

Setting static IP for VM (Ubuntu)

Author: Xu Jiachen

Email: liyingxujiachen@gmail.com e0402032@u.nus.edu

Abstract

由于在较小内存中的宿主机中使用VM及其中的应用可能会造成宿主机卡顿等影响使用体验的情况，现采用设立固定ip的方式，从而解决部分可以使用浏览器访问的应用可以在宿主机中访问而不是在VM中访问，以减轻VM的负载

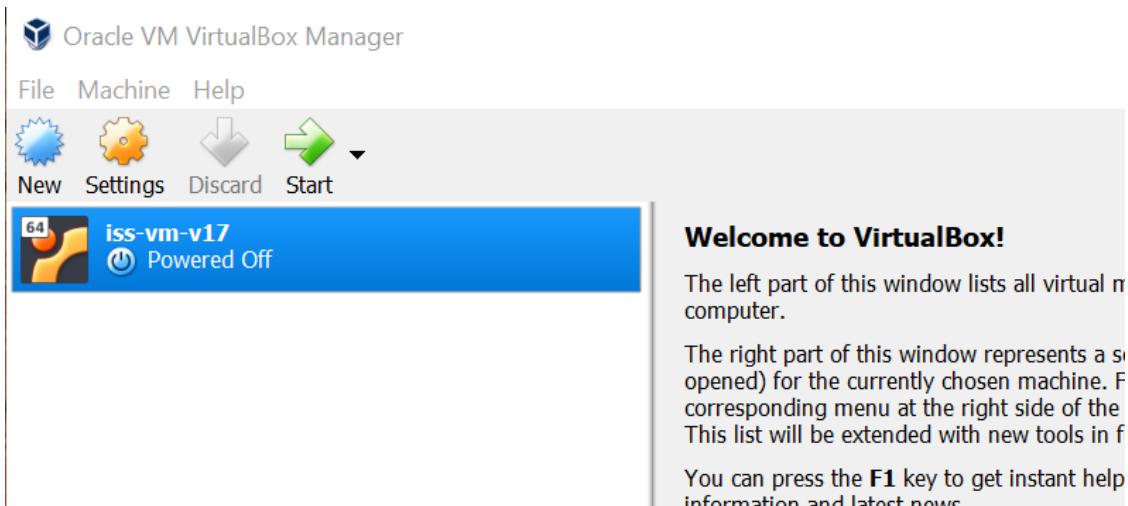
Tools

1. VirtualBox
2. VM (以ISS-VM【Ubuntu】为例)
 - account: iss-user
 - password: iss-user

Procedure

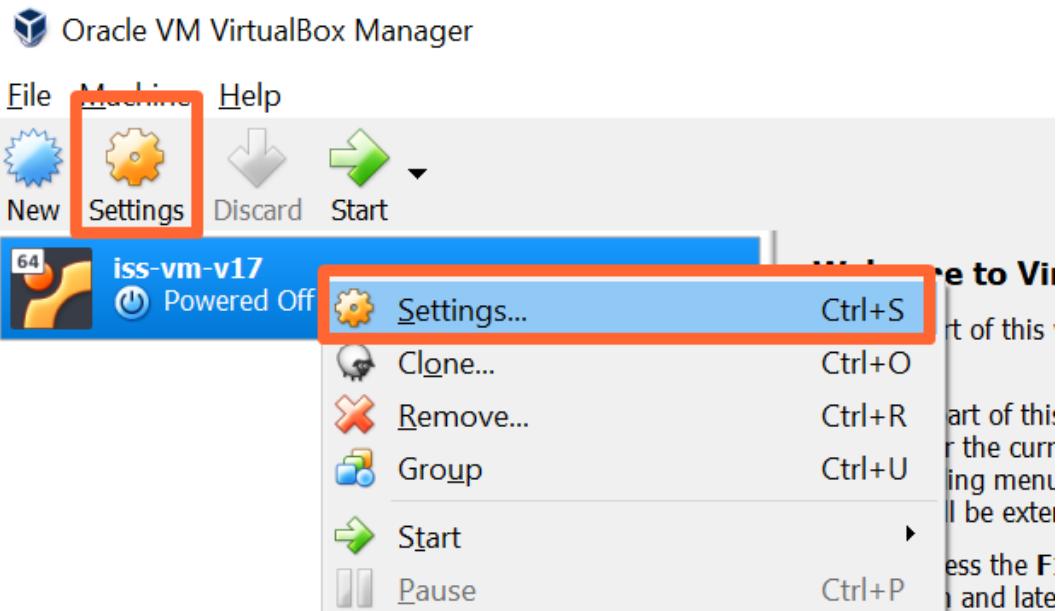
Setting static IP

1. 将VM导入至VirtualBox中

