第四课：openstack排错及使用

- juno版本修改openstack的ip修改需要做的操作
- 各种服务down了的表现（在web页面创建虚拟机失败，能对应感知到出错模块，及如何查看错误日志）

① 创建虚拟机流程：（详见后续ppt）

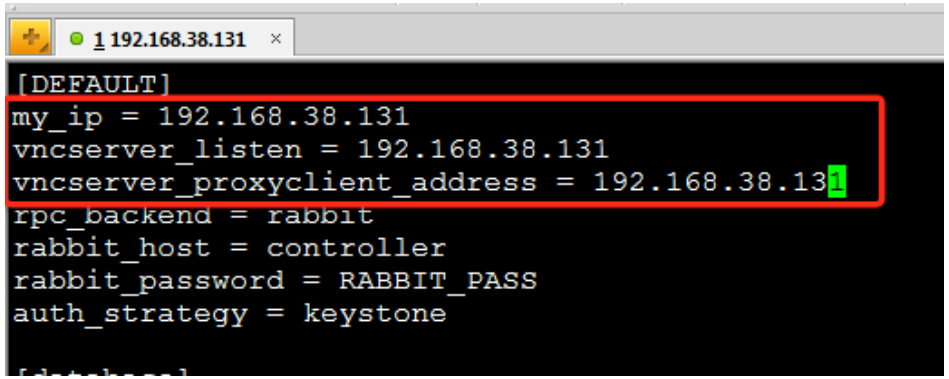
② 前端模块出错、镜像服务出错、网络服务出错、认证服务错误

③ Nova相关：api出错，scheduler错误、compute错误

- openstack管理界面包括哪些基础功能
- openstack环境中创建虚拟机流程，及如何通过openstack来定制特定场景的虚拟机
- 如何配置网络、安全组、能让虚拟机间能正确通信，及虚拟机与外部网络互通

juno版本官方文档all-in-one修改openstack的ip修改需要做的操作

vim /etc/nova/nova.conf



vim /etc/neutron/plugins/ml2/ml2_conf.ini

d. In the [ovs] section, enable tunnels and configure the local tunnel en

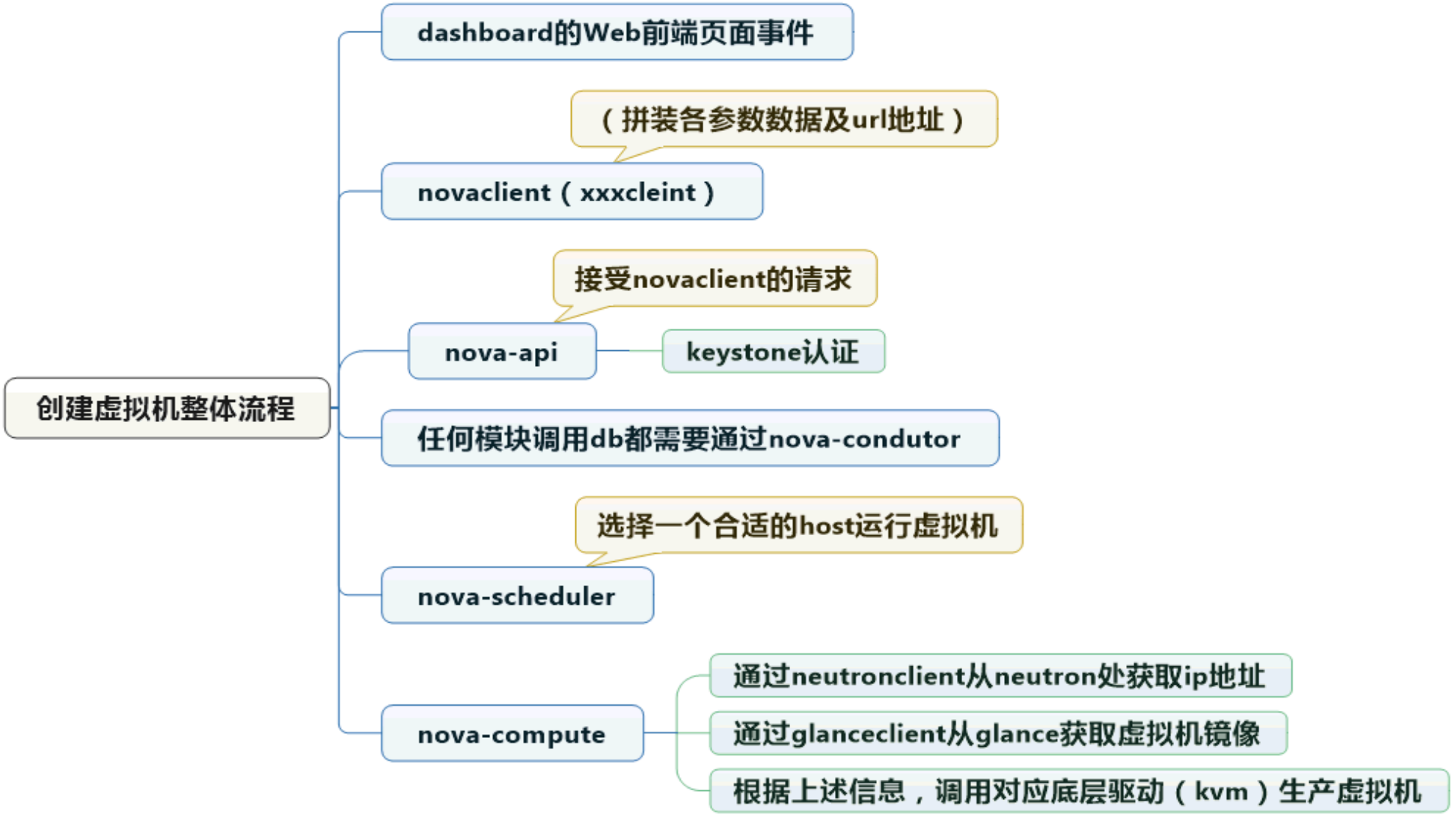
```
[ovs]
...
local_ip = INSTANCE_TUNNELS_INTERFACE_IP_ADDRESS
enable_tunneling = True
```

