T1518.001 - Software Discovery: Security Software Discovery

Description from ATT&CK

> Adversaries may attempt to get a listing of security software, configurations, defensive tools, and sensors that are installed on a system or in a cloud environment. This may include things such as firewall rules and anti-virus. Adversaries may use the information from [Security Software Discovery](https://attack.mitre.org/techniques/T1518/001) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions.
> Example commands that can be used to obtain security software information are netsh, `reg query` with Reg, `dir` with cmd, and Tasklist, but other indicators of discovery behavior may be more specific to the type of software or security system the adversary is looking for. It is becoming more common to see macOS malware perform checks for LittleSnitch and KnockKnock software.
>
> Adversaries may also utilize cloud APIs to discover the configurations of firewall rules within an environment.(Citation: Expel IO Evil in AWS) For example, the permitted IP ranges, ports or user

> accounts for the inbound/outbound rules of security groups, virtual firewalls established within AWS for EC2 and/or VPC instances, can be revealed by the `DescribeSecurityGroups` action with various request parameters. (Citation: DescribeSecurityGroups - Amazon Elastic Compute Cloud)

Atomic Tests

- Atomic Test #1 - Security Software Discovery
- Atomic Test #2 - Security Software Discovery - powershell
- Atomic Test #3 - Security Software Discovery - ps (macOS)
- Atomic Test #4 - Security Software Discovery - ps (Linux)
- Atomic Test #5 - Security Software Discovery - Sysmon Service
- Atomic Test #6 - Security Software Discovery - AV Discovery via WMI
- Atomic Test #7 - Security Software Discovery - AV Discovery via Get-CimInstance and Get-WmiObject cmdlets
- Atomic Test #8 - Security Software Discovery - Windows Defender Enumeration
- Atomic Test #9 - Security Software Discovery - Windows Firewall Enumeration

Preview Code Blame Raw

Atomic Test #1 - Security Software Discovery

Methods to identify Security Software on an endpoint

when sucessfully executed, the test is going to display running processes, firewall configuration on network profiles and specific security software.

Supported Platforms: Windows

auto_generated_guid: f92a380f-ced9-491f-b338-95a991418ce2

Attack Commands: Run with `command_prompt`!

```
netsh.exe advfirewall  show allprofiles
netsh.exe advfirewall firewall dump
netsh.exe advfirewall show currentprofile
netsh.exe advfirewall firewall show rule name=all
netsh.exe firewall show state
netsh.exe firewall show config
sc query windefend
powershell.exe /c "Get-Process | Where-Object { $_.ProcessName -eq 'Sysmon' }"
powershell.exe /c "Get-Service | where-object {$_.DisplayName -like '*sysm*'}"
powershell.exe /c "Get-CimInstance Win32_Service -Filter 'Description = ''System M
tasklist.exe
tasklist.exe | findstr /i virus
tasklist.exe | findstr /i cb
tasklist.exe | findstr /i defender
tasklist.exe | findstr /i cylance
tasklist.exe | findstr /i mc
tasklist.exe | findstr /i "virus cb defender cylance mc"
```

Atomic Test #2 - Security Software Discovery - powershell

Methods to identify Security Software on an endpoint

when sucessfully executed, powershell is going to processes related AV products if they are running. Note that, depending on the privilege of current user, get-process | ?{$_.Description -like "*"} may not return the processes related to AV products of the check. For instance, only with Administrator right, you can see the process description of McAffee processes. Hence, it is better to use get-process | ?{$_.ProcessName -like "*"}, if you know the name of those processes.

Supported Platforms: Windows

auto_generated_guid: 7f566051-f033-49fb-89de-b6bacab730f0

Attack Commands: Run with `powershell`!

```
get-process | ?{$_.Description -like "*virus*"}
get-process | ?{$_.Description -like "*carbonblack*"}
get-process | ?{$_.Description -like "*defender*"}
get-process | ?{$_.Description -like "*cylance*"}
```

```
get-process | ?{$_.Description -like "*mc*"}
get-process | ?{$_.ProcessName -like "*mc*"}
get-process | Where-Object { $_.ProcessName -eq "Sysmon" }
```

Atomic Test #3 - Security Software Discovery - ps (macOS)

Methods to identify Security Software on an endpoint when sucessfully executed, command shell is going to display AV/Security software it is running.

Supported Platforms: macOS

auto_generated_guid: ba62ce11-e820-485f-9c17-6f3c857cd840

Attack Commands: Run with `sh`!

```
ps aux | egrep 'Little\ Snitch|CbOsxSensorService|falcond|nessusd|santad|CbDefense
```

Atomic Test #4 - Security Software Discovery - ps (Linux)

Methods to identify Security Software on an endpoint when sucessfully executed, command shell is going to display AV/Security software it is running.

Supported Platforms: Linux

auto_generated_guid: 23b91cd2-c99c-4002-9e41-317c63e024a2

Attack Commands: Run with `sh`!

```
ps aux | egrep 'falcond|nessusd|cbagentd|td-agent|packetbeat|filebeat|auditbeat|os
```

Atomic Test #5 - Security Software Discovery - Sysmon Service

Discovery of an installed Sysinternals Sysmon service using driver altitude (even if the name is changed).

when sucessfully executed, the test is going to display sysmon driver instance if it is installed.

Supported Platforms: Windows

auto_generated_guid: fe613cf3-8009-4446-9a0f-bc78a15b66c9

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

```
fltmc.exe | findstr.exe 385201
```

Atomic Test #6 - Security Software Discovery - AV Discovery via WMI

Discovery of installed antivirus products via a WMI query.

when sucessfully executed, the test is going to display installed AV software.

Supported Platforms: Windows

auto_generated_guid: 1553252f-14ea-4d3b-8a08-d7a4211aa945

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

```
wmic.exe /Namespace:\\root\SecurityCenter2 Path AntiVirusProduct Get displayName /
```

Atomic Test #7 - Security Software Discovery - AV Discovery via

Get-CimInstance and Get-WmiObject cmdlets

Discovery of installed antivirus products via Get-CimInstance and Get-WmiObject cmdlets of powershell.

when sucessfully executed, information about installed AV software is displayed..

Supported Platforms: Windows

auto_generated_guid: 015cd268-996e-4c32-8347-94c80c6286ee

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

```
powershell Get-CimInstance -Namespace root/securityCenter2 -classname antiviruspro
powershell Get-WmiObject -Namespace root\securitycenter2 -Class antivirusproduct
```

Atomic Test #8 - Security Software Discovery - Windows Defender Enumeration

Windows Defender Enumeration via different built-in windows native tools. when sucessfully executed, information about windows defender is displayed.

Supported Platforms: Windows

auto_generated_guid: d3415a0e-66ef-429b-acf4-a768876954f6

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

```
Get-Service WinDefend #check the service state of Windows Defender
Get-MpComputerStatus #provides the current status of security solution elements, i
Get-MpThreat #threats details that have been detected using MS Defender
```

Atomic Test #9 - Security Software Discovery - Windows Firewall Enumeration

Enumerates windows firewall to retrieves firewall rules from the target computer.

when sucessfully executed, details of windows firewall is displayed.

Supported Platforms: Windows

auto_generated_guid: 9dca5a1d-f78c-4a8d-accb-d6de67cfed6b

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

```
Get-NetFirewallProfile | Format-Table Name, Enabled
Get-NetFirewallSetting
Get-NetFirewallRule | select DisplayName, Enabled, Description
```