



Architecture Guide

Uyuni 4.0

March 16, 2019



Table of Contents

GNU Free Documentation License	1
Introduction	8
Component Legend	9
Types of Components	9
Salt Architecture	11
Salt Architecture	11
Core Salt Components	11
Salt Data and Uyuni	11
Salt Contact Methods	12
Salt Pull	13
Salt SSH Push	13
Salt SSH Push & Tunnel	15
Boostrapping UI	15
Onboarding and Registration	15
Traditional Architecture	16
Traditional Architecture	16
Traditional Contact Methods	16
Traditional Contact Method (rhnsd)	16
Traditional Contact Method (osad)	18
Traditional SSH Push	20
Traditional SSH Pull	21
Repositories	22
Repository Types	22
Pool Repositories	22
Devel Repositories	22
Tool Repositories	22
Maintenance Repositories	22
Repository Synchronization	22
Component Index	23
Introduction	23
Apache	23
Apache Tomcat	23
Python XMLRPC Server	24
Taskomatic	24
Database	25
mgr-sync	25
spacewalk-repo-sync	26
osa-dispatcher	27
jabberd	28
mgr_check	28
zypp-plugin-spacewalk	29
rhnsd	30
osad	30
salt-master	31
salt-api	32
salt-minion	33

DRAFT

GNU Free Documentation License

Copyright © 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA. Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections

then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

-
- D. Preserve all the copyright notices of the Document.
 - E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
 - F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
 - G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
 - H. Include an unaltered copy of this License.
 - I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
 - J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
 - K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
 - L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
 - M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
 - N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
 - O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

Copyright (c) YEAR YOUR NAME.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Component Legend

These diagram components will be used in the following sections explaining the architecture of Uyuni. Components in Uyuni can communicate in three ways:

- One way
- Two way
- Scheduled (time based)

Types of Components

One Way

Components that communicate in only one direction are represented by:

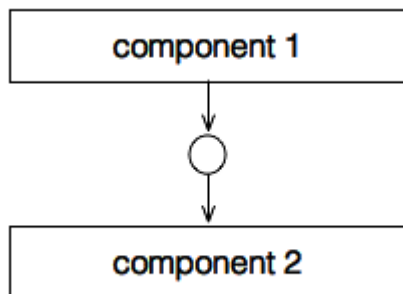


Figure 1. One way communication between components

Two Way

Components that communicate in both directions are represented by:

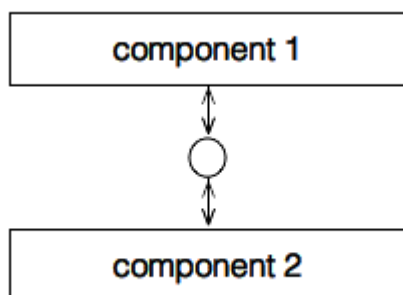


Figure 2. Two way communication between components

Database Connections

A component that reads and writes to the database communicates in both directions and is represented by:

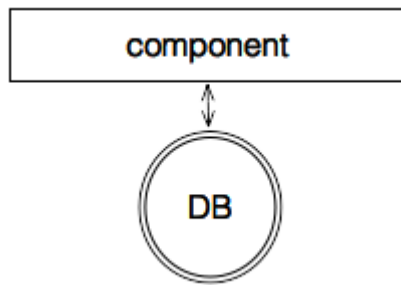


Figure 3. Two way communication between a component and the database(read and write)

Scheduled (Time based)

Components that run on a schedule are represented by:



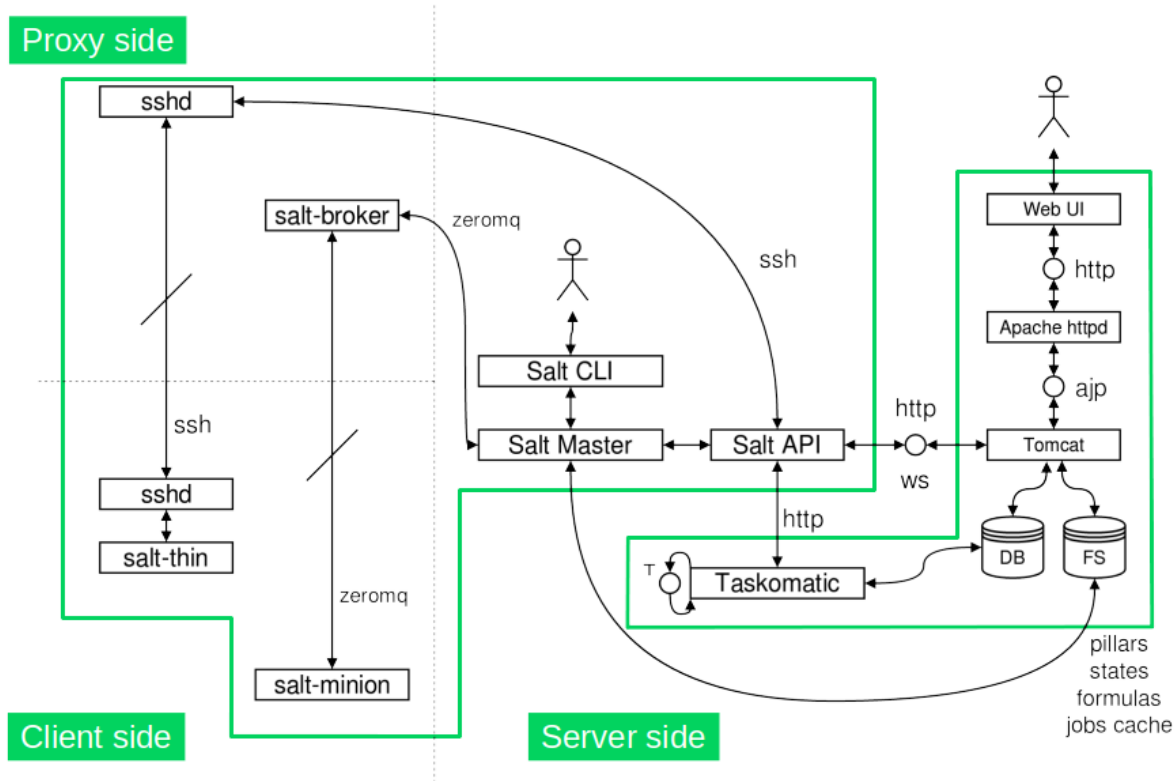
Figure 4. Component that runs on a schedule

