

任务背景

昨天播仔收到邮件，说服务器172.16.45.21/24上的vsftpd程序存在安全漏洞，需要尽快处理。一般这种问题，都是通过打补丁或者升级软件的方式解决。

小王，昨天网络信息安全部门发邮件过来，说我们的文件共享服务器存在安全漏洞，需要我们这边处理一下。这样，我们测试环境有一台服务器目前没有用，你空了把它直接安装成rhel8的系统，配置好软件仓库，安装一下文件共享服务软件vsftpd，因为rhel8系统自带的vsftpd软件版本比较新，所以你先在测试环境下测试一下。



好的，播哥。

任务要求

- ☒ 测试机服务器安装RedHat8操作系统
- ☐ rhel8基础系统环境配置
- ☐ 安装较新版本的vsftpd软件

任务分析

- 测试服务器上安装红帽8操作系统
- 红帽8操作系统基础环境配置
 - 配置主机名
 - 配置网络（静态IP）
 - 直接修改配置文件（建议）
 - 通过nmcli工具（熟练）
 - 通过nmtui工具（了解）
 - 配置防火墙和selinux
 - 配置软件仓库
- 安装vsftpd软件（功能：文件共享，ftp服务）

知识储备

一、RHEL8 Web控制台管理系统

1、RHEL8的Web控制台介绍

(-) rhel8的web控制台是什么？

- RHEL Web控制台是一个基于Web的红帽企业版Linux 8界面，用于管理和监视本地系统以及位于网络环境中的Linux服务器。
- RHEL 8 Web控制台是交互式服务器管理界面，通过浏览器与真实的Linux操作系统交互。

(-) Web控制台可以做什么？

- 监控基本系统功能，例如硬件信息，时间配置，性能配置等
- 检查系统日志文件
- 管理网络接口和配置防火墙
- 管理虚拟机
- 管理用户帐户
- 监视和配置系统服务
- 管理软件包
- 配置SELinux
- 更新软件
- 访问终端

2、安装Web控制台

(-) 系统默认已安装

```
[root@heima ~]# yum list|grep cockpit
cockpit.x86_64                                185-2.el8                                @anaconda
cockpit-bridge.x86_64                        185-2.el8                                @anaconda
cockpit-packagekit.noarch                    184.1-1.el8                              @AppStream
cockpit-storaged.noarch                      184.1-1.el8                              @AppStream
cockpit-system.noarch                       185-2.el8                                @anaconda
cockpit-ws.x86_64                            185-2.el8                                @anaconda
subscription-manager-cockpit.noarch          1.23.8-35.el8                            @anaconda
[root@heima ~]#
```

(-) 设置否开机自启动

查看是否开机自启动：

```
[root@heima ~]# systemctl list-unit-files|grep cockpit
cockpit-motd.service          static
cockpit.service               static
cockpit.socket                disabled-->说明开机不自动启动
```

设置开机自启动

```
[root@heima ~]# systemctl enable --now cockpit.socket
Created symlink /etc/systemd/system/sockets.target.wants/cockpit.socket →
/usr/lib/systemd/system/cockpit.socket.
```

```
[root@heima ~]# systemctl list-unit-files|grep cockpit
cockpit-motd.service          static
cockpit.service               static
cockpit.socket                enabled -->说明开机自动启动
```

启动cockpit服务

```
[root@heima ~]# systemctl start cockpit.service
```

查看状态

```
[root@heima ~]# systemctl status cockpit.service
```

(三) 设置防火墙策略 (可选)

说明: 如果系统防火墙开启, 则需要执行以下操作, 添加cockpit服务到防火墙以打开9090端口

```
[root@heima ~]# firewall-cmd --add-service=cockpit --permanent
```

```
Warning: ALREADY_ENABLED: cockpit
```

```
success
```

```
[root@heima ~]# firewall-cmd --reload
```

```
success
```

3、登录Web控制台

说明: 默认情况下, cockpit服务是启动的, 我们可以使用以下命令检查9090端口是否监听

```
[root@heima ~]# lsof -i :9090
```

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
systemd	1	root	24u	IPV6	72770	0t0	TCP	*:websm (LISTEN)

