

**【内部资料，请勿外传!!!!】****环境说明：****#### 真机 (RHEL8.2):**

```
server1.net0.example.com 172.25.0.254/24
# 预设 root 密码为 tedu
# 提供 RHEL8 软件源 http://server1.net0.example.com/rhel8/BaseOS
# 提供 RHEL8 软件源 http://server1.net0.example.com/rhel8/AppStream
# 提供 DNS 服务，为区域 net0.example.com 中相关站点提供解析
# 提供 NTP 网络时间服务；提供 NFS 文件服务，共享 /rhome/ldapuser0 目录
# 容器仓库位于 http://registry.lab.example.com，账户 admin，密码 redhat321
```

**#### 虚拟机 red (RHEL8.2):**

```
# 预设 root 密码为 redhat
```

**#### 虚拟机 blue (RHEL8.2):**

```
# 提供 2 块额外的磁盘 /dev/vdb、/dev/vdc，大小均为 10GB
# 预设大小为 200MiB 的逻辑卷 /dev/test/vo，格式化为 xfs 文件系统
```

**一、虚拟机 red****01. 配置网络地址**

虚拟机 red 的管理口令为 redhat，请为此虚拟机配置以下网络参数：

- 1) 主机名：red.net0.example.com
- 2) IP 地址：172.25.0.25/24
- 3) 默认网关：172.25.0.254
- 4) DNS 服务器：172.25.0.254

**解题参考：**

```
[root@red ~]# hostnamectl set-hostname red.net0.example.com //设置固定主机名
[root@red ~]# nmcli connection show //找出网卡对应的连接名
[root@red ~]# nmcli connection modify "网卡名" ipv4.method manual ipv4.addresses
172.25.0.25/24" ipv4.gateway 172.25.0.254 ipv4.dns 172.25.0.254
[root@red ~]# nmcli connection modify "网卡名" connection.autoconnect yes //配置地址，设置自启
[root@red ~]# nmcli connection up "网卡名" //激活连接
```

**02. 配置默认软件仓库**

YUM 软件仓库已可从以下地址使用：

<http://server1.net0.example.com/rhel8/BaseOS>

<http://server1.net0.example.com/rhel8/AppStream>

请配置你的虚拟机，以将这些位置作为默认软件源。

**解题参考：**

```
[root@red ~]# vim /etc/yum.repos.d/el8.repo
[BaseOS]
name = BaseOS
```

```
baseurl = http://server1.net0.example.com/rhel8/BaseOS
gpgcheck = 0

[AppStream]
name = AppStream
baseurl = http://server1.net0.example.com/rhel8/AppStream
gpgcheck = 0

[root@red ~]# yum repolist //确认仓库列表
[root@red ~]# yum -y install net-tools bind-utils //安装常用工具，顺便测试源
```

### 03. 调试 SELinux

配置 httpd 在 82 端口上提供 Web 服务，满足以下要求：

- 1) 此 Web 服务器能够提供/var/www/html/中所有现有的 HTML 文件
- 2) 此 Web 服务器在系统启动时自动启动
- 3) 确保 SELinux 保护机制运行在 Enforcing 模式

解题参考：

```
[root@red ~]# yum -y install setroubleshoot-server //安装排错包

[root@red ~]# systemctl restart httpd //SELinux 试错操作
[root@red ~]# journalctl -xe //查看日志，根据提示排错
```

比如：

```
[root@red ~]# semanage port -a -t http_port_t -p tcp 82 //开放非标准 Web 端口

[root@red ~]# systemctl restart httpd //确保服务启动
[root@red ~]# rm /etc/httpd/conf.d/welcome.conf //删除欢迎配置，允许索引

[root@red ~]# systemctl enable httpd //设置 httpd 服务开机自启
[root@red ~]# systemctl disable firewalld --now //禁用防火墙服务
```

### 04. 配置用户账户

创建用户 tammy，其用户 ID 为 2020，此用户的密码应当是 ilovelinux。

解题参考：

```
[root@red ~]# useradd -u 2020 tammy
[root@red ~]# echo ilovelinux | passwd --stdin tammy
```