Salt Architecture

Salt Architecture

Some description...

Salt Stack Diagram



Core Salt Components

Comming soon...

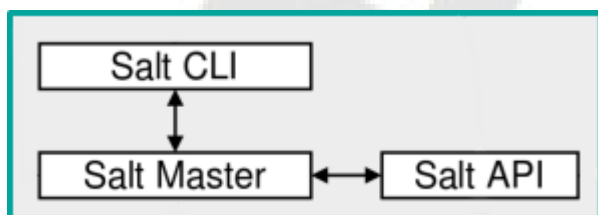


Figure 5. Salt Core

Salt Data and Uyuni

Salt Global Static Data

Global static data **should not** be customized or edited by Uyuni users. This data is generated by the server.

Table 1. Salt Global Static Data

Directory	Function
/usr/share/susemanager/salt/	custom modules, states, grains
/usr/share/susemanager/pillar_data/	global pillar data
/usr/share/susemanager/formulas/	formulas

Generated Data Per Minion

Generated data for minions **should not** be customized or edited by Uyuni users. This data is generated by the server.

Table 2. Salt Generated Data Per Minion

Directory	Function
/srv/susemanager/pillar_data/	custom modules, states, grains
/usr/share/susemanager/pillar_data/	global pillar data
/srv/susemanager/formulas_data/	formulas

Custom Salt Data

The following directories are reserved for use by users **and should be** customized and edited by Uyuni users. The custom salt data place here will be calculated and combined with the content generated listed above when running a highstate.

Table 3. Salt Generated Data Per Minion

Directory	Function
/srv/salt/	user defined custom modules, states, grains
/srv/pillar/	user defined global pillar data
/srv/formula_metadata	user defined formulas

Salt Contact Methods

Choosing a Contact Method for Salt

Uyuni provides several methods for communication between client and server. All commands that the Uyuni server sends to its clients will be routed through one of these contact methods.

The contact method you select for Salt will depend on your network infrastructure. The following sections provide a starting point for selecting a method which best suits your network environment.

- [Salt Pull](#)

- Salt SSH Push
- Salt SSH Push and Tunnel

Salt Pull

Salt SSH Push

Salt SSH Push is intended to be used in environments where your Salt clients cannot reach the Uyuni server directly to regularly checking in and, for example, fetch package updates.



Push via SSH

This feature is not related to Push via SSH for the traditional clients. For Push via SSH, see [xref:bp.contact.methods.ssh.push\[Salt SSH Push\]](#).

