Create or Modify System Process: Systemd Service, Sub-technique T1543.002 - Enterprise

Archived: 2026-04-05 16:57:46 UTC

Adversaries may create or modify systemd services to repeatedly execute malicious payloads as part of persistence. Systemd is a system and service manager commonly used for managing background daemon processes (also known as services) and other system resources.[1] Systemd is the default initialization (init) system on many Linux distributions replacing legacy init systems, including SysVinit and Upstart, while remaining backwards compatible.

Systemd utilizes unit configuration files with the `.service` file extension to encode information about a service's process. By default, system level unit files are stored in the `/systemd/system` directory of the root owned directories (`/`). User level unit files are stored in the `/systemd/user` directories of the user owned directories (`$HOME`).[2]

Inside the `.service` unit files, the following directives are used to execute commands:[3]

- `ExecStart` , `ExecStartPre` , and `ExecStartPost` directives execute when a service is started manually by `systemctl` or on system start if the service is set to automatically start.
- `ExecReload` directive executes when a service restarts.
- `ExecStop` , `ExecStopPre` , and `ExecStopPost` directives execute when a service is stopped.

Adversaries have created new service files, altered the commands a `.service` file's directive executes, and modified the user directive a `.service` file executes as, which could result in privilege escalation. Adversaries may also place symbolic links in these directories, enabling systemd to find these payloads regardless of where they reside on the filesystem.[4][5][6]

The `.service` file's User directive can be used to run service as a specific user, which could result in privilege escalation based on specific user/group permissions.

Systemd services can be created via systemd generators, which support the dynamic generation of unit files. Systemd generators are small executables that run during boot or configuration reloads to dynamically create or modify systemd unit files by converting non-native configurations into services, symlinks, or drop-ins (i.e., Boot or Logon Initialization Scripts).[7][8]