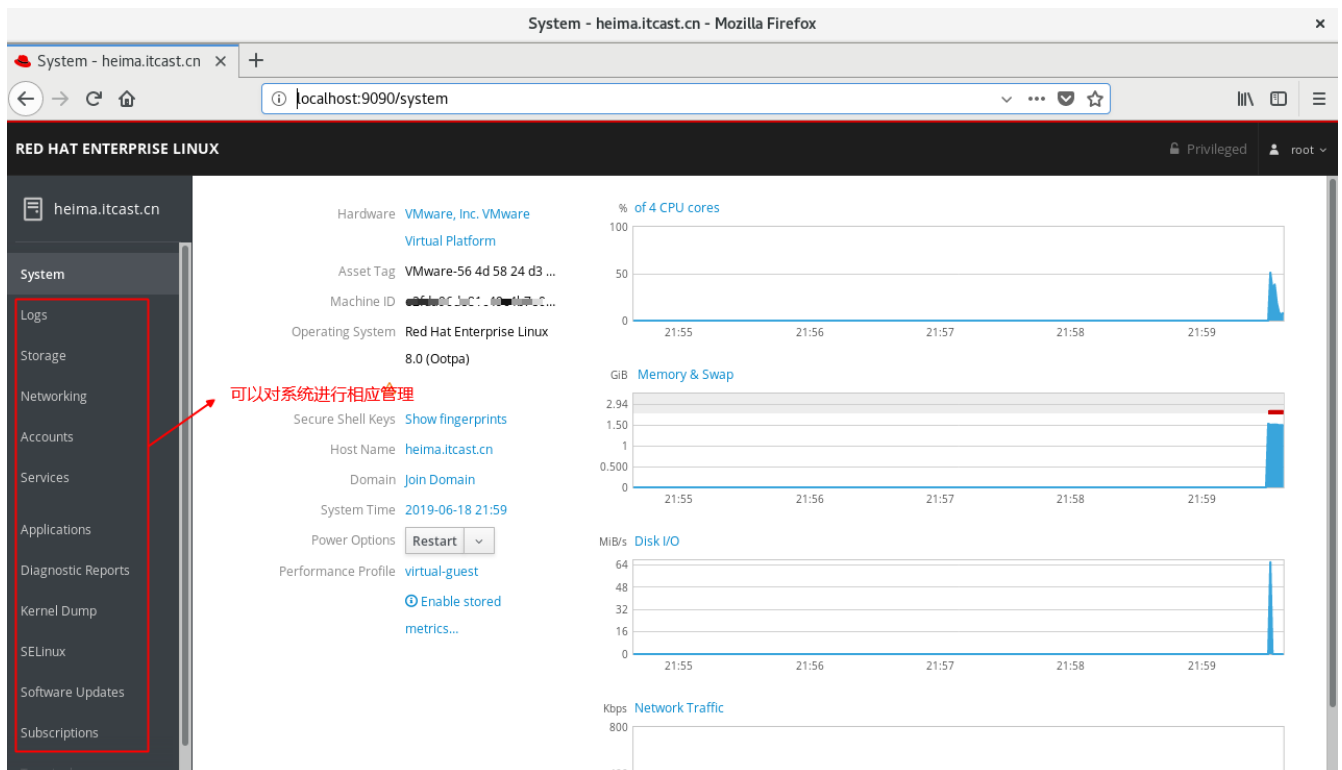
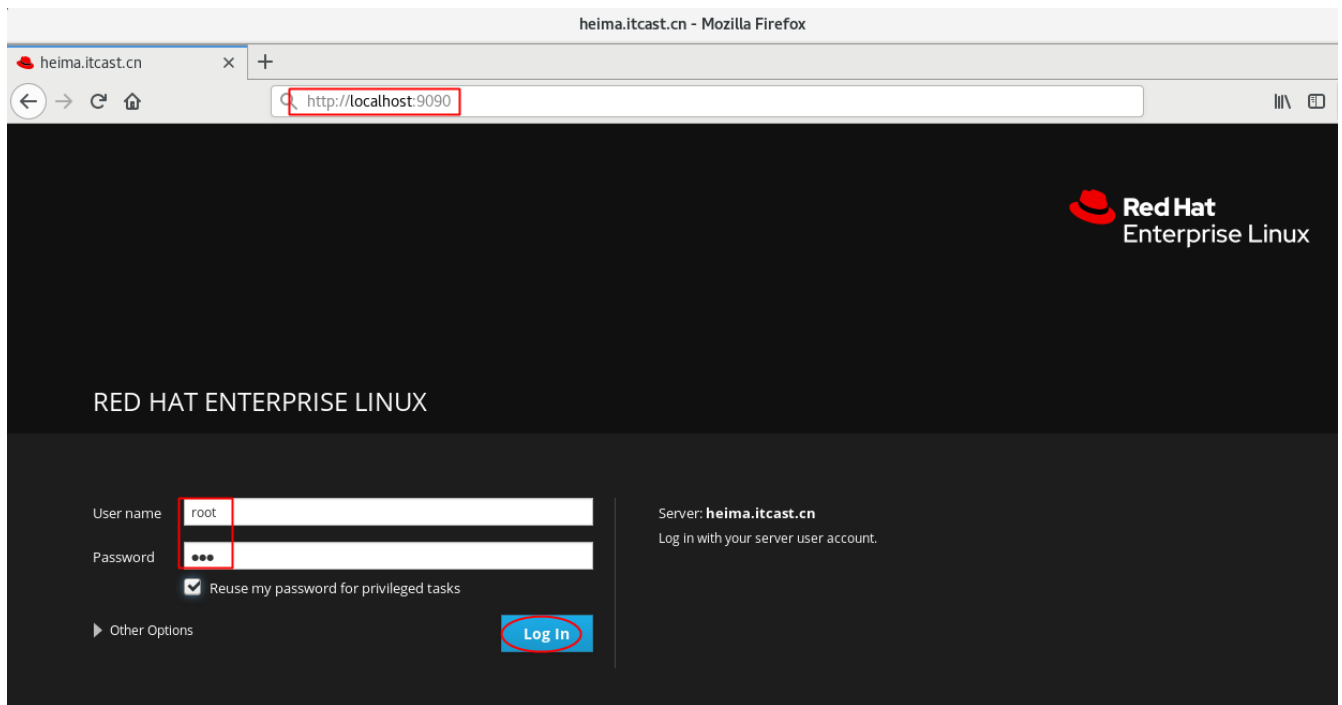
(一) 浏览器版本说明