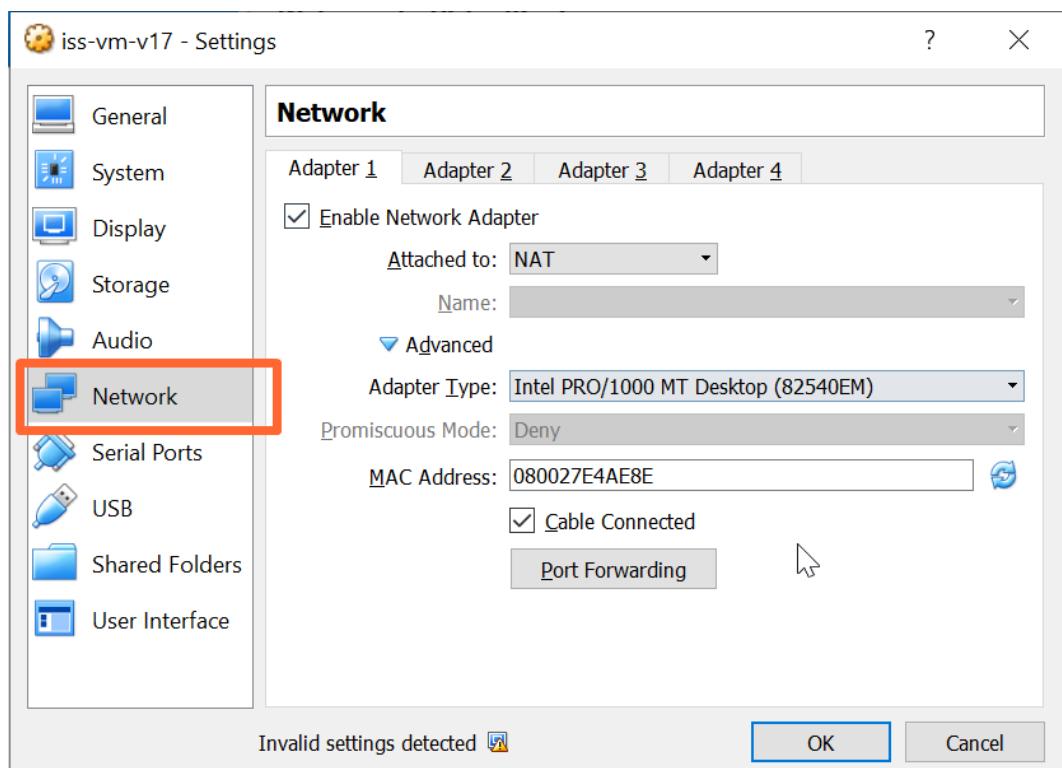


2. (在虚拟机正常关闭的情况下) 修改VM的网卡配置，给VM增加一张host-only的网卡，用于设置固定IP，保留原NAT的网卡以保证VM内部可以访问互联网

1. 选中需要配置的虚拟机，点击工具栏中的Settings按钮或选中需要配置的虚拟机点击鼠标右键在弹出的菜单中选中Settings



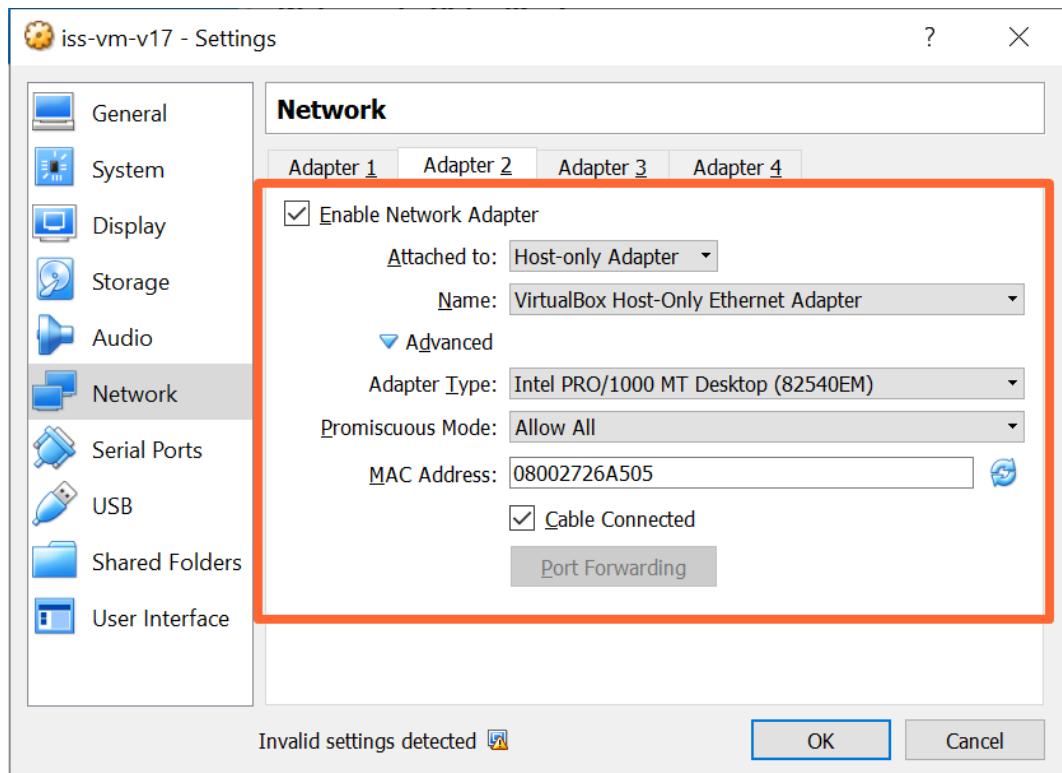
2. 在弹出的对话框的右侧点击Network



3. 对于当前VM，已经默认配置有一个Adapter，设置为NAT模式，用于虚拟机采用NAT模式连接互联网，若没有配置该网卡，请配置该网卡用于后期的端口映射

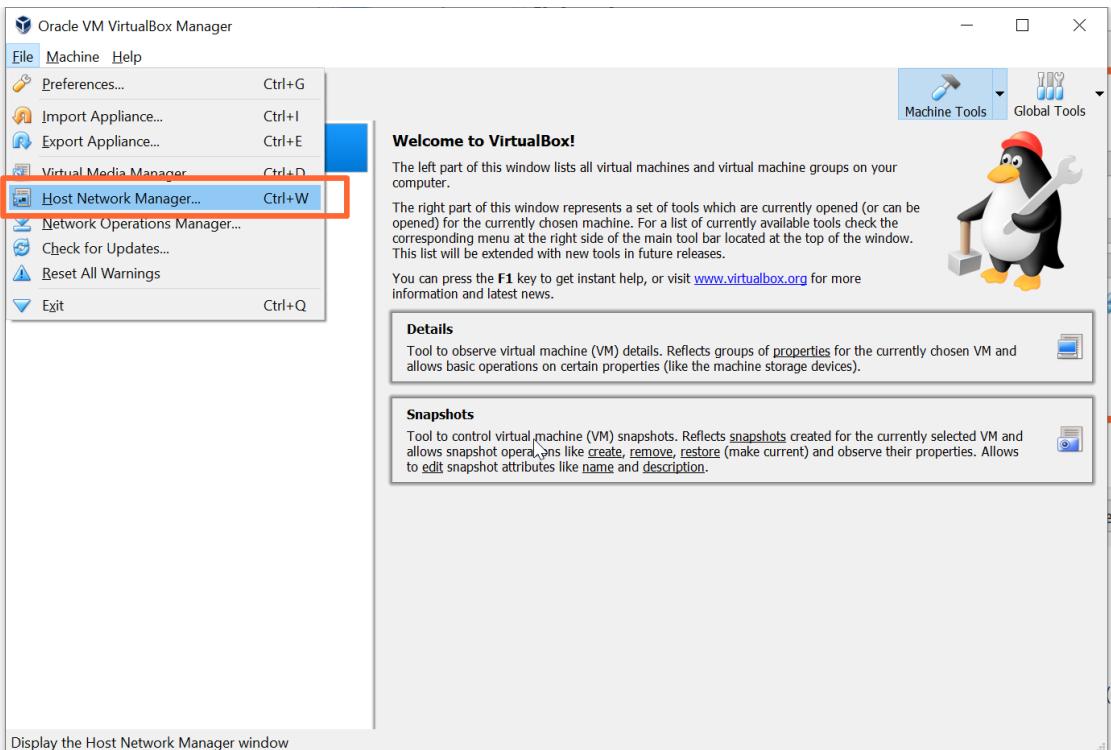
4. 点击Adapter 2 选项卡（若该虚拟机已经配置了Adapter 2，可以使用Adapter 3 or Adapter 4 对结果没有影响）

