

# Contrail 技术文档- Contrail on ESXi 安装篇



SDN Associated Press 首发 [www.sdnnap.com](http://www.sdnnap.com) 网站。本文档描述 Contrail 商业版的基本部署和安装流程，参考外部和内部文档

## 相关资源

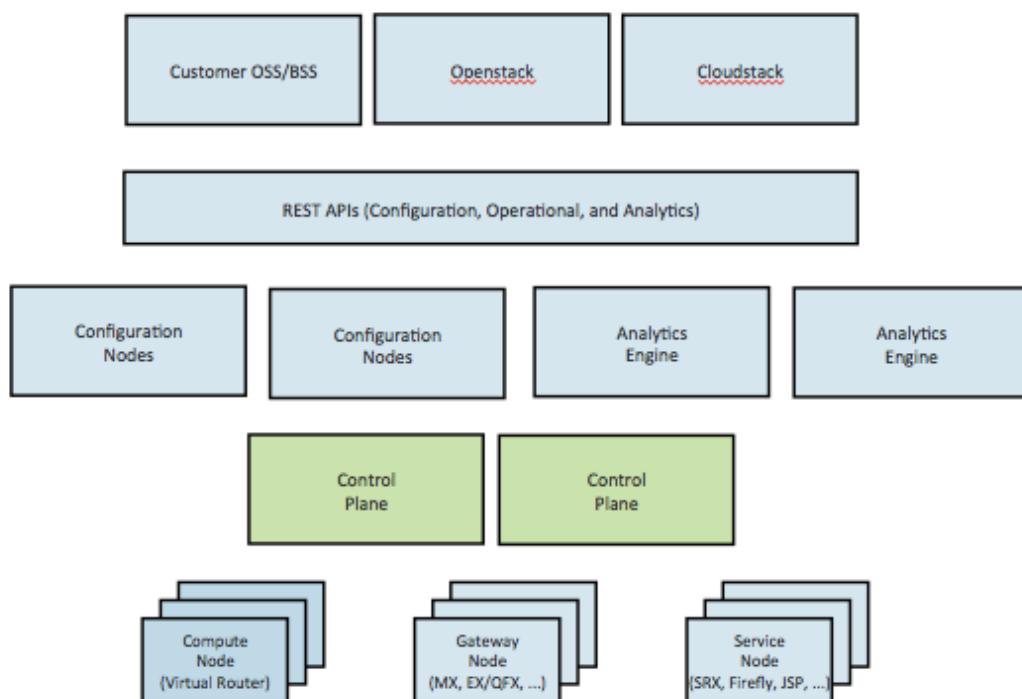
请访问 Juniper 的网站对设备的基本安装有基本了解之后，在进行后续的部署  
[https://techwiki.juniper.net/Documentation/Contrail/Contrail\\_Controller\\_Getting\\_Started\\_Guide](https://techwiki.juniper.net/Documentation/Contrail/Contrail_Controller_Getting_Started_Guide)

安装软件版本，Contrail 1.02 我使用的是 install media for CentOS 6.4 (不要试图问我哪里可以找到商业版的 Contrail，多谢理解) 这样在安装完成系统之后内置 Contrail package 和 Openstack，进行部署就可以了（买一送一有木有）

## 基本环境介绍

硬件环境：某品牌 Server 32G 内存，CPU 貌似至强， 1T 硬盘 预装 ESXi 5.5 ，并且由 Vcenter 进行管理

先看一下整个 Contrail 的基本结构



在整个系统下的基本组件如下

- *cfgm*—Runs Contrail configuration manager (config-node)

*cfgm* 是整个系统的配置管理节点

- *openstack*—Runs OpenStack services such as Nova, Quantum, and the like

这部分不多介绍了，在软件包中已经预制了 *openstack*，也就是说买一送一，装了 *contrail*, *openstack* 也顺道装上去了

- *collector*—Runs monitoring and analytics services

分析节点，运行一些监控协议和分析协议

- *compute*—Runs vRouter service and launches tenant virtual machines (VMs)

