SUSE® Manager 3.2
Technical Comparison
SUSE Manager: The Smart Choice for Managing Linux

These days, IT departments manage highly dynamic and heterogeneous networks under constantly changing requirements. One important trend that has contributed to the growing complexity is the rise of software-defined infrastructures (SDIs). An SDI consists of a single pool of virtual resources that system administrators can manage efficiently and always in the same way, regardless of whether the resources reside on premises, or in the cloud.

SUSE® Manager is a powerful tool that brings the promise of SDI to Linux server management. You can use SUSE Manager to manage a diverse pool of Linux systems throughout the complete lifecycle, including deployment, configuration, auditing, and software management. This paper highlights some of the benefits of SUSE Manager and describes how SUSE Manager stacks up against other open source management solutions..

Introducing SUSE Manager
SUSE Manager is a single tool that enables IT staff to provision, configure, manage, and update all the Linux systems on the network in the same way, regardless of how and where they are deployed. From remote installation to cloud orchestration, automatic updates, custom configuration, compliance and security audits, SUSE Manager deftly handles the lifecycle of registered Linux clients.

A clean and efficient web interface (or, if you prefer, an equivalent command line one) provides a single entry point to all management actions, saving time and allowing a single admin to manage a greater share of network resources.

Discovering SUSE Manager
The diversity of Linux systems can add complexity to the management environment. Time spent managing a large, complex Linux estate with dissimilar tools adds significantly to costs. Your IT staff can be much more efficient with a single tool to automate, coordinate, and monitor Linux operations.

SUSE Manager provides unified management of Linux systems regardless of whether the system is running on bare metal, a virtual machine, or a container environment in a server room, private cloud, or public cloud. SUSE Manager will even manage Linux systems running on IoT devices, including legacy devices where agents cannot be installed. The ZeroMQ protocol provides parallel communication with client systems, which scales much more efficiently than alternatives that talk to each client one at a time.

SUSE Manager is tightly integrated with SUSE Linux Enterprise. You can also use it to administer Red Hat, CentOS, OEL, and other RPM-based systems. The SUSE Manager client-side agent is written in Python and is therefore quite portable. Accompanying APIs allow easy integration with third-party tools.
SUSE Manager consolidates all the following management tasks into a single tool:

- **Deployment**—Declare how many Linux systems you need and what you need them for, and SUSE Manager does the rest. Admins can build their own ISO images for bare metal, containers, or VMs, using either AutoYaST or Kickstart, and installation can be in attended or fully unattended fashion. Integration with the Cobbler integration server allows for efficient deployment using PXE (Pre-Execution Environment).

- **Software updates**—SUSE Manager automates software updates for whole systems or individual packages. A powerful security system guarantees that every package is centrally authorized. You can schedule and execute multiple software updates at once, using one command.

- **Configuration management**—SUSE Manager supports file-based configuration, as well as state-based configuration management, using Salt. The configuration and provisioning tools included with SUSE Manager enable you to define system prototypes and then adapt prototype definitions for easy automation and complex environments.

- **Security**—SUSE Manager supports automatic, system-wide configuration and vulnerability scans, using either CVE lists or the OpenSCAP protocol.

- **Compliance monitoring**—SUSE Manager creates a unified inventory of all systems within the organization, reporting (see Figure 1) on any deviation from configuration or security requirements and eliminating “shadow IT” activities from uncontrolled or undocumented systems.

An intuitive GUI interface offers a complete view of the network at a glance, including features such as “Formulas with Forms” (see Figure 2), that make SUSE Manager the ideal tool for consistent, highly efficient management of hundreds or thousands of servers. Expert Linux and Unix admins who prefer to work at the command line will find a rich set of text-based commands. The SUSE command-line tool “spacecmd” makes it easy to integrate SUSE Manager functions into admin scripts and homegrown utilities. And, SUSE Manager is ready for Nagios-compatible monitoring with Icinga.

