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.
>
> On network devices, Network Device CLI commands such as `show processes` can be used to display current running processes.(Citation: US-CERT-TA18-106A)(Citation: show_processes_cisco_cmd)

Atomic Tests

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`!

```
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
```

Atomic Test #6 - Discover Specific Process - tasklist

Adversaries may use command line tools to discover specific processes in preparation of further attacks. Examples of this could be discovering the PID of lsass.exe to dump its memory or discovering whether specific security processes (e.g. AV or EDR) are running.

Supported Platforms: Windows

auto_generated_guid: 11ba69ee-902e-4a0f-b3b6-418aed7d7ddb

Inputs:

Name	Description	Type	Default Value
process_to_enumerate	Process name string to search for.	string	lsass

Attack Commands: Run with `command_prompt`!

```
tasklist | findstr #{process_to_enumerate}
```