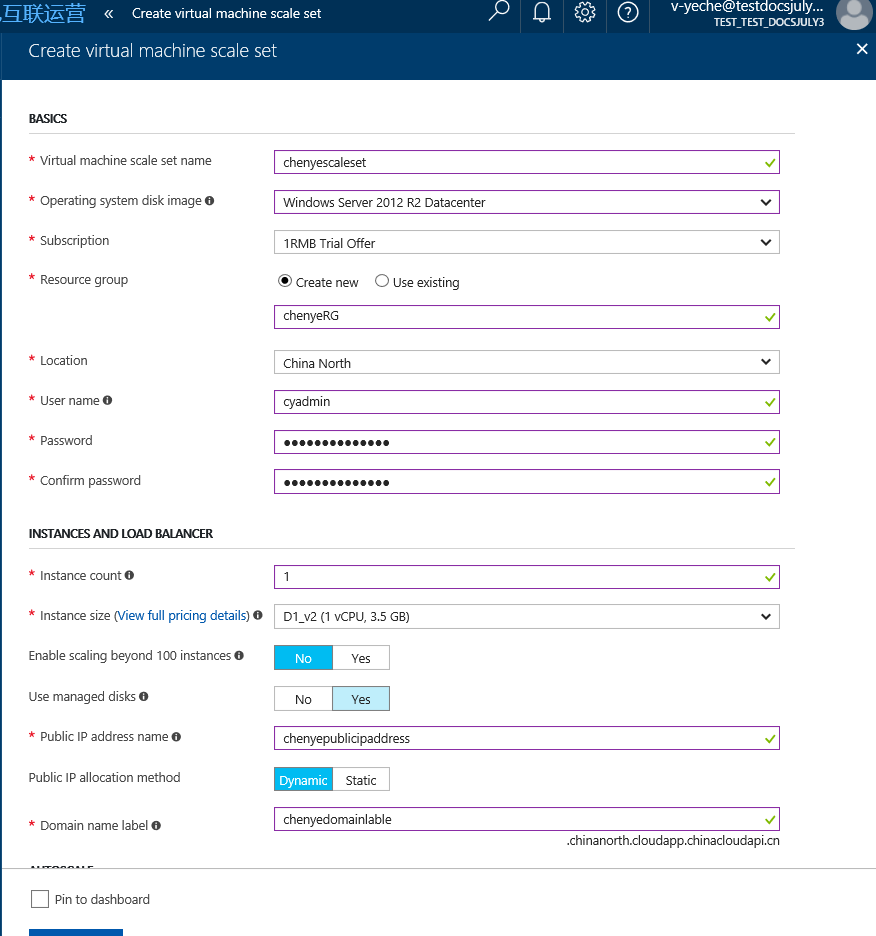
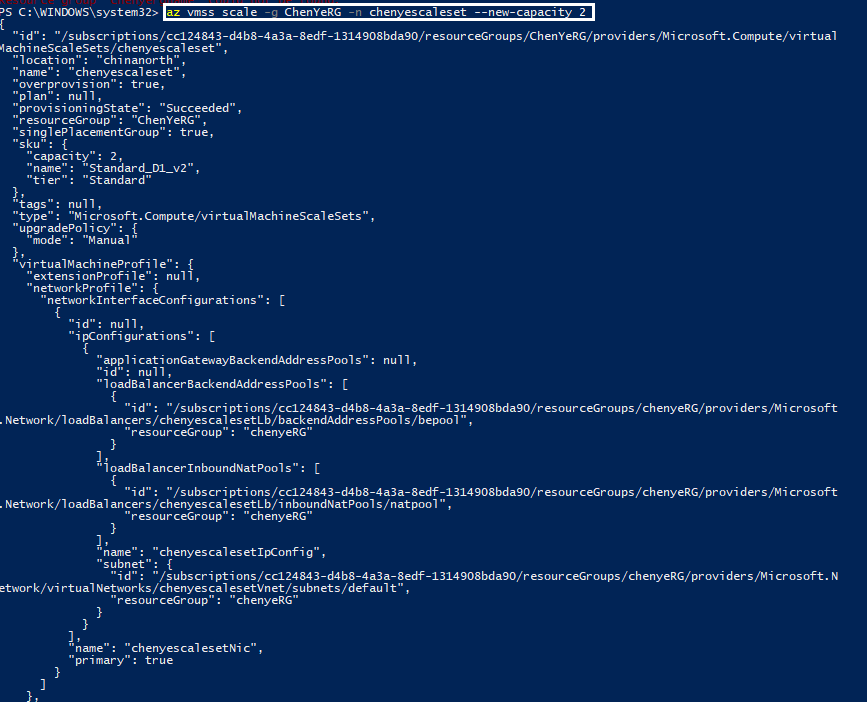
1. Create VM Scale Set

Virtual Machine🡪 Virtual Machine Scale Set



Scale manual :

az vmss scale -g ChenYeRG -n chenyescaleset --new-capacity 2



Step : How to create the VM

az cloud set -n AzureChinaCloud

az login

azure config mode asm

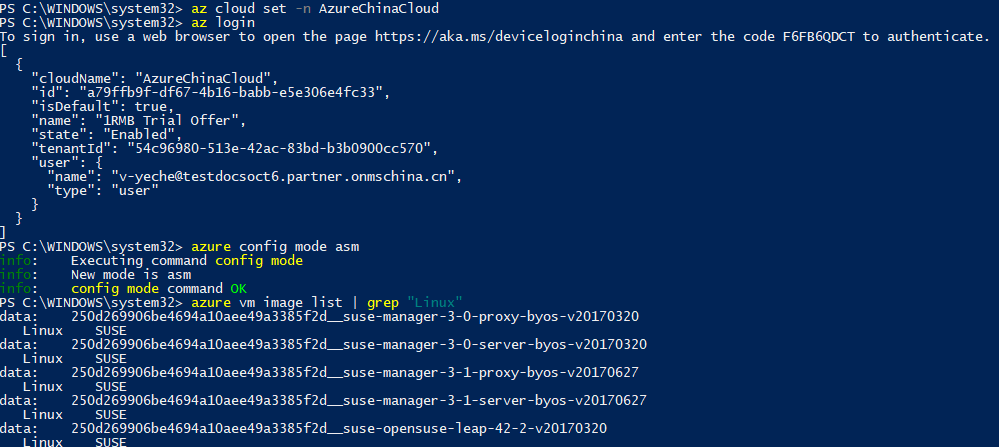
azure vm image list | grep "Linux"

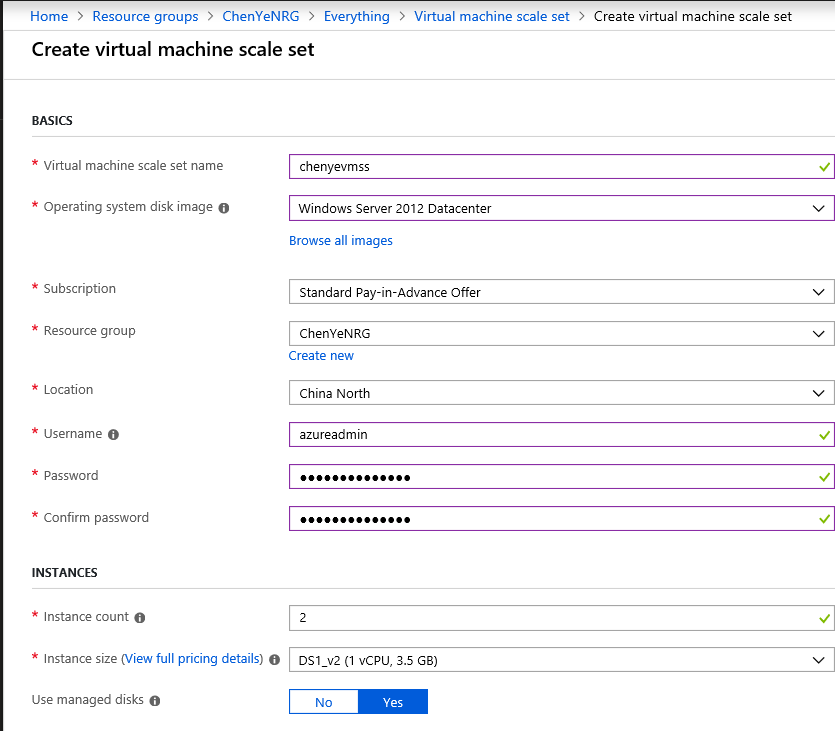
azure vm create myVM `

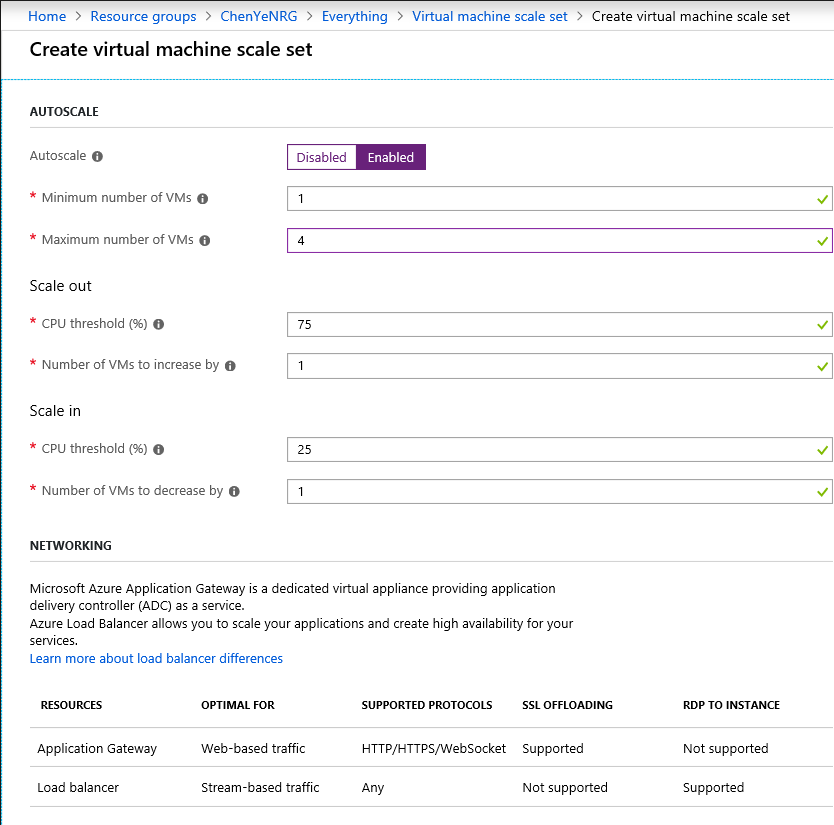
b549f4301d0b4295b8e76ceb65df47d4\_\_Ubuntu-12\_04\_4-LTS-amd64-server-20140529-en-us-30GB `