- Mozilla Firefox 52及更高版本
- 谷歌Chrome 57及更高版本
- Microsoft Edge 16及更高版本

(二) 登录账号说明

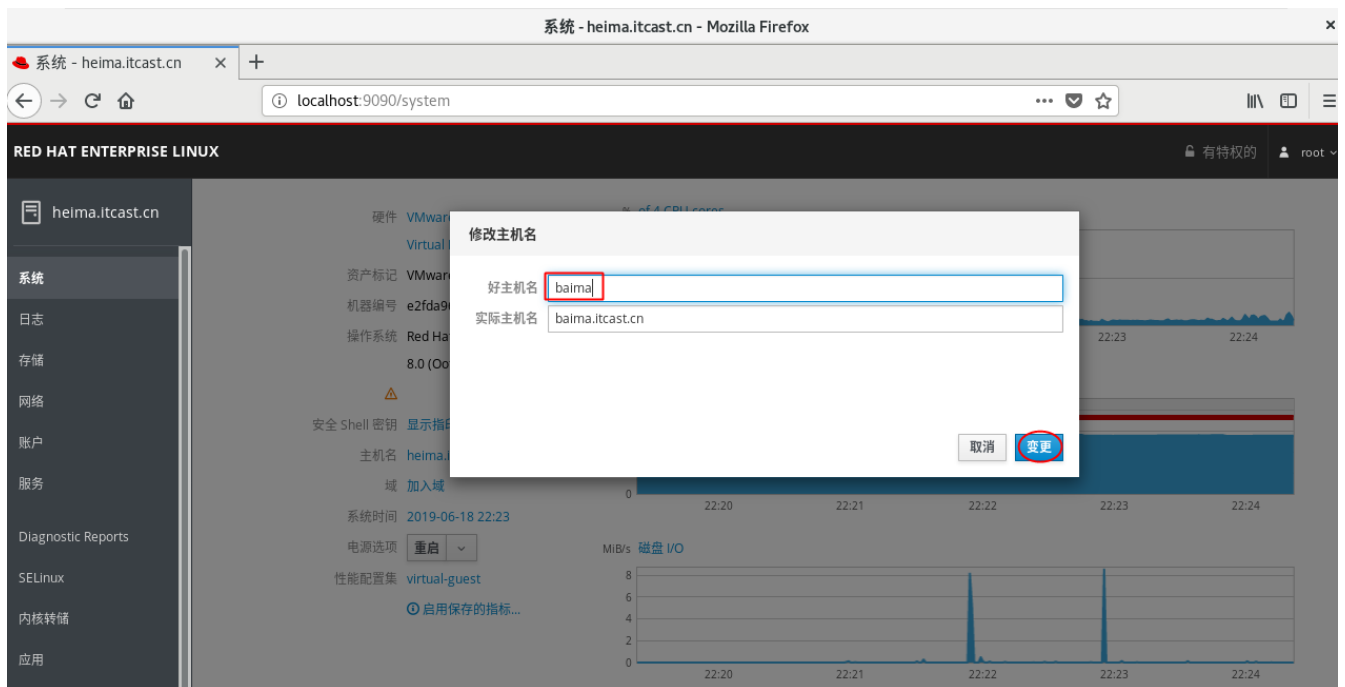
- web控制台登录账号认证文件位于 `/etc/pam.d/cockpit`
- 允许系统上任何本地帐户的用户名和密码登录

