atomic-red-team/atomics/T1057/T1057.md at f339e7da7d05f6057fdfcdd3742bfcf365fee2a9 · redcanaryco/atomic-red-team · GitHub - 02/11/2024 10:11
https://github.com/redcanaryco/atomic-red-team/blob/f339e7da7d05f6057fdfcdd3742bfcf365fee2a9/atomics/T1057/T1057.md

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

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

T1057 - Process Discovery

Description from ATT&CK

> Adversaries may attempt to get information about running processes on a system. Information obtained could be used to gain an understanding of common software/applications running on systems within the network. Adversaries may use the information from [Process Discovery](https://attack.mitre.org/techniques/T1057) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions.
> In Windows environments, adversaries could obtain details on running processes using the Tasklist utility via cmd or `Get-Process` via PowerShell. Information about processes can also be extracted from the output of Native API calls such as `CreateToolhelp32Snapshot`. In Mac and Linux, this is accomplished with the `ps` command. Adversaries may also opt to enumerate processes via /proc.

Atomic Tests

- Atomic Test #1 - Process Discovery - ps
- Atomic Test #2 - Process Discovery - tasklist
- Atomic Test #3 - Process Discovery - Get-Process
- Atomic Test #4 - Process Discovery - get-wmiObject
- Atomic Test #5 - Process Discovery - wmic process

Atomic Test #1 - Process Discovery - ps

Utilize ps to identify processes.

Upon successful execution, sh will execute ps and output to /tmp/loot.txt.

Supported Platforms: macOS, Linux

auto_generated_guid: 4ff64f0b-aaf2-4866-b39d-38d9791407cc

Inputs:

Name	Description	Type	Default Value
output_file	path of output file	path	/tmp/loot.txt

Attack Commands: Run with `sh`!

- T1037.001
- T1037.002
- T1037.004
- T1037.005
- T1039
- T1040

```
ps >> #{output_file}
ps aux >> #{output_file}
```

Cleanup Commands:

```
rm #{output_file}
```

Atomic Test #2 - Process Discovery - tasklist

Utilize tasklist to identify processes.

Upon successful execution, cmd.exe will execute tasklist.exe to list processes. Output will be via stdout.

Supported Platforms: Windows

auto_generated_guid: c5806a4f-62b8-4900-980b-c7ec004e9908

Attack Commands: Run with `command_prompt`!

```
tasklist
```

Atomic Test #3 - Process Discovery - Get-Process

Utilize Get-Process PowerShell cmdlet to identify processes.

Upon successful execution, powershell.exe will execute Get-Process to list processes. Output will be via stdout.

Supported Platforms: Windows

auto_generated_guid: 3b3809b6-a54b-4f5b-8aff-cb51f2e97b34

Attack Commands: Run with `powershell`!

```
Get-Process
```

Atomic Test #4 - Process Discovery - get-wmiObject

Utilize get-wmiObject PowerShell cmdlet to identify processes.

Upon successful execution, powershell.exe will execute get-wmiObject to list processes. Output will be via stdout.

Supported Platforms: Windows

auto_generated_guid: b51239b4-0129-474f-a2b4-70f855b9f2c2

Attack Commands: Run with `powershell`!

```
get-wmiObject -class Win32_Process
```

Atomic Test #5 - Process Discovery - wmic process

Utilize windows management instrumentation to identify processes.

Upon successful execution, WMIC will execute process to list processes. Output will be via stdout.

Supported Platforms: Windows

auto_generated_guid: 640cbf6d-659b-498b-ba53-f6dd1a1cc02c

Attack Commands: Run with `command_prompt`!

```
wmic process get /format:list
```