T1049 - System Network Connections Discovery

Description from ATT&CK

> Adversaries may attempt to get a listing of network connections to or from the compromised system they are currently accessing or from remote systems by querying for information over the network.
> An adversary who gains access to a system that is part of a cloud-based environment may map out Virtual Private Clouds or Virtual Networks in order to determine what systems and services are connected. The actions performed are likely the same types of discovery techniques depending on the operating system, but the resulting information may include details about the networked cloud environment relevant to the adversary's goals. Cloud providers may have different ways in which their virtual networks operate.(Citation: Amazon AWS VPC Guide)(Citation: Microsoft Azure Virtual Network Overview)(Citation: Google VPC Overview) Similarly, adversaries who gain access to network devices may also perform similar discovery activities to gather information about connected systems and services.
>
> Utilities and commands that acquire this information include netstat, "net use," and "net session" with Net. In Mac and Linux, netstat and `lsof` can be used to list current connections. `who -a` and `w` can be used to show which users are currently logged in, similar to "net session". Additionally, built-in features native to network devices and Network Device CLI may be used.(Citation: US-CERT-TA18-106A)

Atomic Tests

- Atomic Test #1 - System Network Connections Discovery
- Atomic Test #2 - System Network Connections Discovery with PowerShell
- Atomic Test #3 - System Network Connections Discovery Linux & MacOS
- Atomic Test #4 - System Discovery using SharpView

Atomic Test #1 - System Network Connections Discovery

Get a listing of network connections.

Upon successful execution, cmd.exe will execute `netstat`, `net use` and `net sessions`. Results will output via stdout.

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

Attack Commands: Run with `command_prompt`!

```
netstat
net use
net sessions
```

Atomic Test #2 - System Network Connections Discovery with PowerShell

Get a listing of network connections.

Upon successful execution, powershell.exe will execute `get-NetTCPConnection`. Results will output via stdout.

Supported Platforms: Windows

auto_generated_guid: f069f0f1-baad-4831-aa2b-eddac4baac4a

Attack Commands: Run with `powershell`!

```
Get-NetTCPConnection
```

Atomic Test #3 - System Network Connections Discovery Linux & MacOS

Get a listing of network connections.

Upon successful execution, sh will execute `netstat` and `who -a`. Results will output via stdout.

Supported Platforms: Linux, macOS

auto_generated_guid: 9ae28d3f-190f-4fa0-b023-c7bd3e0eabf2

Attack Commands: Run with `sh`!

```
netstat
who -a
```

Dependencies: Run with `sh`!

Description: Check if netstat command exists on the machine

Check Prereq Commands:

```
if [ -x "$(command -v netstat)" ]; then exit 0; else exit 1; fi;
```

Get Prereq Commands:

```
echo "Install netstat on the machine."; exit 1;
```

Atomic Test #4 - System Discovery using SharpView

Get a listing of network connections, domains, domain users, and etc.
sharpview.exe located in the bin folder, an opensource red-team tool. Upon successful execution, cmd.exe will execute sharpview.exe . Results will output via stdout.

Supported Platforms: Windows

auto_generated_guid: 96f974bb-a0da-4d87-a744-ff33e73367e9

Inputs:

Name	Description	Type	D
SharpView_url	sharpview download URL	Url	https://github.com/tevora-threat/SharpView/blob/b60456286b41bb055 raw=true
SharpView	Path of the executable opensource redteam tool used for the performing this atomic.	Path	PathToAtomicsFolder\T1049\bin\SharpView.e
syntax	Arguements method used along with SharpView to get listing of network connections, domains, domain users, and etc.	String	"Invoke-ACLScanner", "Invoke-Kerberoast", "F

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

```
$syntaxList = #{syntax}
foreach ($syntax in $syntaxList) {
#{SharpView} $syntax -}
```

Dependencies: Run with `powershell`!

Description: Sharpview.exe must exist on disk at specified location (#{SharpView})

Check Prereq Commands:

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

Get Prereq Commands:

```
New-Item -Type Directory (split-path #{SharpView}) -ErrorAction ignore |
Invoke-WebRequest #{SharpView_url} -OutFile "#{SharpView}"
```