

Re: Remove programs with a script?

Source:

<http://www.tech-archive.net/Archive/Scripting/microsoft.public.scripting.wsh/2007-05/msg00080.html>

- *From:* "Al Dunbar" <AlanDrub@xxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Thu, 24 May 2007 19:46:13 -0600
-

"Michael Harris (MVP)" <mikhar.at.mvps.dot.org> wrote in message news:%23rph%23wlnHHA.4220@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

William Stokes wrote:

I found this on running scripts on remote machine:

My purpose is to use the script below to pass "uninst_notes.bat" to a remote machine and run it there. While I execute this I get error:
Error: ActiveX component can't create object
Code: 800A01AD
Source: Microsoft VBScript runtime error

What could cause this? ...

- 1) See below for items for WshController success.
- 2) The script passed to CreateScript must be a WSH hosted script (vbs/js/wsf), not a bat/cmd hosted by cmd.exe.

My WshController/WshRemote remote scripting checklist:

You need WSH 5.6 on both sides (local and remote).

Both machines must be NT, W2K, XP (Pro, not Home), or higher.

The remote machine needs to have WshRemote enabled:

HKLM
 \Software
 \Microsoft
 \Windows Script Host

named string value: Remote

Re: Remove programs with a script?

Set Remote to "1" to enable and "0" to disable.

The remote machine needs to have DCOM enabled. You can run DCOMCNFG to check, but it's normally enabled by default.

You (meaning the account under which the local wshController script executes) must have administrator privileges on the remote machine.

If the remote machine is XP (pre-sp1 only) you probably need to run:

```
wscript.exe -regserver
```

to register wscript.exe as an out of process COM server. Original XP installs have a bug that don't do this registration.

The default DCOMCNFG configuration of WshRemote Security does not allow Launch permissions for ordinary domain users (unless logged on locally). It also runs by impersonating the launching user. Impersonation doesn't allow the WshRemote instance (i.e., your remotely executed script) to access other remote machine resources.

Scripts executed via WshController/WshRemote can't see mapped network drives, UNC paths, and in general can't access any kind of remote machine resources. That's simply a constraint of processes that run under an impersonated account.

2 possible solutions:

1) (*NOT RECOMMENDED* unless LAUNCH permissions remain restricted) Use DCOMCNFG on the remote machine and reconfigure the Identity of WshRemote to run under an explicit domain account/password.

2) (Recommended but untested) Change the calling script to prompt for username/password and pass them to the remote script as commandline arguments. Change the remote script to map a network drive on the fly using the credentials using an unused drive letter. Before exiting the remote script, unmap the drive. This is not strictly secure since credentials are passed in clear text over the wire.

WshRemote instances do not run in the context of the visible, interactive desktop of a logged on user, assuming there even is anyone logged on on the remote machine. In that Context, SendKeys will not work.

Re: Remove programs with a script?

In general, WshRemote instances should never attempt to produce any visible windows or dialog boxes or depend on any interactive user response. At best you will get a 'permission denied' on the attempt or it will silently fail. At worst the invisible instance will hang waiting for a user response it can never get.

The WshRemote instance is also constrained with respect to the network resources it can access. This is a normal limitation of NTLM and account impersonation. WshRemote instances, by default, run in a process that is impersonating the account of the launching user. Unless WshRemote is reconfigured via DCOMCNFG to use explicit domain account credentials, it cannot access remote machine resource *UNC paths, et.al.).

The WshRemote does not have the clear text credentials (username/password) to respond to an NT challenge/response from a remote machine (assuming no DCOMCNFG reconfiguration). This also applies UNC path access to the local machine.

Alternately, you could create another batch script that would run "uninst_notes.bat" on the other workstations using PSEXEC from the PSTools "SysInternals" package (see <http://www.microsoft.com/technet/sysinternals/default.mspx>). Whether or not this will work for you will depend to some extent on the content of the "uninst_notes.bat" file.

/AI

.