5. 配置Adapter 2 网卡

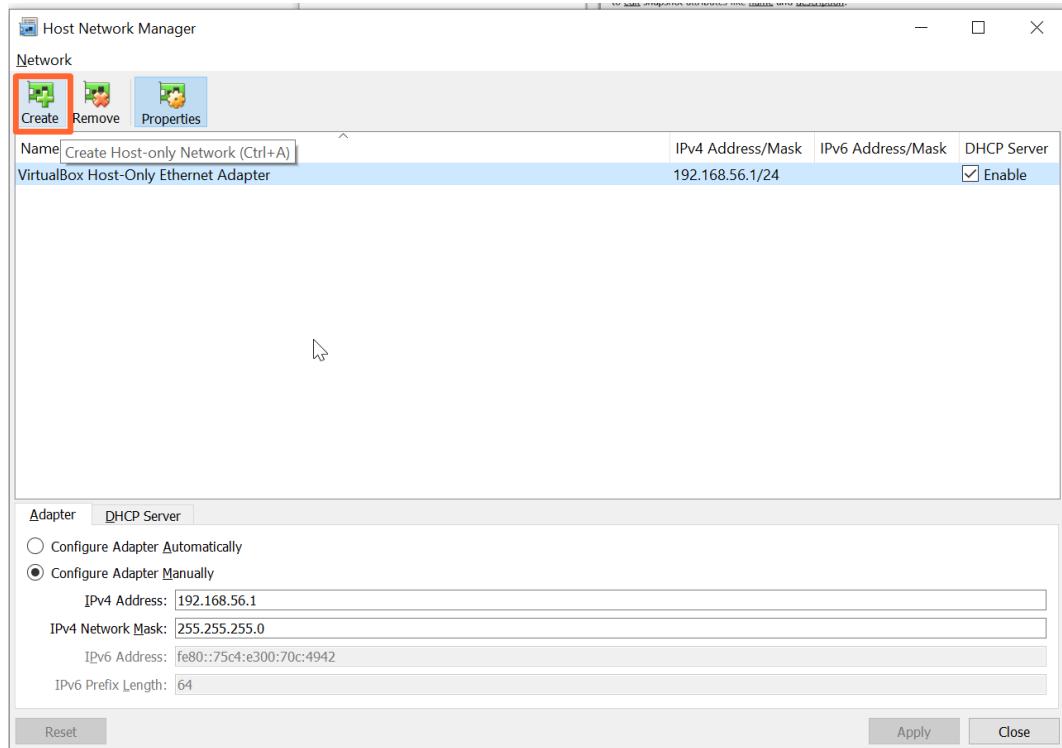


1. 激活该网卡
 2. Attached to: Host-only Adapter
 3. Name: VirtualBox Host-Only Ethernet Adapter
 1. 对于MAC需要在全局的网络管理界面增加一张本地的虚拟网卡，然后再进行上述配置(*)
 4. Promiscuous Mode: Allow All
 5. 点击确定按钮
3. (*) MacOS可能默认没有Host-only Adapter，如果没有请在VirtualBox窗口中，请按照如下步骤创建该网卡