计算节点，这个实际上是安装在每一个 server 上的

- *control*—Runs the control plane service

控制节点，千万别以为他是管理用的，这货就是一个 BGP 的 RR 反射器，不要误会，*cfgm* 才是 Boss

- *database*—Runs analytics and configuration database services

数据库啊数据库

- *webui*—Runs the administrator web-based user interface service

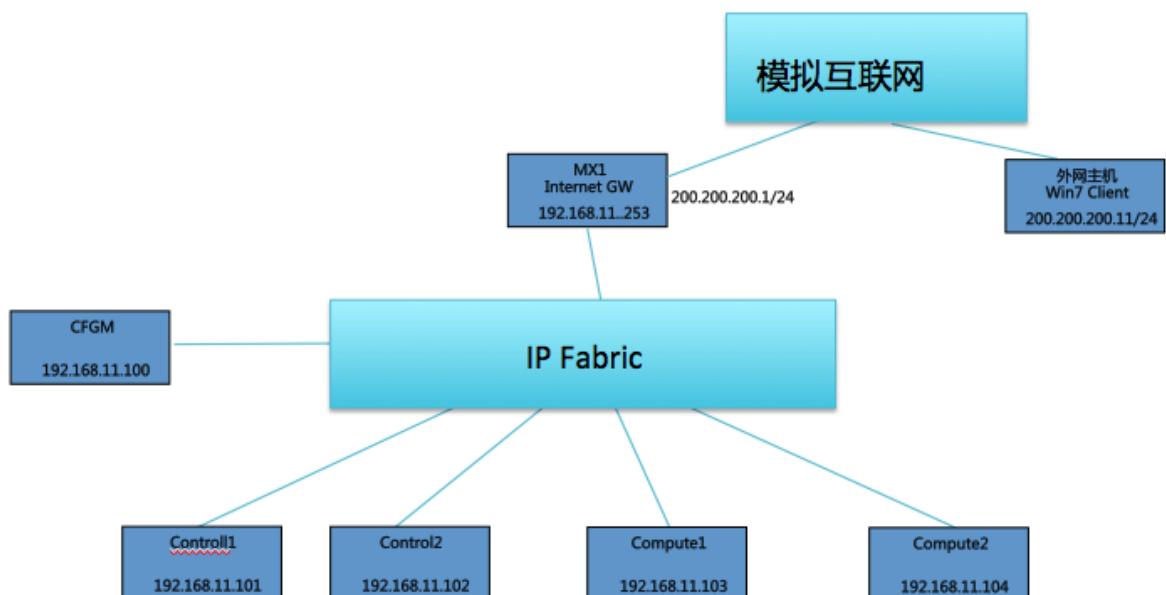
Web 接口，也是在 *cfgm* 上面启用的，访问就从这里开始

- *GW*—连接外部互联网的网关，这个也是很有意思，先让大家看到，在第二篇里面会详细说明

对架构还有希望进一步了解的，可以参考SDNAP上的文档

<http://www.sdnap.com/sdnap-post/2886.html>

基于此，我们可以搭建这样一个基本的环境



从这个基本拓扑中可以看到，我们需要部署五个服务器，服务器之间三层互联，其中一个 Juniper 的 MX 路由器作为出口的网关（有客官问为什么是 Juniper 的，不是思科华为的，原因是我没有，谁问谁送我，多谢）。

根据上面的组件和拓扑的介绍，我们可以看到

- Server1(cfgm): 含有 Horizon 组件的 openstack、webui、config-node, api-server、collector 和 analytics
- Server2 (control node1) : 控制器节点
- Server3 (control node2) : 控制器节点
- Server4 (compute node1) : 计算节点，包括 vrouter、vagent 和 openstack nova
- Server5 (compute node2) : 计算节点，包括 vrouter、vagent 和 openstack nova

