T1560.001 - Archive via Utility

Description from ATT&CK

> Adversaries may use utilities to compress and/or encrypt collected data prior to exfiltration. Many utilities include functionalities to compress, encrypt, or otherwise package data into a format that is easier/more secure to transport.
> Adversaries may abuse various utilities to compress or encrypt data before exfiltration. Some third party utilities may be preinstalled, such as `tar` on Linux and macOS or `zip` on Windows systems. On Windows, `diantz` or `makecab` may be used to package collected files into a cabinet (.cab) file. `diantz` may also be used to download and compress files from remote locations (i.e. Remote Data Staging).(Citation: diantz.exe_lolbas) Additionally, `xcopy` on Windows can copy files and directories with a variety of options.
>
> Adversaries may use also third party utilities, such as 7-Zip, WinRAR, and WinZip, to perform similar activities.(Citation: 7zip Homepage)(Citation: WinRAR Homepage)(Citation: WinZip Homepage)

Atomic Tests

Atomic Test #1 - Compress Data for Exfiltration With Rar

An adversary may compress data (e.g., sensitive documents) that is collected prior to exfiltration. When the test completes you should find the txt files from the %USERPROFILE% directory compressed in a file called T1560.001-data.rar in the %USERPROFILE% directory

Supported Platforms: Windows

auto_generated_guid: 02ea31cb-3b4c-4a2d-9bf1-e4e70ebcf5d0

Inputs:

Name	Description	Type	Default Value
input_path	Path that should be compressed into our output file	Path	%USERPROFILE%
file_extension	Extension of files to compress	String	.txt
output_file	Path where resulting compressed data should be placed	Path	%USERPROFILE%\T1560.001-data.rar
rar_installer	Winrar installer	Path	%TEMP%\winrar.exe

rar_exe	The RAR executable from Winrar	Path	%programfiles%/WinRAR/Rar.exe

Attack Commands: Run with `command_prompt`!

```
"#{rar_exe}" a -r #{output_file} #{input_path}\*#{file_extension}
```

Cleanup Commands:

```
del /f /q /s #{output_file} >nul 2>&1
```

Dependencies: Run with `command_prompt`!

Description: Rar tool must be installed at specified location (#{rar_exe})

Check Prereq Commands:

```
if not exist "#{rar_exe}" (exit /b 1)
```

Get Prereq Commands:

```
echo Downloading Winrar installer
bitsadmin /transfer myDownloadJob /download /priority normal "https://www.win-rar.
#{rar_installer} /S
```

Atomic Test #2 - Compress Data and lock with password for Exfiltration with winrar

Note: Requires winrar installation rar a -p"blue" hello.rar (VARIANT)

Supported Platforms: Windows

auto_generated_guid: 8dd61a55-44c6-43cc-af0c-8bdda276860c

Inputs:

Name	Description	Type	Default Value
rar_installer	Winrar installer	Path	%TEMP%\winrar.exe
rar_exe	The RAR executable from Winrar	Path	%programfiles%/WinRAR/Rar.exe

Attack Commands: Run with `command_prompt`!

```
mkdir .\tmp\victim-files
cd .\tmp\victim-files
echo "This file will be encrypted" > .\encrypted_file.txt
"#{rar_exe}" a -hp"blue" hello.rar
dir
```

Dependencies: Run with `command_prompt`!

Description: Rar tool must be installed at specified location (#{rar_exe})

Check Prereq Commands:

```
if not exist "#{rar_exe}" (exit /b 1)
```

Get Prereq Commands:

```
echo Downloading Winrar installer
bitsadmin /transfer myDownloadJob /download /priority normal "https://www.win-rar.
#{rar_installer} /S
```

Atomic Test #3 - Compress Data and lock with password for Exfiltration with winzip

Note: Requires winzip installation wzzip sample.zip -s"blueblue" *.txt (VARIANT)

Supported Platforms: Windows

auto_generated_guid: 01df0353-d531-408d-a0c5-3161bf822134

Inputs:

Name	Description	Type	Default Value
winzip_exe	Path to installed Winzip executable	Path	%ProgramFiles%\WinZip\winzip64.exe
winzip_url	Path to download Windows Credential Editor zip file	Url	https://download.winzip.com/gl/nkln/winzip24-home.exe
winzip_hash	File hash of the Windows Credential Editor zip file	String	B59DB592B924E963C21DA8709417AC0504F6158CFCB12

Attack Commands: Run with `command_prompt`!

```
path=%path%;"C:\Program Files (x86)\winzip"
mkdir .\tmp\victim-files
cd .\tmp\victim-files
echo "This file will be encrypted" > .\encrypted_file.txt
"#{winzip_exe}" -min -a -s"hello" archive.zip *
dir
```

Dependencies: Run with `powershell`!