1. 点击位于上方工具栏中的File按钮，在弹出的菜单中选择 Host Network Manager

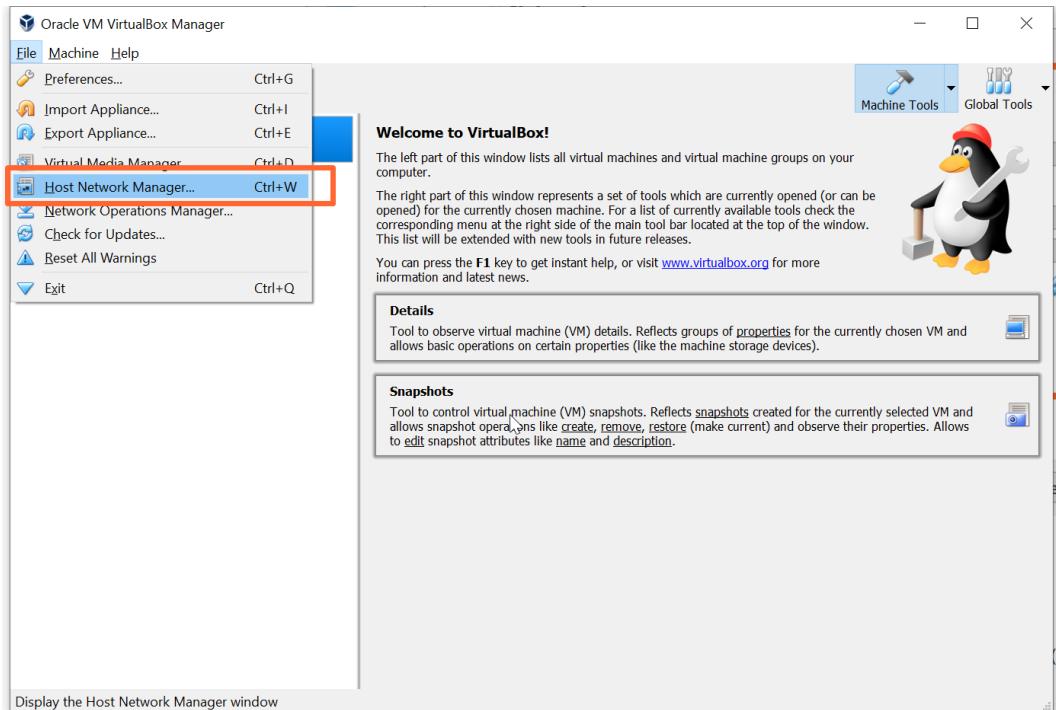


2. 在弹出的窗口中，点击上方工具栏中的Create按钮，创建一张网卡用于给虚拟机分配，并打开该网卡的DHCP Server

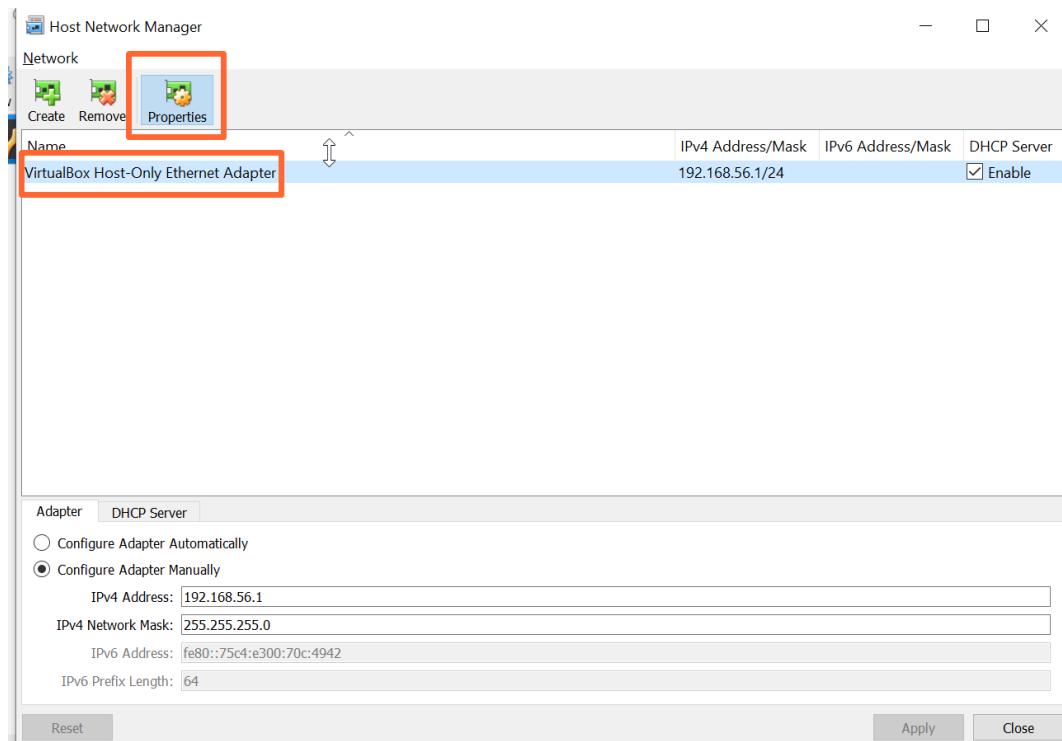


4. 查看分配的网卡可以使用的IP地址范围

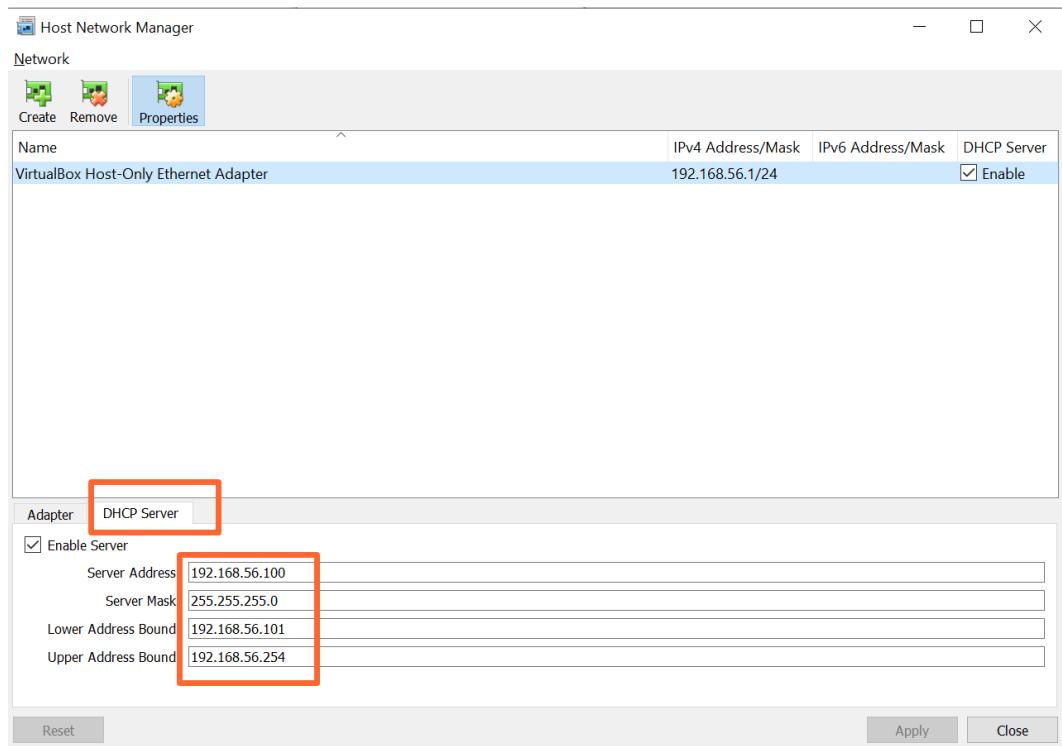
1. 点击VirtualBox 菜单栏中的File按钮，在弹出的菜单中选择Host Network Manager



2. 在弹出的窗口中选择刚刚给虚拟机加上的网卡：VirtualBox Host-Only Ethernet Adapter，并点击位于上方工具栏中的Properties按钮，即可在下方查看当前网卡的属性



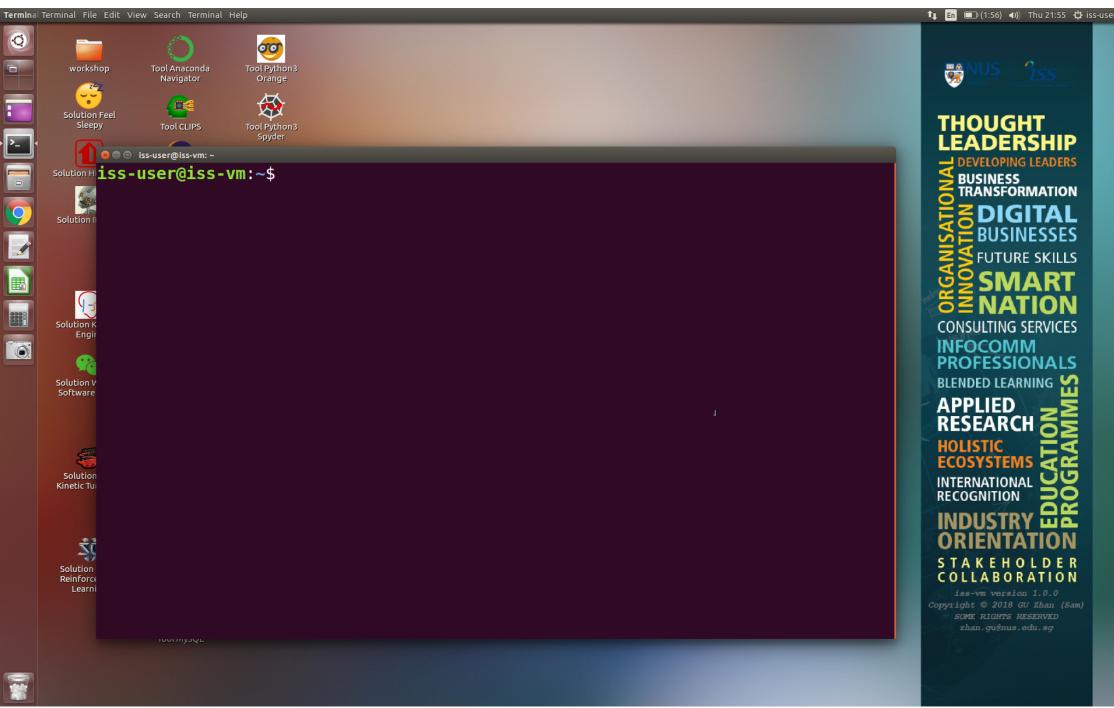
3. 点击下方属性栏中的DHCP Server选项卡，查看该网卡可配置的IP地址范围