### 05. 创建用户账户

根据下列要求创建用户及组账号：

- 1) 名为 admins 的组
- 2) 用户 zhsan，其附属组为 admins 3)
- 用户 lisi，其附属组还属于 admins
- 4) 用户 wangwu，没有可交互的登录 Shell，且不属于 admins 组
- 5) zhsan、lisi、wangwu 的密码都应该是 ilovelinux

解题参考：

```
[root@red ~]# groupadd admins

[root@red ~]# useradd -G admins zhsan
[root@red ~]# useradd -G admins lisi
[root@red ~]# useradd -s /sbin/nologin wangwu

[root@red ~]# echo ilovelinux | passwd --stdin zhsan
[root@red ~]# echo ilovelinux | passwd --stdin lisi
[root@red ~]# echo ilovelinux | passwd --stdin wangwu
```

## 06. 配置 cron 计划任务

配置计划任务，以用户 zhsan 的身份每 5 分钟执行一次命令 logger "RH200 Test"。

解题参考：

```
[root@red ~]# systemctl enable crond           //确保 crond 开机自动运行
[root@red ~]# systemctl restart crond          //确保 crond 服务已启动

[root@red ~]# crontab -e -u zhsan              //添加计划任务
*/5 * * * * logger "RH200 Test"
```

## 07. 配置文件权限

将文件/etc/hosts 复制为/var/tmp/hosts，并按要求配置/var/tmp/hosts 的权限：

- 1) 文件/var/tmp/hosts 属于 root 用户
- 2) 文件/var/tmp/hosts 属于 root 组
- 3) 任何用户对/var/tmp/hosts 都没有可执行权限
- 4) 用户 zhsan 能够读取和写入/var/tmp/hosts 文件
- 5) 用户 lisi 无法读取或写入/var/tmp/hosts 文件
- 6) 所有其他用户（当前或未来）能够读取/var/tmp/hosts 文件

解题参考：

```
[root@red ~]# cp /etc/hosts /var/tmp/hosts      //复制文件
[root@red ~]# setfacl -m u:zhsan:rw /var/tmp/hosts //设置白名单用户权限
[root@red ~]# setfacl -m u:lisi:- /var/tmp/hosts //设置黑名单用户权限
```

## 08. 创建共用目录

创建具有以下特点的共用目录：

- 1) /home/tools 的组的所有权是 admins
- 2) 此目录能被 admins 组的成员读取、写入和访问，除 root 外其他用户没有这些权限
- 3) 在此目录下创建的文件，其组的所有权自动设置为 admins 组

解题参考：

```
[root@red ~]# mkdir /home/tools                 //创建目录
[root@red ~]# chown :admins /home/tools          //更改属组
[root@red ~]# chmod g+rw, o-rwx /home/tools     //设置权限

[root@red ~]# chmod g+s /home/tools              //添加 SetGID 附加权限
```

## 09. 配置 NTP 时间客户端

配置你的系统，使其成为 server1.net0.example.com 的 NTP 客户端。

解题参考：

```
[root@red ~]# vim /etc/chrony.conf
server server1.net0.example.com iburst           //设置为指定的 NTP 服务器

[root@red ~]# systemctl restart chronyd          //重启 chronyd 服务
[root@red ~]# systemctl enable chronyd           //设置开机自启
[root@red ~]# chronyc sources -v                 //检查当前 NTP 源(^*表示已同步过)
```

## 10. 配置 autofs

按照以下要求，通过 autofs 自动挂载远程用户的主目录：1) server1.net0.example.com(172.25.0.254)通过 NFS 共享目录/rhome 到你的系统，此文件系统中包含为用户 ldapuser0 预配置的主目录  
2) 预设用户 ldapuser0 的登录密码是 password  
3) 预设用户 ldapuser0 的主目录是 server1.net0.example.com:/rhome/ldapuser0  
4) 预设用户 ldapuser0 的主目录应自动挂载到本地的/rhome/ldapuser0 目录5) 挂载后的主目录可供用户 ldapuser0 写入