Overview

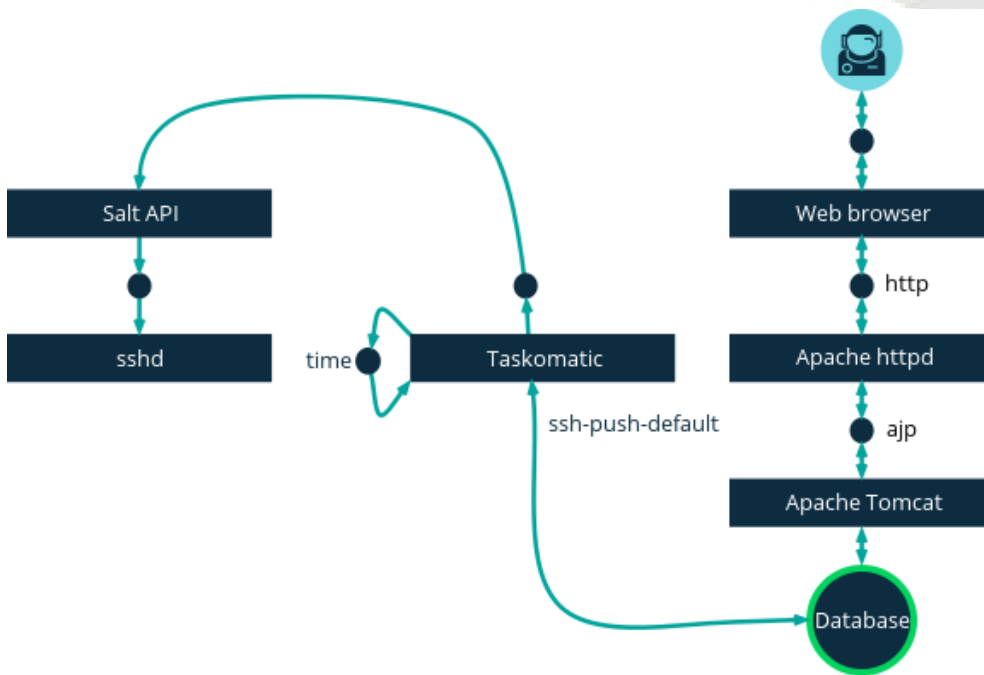


Figure 6. Push via Salt SSH Contact Method

Salt provides “Salt SSH” ([salt-ssh](#)), a feature to manage clients from a server. It works without installing Salt related software on clients. Using Salt SSH there is no need to have minions connected to the Salt master. Using this as a Uyuni connect method, this feature provides similar functionality for Salt clients as the traditional Push via SSH feature for traditional clients.

This feature allows:

- Managing Salt entitled systems with the Push via SSH contact method using Salt SSH.
- Bootstrapping such systems.

Requirements

- SSH daemon must be running on the remote system and reachable by the `salt-api` daemon (typically running on the Uyuni server).
- Python must be available on the remote system (Python must be supported by the installed Salt). Currently: python 2.6.



Unsupported Systems

Red Hat Enterprise Linux and CentOS versions ≤ 5 are not supported because they do not have Python 2.6 by default.

Bootstrapping

To bootstrap a Salt SSH system, proceed as follows:

1. Open the **Bootstrap Minions >]** dialog in the Web UI (menu:Systems[Bootstrapping]).
2. Fill out the required fields. Select an **Activation Key >]** with the menu:Push via SSH[contact method configured. For more information about activation keys, see: xref:ref.webui.systems.activation-keys.
3. Check the **Manage system completely via SSH** option.
4. Confirm with clicking the **Bootstrap** button.

Now the system will be bootstrapped and registered in Uyuni. If done successfully, it will appear in the **Systems** list.

Configuration

There are two kinds of parameters for Push via Salt SSH:

- Bootstrap-time parameters - configured in the **Bootstrapping** page:
 - Host
 - Activation key
 - Password - used only for bootstrapping, not saved anywhere; all future SSH sessions are authorized via a key/certificate pair
- Persistent parameters - configured Uyuni-wide:
 - sudo user - same as in bp.contact.methods.ssh.push.sudo.

Action Execution

The Push via Salt SSH feature uses a taskomatic job to execute scheduled actions using `salt-ssh`. The taskomatic job periodically checks for scheduled actions and executes them. While on traditional clients with SSH push configured only `rhncertcheck` is executed via SSH, the Salt SSH push job executes a

complete `salt-ssh` call based on the scheduled action.

Known Limitation

- OpenSCAP auditing is not available on Salt SSH minions.
- Beacons do not work with Salt SSH.
 - Installing a package on a system using `zypper` will not invoke the package refresh.
 - Virtual Host functions (for example, a host to guests) will not work if the virtual host system is Salt SSH-based.

For More Information

For more information, see

- https://wiki.microfocus.com/index.php/SUSE_Manager/SaltSSHServerPush
- <https://docs.saltstack.com/en/latest/topics/ssh/>