Replace *INSTANCE_TUNNELS_INTERFACE_IP_ADDRESS* with the IP
the instance tunnels network interface on your compute node.

注：

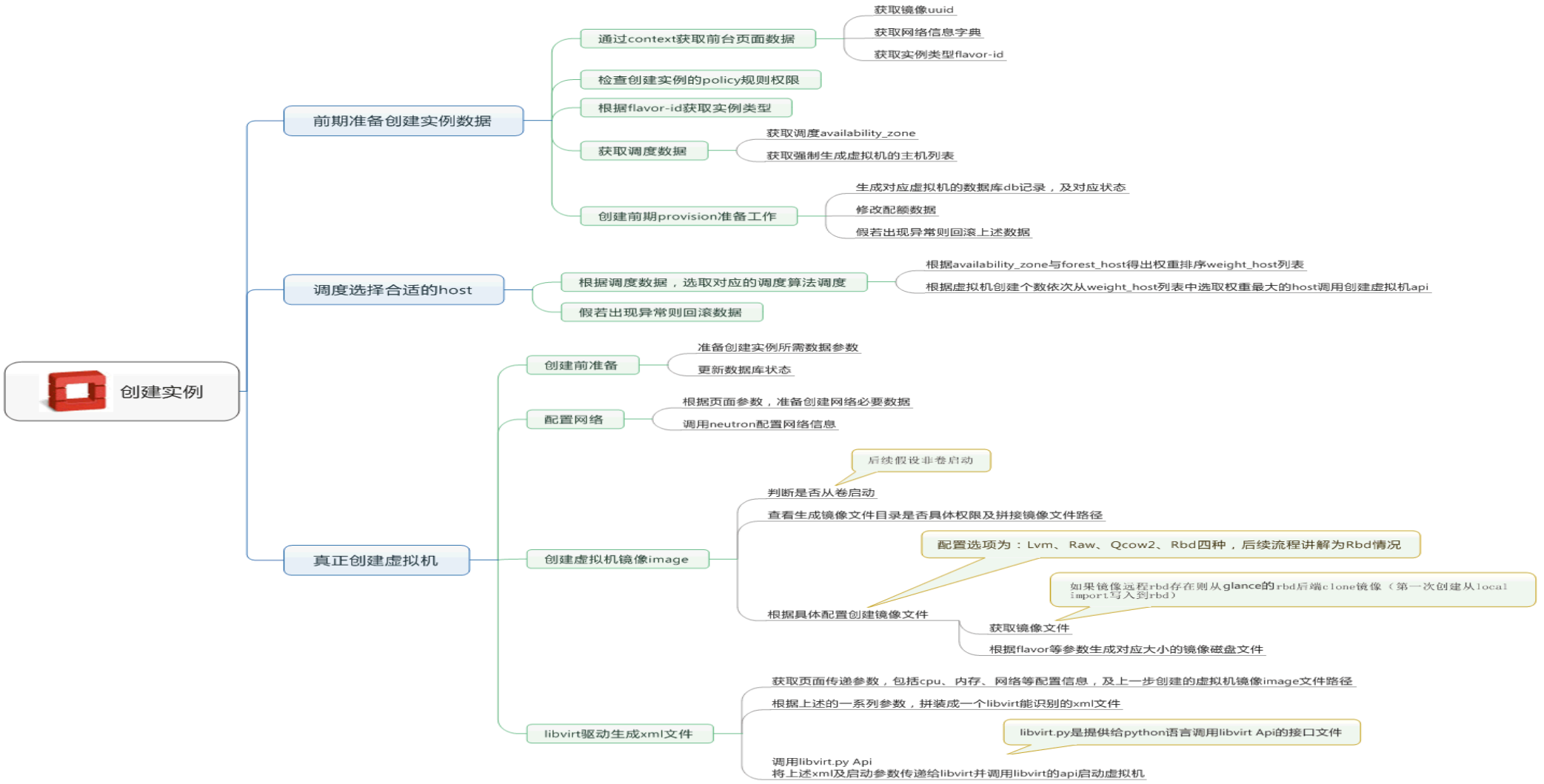
- ✓ 前提是根据官方文档部署（官方文档很多地方使用了controller来代替硬编码的ip地址）
- ✓ 假若使用的是ip地址那么则需要将所有使用ip地址的地方全部做替换（keystone也需要）

创建虚拟机在openstack中主要流程步骤：



Openstack中创建虚拟机主要流程思维导图：

http://blog.csdn.net/tantexian/article/details/44239151



常见排错及服务状态查看

页面登录不进去，关闭iptables 关闭防火墙 iptables –F

查看nova所有服务状态：nova-manage service list或者nova service-list

查看网络代理组件服务状态：neutron agent-list

开启调试log：/etc/xxx/xxx.conf 配置文件的debug=True verbose=True

日志输出路径：/var/log/xxx/xxx.log

注：具体nova、neutron服务状态原理请查看博文：http://blog.csdn.net/tantexian/article/details/39204993

部署过程中排错-1（打开调试模式及keystone 401错误）

Debug模式打开：将配置文件中的verbose = True，debug=true，在命令执行中插入—debug参数，如下：

```
[root@controller ~]# nova --debug list
REQ: curl -i 'http://controller:35357/v2.0/tokens' -X POST -H "Accept: application/json" -H "Content-Type: application/json"
-H "User-Agent: python-novaclient" -d '{"auth": {"tenantName": "admin", "passwordCredentials": {"username": "admin", "passwor
d": "{SHA1}5705cc2e5fda0ab7529d5093c5e389fffe45d615"}}}'
INFO (connectionpool:187) Starting new HTTP connection (1): controller
DEBUG (connectionpool:357) "POST /v2.0/tokens HTTP/1.1" 200 1710
RESP: [200] {'date': 'Wed, 24 Jun 2015 13:10:46 GMT', 'content-type': 'application/json', 'content-length': '1710', 'vary': '
X-Auth-Token'}
```

插入--debug 开启调试模式，将输出更多调试信息

哪些情况会出现401 keystone权限错误？如何排查

```
[keystone_authtoken]
auth_uri = http://controller:5000/v2.0
identity_uri = http://controller:35357
admin_tenant_name = service
admin_user = nova
admin_password = NOVA_PASS
```