A sensible security system enables you to distribute Linux administration work among the staff, according to each employee’s skills and responsibilities. The main administrator of a SUSE Manager server can delegate operations to users at different levels, creating accounts for tasks such as key activation, images, configuration, and software distribution.
A fully open source development model improves code quality and prevents vendor lock-in. The upstream project for SUSE Manager, Uyuni, is 100% Open Source. The software (whose architecture is shown in Figure 3) is developed in the open, on GitHub, with frequent releases and solid, automated testing. Although Uyuni is not commercially supported by SUSE and does not receive the same rigid QA and product lifecycle guarantees, it is not stripped down in any way. Unlike other vendors, whose commercial products heavily rely on extra features not available in the basic, open source version, SUSE keeps the same, full feature set available in both the community-based and subscription-based variants.

Adopting SUSE Manager, or migrating to it, does not mean that you should abruptly renounce your previous configuration management systems. For instance, SUSE Manager may act as an External Node Qualifier (i.e., configuration database) for Puppet or Chef.

**Salt on the Inside**

SUSE supports the powerful Salt configuration management system. Salt is state-based. A client agent, known as a Salt “minion,” can find the Salt master without the need for additional configuration (see Figure 4). If the client does not have an agent, Salt is capable of acting in "agentless" mode, sending Salt-equivalent commands through an SSH connection. The ability to operate in agent or agentless mode is an important benefit for a diverse network.

SUSE Manager extends the automatic configuration capabilities of Salt through its support for action chains. Action chains are sequences of actions that are executed as if they were a single command. An action chain makes it possible to use a single command to specify a complex task that consists of several steps. Examples of chainable actions include rebooting the system (even between other configuration steps of the same system!), installing or updating software packages, and building system images.
Comparing the Alternatives

SUSE Manager is one of several open source tools that inhabit the Linux space. Although the benefits of each depend on the details of your network and the needs of your organization, the following analysis offers a quick look at how SUSE Manager compares with the competition.

SUSE Manager vs. Puppet

The Puppet cross-platform orchestration tool comes in an open source version, as well as in a commercially supported Enterprise edition (which however is not entirely open source, however).

By default, Puppet requires an agent on each client, which adds complexity and additional effort to configuration and rollout for new systems. In the original Puppet working mode, changes are not implemented immediately. Instead, they are implemented the next time the agent asks the server for an update—that is, after an interval configured by the administrator. Overcoming these limits is the task of newer, add-on tools such as Puppet Tasks and Puppet Bolt, which are not yet fully integrated with the rest of the Puppet environment. SUSE Manager can natively operate in both declarative mode (“this is what the system should look like”) and imperative (“do this now”) mode, with or without agents, thus allowing more versatile and direct control of the managed environment.

Puppet configuration directives require advanced knowledge of the custom Domain Specific Language (DSL). Many of the advanced Puppet features required for full functionality are found in additional modules, either from the official Puppet Forge website or from the larger Puppet Community. Interaction of modules from independent developers can add complexity and lead to uncertainty or unpredictability in long-term support. Regardless of modules issues, several advanced tasks still demand input from the command line, even in Puppet Enterprise.

Puppet's support for containers is limited, making it more difficult to manage bare metal, VMs, containers, and cloud instances through a single interface. The Puppet model relies on users setting up their machines before bringing them under Puppet's control, thus reducing the possibilities for automation.

To summarize, Puppet has a significantly higher learning curve than SUSE Manager, and Puppet requires users to spend more time configuring the system in order to achieve an equivalent level of functionality.

SUSE Manager vs. Chef

Chef is a cross platform, open source tool that is also available in a commercial version called Chef Automate. Like Puppet, Chef requires an agent on each node, and the “recipes” used to define client configurations require developer-level knowledge of Ruby-based DSL.

By default, a Chef installation requires an agent on each managed node. The same installation, in addition to the server that directly serves the recipes to those agents, also needs a separate, dedicated machine (called Chef Workstation). The purpose of the Workstation is to host all the configuration recipes, which are first tested there and then pushed to the central Chef server. A Chef Workstation can apply configuration updates directly over SSH, and the web interface of Chef Automate supports agentless compliance scans. However, the correct interaction of Chef server, Workstation, and nodes is difficult to understand for beginners and requires a lot of initial setup, supported by preliminary study.

