Date: 24.07.2025



Copyright @ 2024 PibyThree.com All Rights Reserved

# Automate Service Management and File Operations using SaltStack States (.sls files)

Contents

[Automate Service Management and File Operations using SaltStack States (.sls files) 1](#_Toc204944253)

[1. Git Repository for all salt state files 2](#_Toc204944254)

[2. File Automation 2](#_Toc204944255)

[3. Service Automation 2](#_Toc204944256)

[Starting a service and stopping another service in Linux 2](#_Toc204944257)

[Stopping and starting the same service in Windows 2](#_Toc204944258)

# 1. Git Repository for all salt state files

This repository contains all the salt state files used for salt operations: [salt files](https://github.com/aryan-madhavi/saltstack-demo.git)

These files are to be stored in `/srv/salt`.

# 2. File Automation

This is a simple .sls file that creates a directory and initializes a file with default text.

The file is stored in the `files\_mgmt` directory and is named `init.sls`. When using this naming convention, you can apply the state without specifying the full filename:

$ salt ‘\*’ state.apply files\_mgmt

If the file were instead named files.sls and placed in the files\_mgmt directory, you would need to specify the filename (without the .sls extension) when applying the state:

$ salt ‘\*’ state.apply files\_mgmt.files

This naming convention applies for all files.

# 3. Service Automation

## Starting a service and stopping another service in Linux

$ salt ‘linux-minion\*’ state.apply linux\_srvc\_conf

Over here the `linux-minion\*` is a targeted expression that tells salt to apply this state to any minion name that starts with `linux-minion`. For eg. ‘linux-minion-1’, ‘linux-minion2’, ‘linux-minion-devl’, etc. Same applies for all similar targeted expressions.

Stopping and starting the same service in Windows (essentially restarting service)

$ salt ‘win-minion\*’ state.apply win\_srvc\_conf