此处为nova.conf连接keystone的认证信息，如果输入错误会报401错误

a. Create the nova user:

```
$ keystone user-create --name nova --pass NOVA_PASS
+----------+----------------------------------+
| Property |              Value               |
+----------+----------------------------------+
```

此处为keystone中设置nova的认证账号密码信息

上述两图中账号密码信息不匹配则会报401认证错误。

假若配置账号密码正确还是报错，则尝试用下述命令

更新keystone保存密码：

keystone user-password-update neutron --pass NEUTRON_PASS

```
[root@controller ~]# vim admin-openrc.sh
export OS_TENANT_NAME=admin
export OS_USERNAME=admin
export OS_PASSWORD=ADMIN_PASS
export OS_AUTH_URL=http://controller:35357/v2.0
```

此处不对也可能导致401权限错误，此次admin和ADMIN_PASS及为dashboard登录账号密码

部署过程中排错-2（消息队列出错）

2.消息队列出现问题，连接不上错误：

```
2015-06-24 20:59:38.396 50051 INFO nova.virt.driver [-] Loading compute driver 'libvirt.LibvirtDriver'
2015-06-24 20:59:38.452 50051 INFO oslo.messaging._drivers.impl_rabbit [req-12d91c85-b4ae-4103-a714-2fcace0a160d ] Connecting
 to AMQP server on controller:5672
2015-06-24 20:59:46.395 50051 ERROR oslo.messaging._drivers.impl_rabbit [req-12d91c85-b4ae-4103-a714-2fcace0a160d ] AMQP serv
er controller:5672 closed the connection. Check login credentials: Socket closed
2015-06-24 20:59:47.398 50051 INFO oslo.messaging._drivers.impl_rabbit [req-12d91c85-b4ae-4103-a714-2fcace0a160d ] Delaying r
econnect for 1.0 seconds...
```

```
rpc_backend = rabbit
rabbit_host = controller
rabbit_password = RABBIT_PASS
auth_strategy = keystone
```

nova.conf文件连接rabbit配置，与之前设置必须一致

rabbitmqctl change_password guest *RABBIT_PASS*

rabbitmqctl status | grep rabbit

rabbitmqctl list_queues

systemctl restart rabbitmq-server.service

hostnamectl --static set-hostname controller

```
[root@controller ~]# rabbitmqctl status | grep rabbit
Status of node rabbit@controller ...
Error: unable to connect to node rabbit@controller: nodedown

DIAGNOSTICS
===========

attempted to contact: [rabbit@controller]

rabbit@controller:
  * connected to epmd (port 4369) on controller
  * epmd reports node 'rabbit' running on port 25672
  * TCP connection succeeded but Erlang distribution failed
  * suggestion: hostname mismatch?
  * suggestion: is the cookie set correctly?

current node details:
- node name: 'rabbitmqctl50264@controller-test'
- home dir: /var/lib/rabbitmq
- cookie hash: S4jXqbspw5wi+FcApPTwwQ==

[root@controller ~]#
```

部署过程中排错-3（计算节点down掉）

```
[root@controller ~]# nova-manage service list
Binary           Host                                 Zone             Status     State Updated_At
nova-cert        controller                           internal         enabled    :-)   2015-06-24 13:30:04
nova-consoleauth controller                           internal         enabled    :-)   2015-06-24 13:30:05
nova-scheduler   controller                           internal         enabled    :-)   2015-06-24 13:30:07
nova-conductor   controller                           internal         enabled    :-)   2015-06-24 13:30:08
nova-compute     controller                           nova             enabled    XXX   2015-06-24 13:28:49
[root@controller ~]#
```


```
[root@controller ~]# tailf /var/log/nova/nova-scheduler.log

2015-06-24 21:36:34.411 54184 WARNING nova.scheduler.filters.compute_filter [req-088b3065-c521-47ee-a146-9b693ba24f2e None] (
controller, controller) ram:3273 disk:33792 io_ops:0 instances:0 has not been heard from in a while
2015-06-24 21:36:34.411 54184 INFO nova.filters [req-088b3065-c521-47ee-a146-9b693ba24f2e None] Filter ComputeFilter returned
 0 hosts
```