(三) 本地登录

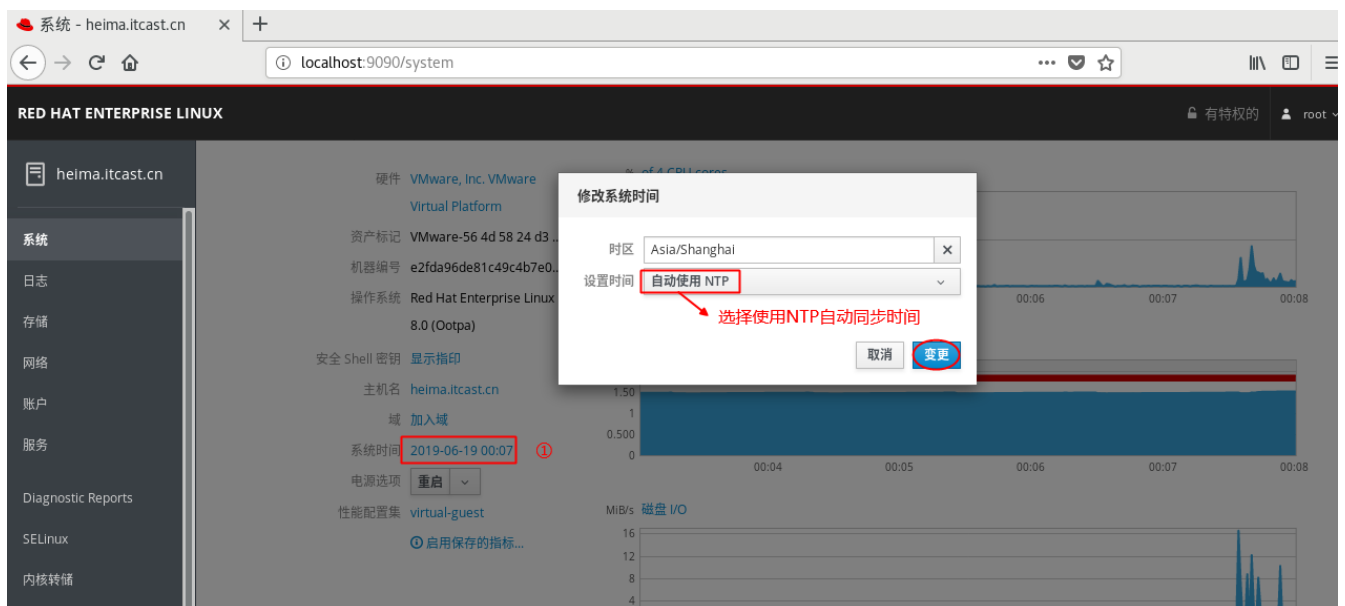


4、Web控制台对系统进行基本配置

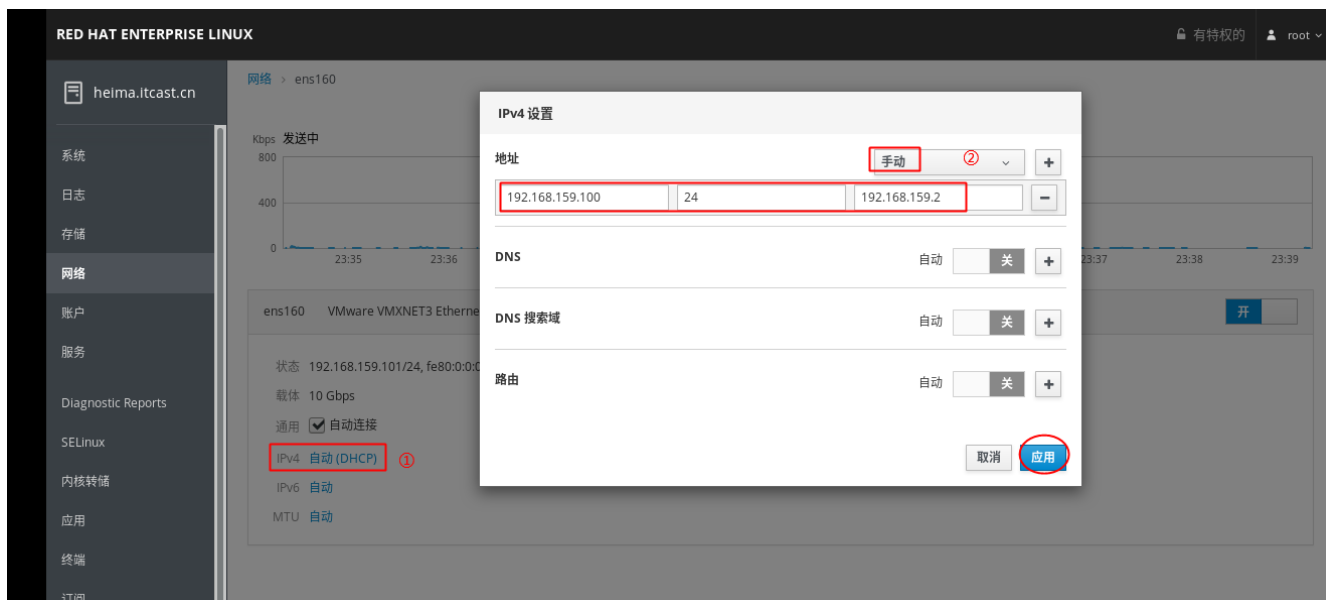
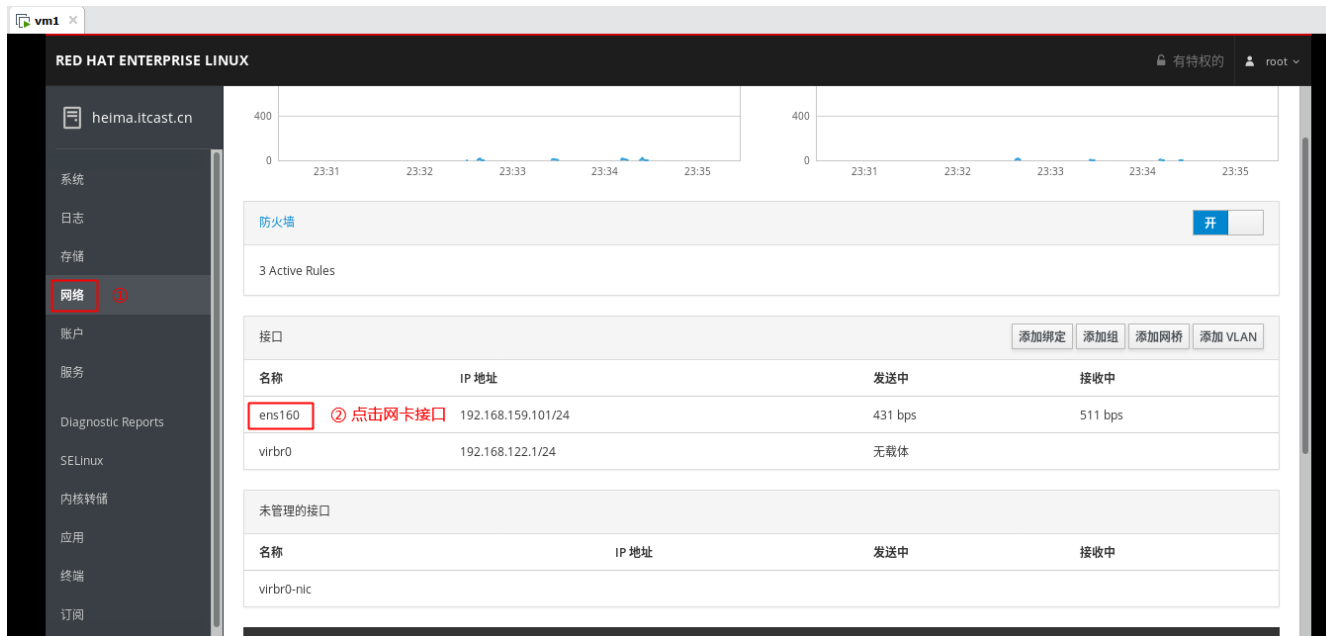
(-) 主机名配置



(二) 系统时间配置



(三) 网络配置



(四) 防火墙和SELinux配置

说明：初次接触Linux，由于是学习实验环境，建议先关闭防火墙和SELinux



二、徒手使用命令终端管理系统

1、主机名配置

```
[root@heima ~]# hostnamectl set-hostname RedHat8.itcast.cn
[root@heima ~]# cat /etc/hostname
RedHat8.itcast.cn
[root@RedHat8 ~]#
```