-g ops -p P@ssw0rd! -z "Small" -e -l "china north"azure vm show myVM

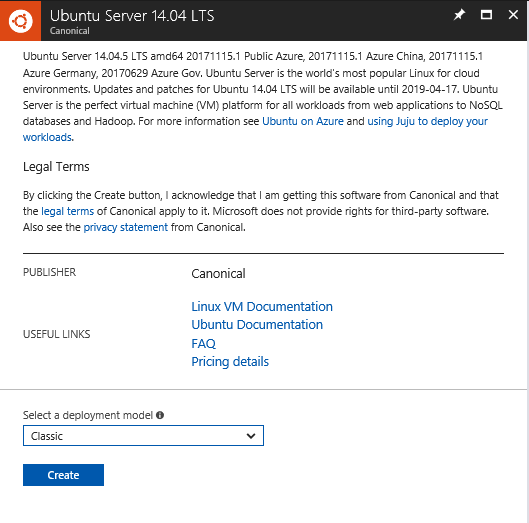
azure vm start myVM

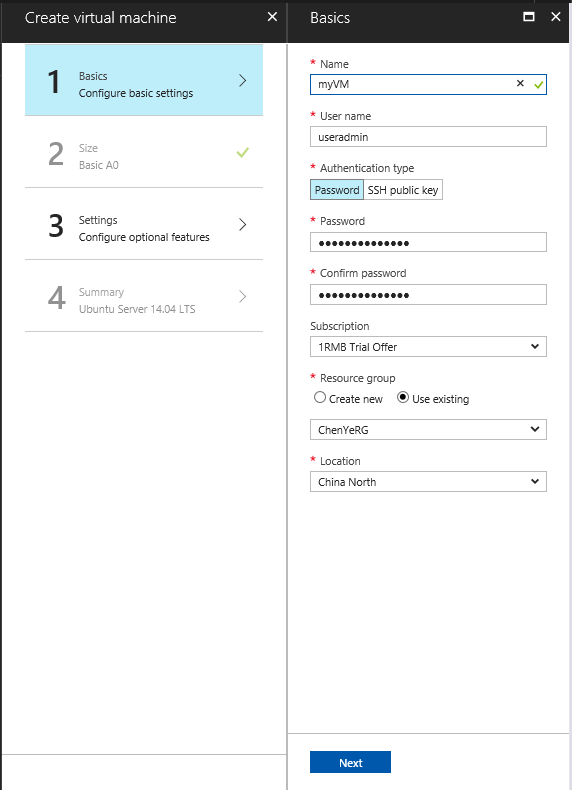


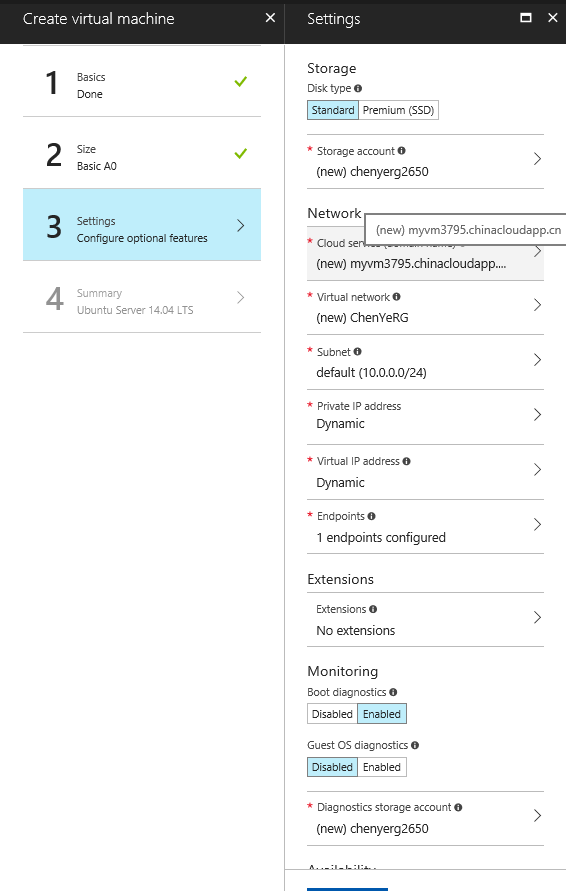


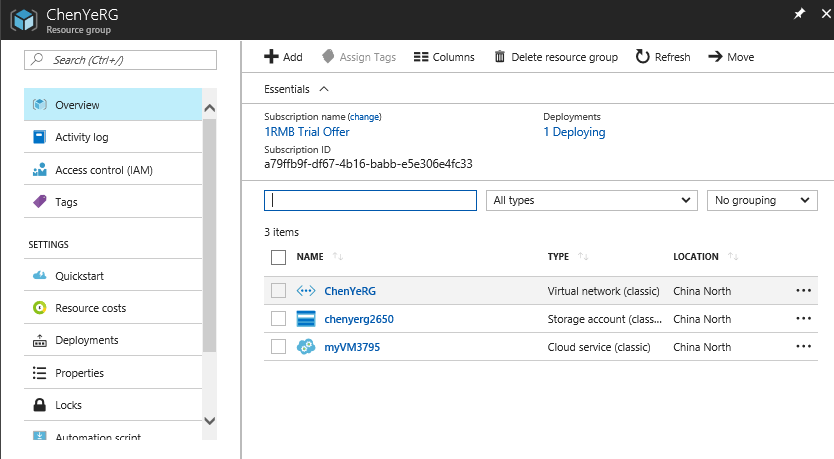


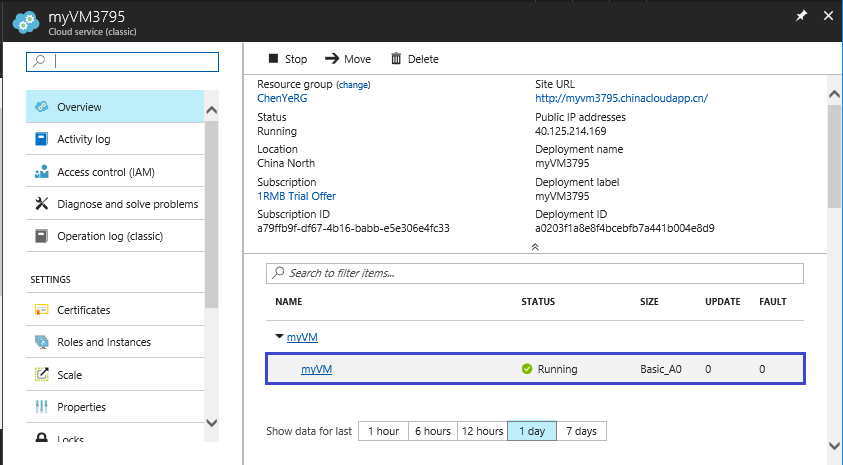
Create Classic Model VM

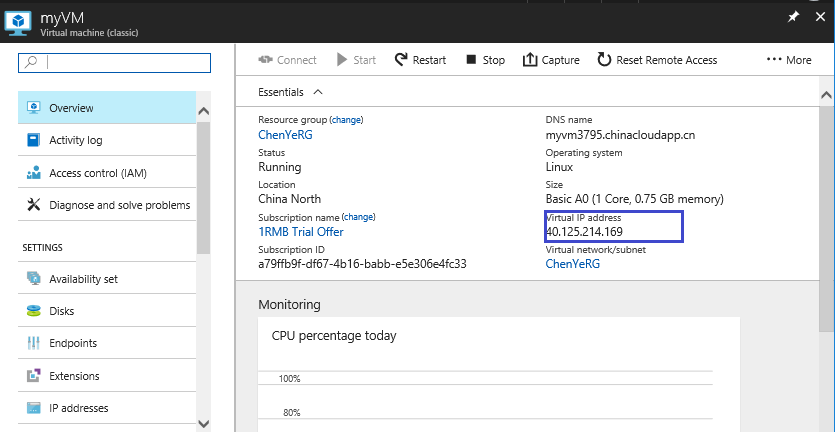




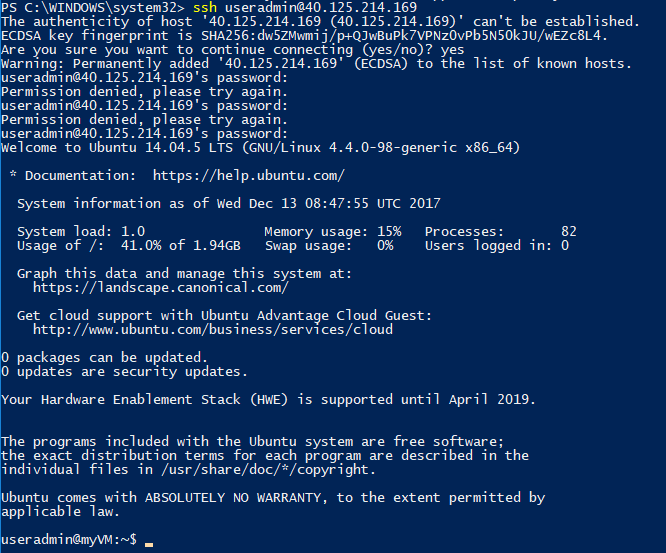


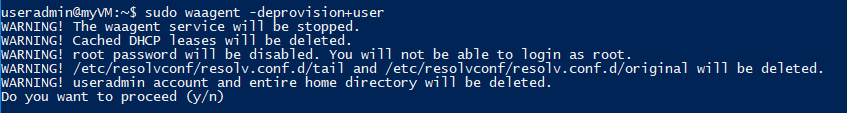






ssh [useradmin@40.125.214.169](mailto:useradmin@40.125.214.169)





sudo waagent -deprovision+user

上述命令将尝试清除系统并使其适合重新预配。 此操作执行以下任务：

* 删除 SSH 主机密钥（如果在配置文件中 Provisioning.RegenerateSshHostKeyPair 为“y”）
* 清除 /etc/resolv.conf 中的 nameserver 配置
* 从 /etc/shadow 中删除 root 用户的密码（如果在配置文件中 Provisioning.DeleteRootPassword 为“y”）
* 删除缓存的 DHCP 客户端租赁
* 将主机名重置为 localhost.localdomain
* 删除上次预配的用户帐户（从 /var/lib/waagent 获得）**和关联的数据**。

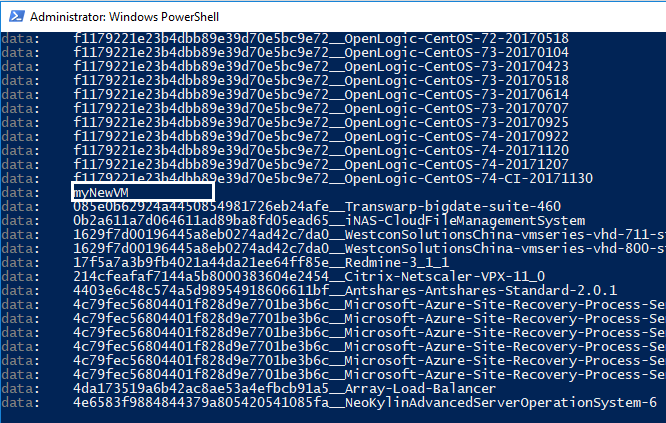
Exit SSH

azure config mode asm

azure vm shutdown myVM

azure vm capture -t myVM myNewVM (-t means to remove the myVM machine)

azure vm image list



Configure Cassandra in Linux.

