# TTP Detail – T1562.003

## TTP Information

Name: Impair Command History Logging

Description: Adversaries may impair command history logging to hide commands they run on a compromised system. Various command interpreters keep track of the commands users type in their terminal so that users can retrace what they've done.

On Linux and macOS, command history is tracked in a file pointed to by the environment variable <code>HISTFILE</code>. When a user logs off a system, this information is flushed to a file in the user's home directory called <code>~/.bash\_history</code>. The <code>HISTCONTROL</code> environment variable keeps track of what should be saved by the <code>history</code> command and eventually into the <code>~/.bash\_history</code> file when a user logs out. <code>HISTCONTROL</code> does not exist by default on macOS, but can be set by the user and will be respected. The `HISTFILE` environment variable is also used in some ESXi systems.(Citation: Google Cloud Threat Intelligence ESXi VIBs 2022)

Adversaries may clear the history environment variable (<code>unset HISTFILE</code>) or set the command history size to zero (<code>export HISTFILESIZE=0</code>) to prevent logging of commands. Additionally, <code>HISTCONTROL</code> can be configured to ignore commands that start with a space by simply setting it to "ignorespace". <code>HISTCONTROL</code> can also be set to ignore duplicate commands by setting it to "ignoredups". In some Linux systems, this is set by default to "ignoreboth" which covers both of the previous examples. This means that “ ls” will not be saved, but “ls” would be saved by history. Adversaries can abuse this to operate without leaving traces by simply prepending a space to all of their terminal commands.

On Windows systems, the <code>PSReadLine</code> module tracks commands used in all PowerShell sessions and writes them to a file (<code>$env:APPDATA\Microsoft\Windows\PowerShell\PSReadLine\ConsoleHost\_history.txt</code> by default). Adversaries may change where these logs are saved using <code>Set-PSReadLineOption -HistorySavePath {File Path}</code>. This will cause <code>ConsoleHost\_history.txt</code> to stop receiving logs. Additionally, it is possible to turn off logging to this file using the PowerShell command <code>Set-PSReadlineOption -HistorySaveStyle SaveNothing</code>.(Citation: Microsoft PowerShell Command History)(Citation: Sophos PowerShell command audit)(Citation: Sophos PowerShell Command History Forensics)

Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to disable historical command logging (e.g. <code>no logging</code>).

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Kill Chain Phases

**•** mitre-attack: defense-evasion

## Malware

* BPFDoor
* Line Dancer

## Tools

* SILENTTRINITY

## APTs (Intrusion Sets)

* APT38
* Sea Turtle