说明：

- 1) 通过命令hostnamectl修改会写到/etc/hostname文件，故也可以直接修改该文件
- 2) 退出重新登录立马生效，不需要重启系统

2、静态IP配置

(一) 了解Vmware三种网络模式

① 了解虚拟网络设备

VMnet0：用于虚拟桥接网络下的虚拟交换机

VMnet1：用于虚拟Host-Only网络下的虚拟交换机

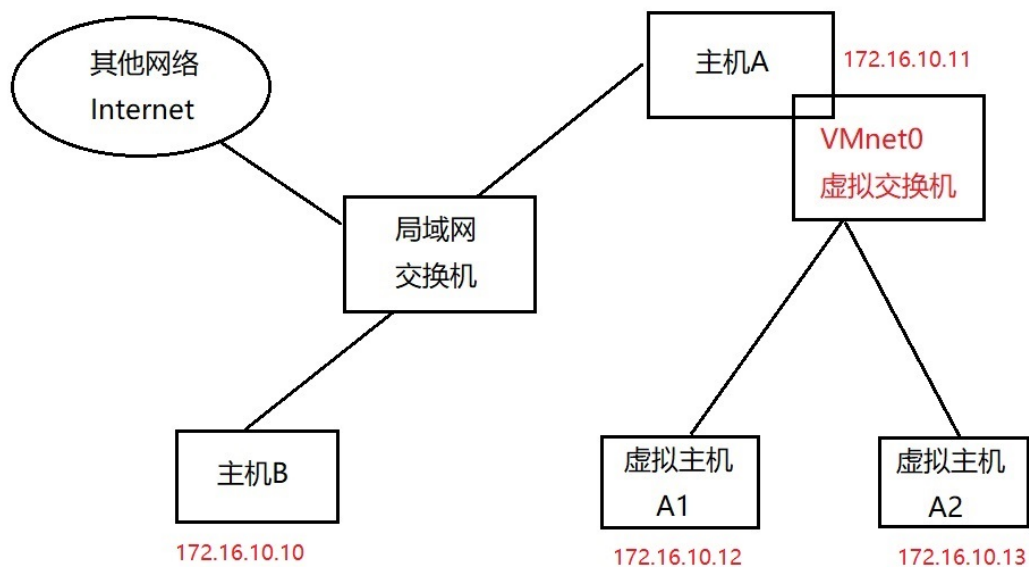
VMnet8：用于虚拟NAT网络下的虚拟交换机

VMware Network Adepter VMnet1：Host用于与Host-Only虚拟网络进行通信的虚拟网卡 VMware Network Adepter VMnet8：Host用于与NAT虚拟网络进行通信的虚拟网卡

② 了解三种网络模式

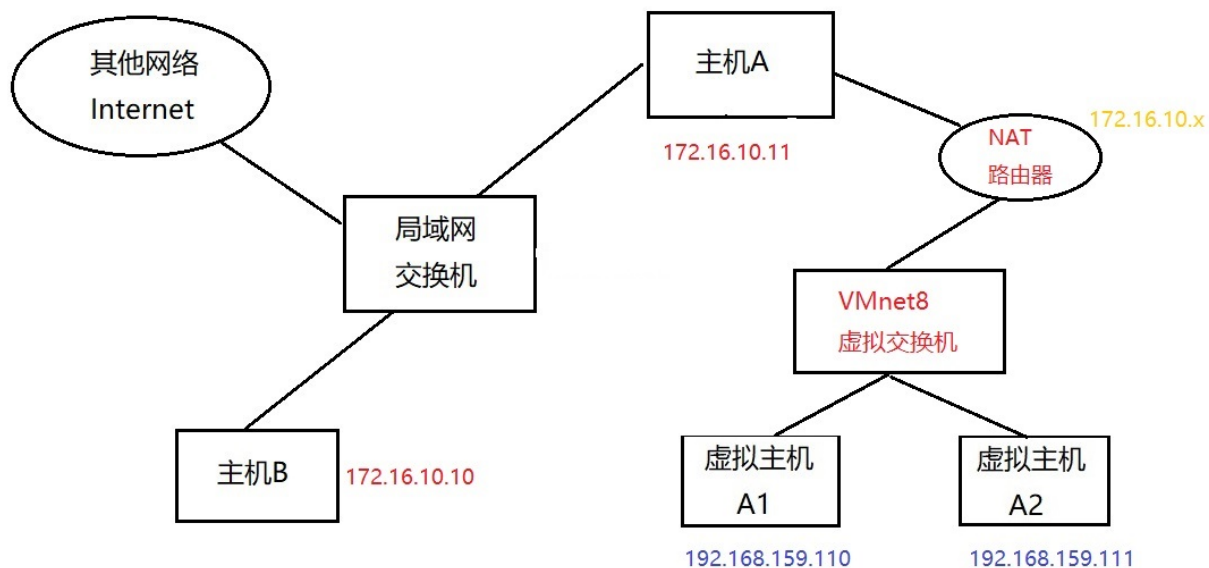
• 桥接网络

桥接网络是指虚拟网卡通过VMnet0虚拟交换机和本地物理网卡进行桥接，那么物理网卡和虚拟网卡就相当于处于同一个网段，虚拟交换机就相当于一台现实网络中的交换机。所以要想虚拟机也可以连接到互联网中，那么两个网卡的IP地址也要设置为同一网段。



• NAT网络

在NAT网络中，会用到VMware Network Adepter VMnet8虚拟网卡，主机上的VMware Network Adepter VMnet8虚拟网卡被直接连接到VMnet8虚拟交换机上与虚拟网卡进行通信。VMware Network Adepter VMnet8虚拟网卡的作用仅限于和VMnet8网段进行通信，它不给VMnet8网段提供路由功能，所以虚拟机虚拟一个NAT服务器，使虚拟网卡可以连接到Internet。VMware Network Adepter VMnet8虚拟网卡的IP地址是在安装VMware时由系统指定生成的，我们尽量不要修改这个数值，否则可能会使主机和虚拟机无法通信。