Salt SSH Push & Tunnel

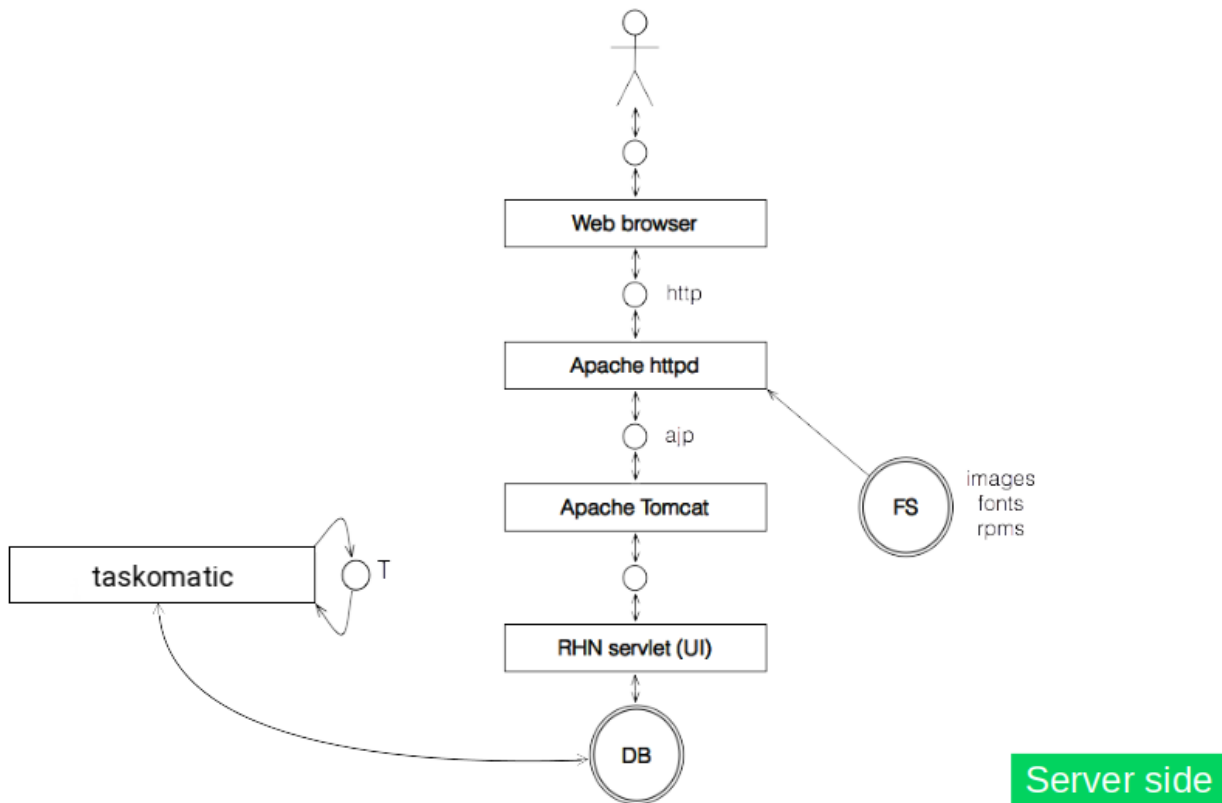
Boostrapping UI

Onboarding and Registration

Traditional Architecture

Traditional Architecture

The Traditional Stack



Traditional Contact Methods

Selecting a Contact Method

Uyuni provides several methods for communication between client and server. All commands that the Uyuni server sends to its clients will be routed through one of these contact methods.

The contact method you select will depend on your network infrastructure. The following sections provide a starting point for selecting a method which best suits your network environment.

Traditional Contact Method (rhnsd)

The Default Contact Method

The Uyuni **rhnsd** daemon runs on client systems and periodically connects with Uyuni to check for new updates and notifications. The daemon, which runs in the background, is started by **rhnsd.service**. By default, it will check every 4 hours for new actions, therefore it may take some time for your clients to begin updating after actions have been scheduled for them.

To check for updates, **rhnmd** runs the external **mgr_check** program located in `/usr/sbin/`. This is a small application that establishes the network connection to Uyuni. The SUSE Manager daemon does not listen on any network ports or talk to the network directly. All network activity is done via the **mgr_check** utility.



Auto accepting (EULAs)

When new packages or updates are installed on the client using Uyuni, any end user licence agreements (EULAs) are automatically accepted. To review a package EULA, open the package detail page in the Web UI.

This figure provides an overview of the default **rhnmd** process path. All items left of the **Python XMLRPC server** block represent processes running on an Uyuni client.

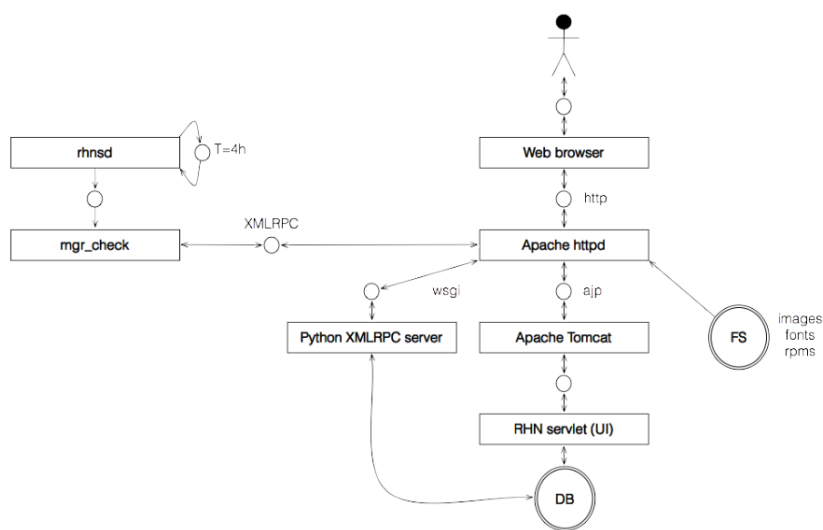


Figure 7. *rhnmd* Contact Method

Configuring Uyuni rhnmd Daemon

The Uyuni daemon can be configured by editing the file on the client:

```
/etc/sysconfig/rhn/rhnmd
```

This is the configuration file the **rhnmd** initialization script uses. An important parameter for the daemon is its check-in frequency. The default interval time is four hours (240 minutes). If you modify the configuration file, you must as root restart the daemon with:

```
systemctl rhnmd restart
```



Minimum Allowed Check-in Parameter

The minimum allowed time interval is one hour (60 minutes). If you set the interval below one hour, it will change back to the default of 4 hours (240 minutes).

Viewing rhnsd Daemon Status

As the root you may view the status of rhnsd by typing the command:

```
systemctl status rhnsd
```

Traditional Contact Method (osad)

OSAD is an alternative contact method between Uyuni and its clients. By default, Uyuni uses **rhnsd**, which contacts the server every four hours to execute scheduled actions. OSAD allows registered client systems to execute scheduled actions immediately.

OSAD has several distinct components:

- The **osa-dispatcher** service runs on the server, and uses database checks to determine if clients need to be pinged, or if actions need to be executed.
- The **osad** service runs on the client. It responds to pings from **osa-dispatcher** and runs **mgr_check** to execute actions when directed to do so.
- The **jabberd** service is a daemon that uses the **XMPP** protocol for communication between the client and the server. The **jabberd** service also handles authentication.
- The **mgr_check** tool runs on the client to execute actions. It is triggered by communication from the **osa-dispatcher** service.

