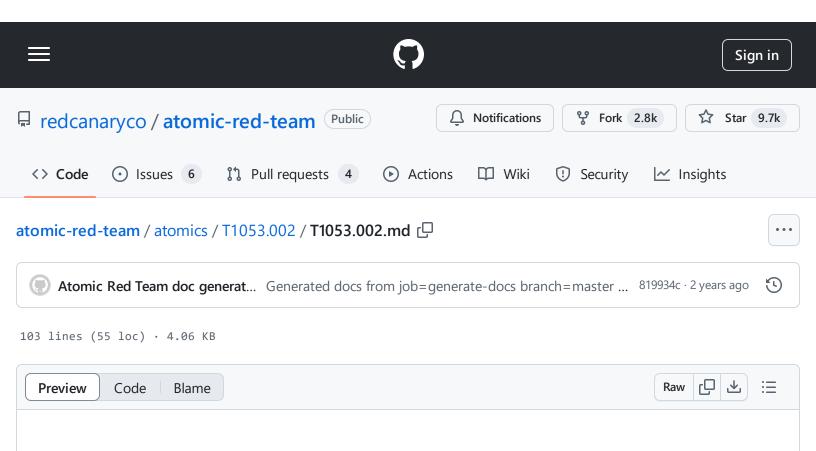
atomic-red-team/atomics/T1053.002/T1053.002.md at f339e7da7d05f6057fdfcdd3742bfcf365fee2a9 · redcanaryco/atomic-red-team · GitHub - 31/10/2024 14:51 https://github.com/redcanaryco/atomic-red-team/blob/f339e7da7d05f6057fdfcdd3742bfcf365fee2a9/atomics/T1053.002/T1053.002.md



# T1053.002 - At

# **Description from ATT&CK**

Adversaries may abuse the [at](https://attack.mitre.org/software/S0110) utility to perform task scheduling for initial or recurring execution of malicious code. The [at] (https://attack.mitre.org/software/S0110) utility exists as an executable within Windows, Linux, and macOS for scheduling tasks at a specified time and date. Although deprecated in favor of [Scheduled Task](https://attack.mitre.org/techniques/T1053/005)'s [schtasks] (https://attack.mitre.org/software/S0111) in Windows environments, using [at] (https://attack.mitre.org/software/S0110) requires that the Task Scheduler service be running, and the user to be logged on as a member of the local Administrators group.

On Linux and macOS, at may be invoked by the superuser as well as any users added to the at.allow file lf the at.allow file does not exist, the at.deny file is checked. Every username not listed in at.deny is allowed to invoke at. If the at.deny exists and is empty, global use of at is permitted. If neither file exists (which is often the baseline) only the superuser is allowed to use at.(Citation: Linux at)

atomic-red-team/atomics/T1053.002/T1053.002.md at f339e7da7d05f6057fdfcdd3742bfcf365fee2a9 redcanaryco/atomic-red-team · GitHub - 31/10/2024 14:51 https://github.com/redcanaryco/atomic-red-team/blob/f339e7da7d05f6057fdfcdd3742bfcf365fee2a9/atomics/T1053.002/T1053.002.md

Adversaries may use <u>at</u> to execute programs at system startup or on a scheduled basis for <u>Persistence</u>. <u>at</u> can also be abused to conduct remote <u>Execution</u> as part of <u>Lateral Movement</u> and/or to run a process under the context of a specified account (such as SYSTEM).

In Linux environments, adversaries may also abuse <u>at</u> to break out of restricted environments by using a task to spawn an interactive system shell or to run system commands. Similarly, <u>at</u> may also be used for <u>Privilege Escalation</u> if the binary is allowed to run as superuser via <u>sudo</u>.(Citation: GTFObins at)

### **Atomic Tests**

- Atomic Test #1 At.exe Scheduled task
- Atomic Test #2 At Schedule a job

### Atomic Test #1 - At.exe Scheduled task

Executes cmd.exe Note: deprecated in Windows 8+

Upon successful execution, cmd.exe will spawn at.exe and create a scheduled task that will spawn cmd at a specific time.

Supported Platforms: Windows

auto\_generated\_guid: 4a6c0dc4-0f2a-4203-9298-a5a9bdc21ed8

Attack Commands: Run with command\_prompt!

at 13:20 /interactive cmd

Q

## Atomic Test #2 - At - Schedule a job

This test submits a command to be run in the future by the at daemon.

Supported Platforms: Linux

auto\_generated\_guid: 7266d898-ac82-4ec0-97c7-436075d0d08e

#### Inputs:

Name	Description	Туре	Default Value
time_spec	Time specification of when the command should run	String	now + 1 minute
at_command	The command to be run	String	echo Hello from Atomic Red Team

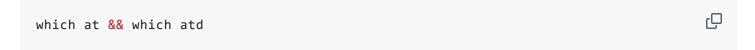
#### Attack Commands: Run with sh!

```
echo "#{at_command}" | at #{time_spec}
```

Dependencies: Run with sh!

Description: The at and atd executables must exist in the PATH

**Check Prereq Commands:** 



#### **Get Prereq Commands:**

```
echo 'Please install `at` and `atd`; they were not found in the PATH (Package name \Box
```

Description: The atd daemon must be running

#### **Check Prereq Commands:**

```
systemctl status atd || service atd status
```

#### **Get Prereq Commands:**

 $atomic-red-team/atomics/T1053.002/T1053.002.md\ at\ f339e7da7d05f6057fdfcdd3742bfcf365fee2a9\cdot redcanaryco/atomic-red-team\cdot GitHub$  - 31/10/2024 14:51 https://github.com/redcanaryco/atomic-red-team/blob/f339e7da7d05f6057fdfcdd3742bfcf365fee2a9/atomics/T1053.002/T1053.002.md

echo 'Please start the `atd` daemon (sysv: `service atd start` ; systemd: `systemc 🚨