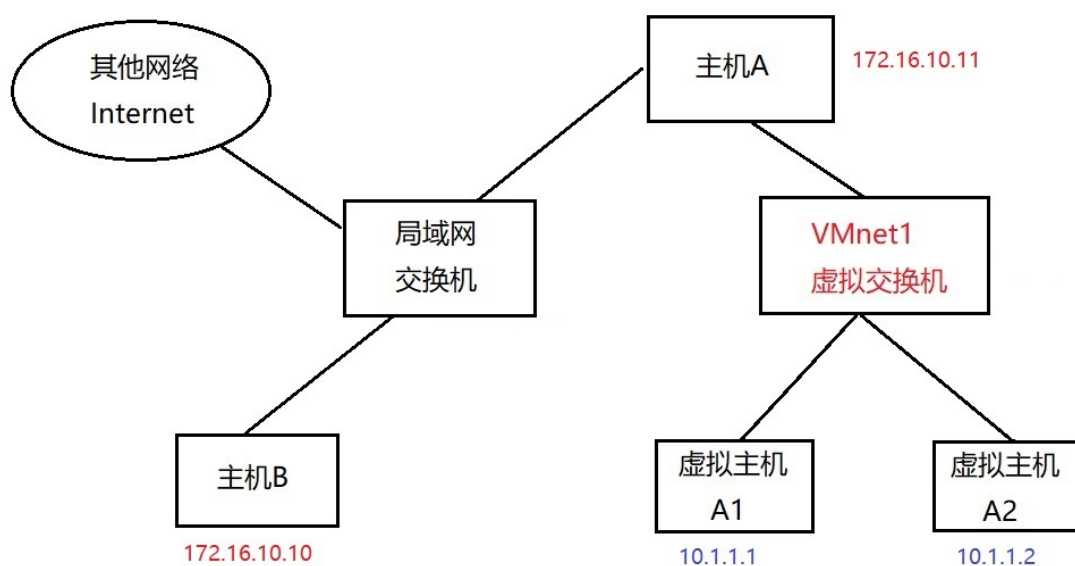


思考:

1. 虚拟主机A1是否可以访问物理真机主机B?
2. 虚拟主机A1是否可以访问物理真机主机 B 下的NAT模式下的虚拟机?

• 仅主机网络

在Host-Only模式下，虚拟网络是一个全封闭的网络，它唯一能够访问的就是物理真机。其实Host-Only网络和NAT网络很相似，不同的地方就是Host-Only网络没有NAT服务，所以虚拟网络不能连接到Internet。主机和虚拟机之间的通信是通过VMware Network Adapter VMnet1虚拟网卡来实现的。



总结:

1. VMware workstation带来哪些网络设备

- 三种网络模式下的虚拟交换机
- 两张虚拟网卡 (vmnet1和vmnet8) ——>作用: 用于物理主机和虚拟机通讯

2. 三种网络模式

- 桥接网络
 - 默认情况下可以访问互联网
 - 桥接网络的虚拟机IP地址和物理真机在同一个网段
- NAT网络
 - 默认情况下可以访问互联网
 - NAT网络的虚拟IP地址和物理真机不在同一个网段
 - 为什么NAT网络可以访问互联网? 因为NAT路由转换功能 (地址转换技术)
- 仅主机网络
 - 默认情况下不可以访问互联网
 - 仅主机模式下虚拟机IP地址和物理真机不在同一个网段

(二) 静态IP地址配置

Linux下一切皆文件!必然通过修改配置文件生效!

方法1: 直接修改网卡配置文件

- 配置静态IP地址

```
[root@heima ~]# cd /etc/sysconfig/network-scripts/
[root@heima network-scripts]# ls
ifcfg-ens160
[root@heima network-scripts]# cat ifcfg-ens160
TYPE=Ethernet                以太网
BOOTPROTO=none              IP获取方式, none和static表示静态, dhcp动态
NAME=ens160                  网卡名称
UUID=63b0b6ee-fbee-4b17-80be-e3b36ff27493 网卡UUID, 唯一标识
DEVICE=ens160                网卡设备名
ONBOOT=yes                   激活网卡
IPADDR=192.168.159.100        IP地址
PREFIX=24                     子网掩码
NETMASK=255.255.255.0
GATEWAY=192.168.159.2         网关
DNS1=8.8.8.8                  dns服务器
```

- 重载网卡配置文件

```
[root@heima network-scripts]# nmcli connection reload ens160
```

- 激活网卡连接

```
[root@heima network-scripts]# nmcli connection up ens160
```

方法2: 使用nmcli工具配置

- 查看网络连接情况

查看所有连接的网络信息

```
[root@heima ~]# nmcli connection show
```

NAME	UUID	TYPE	DEVICE
ens160	ea74cf24-c2a2-ecee-3747-a2d76d46f93b	ethernet	ens160
virbr0	e17e3c81-da25-455a-a8db-755ebdf36601	bridge	virbr0

查看已经激活的网络连接信息

```
[root@heima ~]# nmcli connection show --active
```

NAME	UUID	TYPE	DEVICE
ens160	ea74cf24-c2a2-ecee-3747-a2d76d46f93b	ethernet	ens160
virbr0	e17e3c81-da25-455a-a8db-755ebdf36601	bridge	virbr0

- 修改当前网卡IP地址

```
[root@heima ~]# nmcli connection modify ens160 ipv4.addresses 192.168.159.101/24  
ipv4.gateway 192.168.159.2 ipv4.dns 114.114.114.114
```

- 增加/删除IP地址（子接口）

```
[root@heima ~]# nmcli connection modify ens160 +ipv4.addresses 10.1.1.1/24  
[root@heima ~]# nmcli connection modify ens160 -ipv4.addresses 10.1.1.1/24
```

- 增加/删除DNS

```
[root@heima ~]# nmcli connection modify ens160 +ipv4.dns 8.8.8.8  
[root@heima ~]# nmcli connection modify ens160 -ipv4.dns 8.8.8.8
```

- 修改网络后需要重载配置文件并激活连接

```
[root@heima ~]# nmcli connection reload ens160  
[root@heima ~]# nmcli connection up ens160
```

方法3：使用nmtui文本图形工具

```
[root@RedHat8 ~]# nmtui
```

总结：

1. 推荐直接修改配置文件方式配置静态IP，一步到位
2. 从红帽8以后大家要熟悉使用nmcli工具管理网络，红帽7中的network.service即将被废弃

三、软件包管理

1、Linux系统中软件包分类

(一) 软件包类型

① 二进制包

- 什么是二进制包？有什么特点？

1. 二进制包，指的是已经¹好了的软件包，只需要直接安装就可以使用。
2. 二进制包，不需要编译，直接下载安装即可
3. 二进制包，需要根据自己的计算机CPU以及操作系统去选择合适的
4. 二进制包，命名方式一般为：xlockmore-5.31-2.el6.x86_64.rpm