The **osa-dispatcher** periodically runs a query to check when clients last showed network activity. If it finds a client that has not shown activity recently, it will use **jabberd** to ping all **osad** instances running on all clients registered with your Uyuni server. The **osad** instances respond to the ping using **jabberd**, which is running in the background on the server. When the **osa-dispatcher** receives the response, it marks the client as online. If the **osa-dispatcher** fails to receive a response within a certain period of time, it marks the client as offline.

When you schedule actions on an OSAD-enabled system, the task will be carried out immediately. The **osa-dispatcher** periodically checks clients for actions that need to be executed. If an outstanding action is found, it uses **jabberd** to execute **mgr_check** on the client, which will then execute the action.

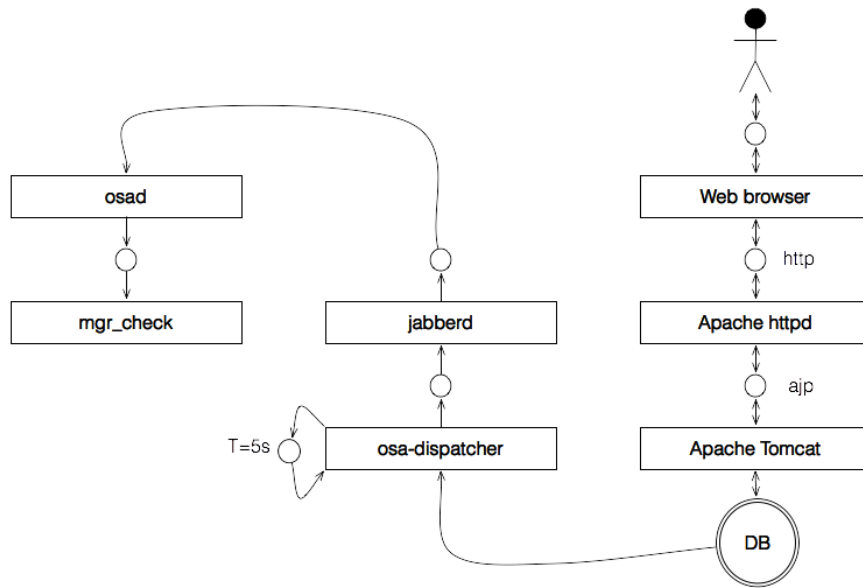


Figure 8. osad Contact Method

Enabling and Configuring OSAD

This section covers enabling the **osa-dispatcher** and **osad** services, and performing initial setup.

OSAD clients use the fully qualified domain name (FQDN) of the server to communicate with the **osa-dispatcher** service.

SSL is required for **osad** communication. If SSL certificates are not available, the daemon on your client systems will fail to connect. Make sure your firewall rules are set to allow the required ports. For more information, see `xref:tab.install.ports.server[Server Ports]`.

Procedure: Enabling OSAD

1. On your Uyuni server, as the root user, start the **osa-dispatcher** service:

```
systemctl start osa-dispatcher
```

2. On each client machine, install the **osad** package from the **Tools** child channel. The **osad** package should be installed on clients only. If you install the **osad** package on your Uyuni Server, it will conflict with the **osa-dispatcher** package.
3. On the client systems, as the root user, start the **osad** service:

```
systemctl start osad
```

Because **osad** and **osa-dispatcher** are run as services, you can use standard commands to

manage them, including `stop`, `restart`, and `status`.

Configuration and Log Files

Each OSAD component is configured by local configuration files. We recommend you keep the default configuration parameters for all OSAD components.

Component	Location	Path to Configuration File
<code>osa-dispatcher</code>	Server	<code>/etc/rhn/rhn.conf</code> Section: OSA configuration
<code>osad</code>	Client	<code>/etc/sysconfig/rhn/osad.conf</code> <code>/etc/sysconfig/rhn/up2date</code>
<code>osad</code> log file	Client	<code>/var/log/osad</code>
<code>jabberd</code> log file	Both	<code>/var/log/messages</code>

Troubleshooting OSAD

If your OSAD clients cannot connect to the server, or if the `jabberd` service takes a lot of time responding to port 5552, it could be because you have exceeded the open file count.

Every client needs one always-open TCP connection to the server, which consumes a single file handler. If the number of file handlers currently open exceeds the maximum number of files that `jabberd` is allowed to use, `jabberd` will queue the requests, and refuse connections.

To resolve this issue, you can increase the file limits for `jabberd` by editing the `/etc/security/limits.conf` configuration file and adding these lines:

```
jabbersoftnofile5100
jabberhardnofile6000
```

Calculate the limits required for your environment by adding 100 to the number of clients for the soft limit, and 1000 to the current number of clients for the soft limit. In the example above, we have assumed 500 current clients, so the soft limit is 5100, and the hard limit is 6000.

You will also need to update the `max_fds` parameter in the `/etc/jabberd/c2s.xml` file with your chosen hard limit:

```
<max_fds>6000</max_fds>
```

Traditional SSH Push

Traditional SSH Pull

DRAFT

Repositories

Repository Types

Description...

Pool Repositories

Devel Repositories

Tool Repositories

Maintenance Repositories

Repository Synchronization

Component Index

Introduction

Description...

Apache

Functions

Apache is a primary component of Uyuni. It performs the following functions in the stack.