部署过程中排错-4（计算节点出错-与down掉不同）

```
[root@controller ~]# nova-manage service list
Binary           Host                                 Zone             Status     State Updated_At
nova-cert        controller                           internal         enabled    :-)   2015-06-24 13:42:36
nova-consoleauth controller                           internal         enabled    :-)   2015-06-24 13:42:37
nova-scheduler   controller                           internal         enabled    :-)   2015-06-24 13:42:29
nova-conductor   controller                           internal         enabled    :-)   2015-06-24 13:42:30
nova-compute     controller                           nova             enabled    :-)   2015-06-24 13:42:30
```

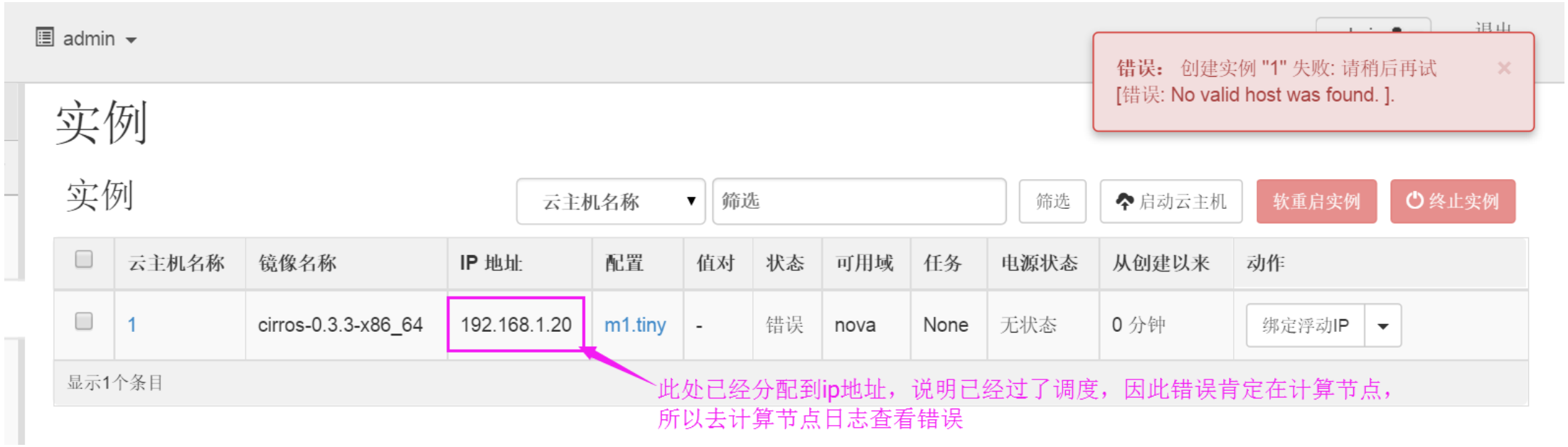
下面人为在计算节点调用的driver.py代码中制造语法错误：

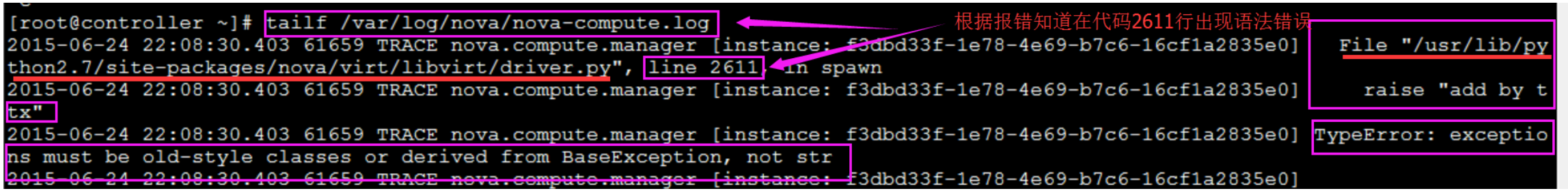
```
    def spawn(self, context, instance, image_meta, injected_files,
              admin_password, network_info=None, block_device_info=None):
        disk_info = blockinfo.get_disk_info(CONF.libvirt.virt_type,
                                            instance,
                                            block_device_info,
                                            image_meta)

        raise "add by ttx"
        self._create_image(context, instance,
                           disk_info['mapping'],
                           network_info=network_info,
                           block_device_info=block_device_info,
                           files=injected_files,
"/usr/lib/python2.7/site-packages/nova/virt/libvirt/driver.py" 6434L, 280437C written
```