Many of the advanced features required for a comparison with SUSE Manager are only available in the commercial Chef Automate edition. For instance, the separate tools for compliance management (InSpec) and application management (Habitat) are only integrated in the commercial version of Chef, whereas these capabilities are fully integrated into the basic, upstream version of SUSE Manager.

SUSE Manager vs. Ansible

The Ansible management tool puts the emphasis on simplicity. It Ansible is best suited for small and relatively simple infrastructures. Part of its simplicity is that, unlike other similar products, it has no notion of state and does not track dependencies. The tool simply executes a series of tasks and stops whenever it fails or encounters an error. When the administrator provides a playbook to Ansible, Ansible compiles it and uses SSH to send the commands to the computers under its control, one at a time.

In small organizations, the performance impact is typically unnoticed, but as the size of the network increases, performance can degrade—and in some cases, commands or upgrades might fail.

In general, this stateless design makes it more difficult for Ansible to execute complex assignments and automation steps. Ansible’s playbooks are easier to create and implement than the DSL rules used with Puppet or Chef, but the YAML markup language used with Ansible is not as versatile. And although
Ansible is written in Python, it does not offer a Python API to support advanced customization and interaction with other products. Ansible also does not provide compliance management or a central directory of the systems it manages.

Ansible has a graphical interface called AnsibleWorks, or AWX for short, which is not as mature as those of SUSE Manager. The commercial version, called Ansible Tower, consists of selected releases of Ansible/AWX, hardened for long-term supportability. As it happens with the other SUSE Manager competitors, not all of the extra features in Ansible Tower are available under open source licenses.

**SUSE Manager and SaltStack**

The SaltStack orchestration and configuration tool comes in an open source edition, as well as through the SaltStack Enterprise commercial version. Like SUSE Manager, SaltStack uses the Salt configuration engine for managing installation and configuration services.

SUSE Manager offers many more features than the open source version of SaltStack. For instance, SUSE Manager supports both state definition and dynamic assignment of configuration via groups through the web interface, as well as offering auditing and compliance features that aren’t available in the open source SaltStack edition.

Like SUSE Manager, SaltStack Enterprise is an enterprise-level management tool based on the Salt configuration engine. In many ways, SaltStack Enterprise is the most similar to SUSE Manager of all the tools described in this paper, so the choice might depend on cost or the details of your environment.

Users who prefer to operate from the command line might prefer SUSE Manager because of its sophisticated command-line interface. And, of course, networks with a large investment in SUSE Linux will appreciate SUSE Manager’s tight integration with the SUSE environment.

In other cases, the choice between SUSE Manager and SaltStack Enterprise might depend on cost, the size of your network, the web interface that best matches your workflow, and other factors. Keep in mind that SUSE Linux is an ideal platform for supporting SaltStack Enterprise. SaltStack Enterprise is meant to serve as a master of Salt masters, a role that doesn’t conflict with SUSE Manager, so it is very possible for the tools to coexist without conflict.

If you are using SaltStack now and want to continue using it, the experts at SUSE can help you with a plan for how to integrate SaltStack with SUSE Manager and the SUSE Linux environment.

**Conclusion**

SUSE Manager provides a single, full-featured interface for managing the whole lifecycle of Linux systems in a diverse network environment, either from a clean graphical interface (see Figure 5) or from the command line. You can manage bare metal, virtual systems, and container-based systems within the same convenient tool, attending to tasks such as deployment, provisioning, software updates, security auditing, and configuration management. The flexible Salt configuration system allows for convenient configuration definition and easy automation, and it is capable of acting in agent or agentless mode. In these ways, SUSE Manager greatly reduces the complexity and risks of dealing with highly dynamic Linux infrastructures and operations.

Unlike several of its competitors, SUSE offers the full feature set of SUSE Manager through its upstream, community-based development project, Uyuni—thus preventing lock-in, simplifying evaluation, and maximizing the benefits of open source development.

![Figure 4.](image-url)
Strong support for customization and complex configurations, along with the ease and convenience of a single-source management solution, make SUSE Manager a powerful option for managing Linux systems in a diverse, enterprise environment.

Talk to the experts at SUSE for more on how you can scale down overhead and scale up efficiency by adding SUSE Manager to your Linux network environment.