JAVA\_HOME=/opt/java/jdk1.8.0\_05

CASS\_HOME= /opt/cassandra/apache-cassandra-2.0.8

PATH=$PATH:$HOME/bin:$JAVA\_HOME/bin:$CASS\_HOME/bin

export JAVA\_HOME

export CASS\_HOME

export PATH

FreeBSD

1. Start DHCP

# echo 'ifconfig\_hn0="SYNCDHCP"' >> /etc/rc.conf

# service netif restart

1. Enable SSH

# echo 'console="comconsole vidconsole"' >> /boot/loader.conf

# echo 'comconsole\_speed="115200"' >> /boot/loader.conf

1. Configure the serial console

# pkg install sudo

1. Azure proxy install

# pkg install python27

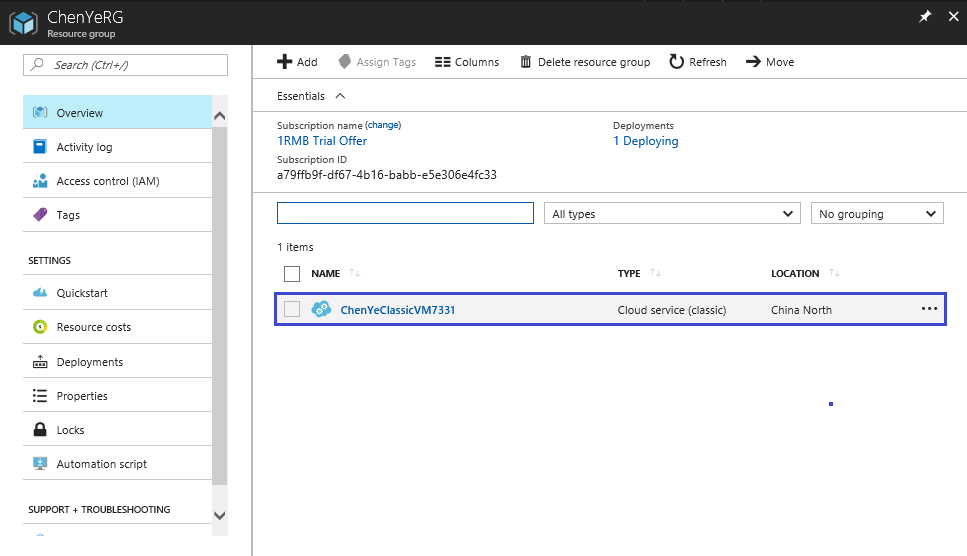
# pkg install Py27-setuptools

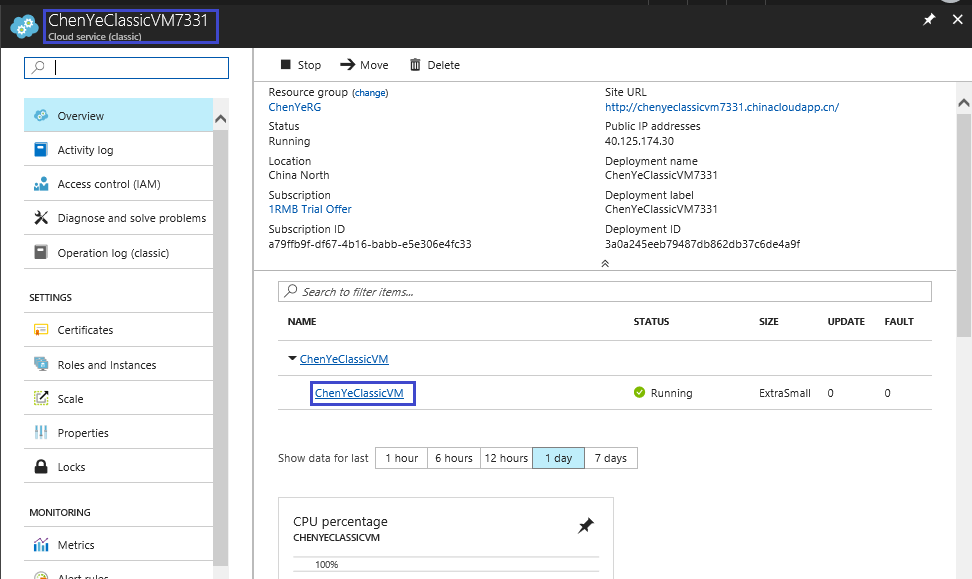
# ln -s /usr/local/bin/python2.7 /usr/bin/python

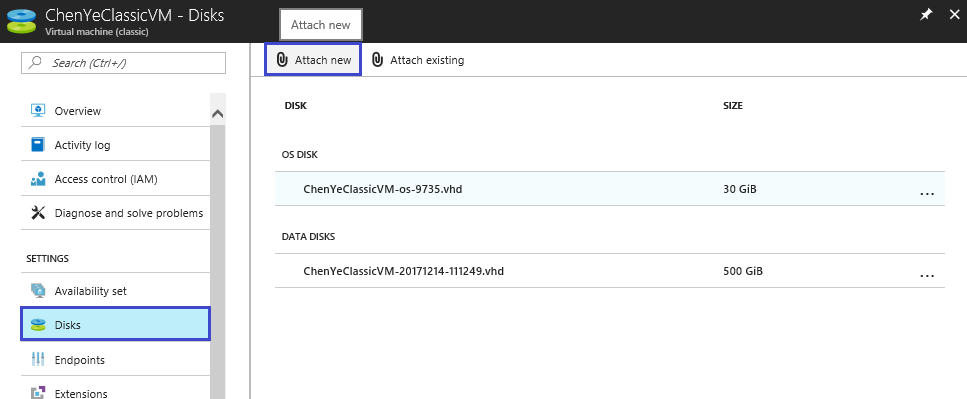
# pkg install git

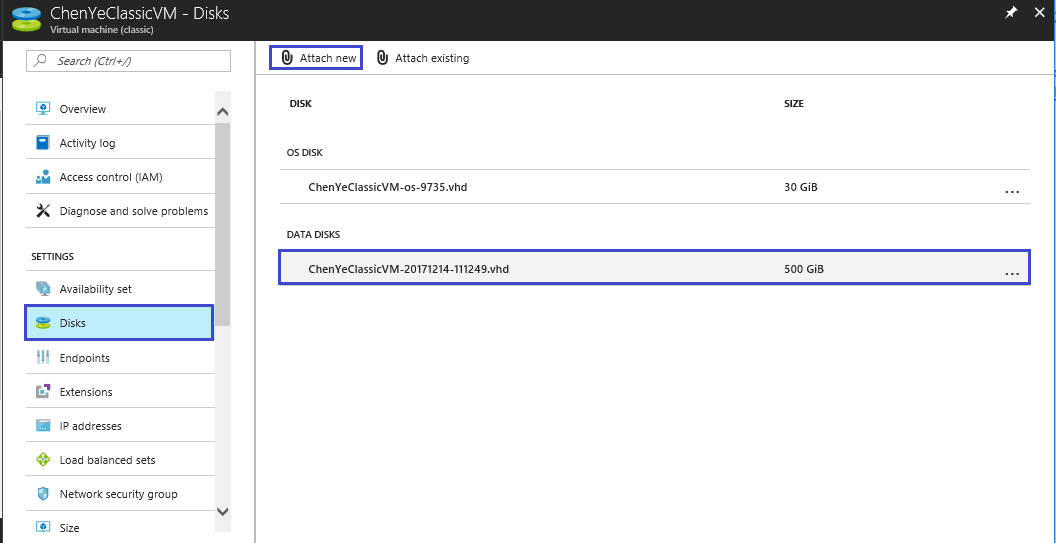
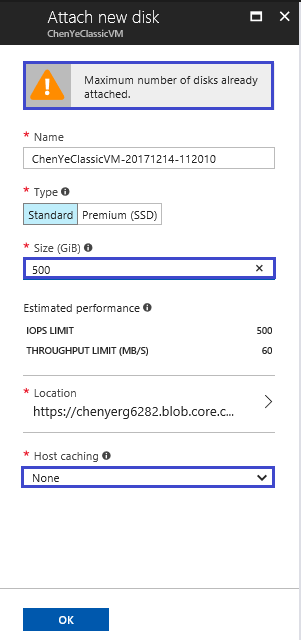
Optimize MySQL

1. Attach RAID Disk









When connect the VM classic , we can use the following

ssh useraccountname@ip-address,

If can not connect it, we can reset the password via portal and try again.

Review the schedule list:

$ cat /sys/block/sda/queue/scheduler

$ noop [deadline] cfq (DISPLAY THE SCHEDULE)

配置系统文件操作设置

最佳做法之一是禁用文件系统上的 *atime* 日志记录功能(RAID盘上的Disk)

例如，编辑 vim /etc/fstab 文件，并按以下示例所示添加 noatime：

复制

# CLOUD\_IMG: This file was created/modified by the Cloud Image build process

UUID=3cc98c06-d649-432d-81df-6dcd2a584d41 / ext4 defaults,discard 0 0

#Add the “noatime” option below to disable atime logging

UUID="431b1e78-8226-43ec-9460-514a9adf060e" /RAID0 xfs defaults,nobootwait, noatime 0 0

/dev/sdb1 /mnt auto defaults,nobootwait,comment=cloudconfig 0 2

# CLOUD\_IMG: This file was created/modified by the Cloud Image build process

UUID=3cc98c06-d649-432d-81df-6dcd2a584d41 / ext4 defaults,discard 0 0

#Add the “noatime” option below to disable atime logging

UUID="431b1e78-8226-43ec-9460-514a9adf060e" /RAID0 xfs defaults,nobootwait, noatime 0 0

/dev/sdb1 /mnt auto defaults,nobootwait,comment=cloudconfig 0 2

增加系统句柄的最大数量以实现高并发

MySQL 是高并发数据库。 对于 Linux，默认的并发句柄数量是 1024 个，这个数量有时候不够用。 通过执行以下步骤来增加系统的最大并发句柄数，以此支持 MySQL 的高并发。

修改 limits.conf 文件

若要增加允许的最大并发句柄数，请在 /etc/security/limits.conf 文件中添加以下四行。 请注意，65536 是系统可以支持的最大数量。

复制

\* <span data-ttu-id="0f513-204">soft nofile 65536</span><span class="sxs-lookup"><span data-stu-id="0f513-204">soft nofile 65536</span></span>

\* <span data-ttu-id="0f513-205">hard nofile 65536</span><span class="sxs-lookup"><span data-stu-id="0f513-205">hard nofile 65536</span></span>

\* <span data-ttu-id="0f513-206">soft nproc 65536</span><span class="sxs-lookup"><span data-stu-id="0f513-206">soft nproc 65536</span></span>

\* <span data-ttu-id="0f513-207">hard nproc 65536</span><span class="sxs-lookup"><span data-stu-id="0f513-207">hard nproc 65536</span></span>

ssss

### \* <span data-ttu-id="0f513-204">soft nofile 65536</span><span class="sxs-lookup"><span data-stu-id="0f513-204">soft nofile 65536</span></

### span>

### \* <span data-ttu-id="0f513-205">hard nofile 65536</span><span class="sxs-lookup"><span data-stu-id="0f513-205">hard nofile 65536</span></span>

### \* <span data-ttu-id="0f513-206">soft nproc 65536</span><span class="sxs-lookup"><span data-stu-id="0f513-206">soft nproc 65536</span></span>

### \* <span data-ttu-id="0f513-207">hard nproc 65536</span><span class="sxs-lookup"><span data-stu-id="0f513-207">hard nproc 65536</span></span>

### **更新系统的新限制**

若要更新系统，请运行以下命令：

复制

ulimit -SHn 65536

ulimit -SHu 65536

## MySQL 数据库优化

若要在 Azure 上配置 MySQL，可以使用在本地计算机上使用的相同性能优化策略。

主要 I/O 优化规则包括：

* 增加缓存大小。
* 减少 I/O 响应时间。

若要优化 MySQL 服务器设置，可以更新 my.cnf 文件，它是服务器和客户端计算机的默认配置文件。

以下配置项是影响 MySQL 性能的主要因素：

* **innodb\_buffer\_pool\_size**：缓冲池包含缓冲数据和索引。 此值通常设置为物理内存的 70%。
* **innodb\_log\_file\_size**：这是重做日志大小。 重做日志用于确保写入操作快速、可靠并且可在出现故障后恢复。 此值设置为 512 MB，这会为记录写入操作提供大量空间。
* **max\_connections**：应用程序有时候不会正确关闭连接。 值越大，服务器就有越多时间回收空闲的连接。 最大连接数为 10,000，但建议的最大值为 5,000。
* **Innodb\_file\_per\_table**：此设置允许或禁止 InnoDB 将表存储在单独的文件中。 启用该选项可确保可以有效地应用多项高级管理操作。 从性能角度来看，它可以提高表空间传输的速度和优化碎片管理性能。 此选项的推荐设置是“打开”。  
  从 MySQL 5.6 开始，默认设置为“打开”，因此不需要任何操作。 对于早期版本，默认设置为“关闭”。 应在加载数据之前更改此选项，因为只有新创建的表才会受影响。
* **innodb\_flush\_log\_at\_trx\_commit**：默认值为 1，范围设置为 0~2。 默认值是最适合独立 MySQL DB 的选项。 设置 2 支持最大程度的数据完整性，适用于 MySQL 群集中的主节点。 设置 0 允许数据丢失，这可能会影响可靠性（但在某些情况下能提供更好的性能），适用于 MySQL 群集中的从属节点。
* **Innodb\_log\_buffer\_size**：借助日志缓冲区，即使在事务提交之前未将日志刷新到磁盘，事务也可以运行。 但是，如果有大型的二进制对象或文本字段，将很快地耗尽缓存，并触发频繁的磁盘 I/O。 如果 Innodb\_log\_waits 状态变量不为 0，最好增加缓冲区大小。
* **query\_cache\_size**：最好是从一开始就禁用。 将 query\_cache\_size 设置为 0（这是 MySQL 5.6 中的默认设置）并使用其他方法来提高查询速度。