这部分配置其实在后面可以在 cfgm 的 testbed 文件上看到，testbed 是什么暂且不表  
开始配置虚拟机

按照实际部署的场景，这些 Server 都应该是物理服务器，但是以做实验为目的的话，  
五台服务器确实有点烧包，所以我们选择 ESXi 作为载体，在这上面虚拟 5 台 Server，然后  
在这些虚拟 Server 上再去起虚拟机，您也可以理解为虚拟机上面再套虚拟机，说的这么繁  
琐是因为这样的功能是需要一个特殊的小配置，对于五台虚拟机的配置我的配置如下

cfgm: 虚拟 CPU 2 个 硬盘 100G 内存 12G-16G 单网卡 (cfgm 的内存不能小于 8G)

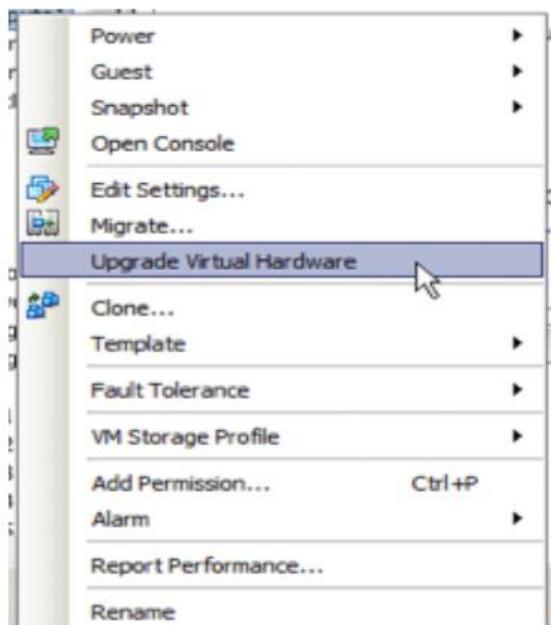
control1: 虚拟 CPU 2 个硬盘 30G 内存 2GG 单网卡

control2: 虚拟 CPU 2 个硬盘 30G 内存 2GG 单网卡

Compute1: 虚拟 CPU 4 个 硬盘 100G 内存 8G 单网卡 (因为上面还是要创建虚拟机，  
所以内存还是要大一些)

Compute2: 虚拟 CPU 4 个 硬盘 100G 内存 8G 单网卡 (因为上面还是要创建虚拟机，  
所以内存还是要大一些)

根据这个硬件配置创建五个虚拟机，启动选择你已经下载好的 DVD Image，  
这里面要注意刚才说到的问题，因为我们需要在虚拟机上面再做一次虚拟环境，  
所以虚拟机需要配置支持硬件虚拟化，在 Vmware 中，如果是用 Vmware Client  
创建的虚拟机的硬件版本为 Version8，是用 Vcenter Web Client 创建的虚拟机是  
Version 10，如果你创建的虚拟机版本是 Version 8 的话，请虚拟机点击右键选择  
升级硬件类型，如下图所示



但是这里有个坑儿，升级了硬件版本之后，这个虚拟机就只能在 Vcenter Web Client 上进行管理了，所以这也是为什么要有 Vcenter 存在的原因

进入 Web Client，点击虚拟机进行编辑

然后展开 CPU 选项，把箭头所指的内容选上

Virtual Hardware		VM Options	SDRS Rules	vApp Options
<div style="background-color: #ffffcc; padding: 5px;"> <input checked="" type="checkbox"/> *CPU      4      ⓘ         </div>				
Cores per Socket: 1      Sockets: 4				
CPU Hot Plug: <input type="checkbox"/> Enable CPU Hot Add				
Reservation: 0 MHz				
Limit: Unlimited MHz				
Shares: Normal 4000				
CPUID Mask: Expose the NX/XD flag to guest      ⓘ Advanced...				
Hardware virtualization (*): <input checked="" type="checkbox"/> Expose hardware assisted virtualization to the guest OS      ⓘ				

这个选项如果不选，在安装过程中不会遇到问题，但是当 compute node 创建虚拟机的时候会出现问题，所以还是选上最好（Compute node 选即可，其他节点都选也无所谓），另外如果这里不选，还有一个解决方法，尽管不推荐但是也写上去，是 openstack 上面的一个配置，如下面介绍

