cloudstack

CloudStack技术沙龙

北京站 2012/11/23

@CloudStack中国

cloudstack

4.0新增功能

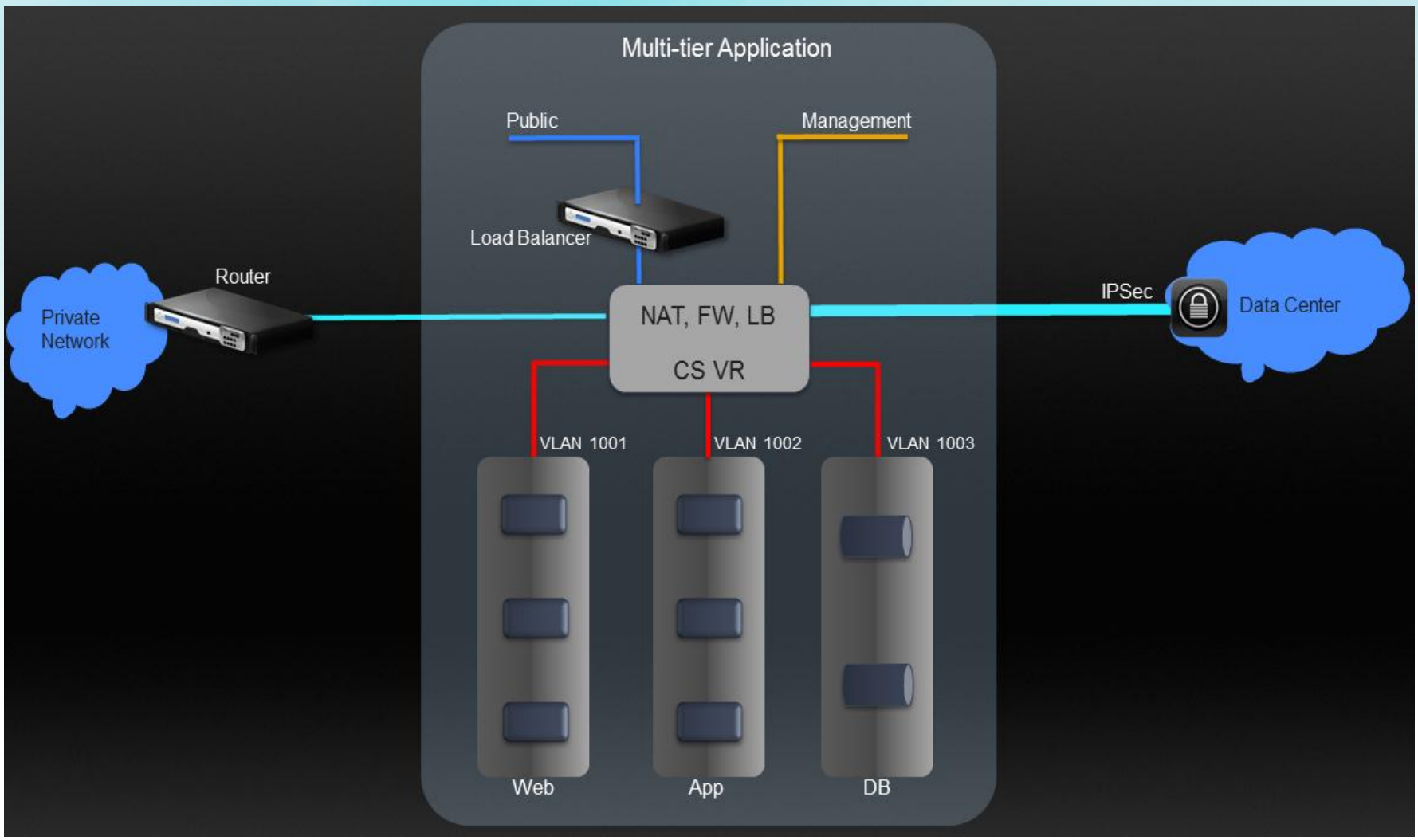
- VLAN之间路由 (VPC)
- Site-to-Site VPN
- 数据盘支持本地存储
- 虚拟资源Tagging
- HA专用主机
- 支持 AWS API
- 支持 Nicira NVP (L2)
- 支持Caringo作二级存储
- KVM Hypervisor 支持升级到 Ubuntu 12.04 和 RHEL 6.3

