A Quick Software Inventory Script

A simple script to determine installed software

by Bob Kelly

February 2007

Software inventory is a very common request in the world of desktop management. Most environments have some sort of asset management system in place that reports on many things including software. However, if you don't have such a system or if you want to base the action of a script on if certain software is installed the method discussed here will get you there.

Rather than digging through the file system looking for known file names or file version information, the best quick and easy way to get this job done is by enumerating the subkeys in the Uninstall subkey of the registry. By enumerate, I mean to get a list of each subkey at this level of the registry structure. Once we have this, we can look inside and dynamically build a list of what is installed by looking for some common registry values that describe the installed items. What we end up with will look suspiciously like what you'd see if you brought up the Add/Remove Programs (ARP) applet and that is because this script and the ARP are working of the same data.

While KiXtart and AutoIt are nice enough to provide nice enumeration functions for the registry, the Windows Script Host has left us hanging. Luckilly there is always more than one way to skin a cat and this time it is WMI that we'll call on to provide us this functionality. It uses hexadecimal values to identify what hive in the registry we are looking to, so we will start by setting a few constants (unchangeable variables) to make the script easier to read:

```
Const HKCR = &H80000000
Const HKCU = &H80000001
Const HKLM = &H80000002
```

Next, we need to get a handle on the WMI registry provider, which we do like so:

```
Set oReg = GetObject("winmgmts:{impersonationLevel=impersonate}!\\" &_
        ".\root\default:StdRegProv")
```

I know, it looks ugly, but that is what VBS demands of us sometimes. Now we will set the path we want to enumerate:

```
sUninstallPath = "SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall"
```

And finally we enumerate all the subkeys beneath this registry path in the HKEY_LOCAL_MACHINE hive and store them in a variable named aSubkeys:

```
oReg.EnumKey HKLM, sUninstallPath, aSubkeys
```

Now aSubkeys is an array, so we will run through each value it has using a For/Each loop. The GetStringValue method of the WMI registry provider is used here to build the path in the registry to each subkey of the Uninstall subkey, and inside this key we collect the value of “DisplayName” in a variable named sAppName. Finally we echo each one. You’ll note I’ve added the statement, “On Error Resume Next” here—this is to ensure the script does not error if it looks inside one of these subkeys and does not find a value named “DisplayName” (which does happen for some applications that don’t like to follow the rules).

```
On Error Resume Next
For Each sSubkey In aSubkeys
        oReg.GetStringValue HKLM, sUninstallPath &"\" & sSubkey, "DisplayName",
sAppName
        WScript.Echo sAppName
Next
```

Want to inventory a remote system? If you have admin access to read their registry remotely, you can replace the period in the line where we set oReg (which means local system) to the name of a remote system. In the example below, I inventory a computer named COMPUTERNAME…

```
Set oReg = GetObject("winmgmts:{impersonationLevel=impersonate}!\\" &_
        "COMPUTERNAME\root\default:StdRegProv")
```

There you go! Just a dozen lines of code (and you could easily reduce even this) and we have ourselves a nice little software inventory of the local machine. I thought this was a good way to cover how to enumerate subkeys in the registry with VBScript, and we have a nice useful example while were at it!

Bob Kelly is president and co-founder of AdminScriptEditor.com, home to an integrated suite of scripting tools and a shared library of scripts and language help. He has authored books on scripting and desktop administration as well as several white papers. Bob also owns and operates AppDeploy.com, where he writes and produces videos on topics related to software deployment. You can contact Bob about "A Quick Software Inventory Script" at bkelly@adminscripteditor.com.