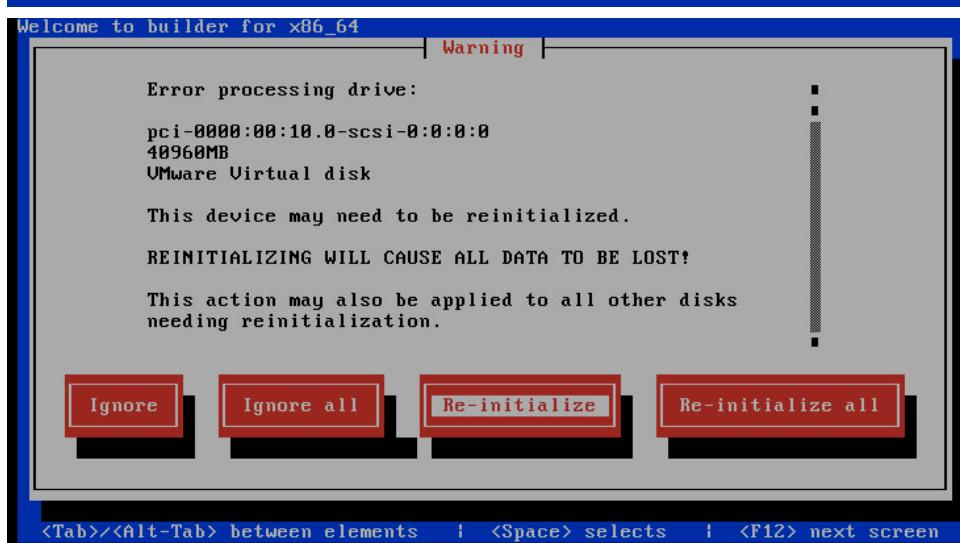
As you are running OpenStack within a virtual machine you need to set this in /etc/nova/nova.conf on your compute host(s) to use QEMU:

libvirt\_type=qemu

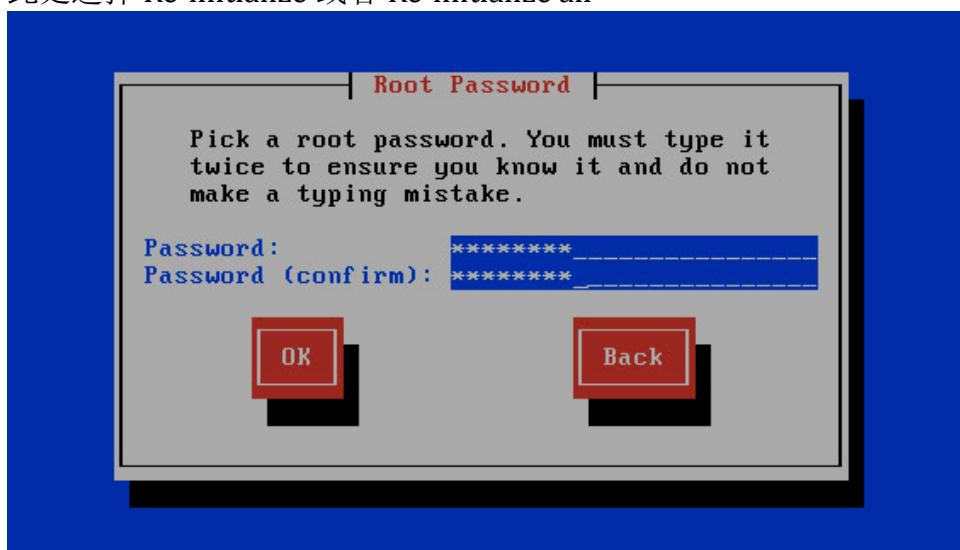
Currently the compute host is trying to use KVM but the virtualization extensions KVM requires are not available, because you are running in a virtual machine and nested virtualization is not enabled/available.

