

Project no. 257438

# CONTRAIL

Integrated Project  
OPEN COMPUTING INFRASTRUCTURES FOR ELASTIC SERVICES

## Contrail Software: Installation Guide

First release v1.0

Contrail Consortium

*Start date of project:* October 1<sup>st</sup> 2010

Release Date: 15 May 2012

Project co-funded by the European Commission within the Seventh Framework Programme		
Dissemination Level		
<b>PU</b>	Public	✓
<b>PP</b>	Restricted to other programme participants (including the Commission Services)	
<b>RE</b>	Restricted to a group specified by the consortium (including the Commission Services)	
<b>CO</b>	Confidential, only for members of the consortium (including the Commission Services)	



# Contents

<b>1</b>	<b>Overview</b>	<b>3</b>
<b>2</b>	<b>Installation</b>	<b>6</b>
2.1	Installation steps . . . . .	6
2.1.1	Package sources . . . . .	6
	Setting up package repositories . . . . .	6
	RabbitMQ . . . . .	8
	Python and Zookeeper . . . . .	8
2.1.2	OpenNebula installation . . . . .	9
	Installing OpenNebula . . . . .	9
2.1.3	OpenNebula Head Node . . . . .	10
	Installing VEP . . . . .	11
2.1.4	OpenNebula Worker Node . . . . .	14
	ONE Sensors . . . . .	14
2.1.5	OpenNebula Head Node Continued . . . . .	15
	ONE Monitor . . . . .	15
	REST Monitoring . . . . .	16
2.1.6	Federation . . . . .	16
	Monitoring Hub . . . . .	17

## CONTENTS

Federation Web . . . . .	18
<b>3 Public resources</b>	<b>20</b>
<b>Bibliography</b>	<b>20</b>
<b>Index</b>	<b>21</b>

## Chapter 1

# Overview

In this document we present a quick set-up guide for the Contrail system. Contrail software packages are divided into four meta-packages:

- `contrail-federation`
- `contrail-provider-common`
- `contrail-provider-one-head`
- `contrail-provider-one-node`

Each meta-package consists of other software packages that need to be installed on the machine in order to join the Contrail system. Package **`contrail-federation`** should be installed on a machine that will act as the federation access point. Installation of this package is given in section [2.1.6](#).

Package **`contrail-provider-common`** should be installed on a machine that will act as the provider's access point to the federation. Installation of this package is given in section [2.1.3](#).

Since Contrail for the first release supports only OpenNebula [4] as IaaS software stack, there are two meta-packages for the IaaS itself: packages **`contrail-provider-one-head`** and **`contrail-provider-one-node`** should be installed on OpenNebula head and node machines, respectively. Refer to section [2.1.2](#) for details on installation of OpenNebula).

The relations between packages are shown in Figure [1.1](#)

In the last chapter 3 we list web resources that can help users to start with Contrail quickly.

In the further sections we provide step-by-step guide to set-up the Contrail. In listings like this:

```
# content of a property file
```

we give e.g. content of the property file.

When a user needs to provide some input such as write a command in command line, we show this in a yellow box:

```
echo "Please, install me Contrail!"
```