4. 记录以下信息以便在后面的步骤中配置固定IP地址

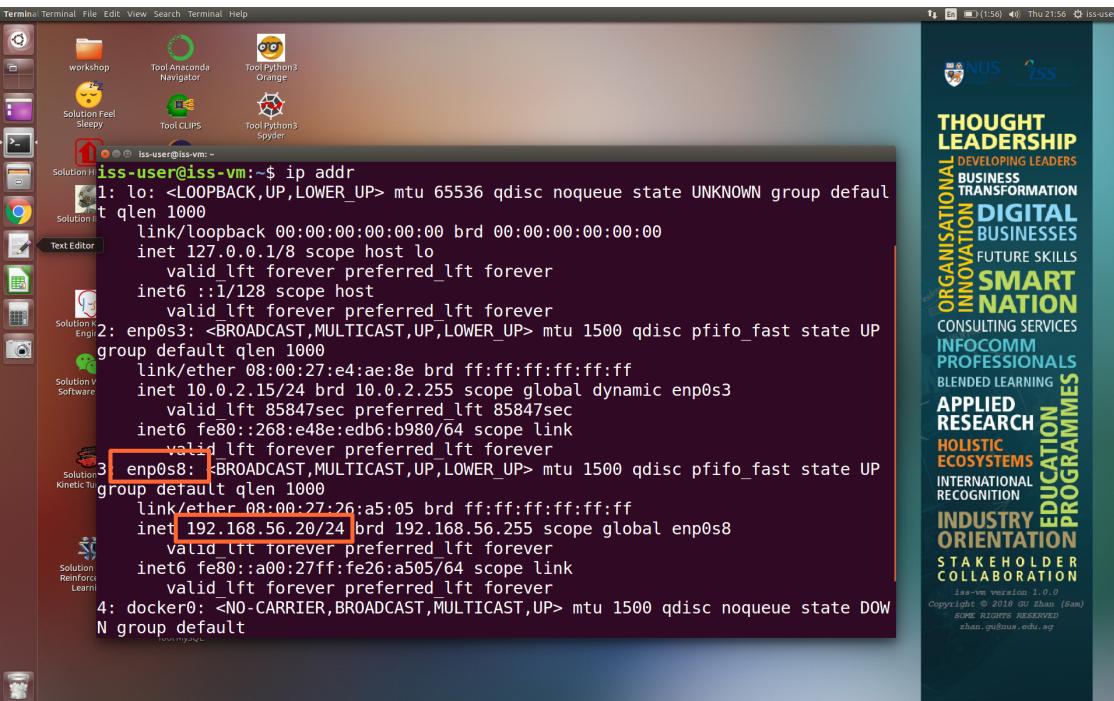
1. Lower Address Bound: 192.168.56.101
2. Upper Address Bound: 192.168.56.254
3. Server Mask: 255.255.255.0

5. 启动虚拟机，启动一个终端窗口



6. 使用 `ip addr` 查看是否已经增加了一张网卡，ip地址以192.168.56.X命名并记录该网卡的名字

192.168.56.101<VM的IP地址<192.168.56.254



7. 设置固定IP地址

1. 使用 `sudo nano /etc/network/interfaces` 修改该网卡的配置文件,增加如下内容

```
auto enp0s8
iface enp0s8 inet static
    address 192.168.56.20
    netmask 255.255.255.0
```

- 此处的 `address` 一定要在之前记录的该网卡可以配置的IP地址的范围之内
(192.168.56.101--192.168.56.254)
- 使用 `ctrl+x` and `y` 保存修改并退出当前文件

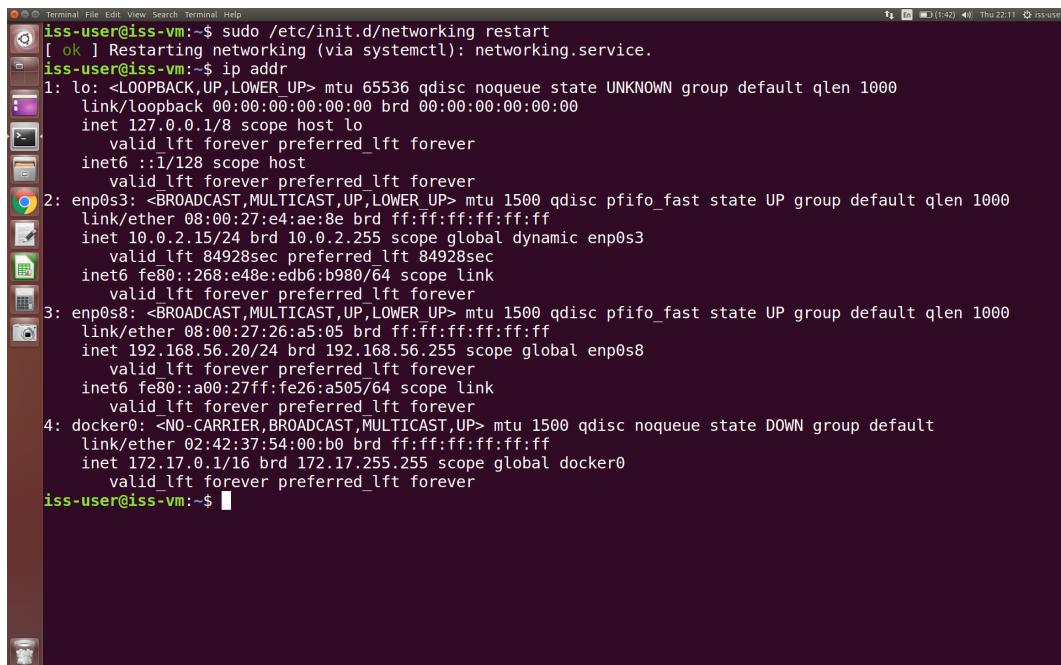
```
# interfaces(5) file used by ifup(8) and ifdown
auto lo
iface lo inet loopback

auto enp0s8
iface enp0s8 inet static
    address 192.168.56.20
    netmask 255.255.255.0
```