我们可以开机并且安装 Contrail Image 的，这个安装就轻车熟路了，一路 Next 即可，截图如下

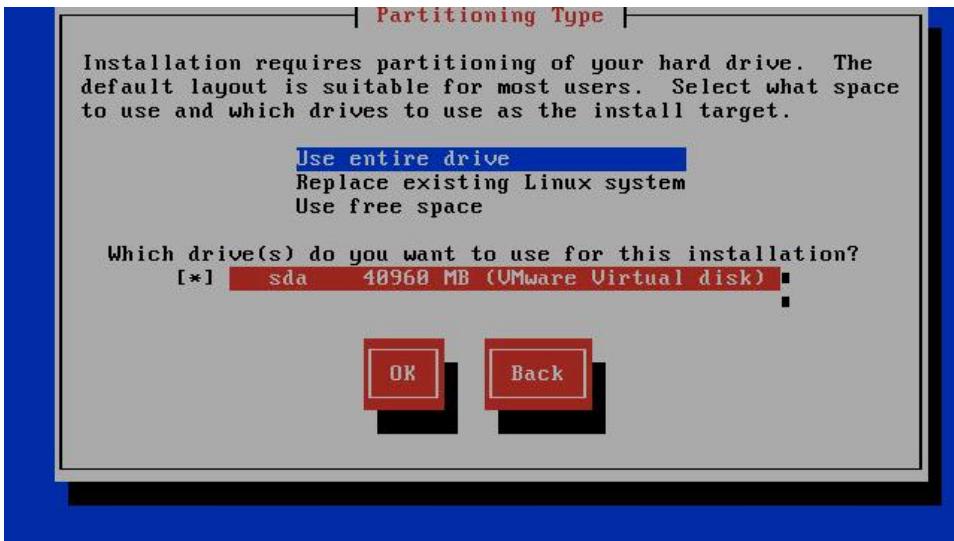




此处选择 Re-initialize 或者 Re-initialize all



设置密码，自己记住



五台设备安装完毕之后，就可以部署 Contrail 了

第一步，我们需要给每个 Server 配置 IP，是用 Web Client 的 console 到每个 Server 上，分别修改如下文件

```
vi /etc/sysconfig/network-scripts/ifcfg-eth0    通常是eth0
以CFG为为例，其他根据这个进行配置
BOOTPROTO="static"
DEVICE="yyy"
ONBOOT="yes"
IPADDR=192.168.11.100
NETMASK=255.255.255.0
GATEWAY=192.168.11.254
NETWORK=192.168.11.0
BROADCAST=192.168.11.255
```

第二步，修改主机名

```
vi /etc/sysconfig/network"    不同的 node 主机名不一致
node1: HOSTNAME=cfgm.dc1
node2: HOSTNAME=control1.dc1
node3: HOSTNAME=control2.dc1
node4: HOSTNAME=compute1.dc1
node5: HOSTNAME=compute2.dc1
```

第三步，修改 hosts 文件

```
vi /etc/hosts
192.168.11.100 cfgm cfgm.dc1
192.168.11.101 control1 control1.dc1
```

```
192.168.11.102 control2 control2.dc1  
192.168.11.103 compute1 compute1.dc1  
192.168.11.104 compute2 compute2.dc1  
192.168.11.254 mx1 mx1.dc1
```

如上三个步骤需要在每台 **Server** 上都进行配置，做了这个配置目的之一是满足设备的连通性，其次，在 **cfgm** 上进行统一部署的时候，这些都非常关键。  
做完之后所有主机都重启一遍

**Note:** 如果这部分配置没问题，相互也可以 ping 通，那么我强烈建议，强烈强烈建议是用 **ESXi** 做一下快照（snapshot），免去日后重装之苦啊

开始正式安装 **Contrail**

安装和部署 **Contrail**，可以直接在 **cfgm** 上进行，  
在**CFGM**上运行下面两条命令

```
cd /opt/contrail/contrail_packages  
. ./setup.sh
```

这部分只是一个洗刷刷的过程，后面的才正式安装

```
cd /opt/contrail/utils/fabfile/testbeds
```

进入这个文件夹，**ls** 之后可以看到很多**.py** 的文件，前面说到 **testbed.py** 是一个很重要的文件，实际上这就是整个 **contrail** 操作和部署的脚本