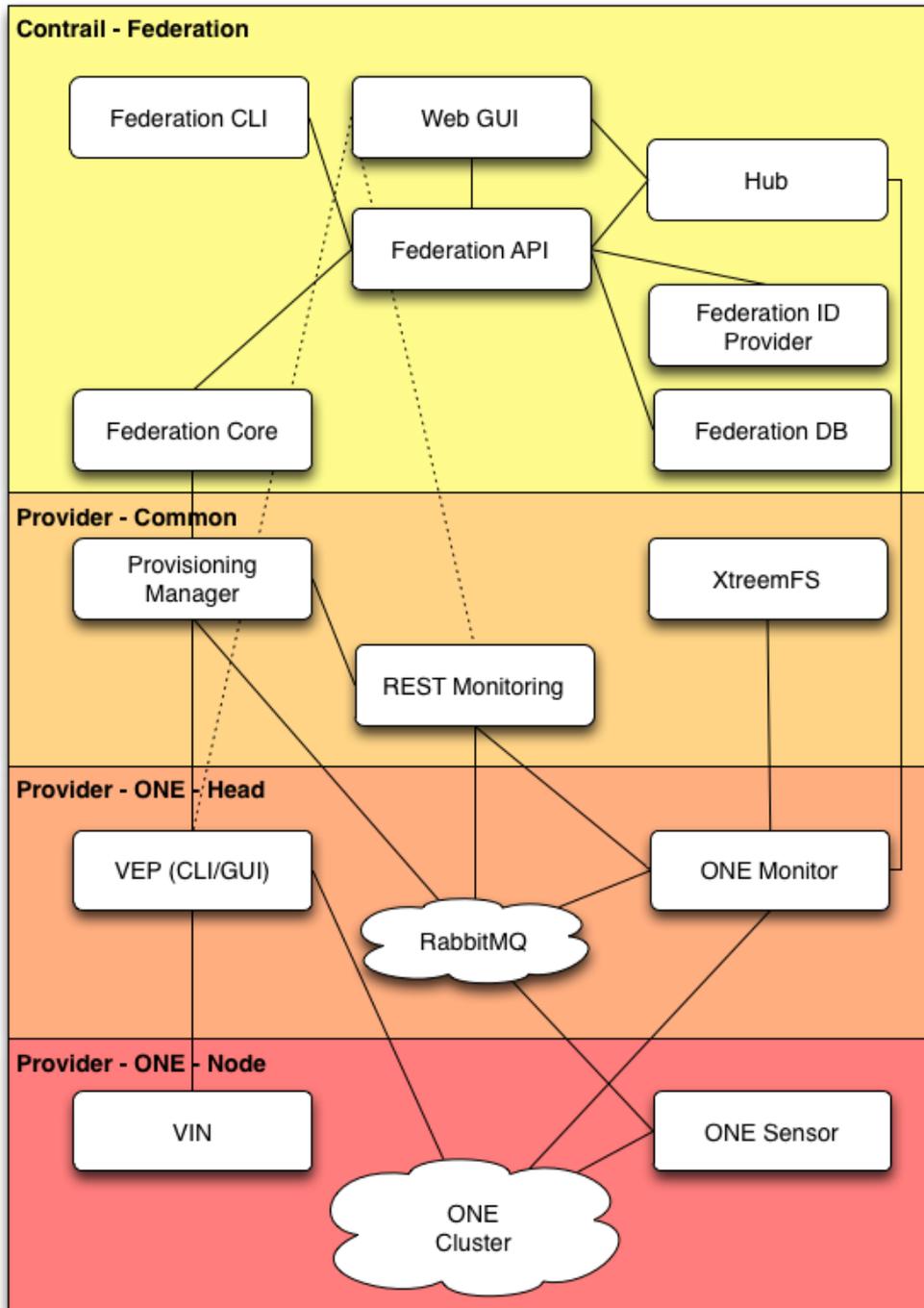


Figure 1.1: Relations and information flow between packages.

---

## 2.1 Installation steps

This is step-by-step installation guide of the Contrail system. In case of any problems, please refer to the documentation of the component for detailed instructions (Contrail AdminGuide, [1]). In order to install the complete system, you need to set up three different parts:

- provider head node (for example, the head node of the ONE cluster),
- provider worker node (for example, the worker node in the ONE cluster, can be the same as head node),
- federation node (different node).

### 2.1.1 Package sources

This chapter is about getting Contrail binaries for installation and how to proceed in different installation scenarios (CD, Download repositories or Virtual Machine). You can find in chapter 3 a quick-reference of network URLs where to get binaries, sources, documentation and support.

#### Setting up package repositories

In order to include Contrail repositories, select one of the following supported distributions (Debian 6.0, Ubuntu 11.04 and Ubuntu 11.10) and add one of the

following lines at the end of the `/etc/apt/sources.list`. Let us note, that the current state of the Contrail stack works best on the newest versions of Ubuntu distributions, also on Debian 6.0 with unstable packaging. The issues with stable version of Debian 6.0 are described in sections 2.1.1 and ??.

Repositories based on official release code:

```
deb http://contrail.ow2.org/repositories/binaries/release/Debian_6.0/ ./
deb http://contrail.ow2.org/repositories/binaries/release/xUbuntu_11.04/ ./
deb http://contrail.ow2.org/repositories/binaries/release/xUbuntu_11.10/ ./
```

Repositories with the nightly builds:

```
deb http://contrail.ow2.org/repositories/binaries/testing/Debian_6.0/ ./
deb http://contrail.ow2.org/repositories/binaries/testing/xUbuntu_11.04/ ./
deb http://contrail.ow2.org/repositories/binaries/testing/xUbuntu_11.10/ ./
```

Once repositories are added to the list, the following command adds a key for the repositories:

```
wget -O - http://contrail.ow2.org/repositories/contrail.pub | sudo apt-key add
```

followed by the updating of the repositories

```
sudo apt-get update
```

If aptitude packages are installed, you can check whether contrail packages are available by executing the following command (in order to install aptitude, use `sudo apt-get install aptitude` first):

```
aptitude search contrail
```

The command should return the list of packages. If the list is empty, either the repositories or the key are not properly set.