2. 使用 `sudo /etc/init.d/networking restart` 重启网络

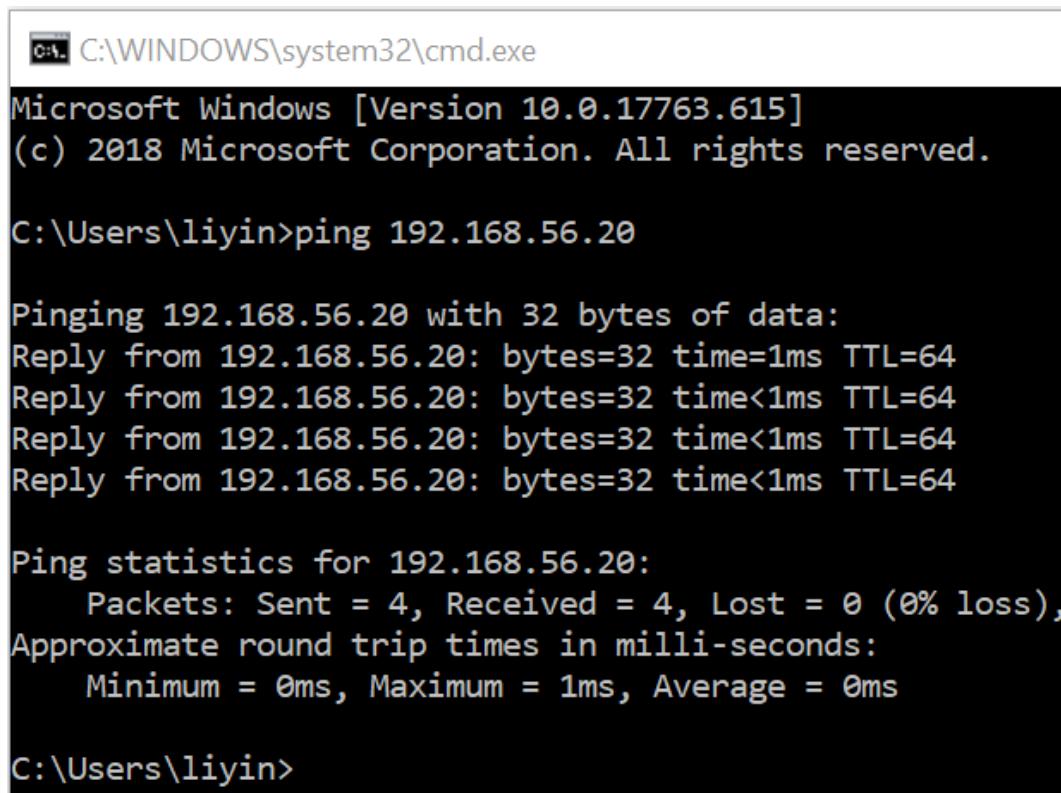
```
iss-user@iss-vm:~$ sudo /etc/init.d/networking restart
[ ok ] Restarting networking (via systemctl): networking.service.
```

3. 使用 `ip addr` 查看当前的是否已经配置好固定ip



```
iss-user@iss-vm:~$ sudo /etc/init.d/networking restart
[ ok ] Restarting networking (via systemctl): networking.service.
iss-user@iss-vm:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:e4:ae:8e brd ff:ff:ff:ff:ff:ff
        inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
            valid_lft 84928sec preferred_lft 84928sec
        inet6 fe80::2e4:edb6:b980/64 scope link
            valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:26:a5:05 brd ff:ff:ff:ff:ff:ff
        inet 192.168.56.20/24 brd 192.168.56.255 scope global enp0s8
            valid_lft forever preferred_lft forever
        inet6 fe80::a00:27ff:fe26:a505/64 scope link
            valid_lft forever preferred_lft forever
4: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:37:54:00:b0 brd ff:ff:ff:ff:ff:ff
        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
            valid_lft forever preferred_lft forever
iss-user@iss-vm:~$
```

4. 在宿主机上 使用 `ping 192.168.56.x` 命令查看是否能够ping通虚拟机



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.17763.615]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\liyin>ping 192.168.56.20

Pinging 192.168.56.20 with 32 bytes of data:
Reply from 192.168.56.20: bytes=32 time=1ms TTL=64
Reply from 192.168.56.20: bytes=32 time<1ms TTL=64
Reply from 192.168.56.20: bytes=32 time<1ms TTL=64
Reply from 192.168.56.20: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.56.20:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Users\liyin>
```

5. VM 固定IP配置完成 (配置的IP地址可能和DHCP Server原来分配的地址相同，但是配置固定IP地址后，该VM的IP地址不会随着VM的重启而改变)

Testing with KIE

8. 测试（将KIE服务映射到出虚拟机）

1. 修改 KIE中的JBoss 配置（因为 JBOSS 默认只能从localhost访问，如果需要被整个局域网都访问，需要修改其配置文件）

1. 找到KIE中的JBoss的配置文件

```

iss-user@iss-vm:~$ pwd
/home/iss-user
iss-user@iss-vm:~$ cd ./Desktop/
iss-user@iss-vm:~/Desktop$ cat Tool\ KIE\ 7.11.desktop | grep Exec
Exec=/home/iss-user/iss-vm-program/is-intelligent-reasoning-systems/jboss/jbpm-server-7.12.0.Final-dist/bin/
standalone.sh
iss-user@iss-vm:~/Desktop$ cd /home/iss-user/iss-vm-program/is-intelligent-reasoning-systems/jboss/jbpm-server-7.12.0.Final-dist/
iss-user@iss-vm:~/iss-vm-program/is-intelligent-reasoning-systems/jboss/jbpm-server-7.12.0.Final-dist$ ls
appclient  copyright.txt  domain  LICENSE.txt  README.txt  welcome-content
bin      docs      jboss-modules.jar  modules  standalone
standalone/configuration/
iss-user@iss-vm:~/iss-vm-program/is-intelligent-reasoning-systems/jboss/jbpm-server-7.12.0.Final-dist$ cd ../
iss-user@iss-vm:~/iss-vm-program/is-intelligent-reasoning-systems/jboss/jbpm-server-7.12.0.Final-dist/standalone/configuration$ ls
application-roles.properties  mgmt-users.properties  standalone-ha.xml  users.properties
application-users.properties  roles.properties  standalone-load-balancer.xml
logging.properties  standalone-full-ha.xml  standalone.xml
mgmt-groups.properties  standalone-full.xml  standalone_xml_history
iss-user@iss-vm:~/iss-vm-program/is-intelligent-reasoning-systems/jboss/jbpm-server-7.12.0.Final-dist/standalone/configuration$ nano standalone.xml

```

2. 使用 `nano standalone.xml` 修改该配置文件

- 需要修改的是JBoss 使用 standalone模式启动时候的IP地址
- 在nano编辑器中使用 `Ctrl+W` 搜索 `jboss.bind.address.management`