默认情况下，未启用此项。 启用慢查询日志可能会占用一些 CPU 资源。 建议临时启用此选项来排除性能瓶颈。 若要启用慢查询日志，请执行以下操作：

1. 通过在末尾添加以下行来修改 my.cnf 文件：

复制

long\_query\_time = 2

slow\_query\_log = 1

slow\_query\_log\_file = /RAID0/mysql/mysql-slow.log

1. 重新启动 MySQL 服务器。

复制

service mysql restart

1. 使用 **show** 命令检查该设置是否生效。

How to install the Docker in Ubutun 14.04

1. 检查Ubuntu的内核版本 (version > 14.04 )

# uname -r   
3.13.0-55-generic

1. 更新系统，确保软件包列表的有效性

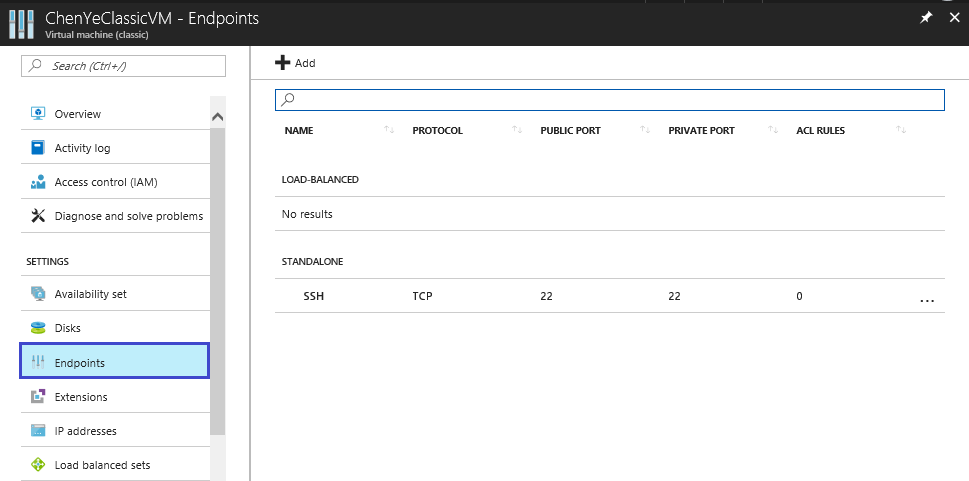
# apt-get update

1. 如果Ubuntu的版本不满足，还需升级Ubuntu

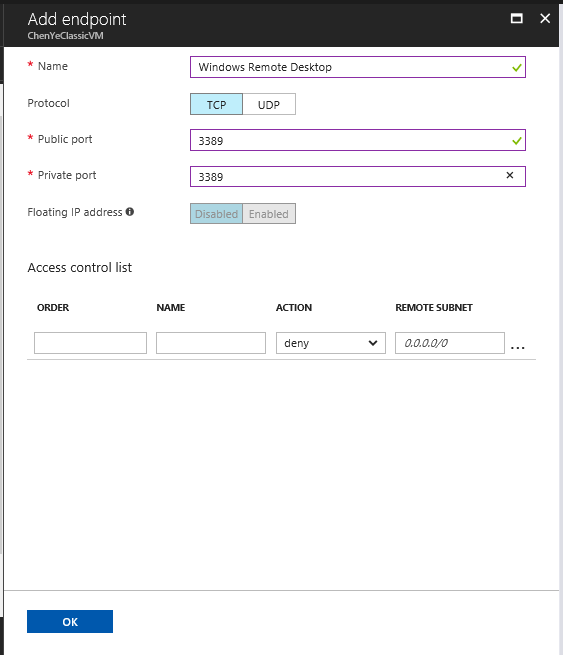
# apt-get -y upgrade

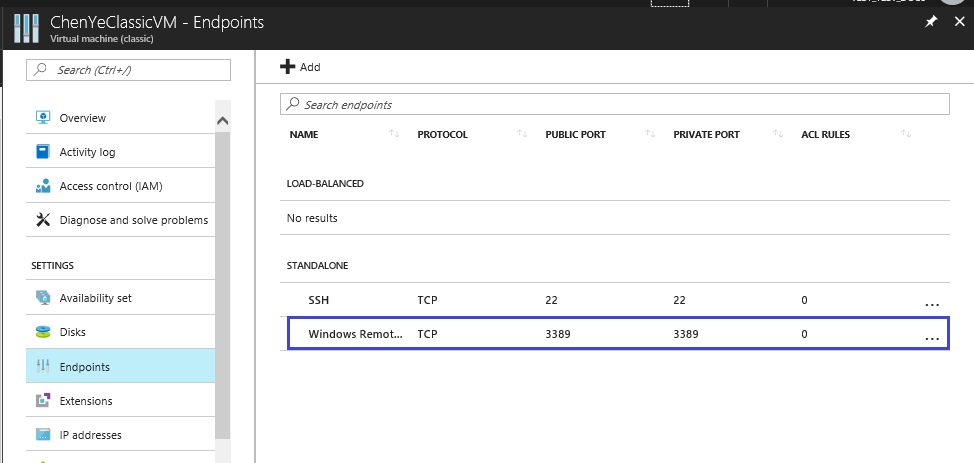
1. Install the Docker.

# sudo apt-get -y install docker.io



Add 3389 endpoint and allow windows to connect classic Linux.





Install Gnome 桌面

For Ubuntu

#sudo apt-get install ubuntu-desktop

For OpenSusu

# sudo zypper install gnome-session

Install xrdp

For Ubuntu

# sudo apt-get install xrdp

For OpenSusu

#sudo zypper in http://download.opensuse.org/repositories/X11:/RemoteDesktop/openSUSE\_13.2/x86\_64/xrdp-0.9.0git.1401423964-2.1.x86\_64.rpm

#sudo zypper install tigervnc xorg-x11-Xvnc xterm remmina-plugin-vnc

## 如果使用的是比 Ubuntu 12.04LTS 更高的 Ubuntu 版本，请使用 xfce

因为 xrdp 的当前版本对高于 Ubuntu 12.04LTS 的 Ubuntu 版本不支持 Gnome 桌面，我们将改而使用 xfce 桌面。

Install xrdp

For Ubuntu

#sudo apt-get install ubuntu-desktop

# echo xfce4-session >~/.xsession

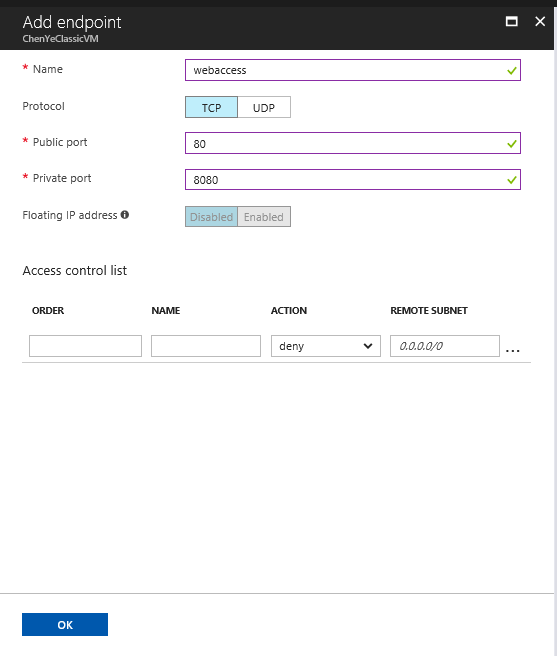
编辑配置文件 /etc/xrdp/startwm.sh：

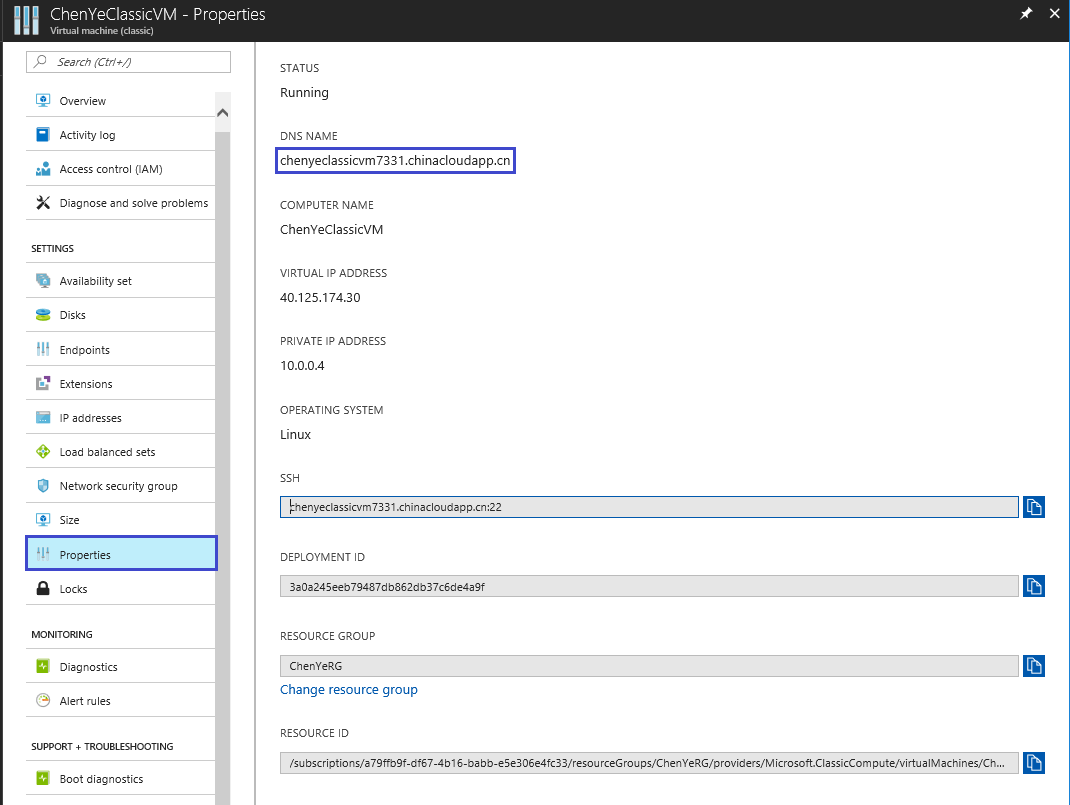
bash 复制

#sudo vi /etc/xrdp/startwm.sh

若要重新启动 xrdp 服务，请使用：

#sudo service xrdp restart





Install Tomcat:

* 1. Install OpenJDK

sudo apt-get update

sudo apt-get install openjdk-7-jre

java -version

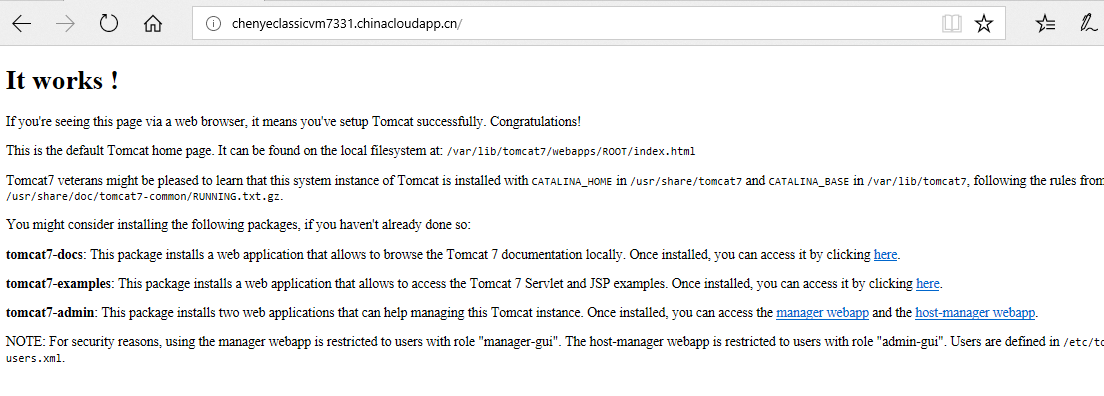
sudo mkdir /usr/lib/jvm

sudo tar -zxf jdk-8u5-linux-x64.tar.gz -C /usr/lib/jvm/

* 1. Install Tomcat7

#sudo apt-get install tomcat7

<http://chenyeclassicvm7331.chinacloudapp.cn>



It show we have install the Tomcat7 successfully.