Apache CloudStack 4.0的工作

- Apache兼容的License(ASL2.0, BSD)
- 调整Plugin及代码架构
- 更改Build系统: ant → maven
- 文档 (docbook, publican, transifex)
- 新功能

VPC (inter-VLAN Routing)

http://wiki.cloudstack.org/display/PM/Inter-VLAN+Routing



VPC (inter-VLAN Routing)

- 在1-N的VPC网络里部署虚机, 虚机里属于N-1的网络可以和属于N的网络通过虚拟路由器进行通信
- 账户A创建的VPC, 只允许账户A的网络加入到这个VPC
- 管理员为账户A创建的VPC, 只允许账户A的网络加入到这个VPC
- 管理员/用户可以添加静态路由(CIDR + GW)来控制网络流量到下列的目的地:
  - —VPN 网关
  - —私有网关
- 网络ACLs - 通过添加网络规则来控制允许/拒绝来宾网络之间的流量

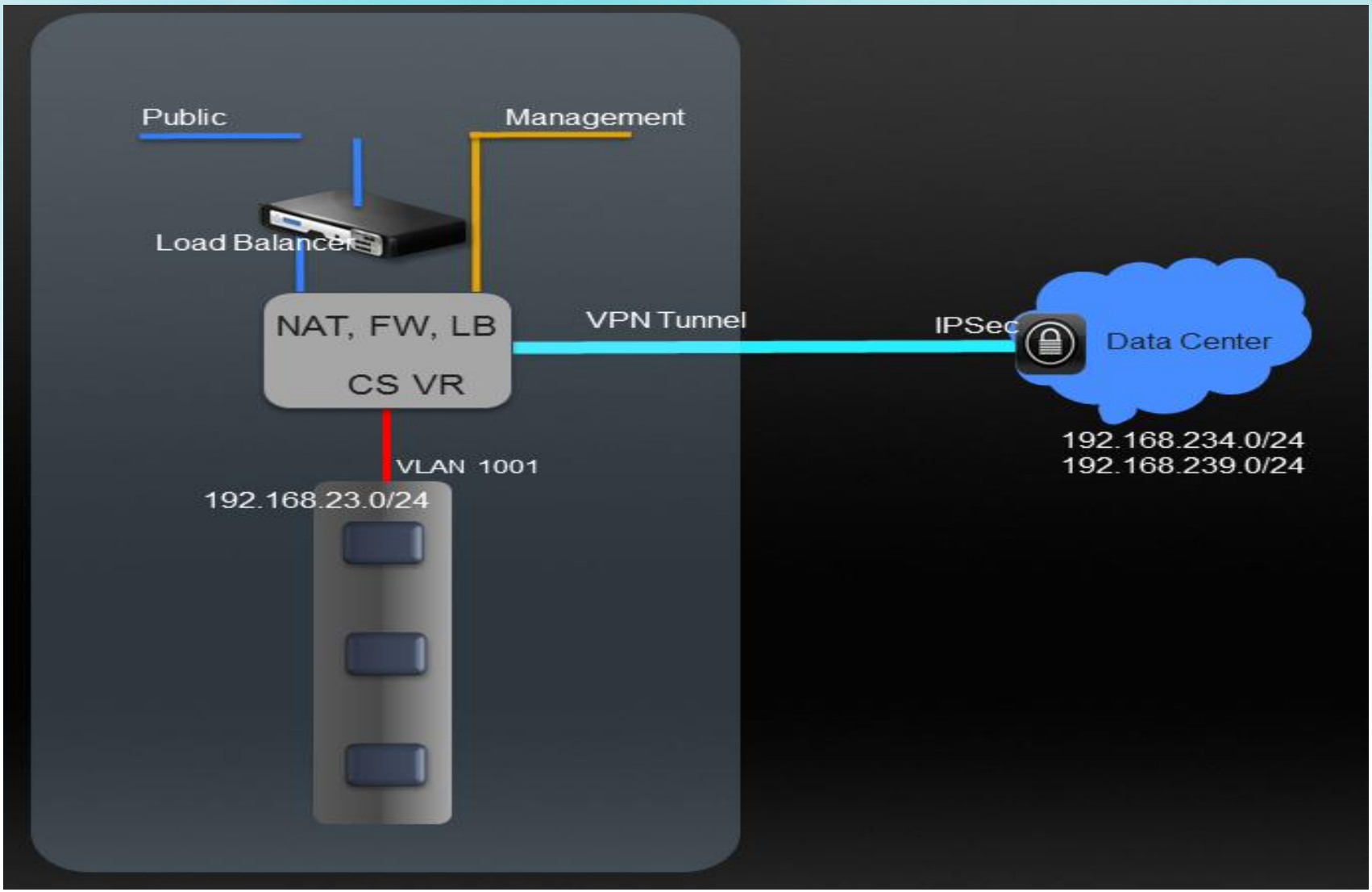
VPC—配置


控制板
实例
存储
网络
模板
事件
帐户
域
基础架构
项目
全局设置
服务方案
网络 - VPC
配置 VPC: MyVPC
MyVPC
虚拟路由器
Applicat...
0 VM
192.168.3.0/24
DB Tier
3 VM
192.168.2.0/24
Web Tier
1 VM
192.168.1.0/24

S-2-S VPN

cloudstack

http://confluence.cloudstack.org/display/PM/Site-to-Site+VPN



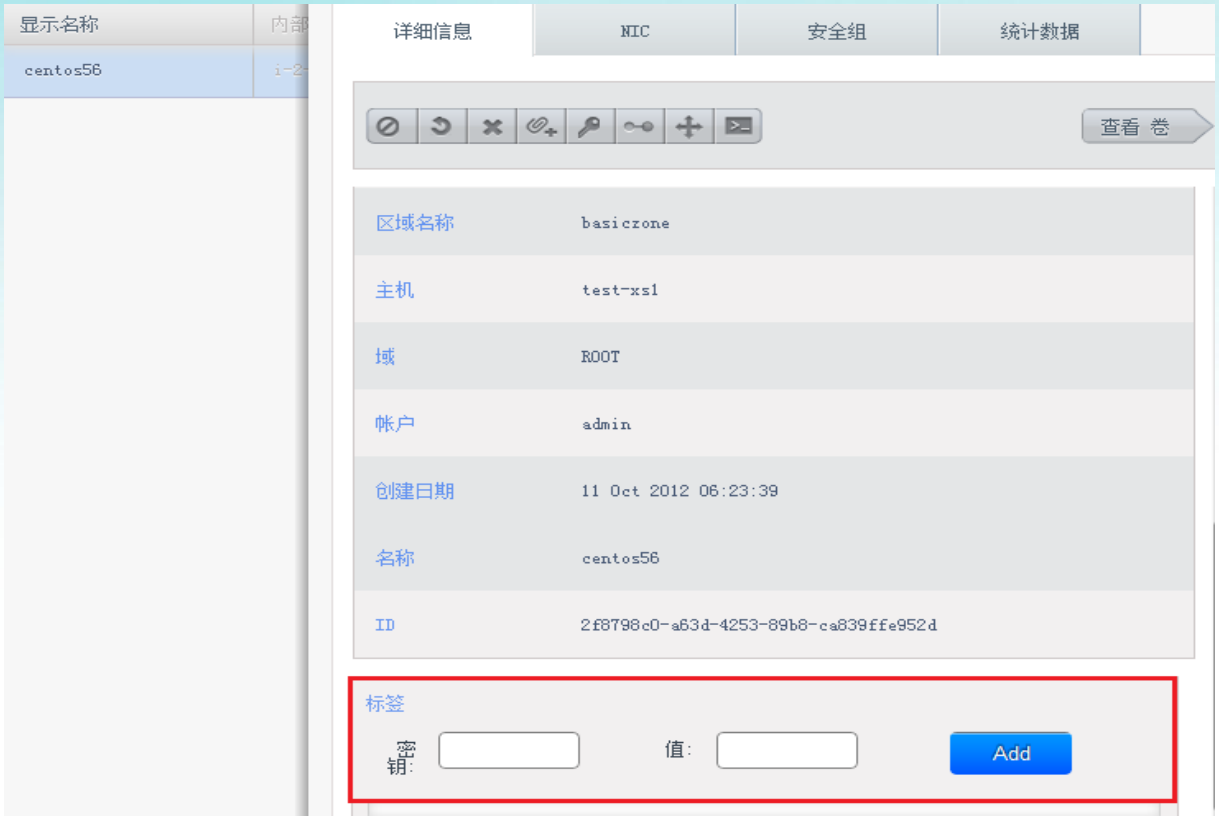
数据卷本地存储支持

1. 建立资源域时启用本地存储
2. 全局设置启用system.vm.use.local.storage
3. 重启管理服务: service cloud-management restart
4. 添加实例或现有虚机增加卷时选择设置本地存储的磁盘服务