② 源码包

- 什么是源码包？有什么特点？
 1. 源码包，指的是程序员写的原始的`程序代码文件`，不能够直接在计算机上运行。
 2. 源码包，需要进行编译，变成二进制的软件包后，才可安装使用
 3. 源码包，一般可以在任何的计算机上安装使用
 4. 源码包，命名方式一般为：
 - 软件包名.tar.gz
 - 软件包名.tar.bz2
 - 软件包名.tar.xz
 - 软件包名.zip

③ 二进制源码包(了解)

- 什么是二进制源码包？有什么特点？
 1. 二进制源码包，是一个半成品，安装后不能直接使用
 2. 二进制源码包，需要使用 `rpmbuild` 工具重建成真正的 `rpm` 包或者重建成源码包才可安装使用
 3. 二进制源码包，命名方式一般为：
 - `mysql-community-5.7.25-1.el6.src.rpm`
 - `mysql-community-5.7.25-1.el7.src.rpm`

(二) 常见的二进制包

系统平台	包类型	工具	在线安装
RedHat/Centos/Fedora/SUSE	rpm	rpm,rpmbuild	yum/dnf
Ubuntu/Debian	deb	dpkg	apt

(三) 总结二进制包和源码包区别

软件包类型	是否编译	安装难易程度	可定制性
二进制包	否	易(直接安装)	差
源码包	是	难(配置—>编译—>安装)	好

2、Linux系统中软件包安装方式

(一) 二进制包

① rpm工具安装

- 首先，需要下载好rpm包到本地
- 然后，直接使用rpm工具安装

② yum/dnf工具安装

- 首先，需要配置软件仓库(里面存放很多软件包，但不一定在本地)
- 然后，使用yum/dnf工具安装

(二) 源码包

① 根据需求配置

功能的定制

② 编译

使用编译器编译成二进制的软件包

③ 安装

将软件包安装到指定位置

④ 源码包安装优点

1. 可以在任意平台上编译安装，编译出来的软件包非常适应所在机器。
2. 可以在编译的时候，通过配置，对某些功能进行定制，开启或关闭相应的功能。

3、二进制rpm包如何管理(重点)

(一) 如何获取rpm包

Linux只是内核，Linux发行版本：GNU/Linux

1. RedHat/Centos光盘
2. 推荐网站
 - www.rpmfind.net
 - rpm.pbone.net
3. 相应软件官方网站
 - <http://www.mysql.com>
 - <http://nginx.org/packages/>

(二) 如何选择合适的rpm包

1. 选择适合当前系统的版本号

- 找不到适合的，才去尝试别的系统版本号
- el6兼容el5；el5无法安装 el6

2. 选择适合当前计算机cpu的架构

- x86_64包，只能安装在64位的系统上
- i386,i586,i686的软件包可以安装在32和64位系统上
- noarch表示这个软件包与硬件构架无关，可以通用
- 32位系统不能安装64位包

建议： 建议不要跨大版本号去安装软件包，尽量使用当前版本自带软件包安装

(三) 如何管理rpm包

1) rpm工具管理

① rpm工具安装rpm包

```
# rpm -ivh 软件包
```

注意：软件包的名字必须写全，xxx.rpm

② rpm工具卸载rpm包

```
# rpm -e 软件包名字
```

注意：卸载软件只需要跟软件包名字即可

③ rpm包的升级rpm包

```
# rpm -Uvh 软件包
```

或者

```
# rpm -Fvh 软件包
```

选项说明：

-v:输出详细信息

-h:打印散列标记，一般和-v一起使用

-U:升级软件包，如果该软件包没安装，会自动帮你安装

-F:升级软件包，如果该软件包没安装，不会自动帮你安装

④ 查看rpm包相关信息

查看已经安装的软件的文件列表

```
rpm -ql 软件包名
```

查看未安装的rpm包里的文件列表

```
rpm -qlp 软件包(xxx.rpm)
```

查看已经安装的所有rpm包

```
rpm -qa 软件包名
```

```
rpm -aq|grep 软件包名字
```

查看已经安装软件的文档列表

```
rpm -qd 软件包名
```

查看已经安装软件的配置文件

```
rpm -qc 软件包名
```

查看已经安装软件的详细信息

```
rpm -qi 软件包名
查看指定文件来自哪个rpm包
rpm -qf 文件名
```

⑤ rpm工具其他安装选项

```
--force          表示强制
rpm -ivh 软件包 --force          强制安装软件包
rpm -e 软件包名 --force          强制卸载软件包

--nodeps          忽略依赖关系
rpm -ivh 软件包 --nodeps 忽略依赖关系安装
rpm -e 软件包 --nodeps          忽略依赖关系卸载
```

其他了解：

```
rpm --import key_file          导入公钥用于检查rpm文件的签名
rpm --checksig package.rpm     检查rpm包的签名
```

