Existing Templates

Help Desk Desktop Shortcut

Notepad

New Template

Copy Remove

To verify that the files and template have been created correctly, deploy the shortcut on a single machine. Identify a machine in the Groups interface. Click the machine, then right-click and choose Deploy Third Party Software. Choose the templates, then click OK to define the job. Click the Run Job button, then click OK to confirm. When the job is finished, review the job status message, then go to the client machine to verify that the icon functions as desired. The files and template can now be used to create a job that deploys the icon on a group of machines, and if desired, it can be set up to run as a recurring job on a regular schedule.

Using a Custom Icon

The same procedure can be used to copy an icon file to the system and use it for the desktop shortcut. The icon file should be placed in the shared directory along with the Visual Basic script file and the Windows batch file.

Modify the Visual Basic script to specify the icon file in the local directory where it will be placed, changing this line:

```
oShellLink.IconLocation = StrPath1& "\\Internet Explorer\iexplore.exe,1"
```

to this:

```
oShellLink.IconLocation = "C:\customicon\custom.ico,0"
```

Before running the Visual Basic script, the Windows batch file should create a directory for the icon and copy the icon to the local directory. The resulting batch file should look like this:

```
c:  
cd \  
mkdir customicon  
copy \\192.168.100.151\apps\custom.ico c:\customicon  
cscript \\192.168.100.151\apps\shortcutsript.vbs  
exit
```

Change the name of the icon file and the local directory, and the location of the network share, from the ones given in our example to the correct ones for your installation.