### **安装其他 Tomcat7 组件**

可以安装其他可选 Tomcat 组件。

使用 **sudo apt-cache search tomcat7** 命令可查看所有可用组件。 使用以下命令安装一些有用的组件。

sudo apt-get install tomcat7-admin #admin web applications

sudo apt-get install tomcat7-user #tools to create user instances

阶段 4：配置 Tomcat7

Start and Stop the Tomcat7

sudo /etc/init.d/tomcat7 start

sudo /etc/init.d/tomcat7 stop

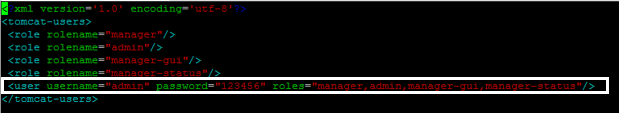
sudo /etc/init.d/tomcat7 status

sudo /etc/init.d/tomcat7 restart

### **Tomcat7 管理**

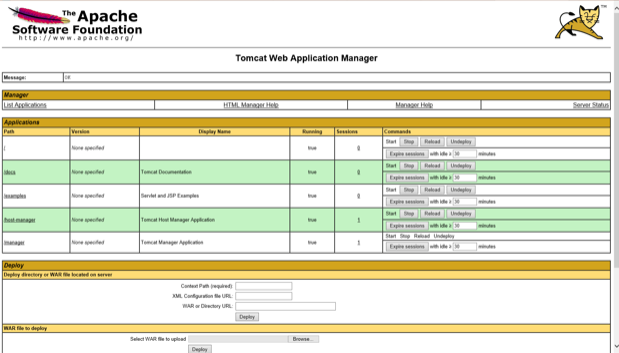
可以通过编辑 Tomcat 用户配置文件来设置管理员凭据。 请使用以下命令：

sudo vi /etc/tomcat7/tomcat-users.xml



sudo /etc/init.d/tomcat7 restarts

[http:// chenyeclassicvm7331.chinacloudapp.cn /manager/html。](http://tomcatexample.cloudapp.net/manager/html。)



#### **症状**

Tomcat 正在运行，但使用浏览器看不到 Tomcat 默认页。

若要确定 Tomcat 侦听端口，请打开 /etc/httpd/conf/httpd.conf（Red Hat 发行版）或 /etc/tomcat7/server.xml（Debian 发行版）。 默认情况下，Tomcat 侦听端口为 8080。 下面是一个示例：  
<Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000" URIEncoding="UTF-8" redirectPort="8443" />

防火墙已禁用 Tomcat 的侦听端口。

只能在本地主机上看到 Tomcat 默认页。 问题的原因很可能是 Tomcat 侦听的端口被防火墙阻止。 可以使用 w3m 工具来浏览网页。 以下命令安装 w3m 并浏览到 Tomcat 默认页：

<span data-ttu-id="d21d9-278">sudo yum install w3m w3m-img</span><span class="sxs-lookup"><span data-stu-id="d21d9-278">sudo yum install w3m w3m-img</span></span>

<span data-ttu-id="d21d9-279">w3m http://localhost:8080</span><span class="sxs-lookup"><span data-stu-id="d21d9-279">w3m http://localhost:8080</span></span>

#### **解决方案**

* 如果 Tomcat 侦听端口与发往虚拟机的流量的终结点专用端口不同，则需要将该专用端口更改为与 Tomcat 侦听端口相同。
* 如果此问题由防火墙/iptables 导致，请将以下行添加到 /etc/sysconfig/iptables。 只有 https 流量才需要第二行：

-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT

-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT

sudo service iptables restart

将项目文件上传到 /var/lib/tomcat7/webapps/ 时，权限被拒绝

#### **可能的根本原因**

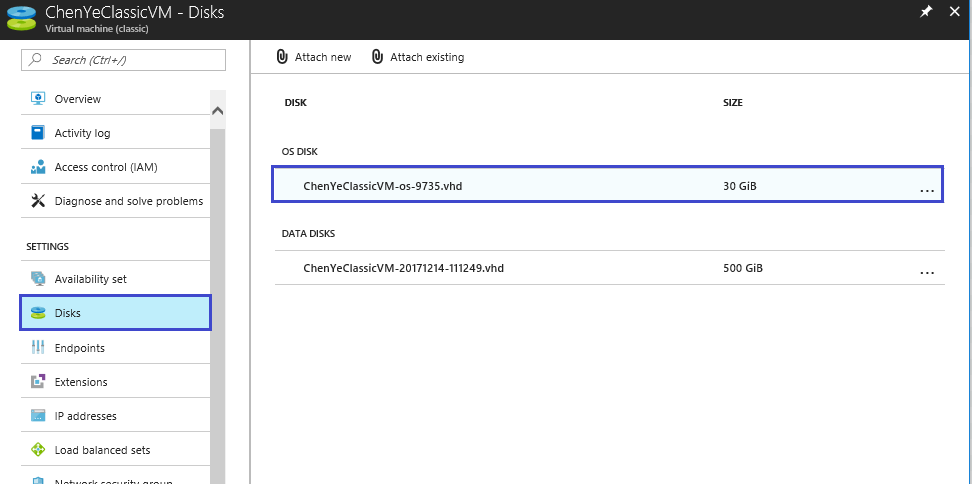
无权访问 /var/lib/tomcat7/webapps 文件夹。

#### **解决方案**

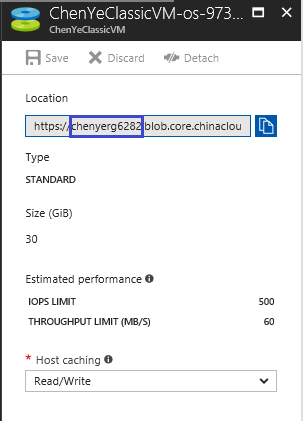
需要获取 root 帐户权限。 可将该文件夹的所有权从 root 更改为在设置计算机时使用的用户名。 下面是使用 azureuser 帐户名称的示例：

sudo chown azureuser -R /var/lib/tomcat7/webapps

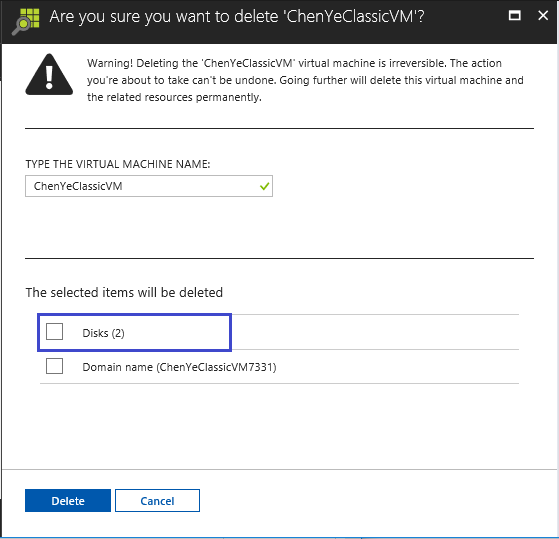
如何排除虚拟硬盘错误



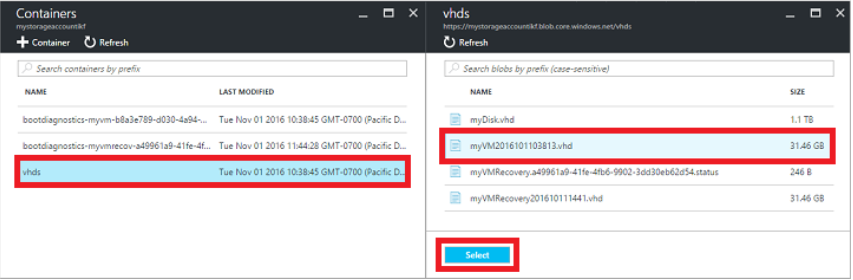
1.Reserve the storage account which linked to the Virtual Disk.

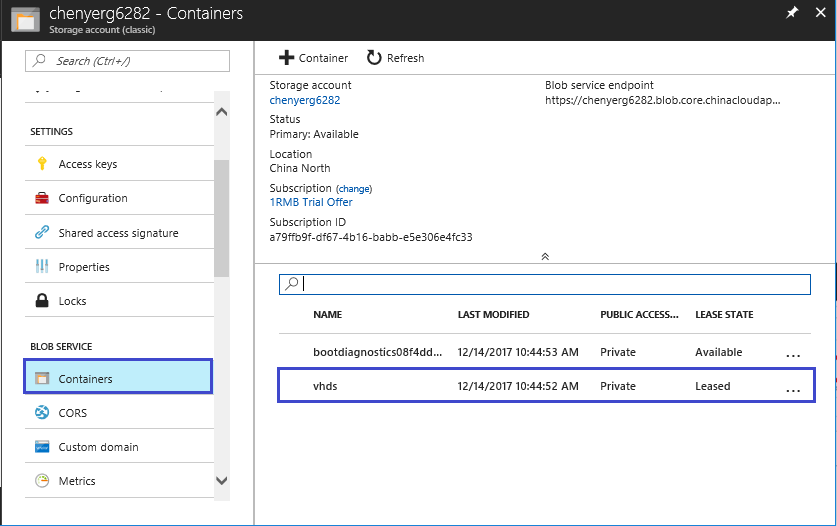


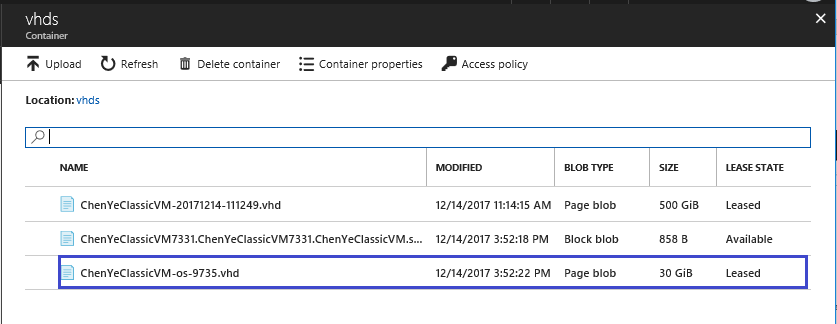
2.Stop the VM and try to delete the VM except the Disk like following:

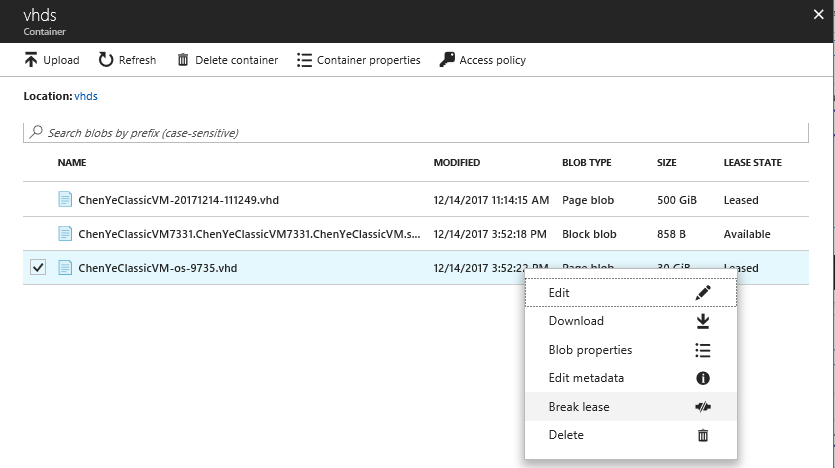


1. Create the new VM and attach the reserved disk on the new VM.
2. Select Storage->VHD container->Virtual disk.









如何在Classical VM安装Ruby

#sudo apt-get update -y

#sudo apt-get upgrade -y

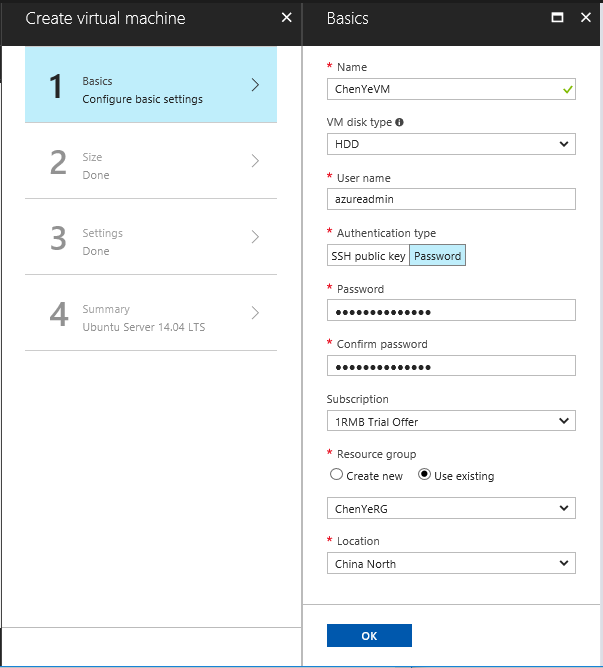
#sudo apt-add-repository ppa:brightbox/ruby-ng