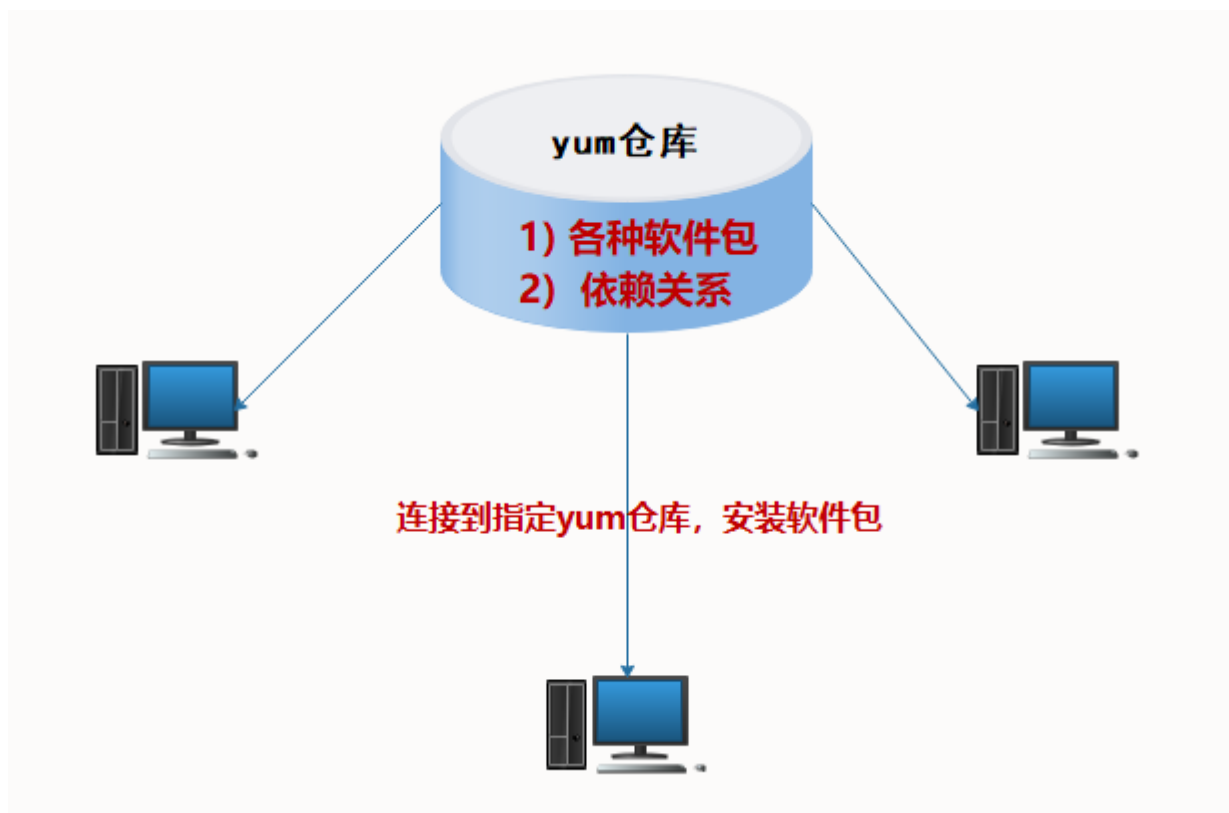
rpm工具管理rpm软件包

- 安装、卸载、升级、查看
- 缺点：有依赖关系需要我们人肉去解决 A---->B和C

2) yum (dnf) 工具管理

yum/dnf优点：能够自动解决依赖关系

核心：需要有一个**软件仓库**，软件仓库指的是来存放**软件包**和**软件包之间的依赖关系**地方。



1. 需要有软件仓库

- 仓库可以在本地——>本地yum源
 - 仓库可以在远程——>网络yum源——>网络必须ok
2. 需要告诉yum工具到哪个仓库里找
- 默认有一个地方，存放了xxx.repo文件——>定义了去哪个仓库里找

① 配置本地yum源

1) 本地仓库的分类

- BaseOS存储库

BaseOS存储库旨在提供一套核心的底层操作系统的功能，为**基础软件**安装库

- AppStream存储库

AppStream存储库中包括额外的**用户空间应用程序、运行时语言和数据库**，以支持不同的工作负载和用例。
AppStream中的内容有两种格式——熟悉的RPM格式和称为模块的RPM格式扩展

2) 配置本地仓库

步骤1：挂载镜像到本地系统

```
[root@RedHat8 ~]# mount -o ro /dev/sr0 /mnt
```

列出BaseOS和AppStream的内容如下说明仓库已准备好

```
[root@RedHat8 ~]# ls /mnt/BaseOS/
```

Packages repodata

```
[root@RedHat8 ~]# ls /mnt/AppStream/
```

Packages repodata

步骤2：修改配置文件指定本地存储库

```
[root@RedHat8 yum.repos.d]# pwd
```

/etc/yum.repos.d

```
[root@RedHat8 yum.repos.d]# cat local.repo
```

[BaseOS]

name=BaseOS

baseurl=file:///mnt/BaseOS

gpgcheck=0

enabled=1

[AppStream]

name=AppStream

baseurl=file:///mnt/AppStream

enabled=1

gpgcheck=0

步骤3：查看是否成功

清空yum缓存

```
[root@RedHat8 yum.repos.d]# yum clean all
```

创建yum缓存

```
[root@RedHat8 yum.repos.d]# yum makecache
```

查看仓库

```
[root@RedHat8 yum.repos.d]# yum repolist
```

仓库标识	仓库名称	状态
AppStream	AppStream	4,672
BaseOS	BaseOS	1,658

② yum (dnf) 工具使用

- 安装软件包

```
# yum -y install 软件包1 软件包2
```

```
# yum -y groupinstall "包组名"
```

注意:

- 1.其中, -y选项表示取消交互
- 2.包组里面包含很多的软件包。

- 卸载软件包

```
# yum -y remove 软件包名
```

```
# yum -y groupremove "包组名"
```

- 升级rpm包

```
# yum update 软件包名
```

任务解决方案

一、配置主机名

二、关闭防火墙和selinux

三、配置静态IP地址

四、配置本地软件仓库

五、安装vsftpd软件包

今日目标打卡

- ☒ 能够使用web控制台对服务器做基本配置
- ☒ 能够使用命令修改主机名
- ☒ 了解VMware三种网络模式
- ☒ 能够使用2种方法配置RedHat8静态IP地址
- ☒ 了解Linux系统中软件包的分类
- ☒ 能够配置RedHat8的本地仓库
- ☒ 能够使用yum（dnf）工具安装、卸载、升级软件包

1. 编译，就是通过编译工具，把高级语言变成计算机可以识别的2进制语言，计算机只认识1和0。编译程序，就是使用编译工具，把高级语言开发的程序变成计算机可以识别的二进制程[程](#)