Tactic: *Discovery: Process Discovery*

Usage

Process Discovery is used to **get information about running processes on a system**. The information can be used to gain knowledge about the common applications, programs, or software running on a machine. This is extremely valuable and important for adversaries as it gives them a better understanding of the machine they are attacking and to shape the malware they are using to match the machine's applications.

Prerequisites

Performing this technique is quite simple. First, the attacker must have at least user-level permissions for the system. It's important to also know the type of operating system the machine is using. If the machine is using Windows, the attacker can use various remote-access tools such as Native API using `CreateToolhelp32Snapshot`, Windows Management Instrumentation (WMI), or the PowerShell using `Get-Process` to inject malicious code and extract data about the computer, including the processes running. The attacker will use commands in the terminal to extract information about the computer's running processes.

Commands

ps

It's important to know the commands associated with process discovery. As Unix is a multi-tasking, multi-user operating system, *ps* is an important command. *ps* will show valuable information about the processes running on the machine. This includes the name, PID, memory usage, user, CPU usage time, etc.

tasklist

tasklist is a Windows command that is the equivalent to *ps* on Unix.

top

top is another important command that shows you the current state of the Unix system. It presents you a list of the top users of system resources, CPU, and memory.

Examples of commands

- **ps**
  - This is an example of *ps* being used MacOS.

```
  PID TTY           TIME CMD
39333 ttys000    0:00.05 -bash
27403 ttys001    0:00.01 /bin/bash --init-file /private/var/folders/1l/nrz6wpb5
24364 ttys002    0:00.02 /bin/bash --init-file /private/var/folders/1l/nrz6wpb5
24390 ttys003    0:00.01 /bin/bash --init-file /private/var/folders/1l/nrz6wpb5
```

- top

  - This is an example of *top* being used on MacOS. As you can see it lists all processes and their usages.

```
PID    COMMAND      %CPU TIME     #TH    #WQ   #PORT MEM    PURG   CMPRS  PGRP
141    WindowServer 41.2 06:53:21 19     8     2021  503M+  0B-    117M+  141
33247  plugin-conta 15.3 38:00.17 31     1     307   607M+  0B     532M-  6573
5822   Adobe CEF He 14.5 02:53:22 27     2     279-  326M-  0B     291M-  5787
5810   Adobe CEF He 12.9 02:32:49 11     1     168-  64M    1024K  17M    5787
0      kernel_task  7.2  03:28:47 203/8 0     0     822M-  0B     0B     0
44151  screencaptur 6.9  00:00.35 4      3     64    4204K+ 620K   0B     488
38137  Discord Help 6.5  61:20.95 45     1     680   523M-  0B     374M-  38123
44149  top          6.5  00:01.24 1/1    0     30+   5080K+ 0B     0B     44149
334    com.apple.Ap 3.0  21:54.90 3      2     253   3284K  0B     2456K  334
6573   librewolf    2.4  04:17:05 86     2     1125  1506M  0B     882M-  6573
5787   Creative Clo 1.1  30:27.58 25     3     387   49M    0B     33M    5787
44152  screencaptur 0.8  00:00.21 8      6     195   13M    48K    0B     44152
190    coreaudiod   0.8  01:45:55 11     3     1536  43M    0B     32M    190
30351  plugin-conta 0.7  15:48.01 30     1     123   292M   0B     254M-  6573
```

My Process

- Since I already have experience with the Unix terminal, practicing Process Discovery was not hard. First, I read the Mitre Att&ck Matrix article on Process Discovery, and then I began to research myself online. While researching, I came across important commands for Process Discovery that I already had used and heard of. Funnily enough, I am even making a clone of one of them in CIS3207. What I did learn from my research is that attackers can use remote-acess tools that interact with Native Windows API to easily extract data about a machine. This is important because many companies and organizations primarily use Windows, making it even easier for attackers to breach their networks through remote-access tools and lateral movement.