**OneStack：Ubuntu 12.04 上一键自动部署 OpenStack**

前几天 [OneStack 项目](http://code.google.com/p/onestack/) 的作者 Kayven [在 vpsee.com 上留言](http://www.vpsee.com/2012/02/use-devstack-to-deploy-openstack-quickly/)谈到了 OneStack，一个国人的 OpenStack 一键安装工具，在 vpsee 的忽悠下 Kayven 终于同意写一篇 OneStack 的介绍性文章，如果大家对[手动安装配置 OpenStack](http://www.vpsee.com/2011/05/install-openstack-nova-on-ubuntu/) 有恐惧的话可以试试这个 OneStack 一键安装工具，类似的项目还有 [DevStack](http://www.vpsee.com/2012/02/use-devstack-to-deploy-openstack-quickly/).

大家如有问题欢迎参与讨论或联系原作者 Kayven (Hily.Hoo@gmail.com). 以下内容来自 Kayven：

在发表了 [OneStack: Ubuntu 12.04 (“Precise”) 一键安装部署云计算平台 OpenStack Essex](http://blog.csdn.net/hilyoo/article/details/7696169) 这篇文章、公布了 OneStack 这个项目后，受到一些人的关注和邀请，诚惶诚恐，非常感谢大家的支持，下面将对这个项目进行更详细的介绍。

**OneStack 的引入**

为什么需要 OpenStack？作为众多云计算项目的一个，OpenStack 很火。

一是因为 OpenStack自身的优势、Apache2.0 授权的开源性以及兼容性、灵活性和可扩展性等优点；  
二是众多企业和组织的参与开发，尤其是世界领军企业的加入，推动了 OpenStack 的高速成长。

为什么需要 OneStack？类似项目有 DevStack，但是使用 DevStack 有如下问题：

部署过错的可定制性和灵活性不是太好，自己只能选择安装哪些服务，如果中间遇到问题或者自己想调整就比较麻烦；  
使用 screen 管理运行 OpenStack，重启服务器需要用 screen 进入，很多人以为有些服务会停止或者希望不使用 screen，于是自己 kill 服务并自己手动开启，容易出各种问题（OpenStack 由很多独立组件和服务组成，注意不要遗漏）；  
没有提供重启、重置、清空数据库等有用功能，还稍显复杂；  
而且，使用 DevStack 后还是不清楚整个部署过程是怎样的，自己不能安装官方安装文档来实验和尝试；  
由于组件独立分散，安装过程过于繁琐，可以抽象成通用的项目供大家方便使用；  
官方提供了一个比较完善的入门文档，但是，按照这几十页的步骤下来需要做很多无用功，容易漏错而引起很多莫名和头疼的问题；  
本项目希望不只是提供实验环境，更可以实际部署使用，可以自己修改配置，按需增加组件和功能，实现一键部署，可扩展、可添加任意计算节点。

为什么需要一键自动部署工具？

很多人首先希望尝试一下 OpenStack，做做实验，弄清楚具体怎么实践。官方文档的一大堆步骤会让人忘而生畏；同时又不想部署好后都不知道到底怎么做的，像 DevStack 这样封装比较难看懂，也就难自己修改。OneStack 能够很好的自动部署，同时又能灵活的实验，对于大部分尝试者是个很好的途径。

为什么使用 Ubuntu 12.04？

OpenStack 官方指定的操作系统是 Ubuntu，当然也可以使用其他的，比如 CentOS，不过安装过程有可能会不同。OpenStack 目前主要是以 Ubuntu 版本 Linux 系统为基础写成的，而且很多测试和文档都是在 Ubuntu 下完成的，所以在 Ubuntu 下部署将会有很多便利。另外，Ubuntu 12.04不仅是LTS（长期支持版本），还可以得到五年的支持，对于开发者是个不错的平台。

**OneStack 的项目结构**

* oneStack.sh（一键部署 all-in-one 的 OneStack，最主要文件）；
* addComputeNode.sh（增加计算节点）；
* delStack.sh（只卸载nova、glance、keystone等）；
* delAll.sh（卸载所有安装的组件和工具）；
* resetStack.sh（清空数据库，镜像、网络和实例等）；
* addClient.sh（添加客户端，nova管理等）；
* setup\_base.sh（安装基本系统）；
* setup\_test.sh（添加镜像和实例）；
* HAStack 目录（OneStack 的高可用性，希望更多人可以提出自己的解决方案）。

**OneStack 的安装部署**

可以一键自动部署 all-in-one 的 OneStack 实验环境，也可以分步骤部署（下次再讨论分步骤部署）。  
一键自动部署最简单，只需要文件 oneStack.sh 把所有服务安装到一个机器。

# wget http://onestack.googlecode.com/files/oneStack.sh && \

chmod +x oneStack.sh && ./oneStack.sh

如果需要更多功能，需要 chechout 整个 svn；当然，安装同样只需要 oneStack.sh  
1、安装 Ubuntu Precise (12.04)；  
2、下载 OneStack 脚本：