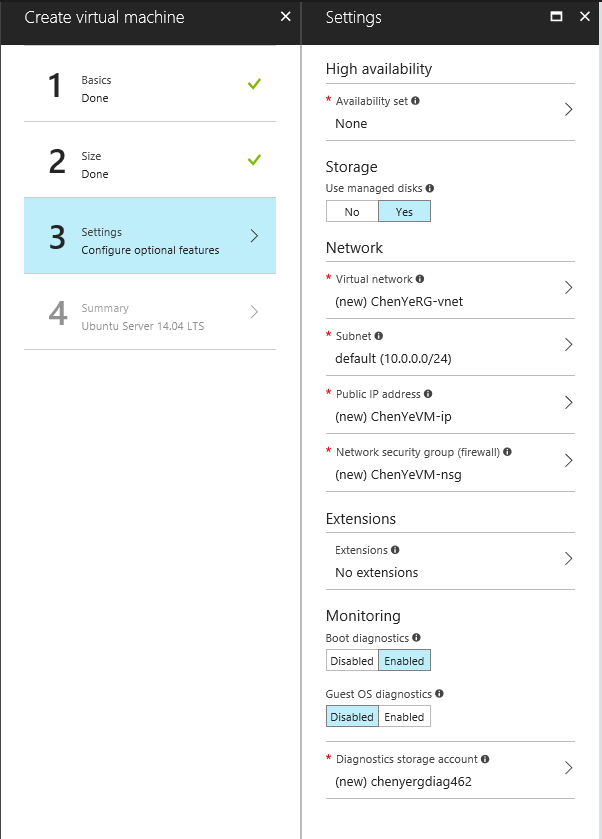
#sudo apt-get update

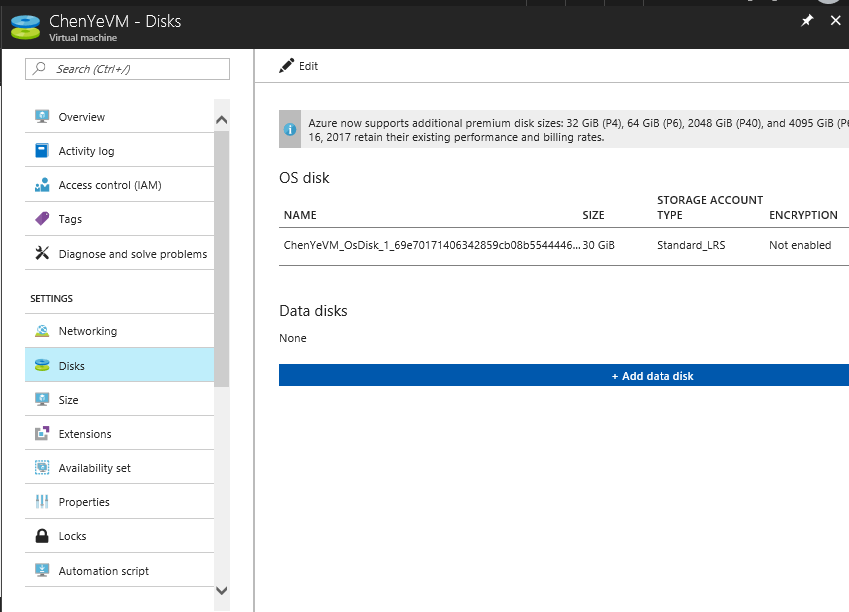
#sudo apt-get install ruby2.4

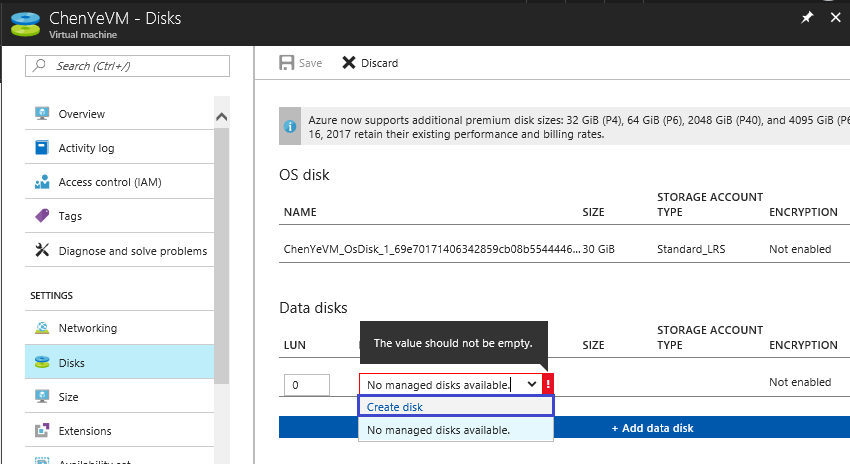
# sudo gem install rails --no-rdoc --no-ri -V

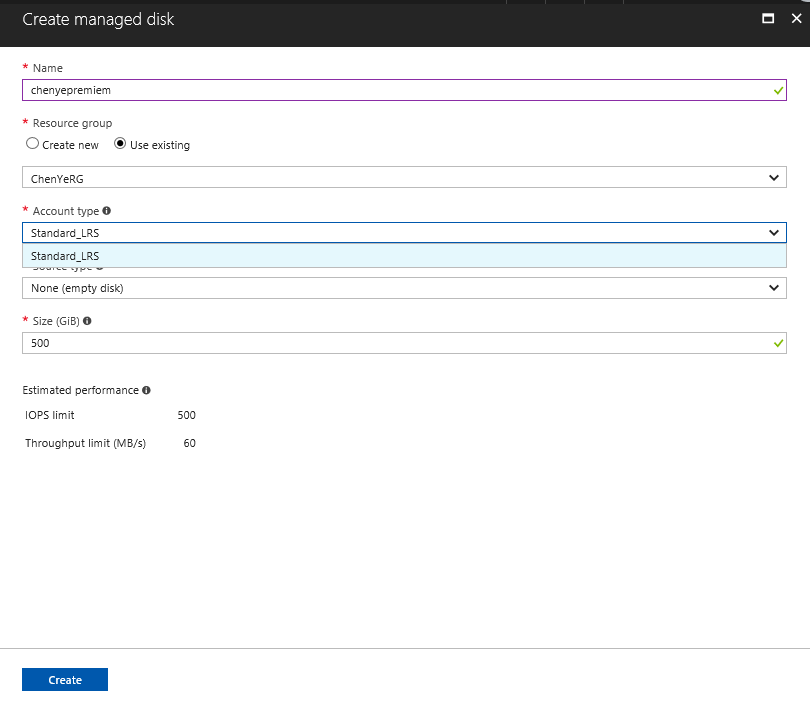
Create ARM model VM



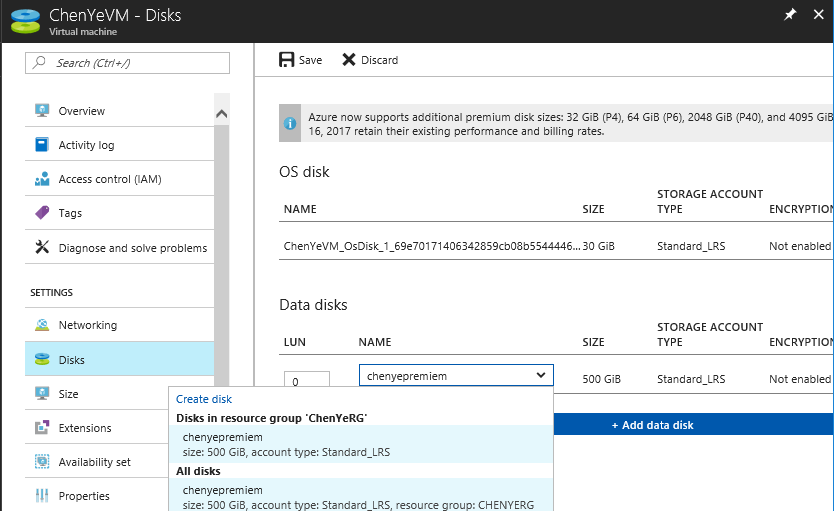








Managed disk not Premium Storage Account in Azure.cn



使用 PowerShell 分离数据磁盘

$VirtualMachine = Get-AzureRmVM -ResourceGroupName "RG11" -Name "MyVM07"

Remove-AzureRmVMDataDisk -VM $VirtualMachine -Name "DataDisk3"

Update-AzureRmVM -ResourceGroupName "RG11" -VM $VirtualMachine

Virtual Machine extensions:

az vm extension image list --location chinanorth -o table

az vm extension image list --location westus -o table

#az vm extension set `

--resource-group exttest `

--vm-name exttest `

--name customScript `

--publisher Microsoft.Azure.Extensions `

--settings '{"fileUris": ["https://gist.github.com/ahmetalpbalkan/b5d4a856fe15464015ae87d5587a4439/raw/466f5c30507c990a4d5a2f5c79f901fa89a80841/hello.sh"],"commandToExecute": "./hello.sh"}'

Install the Mongodb and Nodejs

# azure vm quick-create \

--image-urn CentOS \

--ssh-publickey-file ~/.ssh/id\_rsa.pub

#ssh [azureuser@40.78.23.145](mailto:azureuser@40.78.23.145)

#sudo touch /etc/yum.repos.d/mongodb-org-3.4.repo

打开该 MongoDB 存储库文件进行编辑。 添加以下行：

sh 复制

[mongodb-org-3.4]

name=MongoDB Repository

baseurl=https://repo.mongodb.org/yum/redhat/$releasever/mongodb-org/3.4/x86\_64/

gpgcheck=1

enabled=1

gpgkey=https://www.mongodb.org/static/pgp/server-3.4.asc

#sudo yum install -y mongodb-org

# sudo yum install -y policycoreutils-python

# sudo semanage port -a -t mongod\_port\_t -p tcp 27017

# sudo service mongod start

# mongo

# sudo chkconfig mongod on

Migration from Classic to ARM

#az vm list-usage -l <azure-region> -o jsonc

# az network list-usages -l <azure-region> -o jsonc

# az storage account show-usage

创建资源组

#az group create --name myResourceGroup --location eastus

创建VM

#az vm create --resource-group myResourceGroup --name myVM --image UbuntuLTS --generate-ssh-keys