Tagging

允许用户保存各种资源的元数据信息, AWS风格, Key-Value对, 凡是有Object ID的资源都可以设置

- User Vm
- Template
- ISO
- Volume
- Snapshot
- Guest Network
- LB rule
- PF rule
- Firewall rule
- Security Group
- Public IP Address
- Project
- Vpc
- NetworkACL
- StaticRoute



HA专用主机

- 全局配置: ha.tag : ha_host_tag
- 添加新主机标签设置: ha_host_tag
- 启动或创建VM
  - VM计算服务设置HA
  - VM计算服务主机标签不要写ha_host_tag
- 设置HA标签的主机不能用来启动虚机
- VM实例不能动态迁移到ha_host_tag主机
- 系统虚拟机

AWS EC2 API (test on DevCloud)

python cloudstack-aws-api-register **--apikey**=<APIKEY> \

--secretkey=<SKEY> \

--cert=<cert.pem> \

--url=http://<MS>:7080/awsapi

- 全局配置: enable.ec2.api: true
- 使用7080端口
- 路径: /awsapi
- 安全访问:
  - apikey
  - secretkey
  - certification key

AWS EC2 API (test via DevCloud cont.)

```
region = boto.ec2.regioninfo.RegionInfo(name="ROOT",endpoint="localhost")
apikey=''
secretkey=''
def main():
    '''Establish connection to EC2 cloud'''
    conn =boto.connect_ec2(aws_access_key_id=apikey,
                aws_secret_access_key=secretkey,
                is_secure=False,
                region=region,
                port=7080,
                path="/awsapi",
                api_version="2010-11-15")

    '''Get list of images that I own'''
    images = conn.get_all_images()
    print images
    myimage = images[0]
    '''Pick an instance type'''
    vm_type='m1.small'
    reservation = myimage.run(instance_type=vm_type,security_groups=['default'])
```

AWS S3 API

- 指定文件系统的挂载点
- 全局配置: enable.s3.api: true
- 生成用户的ApiKey及SecretKey
- 注册用户及关联认证信息: cloudstack-aws-api-register
- 使用S3客户端接口,如boto

配置文件: $CATALINA_HOME/conf/cloud-bridge-properties

```
host=http://localhost:7080/awsapi
storage.root=/Users/john1/S3-Mount
storage.multipartDir=__multipart__uploads__
bucket.dns=false
serviceEndpoint=localhost:7080
```

AWS S3 API test

```
conn =boto.connect_s3(aws_access_key_id=apikey,
                      aws_secret_access_key=secretkey,
                      is_secure=False,
                      region=region,
                      port=7080,
                      path="/awsapi",
                      api_version="2010-11-15")
```

- allbuckets = conn.get_all_buckets()
- conn.create_bucket('mybucket1')
- b=conn.lookup('mybucket1')
  k = Key(b)
  k.Key = 'my_upload_1'
  mp = b.initiate_multipart_upload('my_upload_1')
  fp = open('file_split_xaa', 'rb')
  mp.upload_part_from_file(fp,1)
  fp.close()
  mp.complete_upload()
- conn.delete_bucket(b)

Caringo Castor

Caringo 自身保证高性能, 简单管理以及云模块之间的互操作性, 同时降低云环境中的复杂性. Caringo可以在任意的存储硬件上部署

Nicira NVP

- 每个租户任意数量的虚拟私有网络—没有VLan
- 持续的高可用性,包括分布式active-active的集群故障转移
- 网络硬件无关
- 整个云环境为多租户提供企业级安全特性
- 整个云环境中网络服务的可编程及自动化
- 完全可操作性 – logging, monitoring, troubleshooting, 0-downtime upgrading