## RabbitMQ

The version of RabbitMQ that is required by the Contrail is newer than the one available in the Ubuntu 11.04 and Debian 6.0 stable repositories. In order to avoid package conflicts between official old version of RabbitMQ included in the named distribution and ones required by the Contrail system, we recommend to install RabbitMQ directly from the official RabbitMQ page at <http://www.rabbitmq.com/install-debian.html>.

## Python and Zookeeper

The current state of the Contrail stack has some missing functionality. In order for the Web interface to remember the connections between user's applications, deployed VMs, its IDs and IDs on the OpenNebula clusters, we used an additional bookkeeping framework ZooKeeper. However, there are some problems in the packaging of the library in the stable version of Debian 6 and older versions of Ubuntu. In order to solve this problem, we suggest to install the library before installing the *contrail-federation-web* package. To install it manually, perform the following steps.

*Debian* installation (or with python 2.6 by default):

```
sudo apt-get install python-dev python-setuptools
sudo easy_install pip
sudo pip install zc-zookeeper-static
```

For other distributions with python 2.7, the installation script still depends on python 2.6, resulting in:

*Ubuntu 11.04, Ubuntu 11.10* installation (or with python 2.7 by default):

```
sudo apt-get install python-dev2.6 python-setuptools
sudo easy_install-2.6 pip
sudo pip install zc-zookeeper-static
```

Please note that the python *dev* and *setuptools* packages might already be installed on your system.

## 2.1.2 OpenNebula installation

In the following installation scenario, we are going to use `oneadmin` as an OpenNebula user and `oneadmin` as OpenNebula password for the user. Also, we are going to use MySQL user `root` set without password. If your installation is different, please act accordingly. The commands are used for root user. In case of installing the software under the normal user, use `sudo` when necessary.

### Installing OpenNebula

Installation, configuration and administration of OpenNebula is described in detail at [www.opennebula.org](http://www.opennebula.org). After installing a basic, functional OpenNebula (ONE) deployment, follow this guide to join your ONE to a Contrail federation as an infrastructure provider.

At the moment the Contrail software is developed against OpenNebula v2.2, thus these instructions refer to the OpenNebula v2.2. Documentation is available at:

- [OpenNebula v2.2 documentation](#)
- [Planning OpenNebula installation](#)

The guide was tested during the installation of Contrail packages on Ubuntu 11.10 server. On other platforms, different package names and location of configuration files should be expected.

**OpenNebula head** We assume that ONE was already installed, configured and tested. Here it is described the additional Contrail specific configuration.

On OpenNebula head node, we need to add ONE Contrail user account.

```
oneuser create contrail contrailpass
```

Monitoring components send messages about individual VMs to `rabbitmq` server, located on ONE head.

```
apt-get install rabbitmq-server
rabbitmqctl add_user contrail contrailpass
rabbitmqctl set_permissions contrail ".*" ".*" ".*"
```

### 2.1.3 OpenNebula Head Node

We start with the installation of the software stack on ONE head node. In our example ONE head is located at hostname (DNS name) n0004.

To install the software stack that is common for all providers, execute:

```
apt-get install contrail-provider-common
```

The package currently contains the following components:

- contrail-provisioning-manager,
- contrail-rest-monitoring,
- xtremfs-client.

To install software stack that is specific for the ONE cluster, execute:

```
apt-get install contrail-provider-one-head
```

This package currently contains the following components:

- conpaas-scalarix-one-manager,
- contrail-one-monitor.