# svn checkout http://onestack.googlecode.com/svn/trunk/ onestack-read-only

3、运行 OneStack：

# cd onestack-read-only/ && ./oneStack.sh

注意：其实上面的安装还是需要更改网络配置的（其余可以不改，这个是需要改成你自己的）因为，为了简单，在上面的工具里，所有前期工作都加到了文件 oneStack.sh，比如：

* root 用户密码设置（刚安装的 Ubuntu 默认不启用这个 root 用户）；
* apt 源的配置，可以设置为国内的 163、ustc 的源等；
* 网络配置，控制节点是需要外网 ip 的，你需要更改oneStack.sh里面的一些配置：/etc/network/interfaces 里面双网卡的 ip、网关等，在脚本靠前的位置，请查找 interfaces. 参数设置：外网 ip 地址等，这些也都在脚本开头一个块里面。自行检查下面 network/interfaces 的两个网卡设置：
* ## 2、自行检查下面 network/interfaces的两个网卡设置
* + OUT\_IP=192.168.139.50
* + OUT\_IP\_PRE=192.168.139
* ...
* 选择虚拟机技术，裸机使用 kvm，虚拟机使用 qemu 即可  
  ## 选择虚拟技术，裸机使用 kvm，虚拟机里面使用 qemu  
  VIRT\_TYPE=”qemu”
* 数据库的安装和配置，为了自动化部署，参数设置里面设置好帐号和密码，后面就不需要交互；## 配置 /etc/nova/nova.conf，这里与控制节点的配置相同！比如ip是控制节点的ip  
  MYSQL\_PASSWD=${MYSQL\_PASSWD:-“cloud1234″}  
  NOVA\_DB\_USERNAME=${NOVA\_DB\_USERNAME:-“novadbadmin”}  
  NOVA\_DB\_PASSWD=${NOVA\_DB\_PASSWD:-“cloud1234″}
* 系统会安装 Ubuntu 12.04 的镜像，并启动一个实例。这个过程中镜像自动从 Ubuntu 官网下载，可以查找 cloud-images 更换地址或者镜像 precise-server-cloudimg-amd64-disk1.img，也可以注释掉这个步骤，直接使用 dashboard 在 web 添加镜像启动实例。这个镜像有700多 MB，对于网速不好的用户，可能需要较长时间，因此可以先下载好镜像，然后把这里的地址改成本地即可。

总结一下需要设置的参数：

* 设置 root 密码这一步可以删掉，使用 root 执行即可；
* 可选，如果不需要跳过本步骤  
  系统语言设置，可以参考oneStack.sh locale部分，不在此介绍  
  设置apt源 /etc/apt/sources.list
* 设置网络  
  /etc/network/interfaces  
  可以参考oneStack.sh locale部分
* 配置参数，除了网络ip，其它可以不变  
  ## 数据库  
  MYSQL\_PASSWD=${MYSQL\_PASSWD:-“cloud1234″}  
  ## 自行检查下面network/interfaces的两个网卡设置与此处一致  
  OUT\_IP=”192.168.139.50″  
  ## 选择虚拟技术，裸机使用kvm，虚拟机里面使用qemu  
  VIRT\_TYPE=”qemu”  
  ## token, 登录dashboard密码  
  ADMIN\_TOKEN=”admin”
* 然后执行./oneStack.sh安装即可。

**OneStack 的展望**

1、加入高可用性 OpenStack 的部署  
详见[构建 OpenStack 的高可用性（HA，High Availability）](http://blog.csdn.net/hilyoo/article/details/7704280)对高可用性OpenStack的讨论。对照 CAP 理论，OpenStack 的分布式对象存储系统 Swift 满足了可用性和分区容忍性，没有保证一致性（可选的），只是实现了最终一致性。对于 Swift 的研究和学习网上很多，我不做介绍。但是，在整个 OpenStack 架构中，要满足高可用性需要进行很多工作来保证。主要是通过分离、冗余技术实现，也就是 nova-api、nova-network、glance 等可以分别在多节点上工作，RabbitMQ 可以工作在主备模式，MySQL 可以使用冗余的高可用集群。这些组合可能有很多问题，有些也需要加入到 OpenStack 项目。

2、加入对 Ubuntu 以外的操作系统（如 CentOS）的支持  
个人精力有限，所以没有对 CentOS 等其它版本进行支持，也没有对 Ubuntu11 等版本进行测试。但是大家应该只需要把 OneStack 稍加改动就可以用到这些版本的操作系统。因此，如果有人有改好的，可以拿出来分享，别人也也可以顺便帮你改善和讨论。

3、希望更多的有时间的同行参与  
正如上面所说，个人精力有限，业余所做，肯定有诸多不足，而且对其它版本没有添加支持，更主要的，希望对高可用性（HA）这个很关键的要求实现自动化部署，因此希望多提出意见建议、多分享自己的经验和成果，造福别人也提高自己。