Description: Winzip must be installed

Check Prereq Commands:

```
cmd /c 'if not exist "#{winzip_exe}" (echo 1) else (echo 0)'
```

Get Prereq Commands:

```
if(Invoke-WebRequestVerifyHash "#{winzip_url}" "$env:Temp\winzip.exe" #{winzip_hash
  Write-Host Follow the installation prompts to continue
  cmd /c "$env:Temp\winzip.exe"
}
```

Atomic Test #4 - Compress Data and lock with password for Exfiltration with 7zip

Note: Requires 7zip installation

Supported Platforms: Windows

auto_generated_guid: d1334303-59cb-4a03-8313-b3e24d02c198

Inputs:

Name	Description	Type	Default Value
7zip_installer	7zip installer	Path	%TEMP%\7zip.exe
7zip_exe	Path to installed 7zip executable	Path	%ProgramFiles%\7-zip\7z.exe

Attack Commands: Run with `command_prompt`!

```
mkdir $PathToAtomicsFolder\T1560.001\victim-files
cd $PathToAtomicsFolder\T1560.001\victim-files
echo "This file will be encrypted" > .\encrypted_file.txt
"#{7zip_exe}" u archive.7z *txt -pblue
dir
```

Dependencies: Run with `command_prompt`!

Description: 7zip tool must be installed at specified location (#{7zip_exe})

Check Prereq Commands:

```
if not exist "#{7zip_exe}" (exit /b 1)
```

Get Prereq Commands:

```
echo Downloading 7-zip installer
bitsadmin /transfer myDownloadJob /download /priority normal "https://www.7-zip.or
#{7zip_installer} /S
```

Atomic Test #5 - Data Compressed - nix - zip

An adversary may compress data (e.g., sensitive documents) that is collected prior to exfiltration. This test uses standard zip compression.

Supported Platforms: Linux, macOS

auto_generated_guid: c51cec55-28dd-4ad2-9461-1eacbc82c3a0

Inputs:

Name	Description	Type	Default Value
input_files	Path that should be compressed into our output file, may include wildcards	Path	/var/log/{w,b}tmp
output_file	Path that should be output as a zip archive	Path	$HOME/data.zip

Attack Commands: Run with `sh`!

```
zip #{output_file} #{input_files}
```

Cleanup Commands:

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

Dependencies: Run with `sh`!

Description: Files to zip must exist (#{input_files})

Check Prereq Commands:

```
if [ $(ls #{input_files} | wc -l) > 0 ] && [ -x $(which zip) ] ; then exit 0; else
```

Get Prereq Commands:

```
(which yum && yum -y install epel-release zip)||(which apt-get && apt-get install
echo Please set input_files argument to include files that exist
```

Atomic Test #6 - Data Compressed - nix - gzip Single File

An adversary may compress data (e.g., sensitive documents) that is collected prior to exfiltration. This test uses standard gzip compression.

Supported Platforms: Linux, macOS

auto_generated_guid: cde3c2af-3485-49eb-9c1f-0ed60e9cc0af

Inputs:

Name	Description	Type	Default Value
input_file	Path that should be compressed	Path	$HOME/victim-gzip.txt
input_content	contents of compressed files if file does not already exist. default contains test credit card and social security number	String	confidential! SSN: 078-05-1120 - CCN: 4000 1234 5678 9101

Attack Commands: Run with `sh`!

```
test -e #{input_file} && gzip -k #{input_file} || (echo '#{input_content}' >> #{in
```

Cleanup Commands:

```
rm -f #{input_file}.gz
```

Atomic Test #7 - Data Compressed - nix - tar Folder or File

An adversary may compress data (e.g., sensitive documents) that is collected prior to exfiltration. This test uses standard gzip compression.

Supported Platforms: Linux, macOS

auto_generated_guid: 7af2b51e-ad1c-498c-aca8-d3290c19535a

Inputs:

Name	Description	Type	Default Value
input_file_folder	Path that should be compressed	Path	$HOME/$USERNAME
output_file	File that should be output	Path	$HOME/data.tar.gz

Attack Commands: Run with `sh`!

```
tar -cvzf #{output_file} #{input_file_folder}
```

Cleanup Commands:

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

Dependencies: Run with `sh`!

Description: Folder to zip must exist (#{input_file_folder})

Check Prereq Commands:

```
test -e #{input_file_folder}
```

Get Prereq Commands:

```
echo Please set input_file_folder argument to a folder that exists
```

Atomic Test #8 - Data Encrypted with zip and gpg symmetric

Encrypt data for exiltration

Supported Platforms: macOS, Linux

auto_generated_guid: 0286eb44-e7ce-41a0-b109-3da516e05a5f

Inputs:

Name	Description	Type	Default Value
test_folder	Path used to store files.	Path	/tmp/T1560
test_file	Temp file used to store encrypted data.	Path	T1560
encryption_password	Password used to encrypt data.	String	InsertPasswordHere

Attack Commands: Run with `sh`!

```
mkdir -p #{test_folder}
cd #{test_folder}; touch a b c d e f g
zip --password "#{encryption_password}" #{test_folder}/#{test_file} ./*
echo "#{encryption_password}" | gpg --batch --yes --passphrase-fd 0 --output #{tes
ls -l #{test_folder}
```

Cleanup Commands:

```
rm -Rf #{test_folder}
```

Dependencies: Run with `sh`!

Description: gpg and zip are required to run the test.

Check Prereq Commands:

```
if [ ! -x "$(command -v gpg)" ] || [ ! -x "$(command -v zip)" ]; then exit 1; fi;
```

Get Prereq Commands:

```
(which yum && yum -y install epel-release zip gpg)||(which apt-get && apt-get inst
```