```

GNU nano 2.5.3                               File: standalone.xml

<?xml version='1.0' encoding='UTF-8'?>
<server xmlns="urn:jboss:domain:5.0">
  <extensions>
    <extension module="org.jboss.as.clustering.infinispan"/>
    <extension module="org.jboss.as.connector"/>
    <extension module="org.jboss.as.deployment-scanner"/>
    <extension module="org.jboss.as.ee"/>
    <extension module="org.jboss.as.ejb3"/>
    <extension module="org.jboss.as.jaxrs"/>
    <extension module="org.jboss.as.jdr"/>
    <extension module="org.jboss.as.jmx"/>
    <extension module="org.jboss.as.jpa"/>
    <extension module="org.jboss.as.jsf"/>
    <extension module="org.jboss.as.jsr77"/>
    <extension module="org.jboss.as.logging"/>
    <extension module="org.jboss.as.mail"/>
    <extension module="org.jboss.as.naming"/>
    <extension module="org.jboss.as.pojo"/>
    <extension module="org.jboss.as.remoting"/>
    <extension module="org.jboss.as.sar"/>
    <extension module="org.jboss.as.security"/>
    <extension module="org.jboss.as.transactions"/>
    <extension module="org.jboss.as.webservices"/>
    <extension module="org.jboss.as.weld"/>
    <extension module="org.wildfly.extension.batch.jberet"/>
    <extension module="org.wildfly.extension.bean-validation"/>
    <extension module="org.wildfly.extension.core-management"/>
    <extension module="org.wildfly.extension.elytron"/>
  </extensions>
  <search path="jboss.bind.address.management" />

```

- 修改下图中圈出的三处IP地址将其都改为0.0.0.0