分类：[Cloud | Virtualization | Thin | HPC](http://www.vpsee.com/category/cloud/) | [评论(11)](http://www.vpsee.com/2012/07/onestack-all-in-one-installation-tool-for-openstack/#comments)

[**使用 DevStack 快速搭建 OpenStack 实验环境**](http://www.vpsee.com/2012/02/use-devstack-to-deploy-openstack-quickly/)

2012年02月16日 | 标签: [devstack](http://www.vpsee.com/tag/devstack/), [openstack](http://www.vpsee.com/tag/openstack/), [openstack dashboard](http://www.vpsee.com/tag/openstack-dashboard/)

[安装 OpenStack](http://www.vpsee.com/2011/05/install-openstack-nova-on-ubuntu/) 对新手来说是个痛苦的过程，尤其是 OpenStack 和它依赖的一些组件在快速发展中，经常出现这个版本 keystone 对不上那个版本 dashboard 类似的情况。如果只是想看看 OpenStack 的话，使用 DevStack 是个不错的办法。[DevStack](http://devstack.org/) 实际上是个 shell 脚本，可以用来快速搭建 OpenStack 的运行和开发环境，特别适合 OpenStack 开发者下载最新的 OpenStack 代码后迅速在自己的笔记本上搭建一个开发环境出来。正如 DevStack 官方所强调的，devstack 不适合用在生产环境。

Can I use DevStack for production?  
A: No. We mean it. Really. DevStack makes some implementation choices that are not appropriate for production deployments. We warned you!

维护一个支持多 Linux 发行版的脚本需要很多工作，为了保持简单，DevStack 目前只支持 Ubuntu 11.10 发行版。使用 devstack 脚本很简单，安装 git，然后下载 devstack 代码到本地，运行 stack.sh 脚本依次设定 MySQL, RabbitMQ, OpenStack Dashboard 和 Keystone 的密码，密码输入后 stack.sh 脚本会自动开始安装必要的软件包和库并下载最新的 OpenStack 及其组件代码，整个过程自动完成无需干预：

$ sudo apt-get install git

$ git clone git://github.com/openstack-dev/devstack.git

$ cd devstack/

$ ./stack.sh

################################################################################

ENTER A PASSWORD TO USE FOR MYSQL.

################################################################################

This value will be written to your localrc file so you don't have to enter it

again. Use only alphanumeric characters.

If you leave this blank, a random default value will be used.

Enter a password now:

...

horizon is now available at http://192.168.2.240/

keystone is serving at http://192.168.2.240:5000/v2.0/

examples on using novaclient command line is in exercise.sh

the default users are: admin and demo

the password: vpsee

This is your host ip: 192.168.2.240

stack.sh completed in 684 seconds.

$

如果对默认的 tty.tgz 镜像不满意，想用 ubuntu 11.10 镜像的话编辑 stack.sh 脚本，在 for image\_url in ${IMAGE\_URLS//,/ }; do 一行上面加入 IMAGE\_URLS=http://uec-images.ubuntu.com/oneiric/current/oneiric-server-cloudimg-amd64.tar.gz 这行，然后重新运行 stack.sh 脚本：

$ vi stack.sh

...

IMAGE\_URLS=http://uec-images.ubuntu.com/oneiric/current/oneiric-server-cloudimg-amd64.tar.gz

for image\_url in ${IMAGE\_URLS//,/ }; do

# Downloads the image (uec ami+aki style), then extracts it.

IMAGE\_FNAME=`basename "$image\_url"`

if [ ! -f $FILES/$IMAGE\_FNAME ]; then

wget -c $image\_url -O $FILES/$IMAGE\_FNAME

fi

KERNEL=""

RAMDISK=""

...

$ ./stack.sh

还有一点值得注意的地方，如果想使用现有的网络，比如安装 devstack 的这台服务器地址是 192.168.2.240，想让所有运行在上面的 instance 的地址为 192.168.2.241-192.168.2.247 的话，需要修改 stack.sh 里面的 TEST\_FLOATING\_RANGE，然后重新运行 stack.sh 脚本：

$ vi stack.sh

...

#TEST\_FLOATING\_RANGE=${TEST\_FLOATING\_RANGE:-192.168.253.0/29}

TEST\_FLOATING\_RANGE=${TEST\_FLOATING\_RANGE:-192.168.2.240/29}

...

$ ./stack.sh

DevStack 安装结束后打开浏览器访问 http://192.168.2.240/ 就可以看到 OpenStack Dashboard 界面，用户名是 admin，密码是刚才设定的密码，输入正确后就会进入 Dashboard，最新的 OpenStack Dashboard 已经有了很大改善，可以创建或导入 keypairs，启动或终止 instance，分配和注射 IP 给 instance，创建和分配 volume 等，还可以添加和管理用户等。界面如下：

