

# SALTSTACK®

## SaltStack Enterprise II

### Training Syllabus

This is the second course in the SaltStack Enterprise training series. It builds on the concepts of the Salt Open Source Foundations and the SaltStack Enterprise I course by presenting additional topics above the fundamentals of Salt and SaltStack Enterprise administration. The topics are presented with scenario-based labs.

#### PREREQUISITES

Complete the Salt Open Source Foundations or the SaltStack Enterprise I training course.

#### OVERVIEW

This course begins by presenting the different aspects of deploying SaltStack minions. It presents SaltStack support for cloud control, custom deployments of Linux, Windows, and Salt Proxy Minions. Alternative topologies such as implementing master failover strategies. Salt Syndic and other hierarchical solutions are presented outlining the requirements, benefits, and caveats of each type.

Building on the fundamental administrative tasks presented in the Salt Open Source Foundations and the SaltStack Enterprise I course, this course shows how to address administrative tasks in large environments. It provides a detailed outline of various execution paths in the SaltStack architecture. Execution topics include Salt events, state and file rendering, and more. The security of a SaltStack implementation is covered and includes running as a non-root user, using external authentication, and troubleshooting SaltStack.

The course adds to configuration management concepts by outlining best practices for state formulas. There will be scenarios presenting the best use of templating with Jinja. The course also shows how to utilize the Salt state system with a deep-dive into the stages of execution. Salt orchestration scenarios will show how to automate enterprise-wide deployments. The content also includes information about creating a dynamic, event-driven infrastructure with Salt Reactors and Beacons.

#### CERTIFICATION

Completing Salt Open Source Foundations OR SaltStack Enterprise I, AND SaltStack Enterprise II will prepare an attendee to sit for the **SaltStack Certified Engineer (SSCE)** exam.

## OFFERING

All students are provided:

- A digital training manual (PDF)
- A Salt Cloud environment for hands-on labs during training

## FORMAT

This training is offered publically at SaltStack headquarters in Lehi, UT with virtual instructor-led (vILT) access for remote attendees on the five half-day format. When offered privately, on-site training follows a three-day format.

## COURSE TECHNICAL REQUIREMENTS

SaltStack training technical prerequisites include:

**Video conferencing** - If attending remotely, we typically use Zoom for remote attendees. If an alternative video conferencing technology is used, technical requirements will be provided.

**Labs** - Each student will be given a group of Linux and Windows virtual machines hosted in AWS. You will need to be able to browse to the SaltStack Enterprise console (port 443) from your network to access it.



## COURSE OUTLINE

### Introduction

- Welcome
- Objectives
- Getting Started
- Topics Covered
- Summary

### Salt Cloud

- Objectives
- What is Salt Cloud?
- Salt Cloud Components
- The salt-cloud Command
- Configuring Salt Cloud
- Querying Available Data
- Defining Virtual Machine Profiles
- Salt Cloud Options
- Creating VMs with Profiles
- The Salt Cloud Provisioning Process
- Salt Cloud Command Output
- Querying for VMs
- Destroying Virtual Machines in the Cloud
- Managing Multiple VMs Instances
- Provider-Specific Commands
- Salting Existing Systems
- The Class Setup
- Your Salt Lab Environment
- Summary
- Lab - Accessing the Lab Environment
- Lab - Use the EC2 Salt Cloud Driver
- Lab - Create and Destroy Cloud VMs
- Lab - Create a Cloud Map
- Lab - Add Existing Systems to the Salt Master
- Cleanup

### Windows Minions

- Objectives
- Salt Windows Minion Management
- The Salt Windows Management Framework
- Deploying Windows Minions
- User and Group Management
- Managing Windows Permissions
- Software Management
- Managing Windows Features and Roles
- Summary
- Lab - Provision Windows Minions
- Lab - Software Management
- Lab - Managing Windows Roles and Features

- Lab - Windows Updates

### Proxy Minions

- Objectives
- What is a Proxy Minion?
- Salt Proxy Minion Topology
- Using Proxy Minions
- Summary
- Lab - Provision Proxy Minion

### Salt Execution and Architecture

- Objectives
- Salt Execution
- Salt Components
- Calling Modules Locally on a Minion
- Sending Jobs to the Salt Master
- The Event System
- Salt Master Processes
- The "salt" Execution Architecture
- Summary
- Lab - View Master Processes
- Lab - Listening to Events
- Lab - Analyze Salt Activity
- Cleanup

### The Salt State System

- Objectives
- Uses of the Salt State System
- SLS Files
- Salt State Processing
- State Rendering
- State Compile
- State Runtime
- Summary
- Lab - Analyzing State Execution

### Salt Renderers

- Objectives
- Salt State Rendering
- Using the Python Renderer
- Using the Jinja Renderer
- Conditional "if" Statements
- Jinja Variable Assignments
- Leveraging Lookup Lists and Dictionaries
- Jinja Filters
- Calling Execution Modules with Jinja
- Using the GPG Render
- Summary



Lab - Create Users and Groups  
Management State

### **Building Salt Formulas**

Objectives  
Salt Formulas  
Salt State Formulas Best Practices  
Using Inline Pillar Data  
Use Lookup Dictionaries Based on Grains  
The Online Salt Formula Repository  
A Complete State Example  
Summary  
Lab - Create States with Salt Formulas  
Lab - Webstack: Web Server State  
Lab - Webstack: Configuring an HAProxy  
Load Balancer  
Lab - Webstack: Configuring the "top.sls"  
File

### **Salt Orchestration Formulas**

Objectives  
Running Jobs on the Salt Master  
Salt Orchestration  
Orchestration Declarations  
Using Inline Pillar with Orchestration  
Using State Modules in Orchestration  
The Salt Mine  
Summary  
Lab - Webstack: Create the Pillar Data for  
the Web Servers  
Lab - Webstack: Configuring Web Server  
Mine Functions  
Lab - Webstack: Creating the Web Servers  
Orchestration State  
Lab - Windows Orchestration

### **Reactors and Beacons Formulas**

Objectives  
Reactor System  
Beacons  
Reactor Best Practices

Summary  
Lab - Windows Beacons  
Lab - Windows Reactors  
Lab - Webstack: Configure the Load Beacon  
Lab - Webstack: Creating the Web Server  
Reactors  
Lab - Webstack: Triggering the Scale Up  
Event

### **Supporting and Troubleshooting Salt**

Objectives  
Introduction  
Viewing Salt Activity  
Network Settings  
Salt Keys  
Salt States  
Summary  
Overview  
Twelve Lab Troubleshooting Scenarios

### **Salt Security**

Objectives  
More About Keys  
The "pki\_dir"  
Increasing Security  
Decreasing Security  
Client ACLs  
External Authentication System  
Hardening Salt  
Security Issues  
Summary  
Lab - Understanding Salt Keys  
Lab - Configuring Client ACLs  
Lab - External Authentication  
Cleanup

### **Multi-Master**

Objectives  
Multi-Master  
Sharing Files Between Masters  
Summary



## CONTACT INFORMATION

**SaltStack training location**

2801 N. Thanksgiving Way, Suite 150  
Lehi, UT 84043

T +1 801.207.7440

**Email:**

training@saltstack.com

**Web:**

<https://saltstack.com/products/saltstack-training/>  
<https://www.saltstack.com/products/saltstack-services/>

