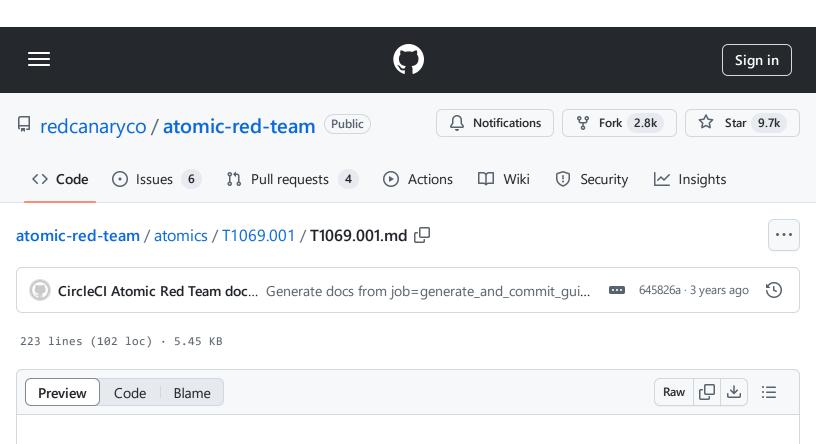
atomic-red-team/atomics/T1069.001/T1069.001.md at f339e7da7d05f6057fdfcdd3742bfcf365fee2a9 · redcanaryco/atomic-red-team · GitHub - 31/10/2024 14:58 https://github.com/redcanaryco/atomic-red-team/blob/f339e7da7d05f6057fdfcdd3742bfcf365fee2a9/atomics/T1069.001/T1069.001.md



T1069.001 - Local Groups

Description from ATT&CK

Adversaries may attempt to find local system groups and permission settings. The knowledge of local system permission groups can help adversaries determine which groups exist and which users belong to a particular group. Adversaries may use this information to determine which users have elevated permissions, such as the users found within the local administrators group.

Commands such as net localgroup of the Net utility, dscl. -list /Groups on macOS, and groups on Linux can list local groups.

Atomic Tests

- Atomic Test #1 Permission Groups Discovery (Local)
- Atomic Test #2 Basic Permission Groups Discovery Windows (Local)
- Atomic Test #3 Permission Groups Discovery PowerShell (Local)
- Atomic Test #4 SharpHound3 LocalAdmin

- Atomic Test #5 Wmic Group Discovery
- Atomic Test #6 WMIObject Group Discovery

Atomic Test #1 - Permission Groups Discovery (Local)

Permission Groups Discovery

Supported Platforms: macOS, Linux

auto_generated_guid: 952931a4-af0b-4335-bbbe-73c8c5b327ae

Attack Commands: Run with sh!

```
if [ -x "$(command -v dscacheutil)" ]; then dscacheutil -q group; else echo "dscacl if [ -x "$(command -v dscl)" ]; then dscl . -list /Groups; else echo "dscl is miss: if [ -x "$(command -v groups)" ]; then groups; else echo "groups is missing from the if [ -x "$(command -v id)" ]; then id; else echo "id is missing from the machine. If [ -x "$(command -v getent)" ]; then getent group; else echo "getent is missing cat /etc/group
```

Atomic Test #2 - Basic Permission Groups Discovery Windows (Local)

Basic Permission Groups Discovery for Windows. This test will display some errors if run on a computer not connected to a domain. Upon execution, domain information will be displayed.

Supported Platforms: Windows

auto_generated_guid: 1f454dd6-e134-44df-bebb-67de70fb6cd8

Attack Commands: Run with command_prompt!

net localgroup

net localgroup "Administrators"

Atomic Test #3 - Permission Groups Discovery PowerShell (Local)

Permission Groups Discovery utilizing PowerShell. This test will display some errors if run on a computer not connected to a domain. Upon execution, domain information will be displayed.

Supported Platforms: Windows

auto_generated_guid: a580462d-2c19-4bc7-8b9a-57a41b7d3ba4

Attack Commands: Run with powershell!

get-localgroup
Get-LocalGroupMember -Name "Administrators"

ιŌ

Atomic Test #4 - SharpHound3 - LocalAdmin

This module runs the Windows executable of SharpHound in order to remotely list members of the local Administrators group (SAMR)

Supported Platforms: Windows

auto_generated_guid: e03ada14-0980-4107-aff1-7783b2b59bb1

Inputs:

Name	Description	Туре	Default Value
domain	FQDN of the targeted domain	string	\$env:UserDnsDomain
sharphound_path	SharpHound Windows executable	path	\$env:TEMP\SharpHound.exe

Attack Commands: Run with powershell!

```
New-Item -Path "#{output_path}" -ItemType Directory > $null
& "#{sharphound_path}" -d "#{domain}" --CollectionMethod LocalAdmin --NoSaveCache
```

Cleanup Commands:

```
Remove-Item -Recurse #{output_path} -ErrorAction Ignore
```

Dependencies: Run with powershell!

Description: SharpHound binary must exist on disk and at specified location (#{sharphound_path}).

And the computer must be domain joined (implicit authentication).

Check Prereq Commands:

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

Get Prereq Commands:

```
Invoke-WebRequest "https://github.com/BloodHoundAD/BloodHound/blob/e062fe73d73c015 🚨
```

Atomic Test #5 - Wmic Group Discovery

Utilizing wmic.exe to enumerate groups on the local system. Upon execution, information will be displayed of local groups on system.

Supported Platforms: Windows

auto_generated_guid: 7413be50-be8e-430f-ad4d-07bf197884b2

Attack Commands: Run with powershell!

wmic.exe group get name

Q

Atomic Test #6 - WMIObject Group Discovery

Utilizing PowerShell cmdlet - get-wmiobject, to enumerate local groups on the endpoint. Upon execution, Upon execution, information will be displayed of local groups on system.

Supported Platforms: Windows

auto_generated_guid: 69119e58-96db-4110-ad27-954e48f3bb13

Attack Commands: Run with powershell!

Get-WMIObject Win32_Group

Q