**cp testbed\_multibox-example.py testbed.py** 我们可以 **copy** 一个模板，然后在这个模板的基础上进行调整，关于这部分文档的参考，请参见

[http://www.juniper.net/techpubs/en\\_US/contrail1.0/topics/task/installation/testbed-file-vnc.html](http://www.juniper.net/techpubs/en_US/contrail1.0/topics/task/installation/testbed-file-vnc.html)

我先把我做的拿出来 share 一下

这部分就不做太多解释了，大家依样画葫芦就可以，后面的 **option** 部分需要的时候在做配置

```
[root@cfgm ~]# cd /opt/contrail/utils/fabfile/testbeds/  
[root@cfgm testbeds]# more testbed.py  
from fabric.api import env  
  
#Management ip addresses of hosts in the cluster  
host1 = 'root@192.168.11.100'  
host2 = 'root@192.168.11.101'  
host3 = 'root@192.168.11.102'
```

```

host4 = 'root@192.168.11.103'
host5 = 'root@192.168.11.104'

#External routers if any
#for eg.
#ext_routers = [('mx1','10.204.216.253')]
ext_routers = [('mx1','192.168.11.253')]

#Autonomous system number
router_asn = 64512

#Host from which the fab commands are triggered to install and provisi
host_build = 'root@192.168.11.100'

#Role definition of the hosts.
env.roledefs = {
    'all': [host1, host2, host3, host4, host5],
    'cfgm': [host1],
    'openstack': [host1],
    'control': [host2, host3],
    'compute': [host4, host5],
    'collector': [host1],
    'webui': [host1],
    'database': [host1],
    'build': [host_build],
}

env.hostnames = {
    'all': ['cfgm', 'control1', 'control2','compute1', 'compute2']
}

env.password = 'jnpr1234'
#Passwords of each host
env.passwords = {
    host1: 'jnpr1234',
    host2: 'jnpr1234',
    host3: 'jnpr1234',
    host4: 'jnpr1234',
    host5: 'jnpr1234',

    host_build: 'jnpr1234',
}

#OPTIONAL BONDING CONFIGURATION

```

```

#=====
#Interface Bonding
#bond= {
#    host2 : { 'name': 'bond0', 'member': ['p2p0p0','p2p0p1','p2p0p2',
#        'mode':'balance-xor' },
#    host5 : { 'name': 'bond0', 'member': ['p4p0p0','p4p0p1','p4p0p2',
#        'mode':'balance-xor' },
#}

#OPTIONAL SEPARATION OF MANAGEMENT AND CONTROL + DATA
#=====
#Control Interface
#control = {
#    host1 : { 'ip': '192.168.10.1/24', 'gw' : '192.168.10.254', 'devi
#        },
#    host2 : { 'ip': '192.168.10.2/24', 'gw' : '192.168.10.254', 'devi
#        p0' },
#    host3 : { 'ip': '192.168.10.3/24', 'gw' : '192.168.10.254', 'devi
#        },
#    host4 : { 'ip': '192.168.10.4/24', 'gw' : '192.168.10.254', 'devi
#        },
#    host5 : { 'ip': '192.168.10.5/24', 'gw' : '192.168.10.254', 'devi
#        1' },
#    host6 : { 'ip': '192.168.10.6/24', 'gw' : '192.168.10.254', 'devi
#        },
#    host7 : { 'ip': '192.168.10.7/24', 'gw' : '192
#        },
#    host8 : { 'ip': '192.168.10.8/24', 'gw' : '192
#        },
#}

#Data Interface
#data = {
#    host2 : { 'ip': '192.161.10.1/24', 'gw' : '192
#        },
#    host5 : { 'ip': '192.161.10.2/24', 'gw' : '192
#        },
#}

#To disable installing contrail interface rename pa
#env.interface_rename = False

#To enable multi-tenancy feature
#multi_tenancy = True

```

```
#To Enable parallel execution of task in multiple nodes
#do_parallel = True
[root@cfgm testbeds]#
```

