

## Description from ATT&CK

## Atomic Tests

- ## Atomic Test #1 - System Network Connections Discovery

Page 1 of 3

Files

f339e7d

Go to file

> .github

> atomic\_red\_team

> atomics

- > Indexes
- > T1003.001
- > T1003.002
- > T1003.003
- > T1003.004
- > T1003.005

auto\_generated\_guid: 0940a971-809a-48f1-9c4d-b1d785e96ee5

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
```

atomic-red-team / atomics / T1049 / T1049.md ↑ Top

Preview | Code | Blame | 176 lines (87 loc) · 5.38 KB | Raw | Copy | Download | Menu

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;
```

Page 2 of 3

- >  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

## 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	Detailed Description
SharpView_url	sharpview download URL	Url	<a href="https://github.com/tevora-threat/SharpView/blob/b60456286b41bb055f9e7d0c8be532029eb32e96/SharpView.exe?raw=true">https://github.com/tevora-threat/SharpView/blob/b60456286b41bb055f9e7d0c8be532029eb32e96/SharpView.exe?raw=true</a>
SharpView	Path of the executable opensource redteam tool used for the performing this atomic.	Path	PathToAtomicsFolder\T1049\bin\SharpView.exe
syntax	Arguements method used along with SharpView to get listing of network connections, domains, domain users, and etc.	String	"Invoke-ACLScanner", "Invoke-Kerberoast", "Invoke-Mimikatz"

**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}"
```