Nicira NVP - TODO

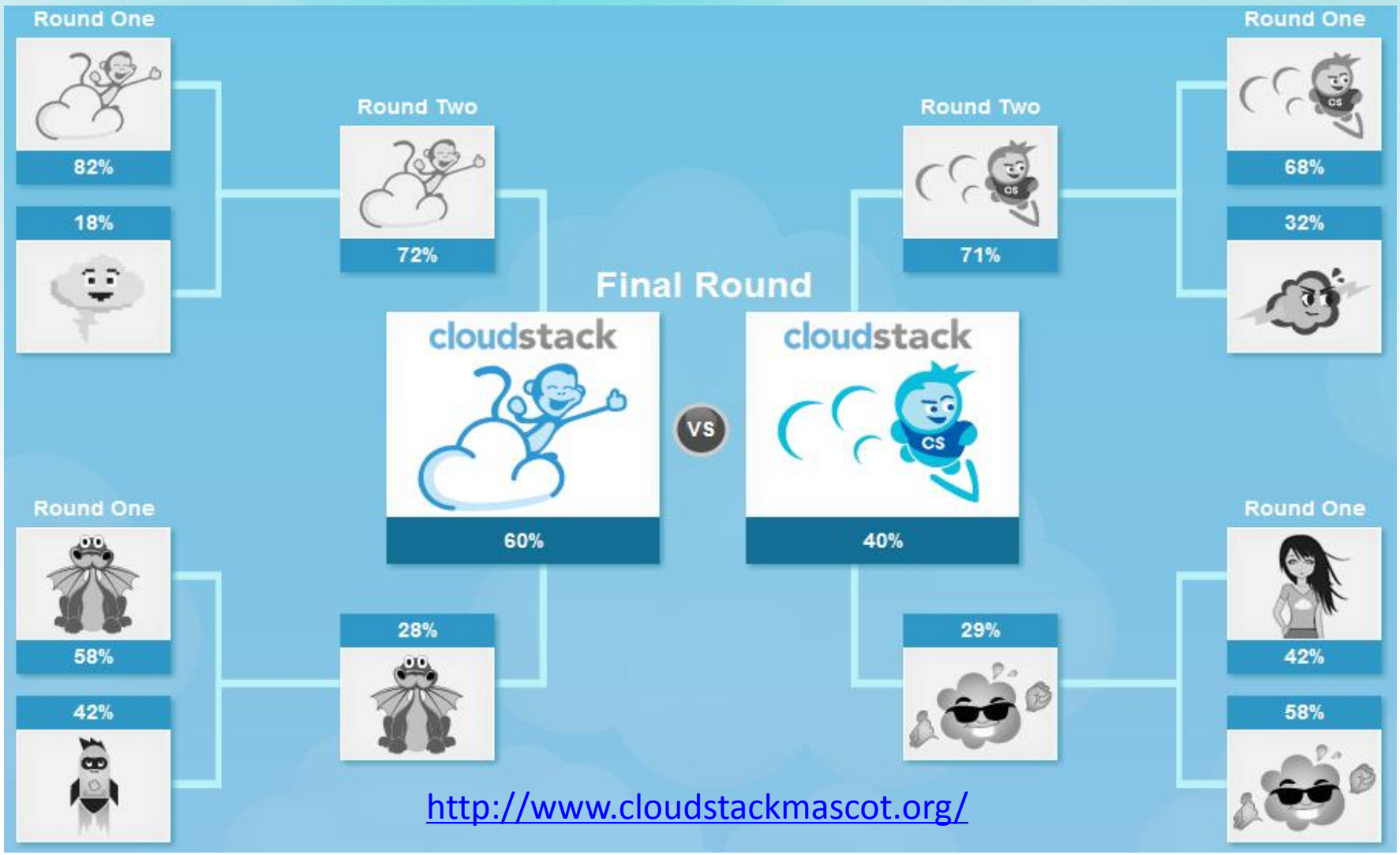
Nicira目前没有GUI可用,目前支持的API:

- addNetworkServiceProvider
- updateNetworkServiceProvider
- addNiciraNvpDevice

第二阶段工作:

- 集成L3 support
- 集成NetworkOfferings

Apache CloudStack Logo



http://www.cloudstackmascot.org/