```

GNU nano 2.5.3                               File: standalone.xml

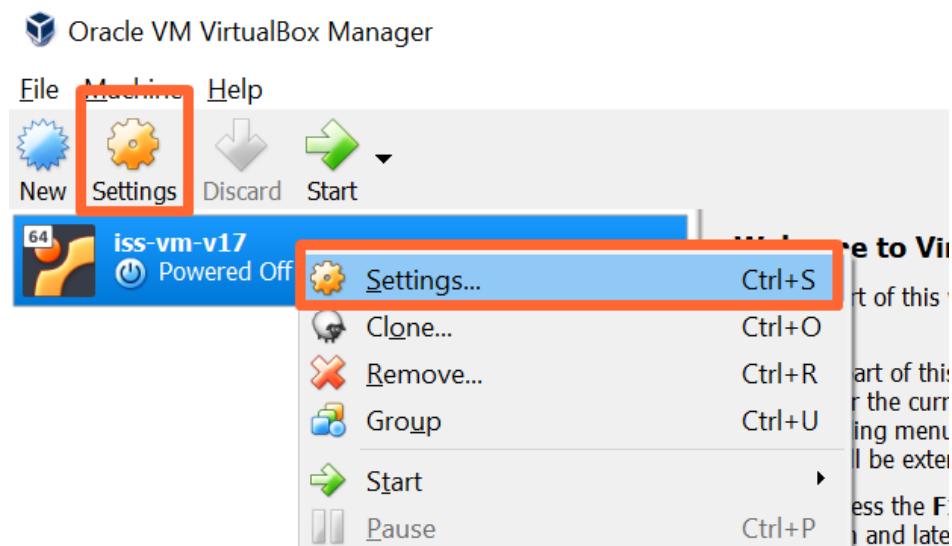
<subsystem xmlns="urn:jboss:domain:webservices:2.0">
  <wsdl-host>${jboss.bind.address:127.0.0.1}</wsdl-host>
  <endpoint-config name="Standard-Endpoint-Config"/>
  <endpoint-config name="Recording-Endpoint-Config">
    <pre-handler-chain name="recording-handlers" protocol-bindings="##SOAP11_HTTP ##SOAP11_HTTPS">
      <handler name="RecordingHandler" class="org.jboss.ws.common.invocation.RecordingServerHandler"/>
    </pre-handler-chain>
  </endpoint-config>
  <client-config name="Standard-Client-Config"/>
</subsystem>
<subsystem xmlns="urn:jboss:domain:weld:4.0"/>
</profile>
<interfaces>
  <interface name="management">
    <inet-address value="${jboss.bind.address.management:0.0.0.0}"/>
  </interface>
  <interface name="public">
    <inet-address value="${jboss.bind.address[0]:0.0.0.0}"/>
  </interface>
  <interface name="unsecure">
    <inet-address value="${jboss.bind.address.unsecure:0.0.0.0}"/>
  </interface>
</interfaces>
<socket-binding-group name="standard-sockets" default-interface="public" port-offset="${jboss.socket.binding.port-offset:0}">
  <socket-binding name="management-http" interface="management" port="${jboss.management.http.port:9990}"/>
  <socket-binding name="management-https" interface="management" port="${jboss.management.https.port:9993}"/>
  <socket-binding name="ajp" port="${jboss.ajp.port:8009}"/>
  <socket-binding name="http" port="${jboss.http.port:8080}"/>
  <socket-binding name="https" port="${jboss.https.port:8443}"/>

```

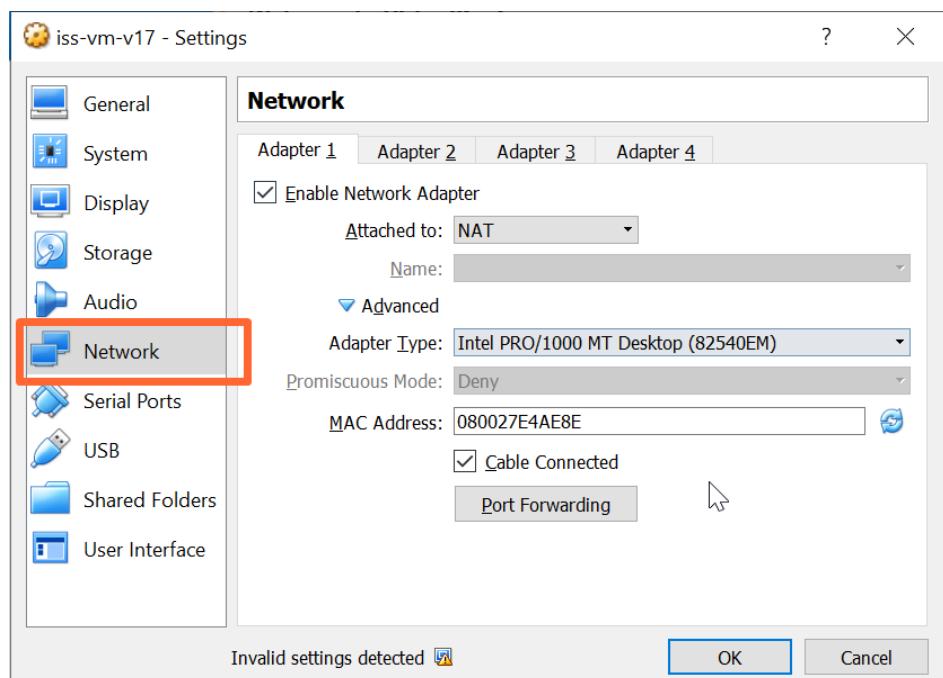
- 使用 `ctrl+x` and `y` 保存修改并当前文件

3. 在虚拟机设置中修改端口映射，将JBoss默认的端口8080，映射到宿主机的8080端口上
(可以映射不同端口)

1. 选中需要配置的虚拟机，点击工具栏中的Settings按钮或选中需要配置的虚拟机点击鼠标右键在弹出的菜单中选中Settings



2. 在弹出的对话框的右侧点击Network

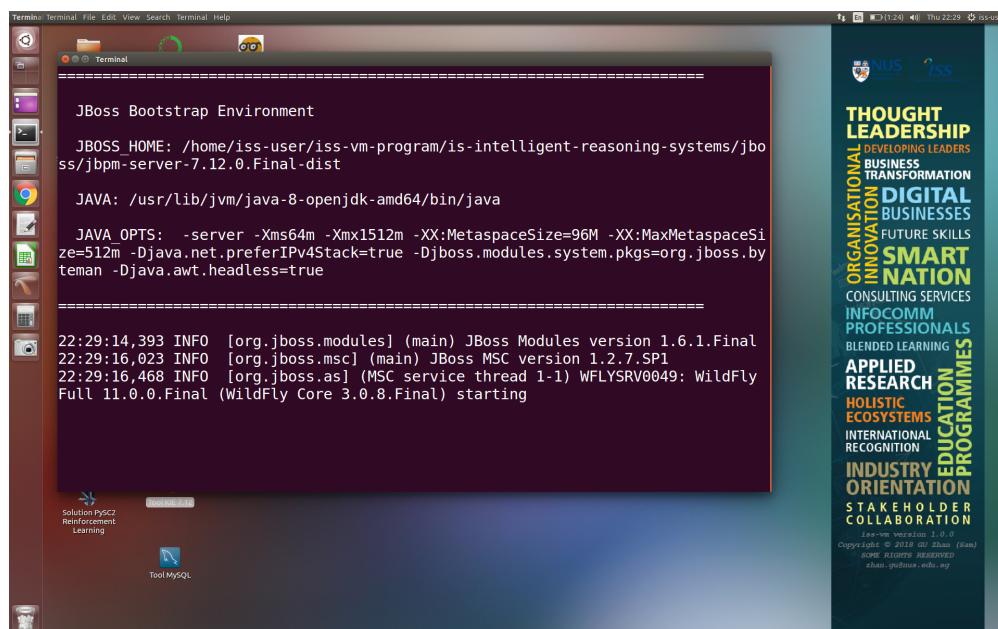


3. 在使用NAT模式的网卡中 点击 Port Forwarding按钮

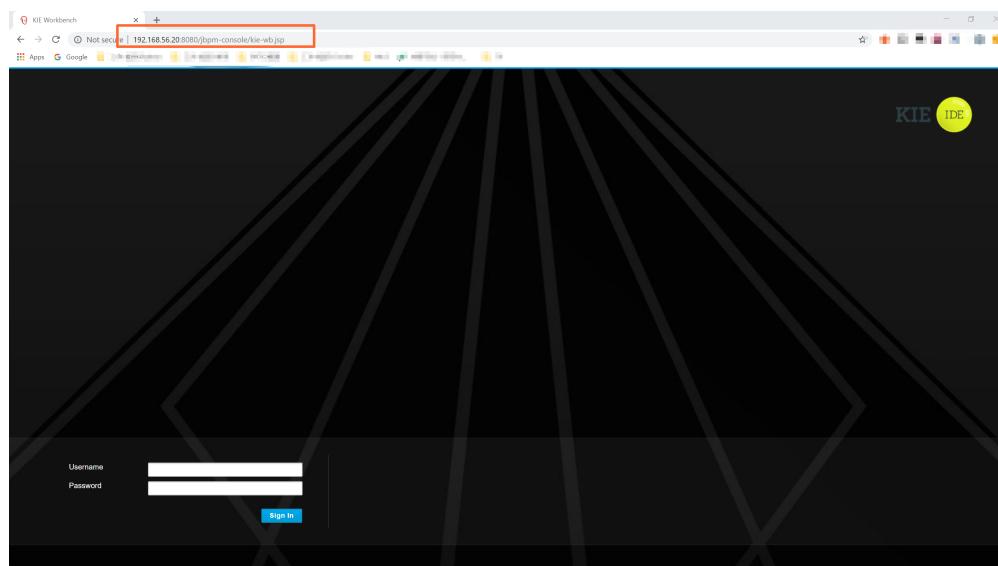
4. 在弹出的对话框中增加一条端口映射规则 8080 --- 8080 (宿主机端口可以随意修改只要保证不与宿主机现有服务所占用的端口冲突即可)

Name	Protocol	Host IP	Host Port	Guest IP	Guest Port
Rule 1	TCP		8888		8888
Rule 2	TCP		8080		8080

4. 启动位于VM桌面的KIE快捷方式



5. 带启动完成后，在宿主机上使用之前设置的IP和映射出的端口进行访问 (<http://192.168.56.20:8080/jbpm-console/>)



6. wbadmin登录成功，测试结束