On some system, the following error might occur during the installation of the contrail-provider-one-head package:

```
ERROR: Could not find a valid gem 'oca' (>= 0) in any repository
ERROR: Possible alternatives: oca
1 gem installed
dpkg: error processing conpaas-scalarix-one-manager (--configure):
subprocess installed post-installation script returned error exit \
status 2
```

To get past that error, use the following command and repeat the procedure.

```
gem install oca
apt-get install contrail-provider-one-head
```

## Installing VEP

Let us start with the component that currently needs most configuration, VEP. There are two packages, one with GUI (`contrail-vep-gui`) and one for CLI (`contrail-vep-cli`). We've used the CLI version of the component. If you prefer to use GUI version, use the documentation that is available in the Contrail user guide [2]. If `contrail-vep-cli` package is not part of the `contrail-provider-one-head`, yet, you need to install it manually<sup>1</sup>.

```
apt-get install contrail-vep-cli
```

To check if vep is running use:

```
ps -x | grep vep
```

If VEP component is not available, the following error might have occurred when running:

```
root@n0004:~# contrail-vep-cli -d
VEP system properties file and VEP logger properties paths were \
    not specified, using the default path for vep properties \
    (/root/.vep-cli/vep.properties) and created a new default \
    logger properties files
SLF4J: Detected both log4j-over-slf4j.jar AND slf4j-log4j12.jar \
    on the class path, preempting StackOverflowError.
SLF4J: See also http://www.slf4j.org/codes.html#log4j\
    DelegationLoop for more details.
Exception in thread "main" java.lang.ExceptionInInitializerError
    at org.apache.log4j.Logger.getLogger(Logger.java:39)
    at org.ow2.contrail.provider.vep.VEPHelperMethods.<init>\
    (VEPHelperMethods.java:32)
    at org.ow2.contrail.provider.vep.VEPStart.<init>(VEPStart.\
    java:85)
    at org.ow2.contrail.provider.vep.VEPStart.main(VEPStart.\
    java:1048)
```

<sup>1</sup>At time of writing this guide, `contrail-vep-cli` still was not part of the meta-package.

```
Caused by: java.lang.IllegalStateException: Detected both \
    log4j-over-slf4j.jar AND slf4j-log4j12.jar on the \
    class path, preempting StackOverflowError. See \
    also http://www.slf4j.org/codes.html#\
    log4jDelegationLoop for more details.
    at org.apache.log4j.Log4jLoggerFactory.<clinit>\
    (Log4jLoggerFactory.java:49)
    ... 4 more
```

To correct this error, edit the script that runs the component and remove the references to `log4j-over-slf4j.jar` and `slf4j-log4j12.jar`:

```
nano /usr/bin/contrail-vep-cli
```

remove the topmost two files from the list and save the file.

We need to copy the following files for VEP to work properly:

```
mkdir /root/.vep-cli/
cp /usr/share/contrail/contrail-vep-cli/vep.properties \
    /root/.vep-cli/
cp /usr/share/contrail/contrail-vep-cli/VEPKeyStore.jks .vep-cli/
```

Once configuration files are copied, edit the `vep.properties` file and set ONE user and password, as well as part of the ONE installation details.

```
cd /root/.vep-cli
nano vep.properties

mysql.pass=contrail
one.ip=
one.port=2633
one.user=oneadmin
one.pass=oneadmin
pdp.use = true/false
contrail.cluster=1
```

VEP currently expects that ONE cluster exists. This can be made with the following command:

```
onecluster create contrail # creates cluster with ID 1
```

Try running VEP again with calling the script `contrail-vep-cli` (do not use switch `-d` now).

If output of the VEP indicates errors, such as:

```
/user/gregorb DEBUG dbHandler - Executing query: select * from \
    user where username='fedadmin'
qtp1482258114-21 DEBUG log - EOF org.eclipse.jetty.io.EOFException
```

This indicates that the problem occurs when there is no `fedadmin` user in MySQL database (or no user in the table `user`). To get past this, the following SQL statements are currently missing when running VEP:

```
root@n0004:~# SQL="use vepdb;"
root@n0004:~# SQL+="insert into ugroup \
    (gname,uid) values ('admin', 1);"
root@n0004:~# SQL+="insert into user (username,uid,vid,oneuser,\
    onepass,oneid,role) values ('fedadmin',1,-1,0,\
    '7bc8559a8fe509e680562b85c337f170956fcb06',-1,'admin');"
root@n0004:~# mysql -u root -e "$SQL"
```

Now, try using VEP with Telnet. As stated in the documentation, VEP REST server listens on port 10500 while telnet server listens on port 10555.

```
# use vep-cli via telnet
telnet localhost 10555
  locadmin
  loc1234
  yes
  ladmin
  l1234
```