- **Handles HTTP(S) communication**
- **Serves Static Files**
- **HTTP gateway to: Apache Tomcat, the Python XMLRPC server and Cobbler**

Log Files

Logs for Apache are located in:

```
/var/log/apache2/error_log
```

Apache Tomcat

Functions

Apache Tomcat is a primary component of Uyuni. It performs the following functions in the stack.

- **Contains servlet (Java) applications**
- **The most important servlet is the RHN servlet:**
- **Handles the majority of the Web UI**
- **Public XMLRPC API**

Log Files

Logs for Apache Tomcat are located in:

```
/var/log/rhn/rhn_web_ui.log  
/var/log/rhn/rhn_web_api.log  
/var/log/tomcat/catalina.out  
/var/log/tomcat/catalina.*.log
```

Python XMLRPC Server

Functions

The Python XMLRPC Server is a primary component of Uyuni. It performs the following functions in the stack.

- **Provides the private XMLRPC API**
- **Used primarily by client tools (mgr_check)**

Log Files

Logs for the Python XMLRPC Server are located in:

```
/var/log/apache2/error_log  
/var/log/rhn/rhn_server_xmlrpc.log
```

Taskomatic

Functions

Taskomatic is a primary component of Uyuni. It performs the following functions in the stack.

- **Taskomatic handles most background jobs**
- **Patch applicability status refresh**
- **Server side scheduling**
- **SSH push**
- **Cobbler database sync**
- **Repository synchronization and repository metadata generation**
- **CVE audit pre-computation**
- **Cleanup Jobs**

Log Files

Log files for taskomatic are located in:

```
/var/log/rhn/rhn_taskomatic_daemon.log  
/var/log/rhn/reposync/*
```

Database

Functions

The database is a primary component of Uyuni. It performs the following functions in the stack.

- **Primarily stores application data**
- **Functions as a data exchange area between components**

Log Files

Logs for the database are located in:

```
/var/lib/pgsql/data/pg_log/*
```

mgr-sync

Functions

mgr-sync is a command line tool for Uyuni. It performs the following function.

- **mgr-sync is a command line tool that synchronizes with SUSE Customer Center(SCC) and retrieves data and package repositories.**



mgr-sync and Open Source Distributions

This tool is designed for use with a support subscription or trial account with SUSE Customer Center. It is not required for open source distributions(OpenSUSE Leap , CentOS, Ubuntu, etc.).

mgr-sync --help

The following options are available for the **mgr-sync** command:

```
mgr-sync --help
usage: mgr-sync [-h] [--version] [-v] [-s] [-d {1,2,3}]
               {list,add,refresh,delete} ...

Synchronize SUSE Manager repositories.

optional arguments:
  -h, --help            show this help message and exit
  --version             Print mgr-sync version
  -v, --verbose         Be verbose
  -s, --store-credentials
                        Store credentials to the local dot file.
  -d {1,2,3}, --debug {1,2,3}
                        Log additional debug information depending on DEBUG

Subcommands:
  {list,add,refresh,delete}
    list                List channels, SCC organization credentials or
                        products
    add                 add channels, SCC organization credentials or products
    refresh             Refresh product, channel and subscription
    delete             Delete SCC organization credentials
```

Log Files

Logs for the mgr-sync tool are located in:

```
/var/log/rhn/mgr-sync.log
/var/log/rhn/rhn_web_api.log
```

spacewalk-repo-sync

Functions

spacewalk-repo-sync is a command line tool for Uyuni. It performs the following functions.

- **Copies a repo's metadata to the database**
- **Copies a repo's RPM files to the filesystem**

mgr-sync --help

*The following options are available for the **spacewalk-repo-sync** tool:*


```
spacewalk-repo-sync --help
Usage: spacewalk-repo-sync [options]

Options:
  -h, --help                show this help message and exit
  -l, --list                List the custom channels with the associated
                           repositories.
  -s, --show-packages      List all packages in a specified channel.
  -u URL, --url=URL        The url of the repository. Can be used multiple times.
  -c CHANNEL_LABEL, --channel=CHANNEL_LABEL
                           The label of the channel to sync packages to. Can be
                           used multiple times.
  -p PARENT_LABEL, --parent-channel=PARENT_LABEL
                           Synchronize the parent channel and all its child
                           channels.
  -d, --dry-run            Test run. No sync takes place.
  --latest                 Sync latest packages only. Use carefully - you might
                           need to fix some dependencies on your own.
  -g CONFIG, --config=CONFIG
                           Configuration file
  -t REPO_TYPE, --type=REPO_TYPE
                           Force type of repository ("yum", "uln" and "deb" are
                           supported)
  -f, --fail               If a package import fails, fail the entire operation
  -n, --non-interactive    Do not ask anything, use default answers
  -i FILTERS, --include=FILTERS
                           Comma or space separated list of included packages or
                           package groups.
  -e FILTERS, --exclude=FILTERS
                           Comma or space separated list of excluded packages or
                           package groups.
  --email                  e-mail a report of what was synced/imported
  --traceback-mail=TRACEBACK_MAIL
                           alternative email address(es) for sync output (--email
                           option)
  --no-errata              Do not sync errata
  --no-packages            Do not sync packages
  --sync-kickstart         Sync kickstartable tree
  --force-all-errata      Process metadata of all errata, not only missing.
  --batch-size=BATCH_SIZE
                           max. batch size for package import (debug only)
  -Y, --deep-verify        Do not use cached package checksums
  -v, --verbose            Verbose output. Possible to accumulate: -vvv
```

Log Files

Logs for the *spacewalk-repo-sync* tool are located in:

```
/var/log/rhn/reposync/*
```

osa-dispatcher

Functions

osa-dispatcher is a component of Uyuni. It performs the following function in the stack.

