Main Process Worknotes

Last Updated 07/08/22: Alex Farr

- Return to Index

Abstract

This article is intended to document the considerations and workflow behind the packaging process.

Questions to guide research:

- How do I check for Firefox's version? Mozilla Docs?
  - Firefox's version is going to be checked by SCCM most effectively. This should not be done in the PowerShell script itself.
- Where is the user's default homepage RegKey? Or how can this be managed through GPO? Autoconfig? What are the pros and cons of implementing each of them?
  - Firefox does not store a registry key for homepage. Group Policy, while effective, ultimately seems more involved than dropping a couple of autoconfig files into the install directory.
- What MSI flags will be needed to correctly perform the silent install/uninstall in Firefox's case?
  - No particular flags are needed for calls within the script itself. The `-NonInteractive` flag should be set while launching `Deploy-Application.exe`
- What changes need to be made to `Deploy-Application.ps1`?
  - See below.

Content

Implementation

After environment and workspace are set up, open `src\FirefoxESR\Deploy-Application.ps1` in VSCode and make the following changes.

Under the variable declaration section, modify lines 64-72 to the relevant application information for Firefox:

```
64: [string]$appVendor = 'Mozilla'
65: [string]$appName = 'Firefox ESR'
66: [string]$appVersion = '102.0'
67: [string]$appArch = 'x64'
. . .
71: [string]$appScriptDate = '07/08/2022'
72: [string]$appScriptAuthor = 'Alex Farr'
```

Add line 120 to act as a dev warning if the `NonInteractive` flag is not passed to the script during deployment:

```
120: If (-not $useDefaultMsi) { Show-InstallationPrompt -Message 'NOTE:
Without executing under NonInteractive there will be progress pop-ups from
Mozilla.' -ButtonRightText 'OK' -Icon Information -NoWait }
```

Modify line 123 in the preinstallation section, and line 153 in pre-uninstallation to attempt to close any currently active Firefox processes:

```
123: Show-InstallationWelcome -CloseApps 'firefox' -CloseAppsCountdown 60
```

```
153: Show-InstallationWelcome -CloseApps 'firefox' -CloseAppsCountdown 60
```

Under the installation section, remove the MSI autorun template materials and add line 132 to run the Firefox MSI with `Execute-MSI` and pass it the flags `-ms`:

```
132: Execute-MSI -Action Install -Path 'Firefox Setup 102.0esr.msi' -
Parameters '-ms'
```

Under the post-installation section, add lines 141 and 142 which use `Copy-File` to drop in the autoconfig files from `src\FirefoxESR\SupportFiles` into `C:\Program Files\Mozilla Firefox\` and `..\defaults\pref\` in order to force default the homepage to ung.edu:

```
141: Copy-File -Path 'autoconfig.js' -Destination "C:\Program Files\Mozilla
Firefox\defaults\pref"
```

```
142: Copy-File -Path 'ung.cfg' -Destination "C:\Program Files\Mozilla Firefox"
```

Modify line 161 in the uninstallation section to run the uninstall helper exe with `Execute-Process`:

```
161: Execute-Process -Path 'C:\Program Files\Mozilla Firefox\uninstall\helper.exe' -Parameters '/S'
```

Before testing, the autoconfig files need to be built. Create `autoconfig.js` and `ung.cfg` in `src\FirefoxESR\SupportFiles`. Edit `autoconfig.js` to contain the following:

```
pref("general.config.filename", "ung.cfg");
pref("general.config.obscure_value", 0);
```

Then modify `ung.cfg` to contain:

```
// Obligatory non-executable first line.
defaultPref("browser.startup.homepage", "data:text/plain,browser.startup.homepage=https://www.ung.edu");
```

Since PSADT has verbose logging, refer to the Firefox logs at `C:\Windows\Logs\Software` to verify that installation and uninstallation processes run correctly during testing.

Issues

The `Execute-Process` call for uninstallation on line 162 is using an absolute path to the helper exe which is less than ideal, however the PSADT `Execute-MSI` route referencing the packaged MSI fails to locate the installed application through registry checking, as does resolving the same literal path through the registry that PSADT should have searched for by using `Get-ItemPropertyValue` and then providing that to the `-Path flag of `Execute-MSI`.

Resources

- [Mozilla Docs]()

- File & Registry locations | Firefox Support Forum - Mozilla
- Setting the Default Firefox Homepage with Autoconfig
- Customizing Firefox Using AutoConfig