在此处故意加入语法错误，来查看生成虚拟机时候计算节点出错，表现。

部署过程中排错-4-续（计算节点出错-与down掉不同）





部署过程中排错-5（全局搜索错误）

其他模块出错，对应表现请自行实验。

如果创建虚拟机出错了，不是很清楚错误出现在哪，可以使用如下全局搜索模块搜索出错ERROR日志：

如果报出错误是比较模糊的，可以自行使用gfsoso.com搜索出错关键字日志。（一般错误都只能在google搜索比较好，但是对于一些低级错误，可能baidu搜索出结果的可能性更大）

通过openstack来定制特定场景的虚拟机

1、openstack镜像制作官方文档：http://docs.openstack.org/image-guide/content/

2、其他镜像制作方法：制作OpenStack CentOS 6.5 & win7镜像

3、通过dashboard做成快照

4、用后台qemu-img命令合并虚拟机文件，再将该镜像文件上传至openstack中。

disk合并base的方法：

qemu-img convert -O raw disk new_img.raw :直接将disk从qcow2转化成raw就可以了

qemu-img commit -f qcow2 disk ： 将disk提交到他的backing_file中，backing_file里将拥有disk里的文件

网络、安全组配置、能让虚拟机间能正确通信，及虚拟机与外部网络互通

1. openstack官方文档配置eth0，eth1（eth1为外网网卡）
2. 根据安装文档，配置好正确的int-net、ext-net
3. 配置安全组
4. 生成虚拟机（选择对应的安全组）
5. 配置防火墙等
6. 虚拟机vm间互相ping、虚拟机ping 真是物理网络

注：由于大部分同学搭建环境出现错误，因此本次课程多花点时间讲解排错。

此页内容放到下一次课中讲解！！！

本次教程中使用的alias自定义列表

http://blog.csdn.net/tantexian/article/details/45058187

```
alias re = 'systemctl restart'
alias echoall='for a in `find /var/log/nova/ -type f`;do echo >
$a;done;
for a in `find /var/log/neutron/ -type f`;do echo > $a;done;'
alias echono='for a in `find /var/log/nova/ -type f`;do echo >
$a;done;'
alias echone='for a in `find /var/log/neutron/ -type f`;do echo >
$a;done;'
alias reli='systemctl restartlibvirtd'
alias reht='systemctl restarthttpd'
alias reap='systemctl restart openstack-nova-api'
alias resc='systemctl restart openstack-nova-scheduler'
alias reco='systemctl restart openstack-nova-compute'
alias rene='systemctl restart neutron-server'
alias reallno='
systemctl restart openstack-nova-api
systemctl restart openstack-nova-cert
systemctl restart openstack-nova-consoleauth
systemctl restart openstack-nova-scheduler
systemctl restart openstack-nova-conductor
systemctl restart openstack-nova-novncproxy
systemctl restart openstack-nova-compute
'
```

```
alias reallne='systemctl restart neutron-openvswitch-agent
neutron-metadata-agent neutron-dhcp-agent neutron-l3-agent
neutron-server'
alias stopall='
systemctl stop openstack-nova-api
systemctl stop openstack-nova-cert
systemctl stop openstack-nova-consoleauth
systemctl stop openstack-nova-scheduler
systemctl stop openstack-nova-conductor
systemctl stop openstack-nova-novncproxy
systemctl stop openstack-nova-compute
'
alias nm='nova-manage service list'
alias grep='grep --color'
alias tailfco='tailf /var/log/nova/nova-compute.log'
alias tailfap='tailf /var/log/nova/nova-api.log'
alias tailfsc='tailf /var/log/nova/nova-scheduler.log'
alias rehe='systemctl start openstack-heat-api.service openstack-
heat-api-cfn.service openstack-heat-engine.service'
```

Thanks