为 Web 流量打开端口 80

#az vm open-port --port 80 --resource-group myResourceGroup --name myVM

update package source

#sudo apt-get -y update

install NGINX

#sudo apt-get -y install nginx

#az group delete --name myResourceGroup

# azure vm quick-create -M ~/.ssh/id\_rsa.pub -Q CoreOS

创建 SSH 密钥对

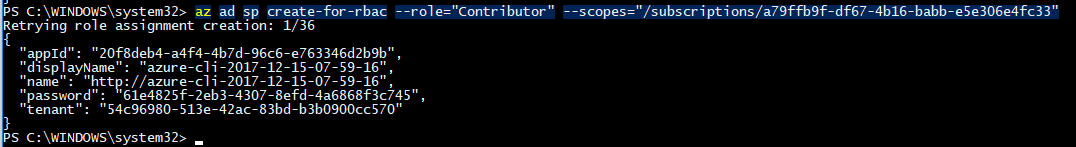
#ssh-keygen -t rsa -b 2048

# az account set --subscription="${SUBSCRIPTION\_ID}"

# az account show --query "{subscriptionId:id, tenantId:tenantId}"

# az ad sp create-for-rbac --role="Contributor" --scopes="/subscriptions/${SUBSCRIPTION\_ID}"

# az ad sp create-for-rbac --role="Contributor" --scopes="/subscriptions/a79ffb9f-df67-4b16-babb-e5e306e4fc33"



{

"appId": "20f8deb4-a4f4-4b7d-96c6-e763346d2b9b",

"displayName": "azure-cli-2017-12-15-07-59-16",

"name": "http://azure-cli-2017-12-15-07-59-16",

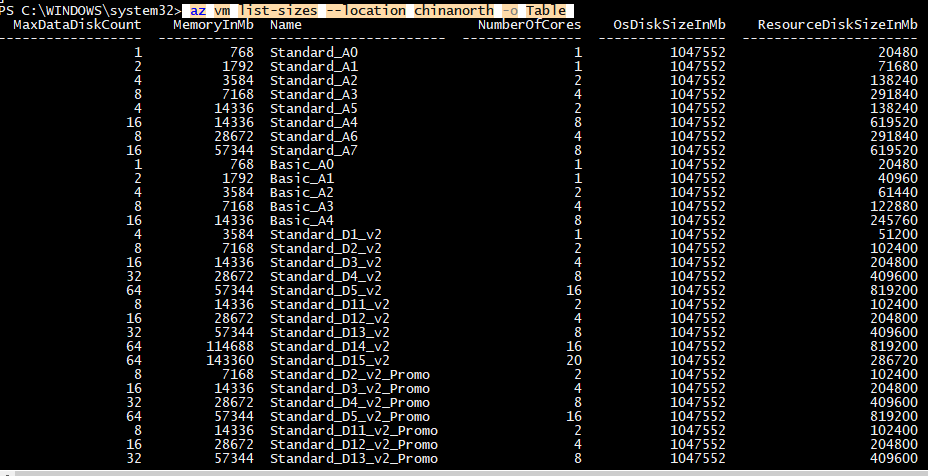
"password": "61e4825f-2eb3-4307-8efd-4a6868f3c745",

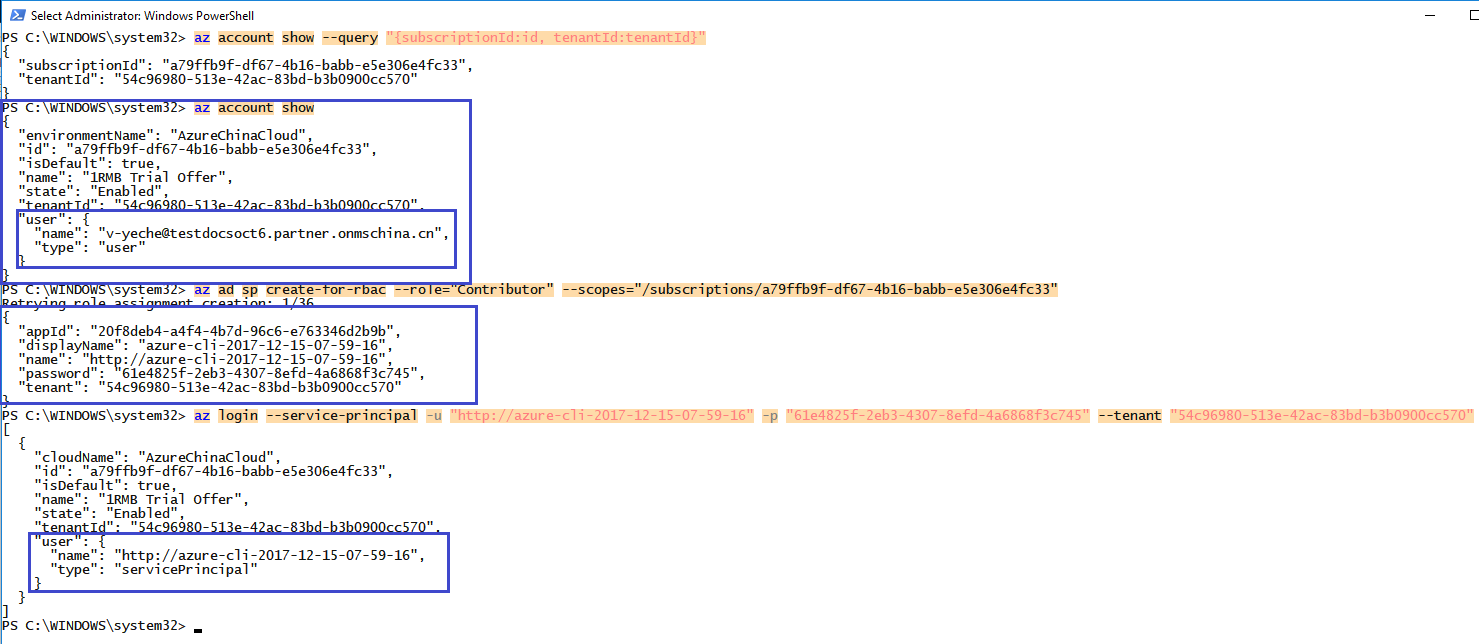
"tenant": "54c96980-513e-42ac-83bd-b3b0900cc570"

}

#az login --service-principal -u SP\_NAME -p PASSWORD --tenant TENANT

#az vm list-sizes --location chinanorth -o Table





创建 Azure 负载均衡器

* 1. Create Resource Group

#az group create --name myResourceGroupLoadBalancer --location chinanorth

* 1. Create Public IP Address

#az network public-ip create \

--resource-group myResourceGroupLoadBalancer \

--name myPublicIP

3. Create load Balancer

#az network lb create \

--resource-group myResourceGroupLoadBalancer \

--name myLoadBalancer \

--frontend-ip-name myFrontEndPool \

--backend-pool-name myBackEndPool \

--public-ip-address myPublicIP

4. 创建运行状况探测器

# az network lb probe create \

--resource-group myResourceGroupLoadBalancer \

--lb-name myLoadBalancer \

--name myHealthProbe \

--protocol tcp \

--port 80

5. 创建负载均衡器规则

# az network lb rule create \

--resource-group myResourceGroupLoadBalancer \

--lb-name myLoadBalancer \

--name myLoadBalancerRule \

--protocol tcp \

--frontend-port 80 \

--backend-port 80 \

--frontend-ip-name myFrontEndPool \

--backend-pool-name myBackEndPool \

--probe-name myHealthProbe

6. 创建网络资源

# az network vnet create \

--resource-group myResourceGroupLoadBalancer \

--name myVnet \

--subnet-name mySubnet

7. 若要添加网络安全组

# az network nsg create \

--resource-group myResourceGroupLoadBalancer \

--name myNetworkSecurityGroup

8. 使用 [az network nsg rule create](https://docs.microsoft.com/zh-cn/cli/azure/network/nsg/rule#create) 创建网络安全组规则

# az network nsg rule create \

--resource-group myResourceGroupLoadBalancer \

--nsg-name myNetworkSecurityGroup \

--name myNetworkSecurityGroupRule \

--priority 1001 \

--protocol tcp \

--destination-port-range 80

9. 以下示例创建三个虚拟 NIC

for i in `seq 1 3`; do

az network nic create \

--resource-group myResourceGroupLoadBalancer \

--name myNic$i \

--vnet-name myVnet \

--subnet mySubnet \

--network-security-group myNetworkSecurityGroup \

--lb-name myLoadBalancer \

--lb-address-pools myBackEndPool

Done

10. 要提高应用的高可用性，请将 VM 放置在可用性集中。

az vm availability-set create \

--resource-group myResourceGroupLoadBalancer \

--name myAvailabilitySet

11. 测试负载均衡器

# az network public-ip show \

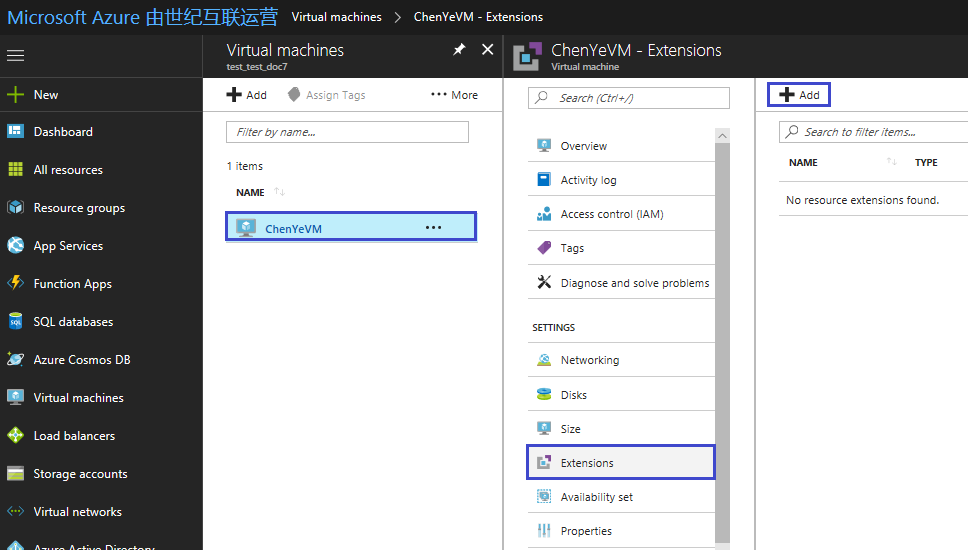
--resource-group myResourceGroupLoadBalancer \

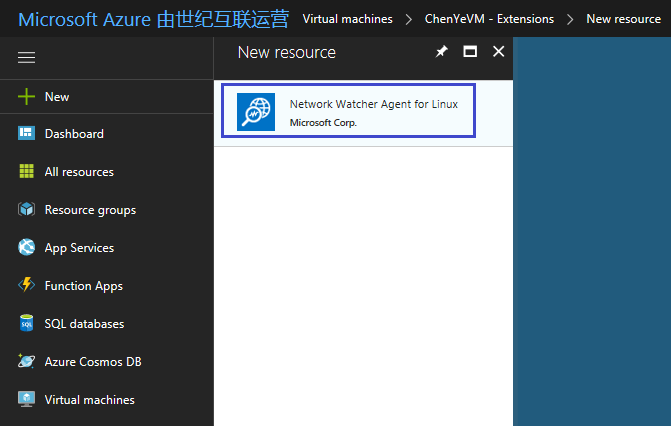
--name myPublicIP \ss

--query [ipAddress] \

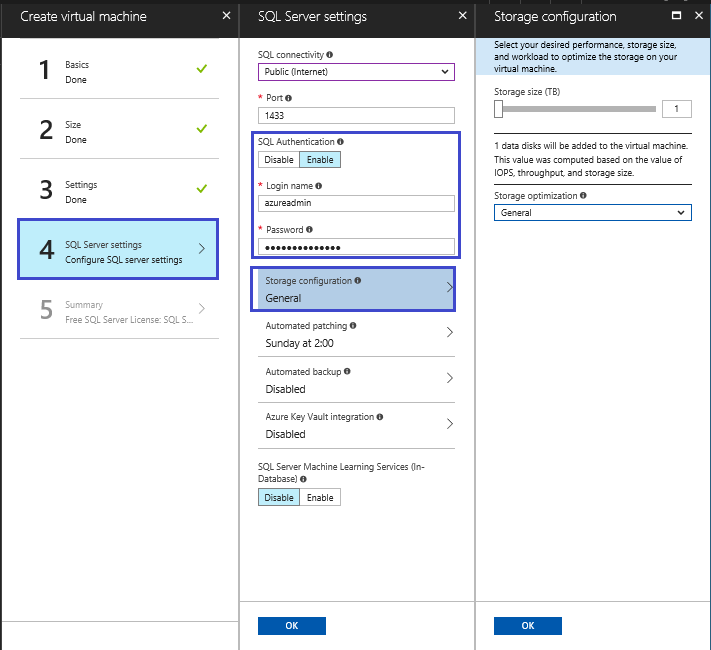
--output tsv

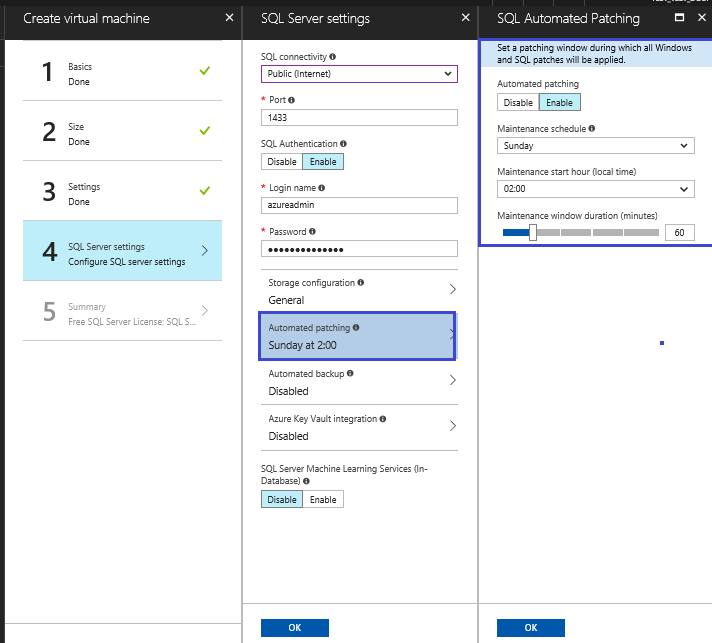
Linux Extentions valid on Network Watcher Agent for Linuxs.