- **Monitors database for actions, informing osad clients when they need to run them**

osa-dispatcher --help

*The following options are available for the **osa-dispatcher**:*

```
osa-dispatcher --help
Usage: osa-dispatcher [options]

Options:
  -v, --verbose          Increase verbosity
  -N, --nodetach         Suppress backgrounding and detachment of the process
  --pid-file=PID_FILE   Write to this PID file
  --logfile=LOGFILE     Write log information to this file
  -h, --help            show this help message and exit
```

Log Files

*Logs for the **osa-dispatcher** are located in:*

```
/var/log/rhn/osa_dispatcher.log
```

jabberd

Functions

jabberd is a component of Uyuni. It performs the following function in the stack.

- **Implements the Jabber (XMPP) protocol that osa-dispatcher uses**

Log Files

Logs for jabberd are located in:

```
/var/log/messages
```

mgr_check

Functions

mgr_check is a primary component of Uyuni. It performs the following functions in the stack.

- **Client-side command line tool for legacy clients that checks for actions on the server and executes them**



mgr_check and rhn_check

mgr_check is symlinked to **rhn_check** in `/usr/sbin/`. Both *mgr_check* and *rhn_check* can be used for checking for actions on the server.

```
lrwxrwxrwx 1 root root          9 Sep  9 09:05 mgr_check ->
rhn_check*
```

mgr_check --help

The following options are available for the **rhn_check** on your legacy clients:

```
mgr_check --help
Usage: rhn_check [options]

Options:
  -v, --verbose          Show additional output. Repeat for more detail.
  --proxy=PROXY          Specify an http proxy to use
  --proxyUser=PROXYUSER  Specify a username to use with an authenticated http
                        proxy
  --proxyPassword=PROXYPASSWORD Specify a password to use with an authenticated http
                        proxy
  --version              show program's version number and exit
  -h, --help             show this help message and exit
```

Log Files

Logs for the **mgr_check** are located on your legacy clients in:

```
/var/log/up2date
```

zypp-plugin-spacewalk

Functions

zypp-plugin-spacewalk is a component of Uyuni. It performs the following functions in the stack.

- **Client-side add-on to zypper for legacy clients**
- **Exposes SUSE Manager channels as zypper repositories**
- **The plugin is not required on salt-minions**

Log Files

Logs for the **zypp-plugin-spacewalk** are located on your legacy clients in:

```
/var/log/zypper.log
/var/log/zypp/*
```

rhnsd

Functions

rhnsd is a primary component of Uyuni. It performs the following functions in the stack.

- **Client-side daemon for legacy clients**
- **Periodically calls mgr_check(symlinked to rhn_check)**
- **Randomizes check time not to overload the server**

rhnsd --help

The following options are available for use with rhnsd on your legacy clients:

```
rhnsd --help
Usage: rhnsd [OPTION...]
Spacewalk Services Daemon
```

-f, --foreground	Run in foreground
-i, --interval=MINS	Connect to Spacewalk every MINS minutes
-?, --help	Give this help list
--usage	Give a short usage message
-V, --version	Print program version

Mandatory or optional arguments to long options are also mandatory or optional for any corresponding short options.

osad

Functions

osad is a primary component of Uyuni. It performs the following functions in the stack.

- **Client-side daemon for legacy clients**
- **Calls mgr_check(rhn_check) when notified by Jabber**

osad --help

The following options are available for use with **osad** on your legacy clients:

```

osad --help
Usage: osad [options]

Options:
  -v, --verbose           Increase verbosity
  -N, --nodetach          Suppress backgrounding and detachment of the process
  --pid-file=PID_FILE     Write to this PID file
  --logfile=LOGFILE       Write log information to this file
  --cfg=CFG               Use this configuration file for defaults
  --jabber-server=JABBER_SERVER
                          Primary jabber server to connect to
  -h, --help              show this help message and exit

```

Log Files

*Logs for **osad** are located in:*

```
/var/log/osad
```

salt-master

Functions

The **salt-master** is a primary component of Uyuni. It performs the following functions in the stack.

- **Core process of Salt on the server side**
- **Provides communication with salt minions**
- **Handles Salt jobs, publishes to the Salt event Bus**
- **Handles minion responses**
- **Manages states, highstates, pillar information, etc**

salt-master --help

The following options are available for the **salt-master**. The following list is not comprehensive, for more information see: [The Saltstack Docs](#)

Options:

```
salt-master --help
Usage: salt-master [options]
```

The Salt Master, used to control the Salt Minions

Options:

```
--version          show program's version number and exit
-V, --versions-report
                    Show program's dependencies version number and exit.
-h, --help          show this help message and exit
--saltfile=SALTFILE Specify the path to a Saltfile. If not passed, one
                    will be searched for in the current working directory.
-c CONFIG_DIR, --config-dir=CONFIG_DIR
                    Pass in an alternative configuration directory.
                    Default: '/etc/salt'.
-u USER, --user=USER Specify user to run salt-master.
-d, --daemon        Run the salt-master as a daemon.
--pid-file=PIDFILE  Specify the location of the pidfile. Default:
                    '/var/run/salt-master.pid'.
```