查看没有问题之后，我们就可以进行安装和部署了

```
cd /opt/contrail/utils
```

这里面我们要执行一个命令叫 **fab**，在做这个之前，大家可以到这个文件夹下看一下介绍

```
/opt/contrail/utils/README.fabric
```

安装 contrail package

```
cfgm下面， cd /opt/contrail/utils
```

```
cfgm下执行 fab install_contrail
```

这时cfgm会登陆到每个server上进行安装，安装完毕后

```
cfgm下执行fab setup_all
```

稍安勿躁，喝杯水吧少年，中间compute node还要重启一下，如果没有问题，安装基本就此完成

如何登陆呢，对于本次案例

OpenStack的登陆地址 <http://192.168.11.100> admin/contrail123

Contrail的登陆地址 <http://192.168.11.100:8080/> admin/contrail123

截图如下Openstack

The screenshot shows the OpenStack dashboard interface. At the top, there's a navigation bar with links for Usage, Overview, OpenStack, Contrail Control, Login - OpenStack, and HFS /OS. Below the navigation bar, the main content area has a large "openstack" logo and a "DASHBOARD" button. On the left, there's a "System Panel" sidebar with tabs for Project (which is selected), Admin, Instances, Volumes, and Flavors. The main content area is titled "Overview". It features a section to "Select a month to query its usage:" with dropdown menus for December and 2013, and a "Submit" button. Below this, it displays "Active Instances: 4 Active RAM: 8GB This Month's VCPU-Hours: 104.99 This Month's Disk: 80GB". A "Usage Summary" table follows, showing data for a single project named "demo":

Project Name	VCPUs	Disk	RAM	VCPU Hours
demo	4	80	8GB	104.99

At the bottom of the summary table, it says "Displaying 1 item".

Contrail

The screenshot shows the Juniper Contrail Controller dashboard. On the left, there's a navigation sidebar with 'Monitor' selected, followed by 'Infrastructure' which is expanded to show 'Dashboard', 'Control Nodes', 'Virtual Routers', 'Analytics Nodes', and 'Config Nodes'. Below this is 'Networking' and 'Debug'. The main area has a title 'Monitor > Infrastructure > Dashboard'. It features several summary boxes: one for 'vRouters' (2), 'Control Nodes' (2), 'Analytics Nodes' (1), and 'Config Nodes' (1). Another box shows 'Instances' (4), 'Interfaces' (4), and 'VNs' (2). To the right is a scatter plot with 'Memory (MB)' on the y-axis (ranging from 230 to 270) and 'CPU (%)' on the x-axis (ranging from 0.00 to 0.15). Two green dots are plotted at approximately (0.08, 255) and (0.13, 255). At the bottom, there are tabs for 'Logs', 'System Information', and 'Alerts', each showing 'No [Logs/Alerts] found.' and 'No of servers 5'.

在contrail上先别关注instance的数量，我的截图是建立了虚拟机和虚拟网络之后的现实，基本上你能够看到下面这个图，并且在Log上和Alerts上没有什么报错，就证明安装成功了

This screenshot shows the same Juniper Contrail Controller dashboard as the previous one, but it appears to be a later state where the system is fully operational. The summary boxes show the same counts: 2 vRouters, 2 Control Nodes, 1 Analytics Nodes, and 1 Config Node. The 'Logs' and 'Alerts' sections at the bottom are both empty, indicating no errors or alerts have been generated.

接下来，可以随便逛逛，看看一些页面是否正常，这里需要注意一下，因为Openstack的Web console（就是建立虚拟机之后查看screen的东东）和Contrail都是基于HTML5的，所以，强烈建议是用chrome登陆配置，我因为这个问题还重装了一遍（是的你没看错，我真的按照这十来页文档又重装了一遍），血泪的教训啊，切记切记。

至此，Contrail Demo搭建完毕，下一步介绍一下基本界面和原理

别忘了做snapshot哦骚年：）

作者: KkBluE ( SDN QQ 群#Kkblue [@KkBLeE 知行合一](#) 微信公众平台 :kkbluepublic )