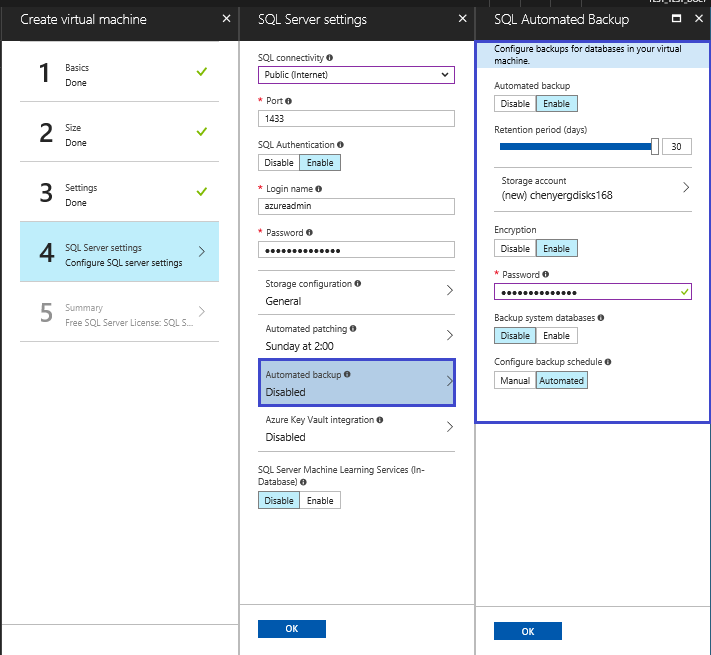


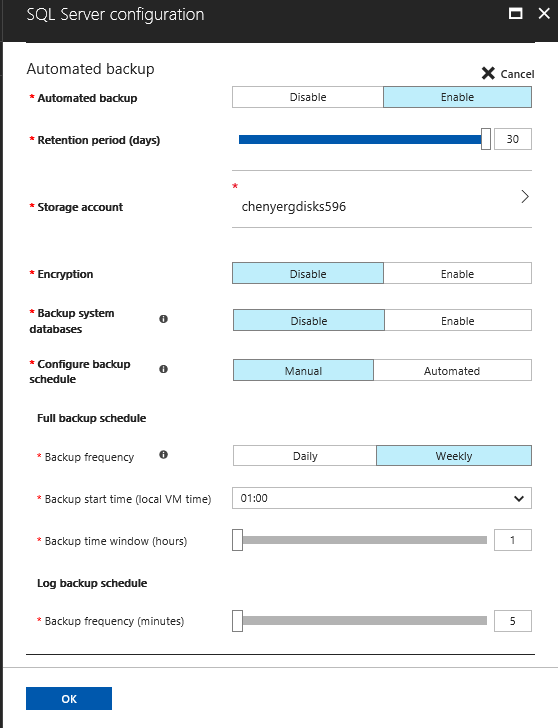


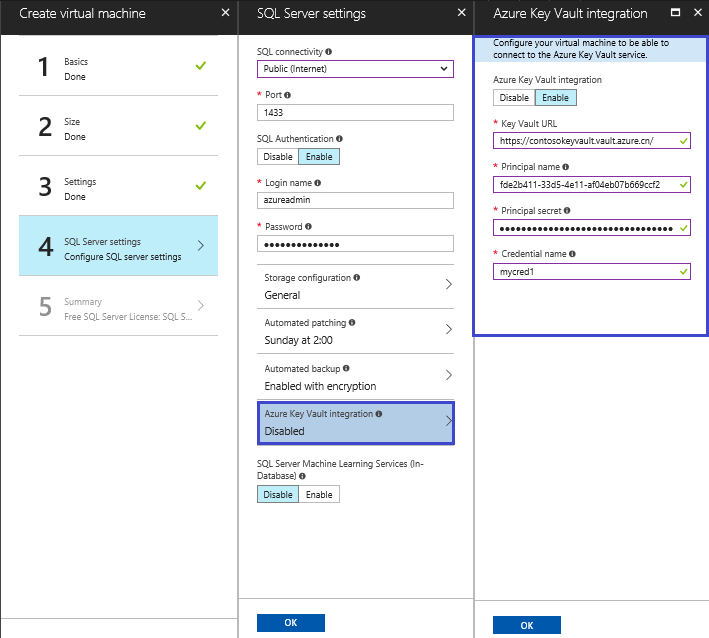
Install SQL 2017 on New VM

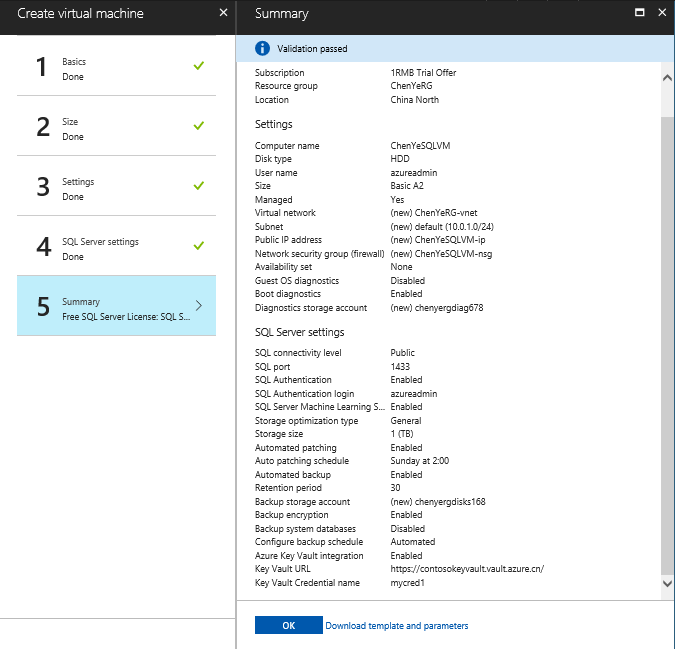




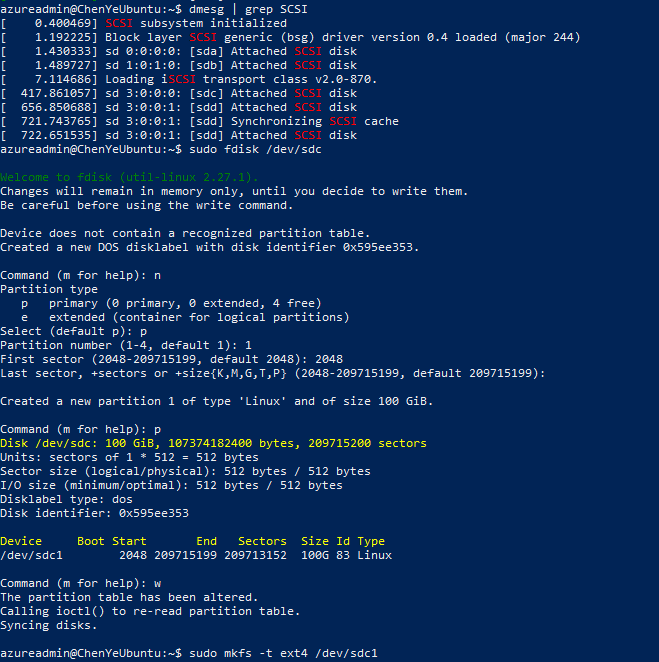
Backup V2

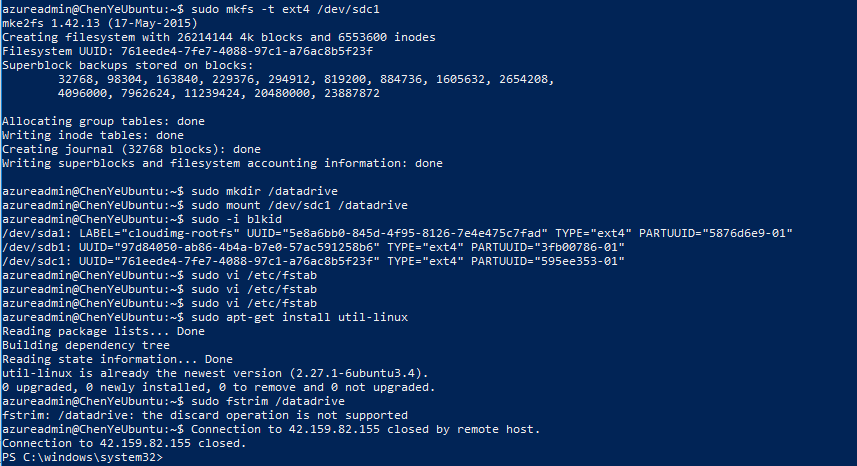






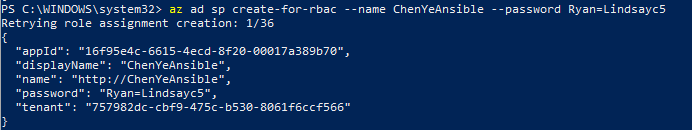
attach-disk-portal.md verify successfully





**How to create Create the service principal**

az ad sp create-for-rbac --name ChenYeAnsible --password Ryan=Lindsayc5



output below:

{

"appId": "16f95e4c-6615-4ecd-8f20-00017a389b70",

"displayName": "ChenYeAnsible",

"name": "http://ChenYeAnsible",

"password": "Ryan=Lindsayc5",

"tenant": "757982dc-cbf9-475c-b530-8061f6ccf566"

}

## Install pre-requisite packages

sudo apt-get update && sudo apt-get install -y libssl-dev libffi-dev python-dev python-pip

## Install Ansible and Azure SDKs via pip

sudo pip install ansible[azure]

create ~/.azure/credential file with following content

[default]

subscription\_id=4e781b17-e02d-4e5c-a9d8-894d5f1532bd

client\_id=16f95e4c-6615-4ecd-8f20-00017a389b70

secret=Ryan=Lindsayc5

tenant=757982dc-cbf9-475c-b530-8061f6ccf566

cloud\_environment=AzureChinaCloud

Or set bash export environment.

export AZURE\_SUBSCRIPTION\_ID=4e781b17-e02d-4e5c-a9d8-894d5f1532bd

export AZURE\_CLIENT\_ID=16f95e4c-6615-4ecd-8f20-00017a389b70

export AZURE\_SECRET= Ryan=Lindsayc5

export AZURE\_TENANT=757982dc-cbf9-475c-b530-8061f6ccf566

Managed disk change from standard HDD to Standard SSD