Logging Options:

Logging options which override any settings defined on the configuration files.

```
-l LOG_LEVEL, --log-level=LOG_LEVEL
                    Console logging log level. One of u'all', u'garbage',
                    u'trace', u'debug', u'profile', u'info', u'warning',
                    u'error', u'critical', u'quiet'. Default: 'warning'.
--log-file=LOG_FILE
                    Log file path. Default: '/var/log/salt/master'.
--log-file-level=LOG_LEVEL_LOGFILE
                    Logfile logging log level. One of u'all', u'garbage',
                    u'trace', u'debug', u'profile', u'info', u'warning',
                    u'error', u'critical', u'quiet'. Default: 'warning'.
```

You can find additional help about salt-master issuing "man salt-master" or on <http://docs.saltstack.com>

Log Files

Logs for **salt-master** are located in:

```
/var/log/salt/master
```

salt-api

Functions

The **salt-api** is a primary component of Uyuni. It performs the following functions in the stack.

- **Internal API communicates the Java side of Uyuni with the salt-master**
- **Provides HTTPS and websocket interfaces with the salt-master**
- **Handles the SSH connections to minions (SSH Push)**

salt-api --help

The following options are available for the **salt-api**. The following list is not comprehensive, for more information see: [The Saltstack Docs](#)

Options:

```
salt-api --help
Usage: salt-api [options]

The Salt API system manages network API connectors for the Salt Master

Options:
  --version            show program's version number and exit
  -V, --versions-report
                        Show program's dependencies version number and exit.
  -h, --help            show this help message and exit
  -c CONFIG_DIR, --config-dir=CONFIG_DIR
                        Pass in an alternative configuration directory.
                        Default: '/etc/salt'.
  -d, --daemon          Run the salt-api as a daemon.
  --pid-file=PIDFILE    Specify the location of the pidfile. Default:
                        '/var/run/salt-api.pid'.

Logging Options:
  Logging options which override any settings defined on the
  configuration files.

  -l LOG_LEVEL, --log-level=LOG_LEVEL
                        Console logging log level. One of u'all', u'garbage',
                        u'trace', u'debug', u'profile', u'info', u'warning',
                        u'error', u'critical', u'quiet'. Default: 'warning'.
  --log-file=API_LOGFILE
                        Log file path. Default: '/var/log/salt/api'.
  --log-file-level=LOG_LEVEL_LOGFILE
                        Logfile logging log level. One of u'all', u'garbage',
                        u'trace', u'debug', u'profile', u'info', u'warning',
                        u'error', u'critical', u'quiet'. Default: 'warning'.

You can find additional help about salt-api issuing "man salt-api" or on
http://docs.saltstack.com
```

Log Files

*Logs for **salt-api** are located in:*

```
/var/log/salt/master
/var/log/salt/api
```

salt-minion

Functions

The **salt-minion** is a primary component of Uyuni. It performs the following functions in the stack.

- **Client-side main process for Salt clients (only pull method)**
- **Communicates the client with salt-master via Salt event bus (ZeroMQ)**
- **Executes the actions received from the Salt master on the client (minion)**

salt-minion --help

The following options are available for the **salt-minion**. The following list is not comprehensive, for more information see: [The Saltstack Docs](#)

Options:

```
salt-minion --help
Usage: salt-minion [options]

The Salt Minion, receives commands from a remote Salt Master

Options:
  --version            show program's version number and exit
  -V, --versions-report
                        Show program's dependencies version number and exit.
  -h, --help          show this help message and exit
  --saltfile=SALTFILE Specify the path to a Saltfile. If not passed, one
                        will be searched for in the current working directory.
  -c CONFIG_DIR, --config-dir=CONFIG_DIR
                        Pass in an alternative configuration directory.
                        Default: '/etc/salt'.
  -u USER, --user=USER Specify user to run salt-minion.
  -d, --daemon        Run the salt-minion as a daemon.
  --pid-file=PIDFILE  Specify the location of the pidfile. Default:
                        '/var/run/salt-minion.pid'.

Logging Options:
  Logging options which override any settings defined on the
  configuration files.

  -l LOG_LEVEL, --log-level=LOG_LEVEL
                        Console logging log level. One of u'all', u'garbage',
                        u'trace', u'debug', u'profile', u'info', u'warning',
                        u'error', u'critical', u'quiet'. Default: 'warning'.
  --log-file=LOG_FILE  Log file path. Default: '/var/log/salt/minion'.
  --log-file-level=LOG_LEVEL_LOGFILE
                        Logfile logging log level. One of u'all', u'garbage',
                        u'trace', u'debug', u'profile', u'info', u'warning',
                        u'error', u'critical', u'quiet'. Default: 'warning'.

You can find additional help about salt-minion issuing "man salt-minion" or on
http://docs.saltstack.com
```

Log Files

*Logs for **salt-minion** are located in:*

```
/var/log/salt/minion
```

salt-broker

Functions

The **salt-broker** is a component of the Uyuni proxy. It performs the following functions in the stack.

- **Used only in the Uyuni Proxy for minions using pull method**
- **Forwards the ZeroMQ Salt channels from Uyuni server to the proxy minions**

Log Files

*Logs for **salt-broker** are located in:*

```
/var/log/salt/broker
```