As described in the section about the VEP component, the following commands add requested to properly set the data (datacenter, cluster, rack, host) that is needed to identify the ONE setup (in the next listing we use "/" to break the line):

```
#noninteractive telnet, check console output if command succeeded
echo -e 'ladmin\nl1234\nadd datacenter\
      ndatac1\nSI\ndc-1-desc\ny\n\n\n' | nc localhost 10555
echo -e 'ladmin\nl1234\nadd cluster\ncl1\
      n001\n1\ncl1-1-desc\ny\n\n\n' | nc localhost 10555
echo -e 'ladmin\nl1234\nadd rack\nrc1\n1\
      nrc-1-desc\ny\n\n\n' | nc localhost 10555
echo -e 'ladmin\nl1234\nadd \
      host\nn0004\nnim_kvm\nvmm_kvm\ntm_nfs\ny\
      nexit\n' | nc localhost 10555
echo -e 'ladmin\nl1234\nadd fedadmin\n0\n1\n1\
      nhost-vep-1\ny\n' | nc localhost 10555
```

To check if all is properly set call:

```
onehost list
onecluster list
```

The final step is to make sure that ONE virtual networks and images which are referenced by the OVF should be public. While `oneimage` can be created on-the-fly by the VEP, `onevnet` virtual network has to be pre-created and explicitly published.

```
onevnet publish 0
```

This finalizes our installation of VEP.

## 2.1.4 OpenNebula Worker Node

### ONE Sensors

When VEP install is finished, we can continue installing `contrail-provider-one-node` package on each ONE node. The package contains the following components:

- contrail-vin,
- scalaris,
- xtreemfs-server,
- contrail-one-sensor

Currently there is a missing dependency on `sysstat` package, therefore the installation for ONE worker node looks like:

```
apt-get install contrail-provider-one-node sysstat
```

Edit the following configuration and set the address of ONE head node for RabbitMQ host:

```
nano /etc/contrail/contrail-one-sensor/one-sensor.config
```

Example of the content of the configuration file (changed values only):

```
rabbitmq_host = n0004  
host_properties_file = /etc/contrail/contrail-one-sensor/hostConfig
```

This concludes the installation of the ONE worker node.

## 2.1.5 OpenNebula Head Node Continued

### ONE Monitor

Once the sensors for OpenNebula are properly configure, we can continue setting up ONE head node. Next step is to set up `contrail-provider-one-monitoring` package by editing the following configuration file:

```
/etc/contrail/contrail-one-monitor/one-monitor.config
```

Change the host of the federation node (in our case, the federation host is `n0005`):

```
federation_finagle_host=n0005
```

Restart the component by calling:

```
/etc/init.d/contrail-one-monitor restart
```

If the `contrail-provider-one-monitoring` component is running or not can be seen by checking the `/tmp/contrail-one-monitor.log` file with

```
cat /tmp/contrail-one-monitor.log
```

If there are errors due to missing/pending connection to the monitoring-hub, ignore that. The connection will be restored once the monitoring-hub is installed and set up on the federation node.

### REST Monitoring

To properly configure `contrail-rest-monitoring` package, edit the following config file:

```
/usr/share/contrail/common/rest-monitoring/config
```

And change the location of the directory with ONE images. The location of the ONE images depends on the ONE deployed. In our case, the changed part of the configuration file looks like:

```
image_cache_dir = /srv/one-images-2
```

### 2.1.6 Federation

Once providers are set, we need to set the federation node. The federation stack is installed through `contrail-federation` package with:

```
sudo apt-get install contrail-federation
```

Components in the contrail-federation package are:

- contrail-federation-api,
- contrail-federation-db,
- contrail-federation-id-prov,
- contrail-federation-web,
- contrail-monitoring-hub.

Please note, that in the following examples of configuration, our federation node is located on host n0005.

