# How do I configure Windows Update programmatically?

devblogs.microsoft.com/oldnewthing/20140716-00

July 16, 2014

Raymond Chen

First of all, normal programs shouldn't be messing with Windows Update configuration. That's something the user (or the user's administrator) decides. If you're an IT administrator, then you can use Group Policy to configure Windows Update on your network.

But maybe you're a special case where the above remarks don't apply. Say you're a data center and all the systems are running inside of virtual machines and you don't want them installing updates or rebooting without your permission, so you want to run a script when you set up the image to disable updates.

You can use the `Microsoft.Update.AutoUpdate` object, known to native code as `IAutomaticUpdates`. Here's a script that prints your current update settings:

```
var AU = new ActiveXObject("Microsoft.Update.AutoUpdate");
var Settings = AU.Settings;
WScript.StdOut.WriteLine(Settings.NotificationLevel);
```

The notification levels are documented as `AutomaticUpdatesNotificationLevel`. If you want to change the notification level, you can update the level in the Settings object, and then save it.

```
var AU = new ActiveXObject("Microsoft.Update.AutoUpdate");
var Settings = AU.Settings;
Settings.NotificationLevel = 1; // aunlDisabled
Settings.Save();
```

All the various settings are documented in MSDN, though you have to dig through `IAutomaticUpdatesSettings`, `IAutomaticUpdatesSettings2`, and `IAutomaticUpdatesSettings3` to find them all.

Raymond Chen

**Follow**