### Monitoring Hub

For basic setup, we need to set the monitoring-hub and the federation-web components. The monitoring-hub must know where to locate the federation RabbitMQ. To set this up, we need to configure the following file:

```
/etc/contrail/contrail-monitoring-hub/config.json
```

And change the values:

```
rabbit,enabled=true  
rabbit,host=n0005
```

When configuration is complete, restart the service by calling:

```
/etc/init.d/contrail-monitoring-hub restart
```

The following output is an example of a properly set monitoring-hub.

```
688731 [New I/O server worker #1-7] INFO org.ow2.contrail.\  
      monitoring.hub.HubServer$ - PUT /metrics/route/host.\  
      n0004-xc2-xlab-lan.disk  
688731 [New I/O server worker #1-7] INFO org.ow2.contrail.\  
      monitoring.hub.HubServer$ - New metric for route host.\  
      n0004-xc2-xlab-lan.disk for provider 1
```

The easiest way to check if the monitoring-hub is properly set is to use the stdout for messages. This is obtained by using the following commands:

```
# To view log file - it is stdout actually, so
/etc/init.d/contrail-monitoring-hub stop
contrail-monitoring-hub
```

### Federation Web

The final component that needs to be configured is contrail-federation-web. Currently, contrail-federation-cli and zookeeper (only a temporary solution) should be installed manually:

```
apt-get install contrail-federation-cli zookeeperd
```

Continue the configuration by setting up the configuration file which is located at:

```
/etc/contrail/contrail-federation-web/federation-web.conf.
```

We need to set up the locations of the federation and provider head nodes. In our case, the changed values are:

```
FEDERATION_API_URL = http://n0005:8080/federation-api
SLA_EXTRACTOR_BASE =
    http://n0004:8080/rest-monitoring/sla/slaextractor
MONITORING_BASE = http://n0004:8080/rest-monitoring/monitoring
```

Some parts of the configuration (user data, not install) are not included in the GUI, yet. The script, which is listed in the following lines, is accessible on-line<sup>2</sup>. The following commands add provider, server, and SLAT (SLA Template) via contrail-federation-cli (provider can be added via GUI; actually only PROVIDER.ID is required for add-server and add-slats commands):

<sup>2</sup>[http://contrail.xlab.si/test-files/init\\_fed.sh](http://contrail.xlab.si/test-files/init_fed.sh)

```

export FEDERATION_IP=n0005

export IP=n0004

export HOSTNAME=n0004

export FEDERATION_CLI_URL="http://n0005:8080/federation-api"

PROVIDER_ADD_DATA=$(contrail-federation-cli add-provider -data '{"name':\
'CloudProvider2', 'email': 'cloudprovider2.com', 'country': 'UK', 'typeId': 42,\
'providerUri': 'http://$IP:10500'}")

export PROVIDER_ID=$(echo $PROVIDER_ADD_DATA | python -c "import sys; import \
json; print json.loads(sys.stdin.read())['headers']['Location']\
.split('/')[-1]")

contrail-federation-cli add-server -providerId $PROVIDER_ID -data "\
{'name': '$HOSTNAME', 'ram_total': '3915', 'ram_used': '1152', \
'ram_free': '2763', 'cpu_cores': '4', 'cpu_speed': '2494.276', \
'cpu_load_one': '0.09', 'cpu_load_five': '0.04'}"

contrail-federation-cli add-slat -providerId $PROVIDER_ID -data "\
{'name': 'XLAB SLAT', 'url': 'http://contrail.xlab.si/test-\
files/ubuntu-test-xlab-SLA.xml'}"

```

Web GUI should now be available at <http://n0005/>.

The CloudCoordinator can login as `coordinator:password`. He creates new users with role `FederationUser`. New users are able to use provider SLATs and to deploy their applications (more on this in the User's Guide).

You can also check if the `contrail-federation-api` is working properly by calling (and obtaining non-empty JSON file in response):

```

apt-get install curl          # In case you do not have 'curl' \
                              installed yet
curl http://localhost:8080/federation-api/users # check, by \
                              querying federation-api rest interface

```

This concludes the installation steps of the Contrail components. The usage of the system is described in the User Manual.

## Chapter 3

# Public resources

Here we list available Contrail public resources:

**Contrail public Wiki** resides here [3]

**Contrail Admin guide** [1]

**Contrail User guide** [2]

**Contrail Installation guide** is this document.

**Contrail example files** reside here: <http://contrail.xlab.si/test-files/>

# Bibliography

- [1] Contrail Consortium. Advance Guide: Administering Contrail. Technical report, 2012.
- [2] Contrail Consortium. Advance Guide: Using Contrail. Technical report, 2012.
- [3] Contrail consortium. Contrail wiki pages. Website, 2012. <http://contrail.projects.ow2.org/xwiki/bin/view/Main/>.
- [4] OpenNebula open-source project. Opennebula 2.2 guides. Website, 2011. <http://www.opennebula.org/documentation:archives:rel2.2>.