解题参考：

```
[root@red ~]# yum -y install autofs nfs-utils      //装包
[root@red ~]# vim /etc/auto.master                //设置监控点（主目录的上一层）
/rhome /etc/rhome.rule                          //由 rhome.rule 文件给出具体挂载策略

[root@red ~]# vim /etc/rhome.rule                 //配置挂载策略
ldapuser0 -fstype=nfs 172.25.0.254:/rhome/ldapuser0

[root@red ~]# systemctl restart autofs            //重启 autofs 服务
[root@red ~]# systemctl enable autofs             //设置开机自启

[root@red ~]# ls /rhome/ldapuser0                 //访问目录以触发自动挂载
ls: cannot open directory '/rhome/ldapuser0': Permission denied
[root@red ~]# ls /rhome/                          //确认结果
ldapuser0
```

## 11. 查找文件

找出/etc/目录下大小超过 5MB 的文件，并将其副本放入/root/findfiles 目录。

解题参考：

```
[root@red ~]# mkdir /root/findfiles
[root@red ~]# find /etc -type f -size +5M -exec cp -p {} /root/findfiles \;
```

## 12. 查找字符串

找出文件/etc/man\_db.conf 中包含字符串 sbin 的所有行，将其副本按原始顺序存放到文件 /root/out.txt 中。文件/root/out.txt 中不得包含空行，且所有行必须是/etc/man\_db.conf 中原始行的准确副本。

### 解题参考：

```
[root@red ~]# grep sbin /etc/man_db.conf > /root/out.txt
```

### 13. 创建归档

创建一个名为 /root/backup.tar.bz2 的 tar 归档，其应该包含 /usr/local/ 的内容。此归档文件必须使用 bzip2 进行压缩。

### 解题参考：

```
[root@red ~]# yum -y install tar bzip2
[root@red ~]# tar -jcpf /root/backup.tar.bz2 /usr/local
```

### 14. 配置容器服务

利用仓库服务器（注册表服务器）上面的 nginx 镜像，创建一个名为 logserver 的容器。

1) 将其配置为以 systemd 服务的形式运行，仅面向用户 tammy 2)

该服务应命名为 container-logserver，并将其设置为开机自动启动

### 解题参考：

```
[root@red ~]# yum module install -y container-tools //安装容器模块及配置
[root@red ~]# vi /etc/containers/registries.conf
30 [registries.search]
31 registries = ['registry.lab.example.com']
42 [registries.insecure]
43 registries = ['registry.lab.example.com']
PS: 找到对应的行数，进行修改
```

以指定用户 tammy 直接 SSH 登录（不要 su 或 sudo）：

### ==> 配置容器环境、下载镜像

```
[root@server1 ~]# ssh tammy@red
Password: //验证 tammy 的密码
...
[tammy@red ~]$ mkdir -p ~/.config/containers //建立容器配置目录
[tammy@red ~]$ vim ~/.config/containers/registries.conf //建立容器仓库配置
unqualified-search-registries = ["registry.lab.example.com"] //默认的仓库搜索地址
[[registry]]
insecure = true //不检查 TLS 证书
blocked = false //允许访问
location = "registry.lab.example.com" //仓库地址
//上述配置可通过 man containers-registries.conf 获得
[tammy@red ~]$ podman login registry.lab.example.com //登录容器仓库
Username: admin //输入仓库账号
Password: ***** //输入仓库密码

[tammy@red ~]$ podman search nginx //搜索镜像
...
[tammy@red ~]$ podman pull registry.lab.example.com/library/nginx //下载镜像
```

### ==> 启动容器，并设置为用户服务

```
[tammy@red ~]$ mkdir /home/tammy/container_logserver //创建资源目录
[tammy@red ~]$ podman run --name logserver -d -p 8080:80 -v
/home/tammy/container_logserver:/usr/share/nginx/html:Z nginx
//启动容器（运行参数、目录位置及映射位置等参见关联题目的要求）
[tammy@red ~]$ mkdir -p ~/.config/systemd/user //创建用户服务目录
[tammy@red ~]$ cd ~/.config/systemd/user
[tammy@red user]$ podman generate systemd --name logserver --files //生成服务配置
[tammy@red user]$ systemctl --user daemon-reload //更新用户服务配置
[tammy@red user]$ podman stop logserver //停用当前手动开启的容器

[tammy@red user]$ systemctl --user enable container-logserver --now //配置容器自启动
[tammy@red user]$ loginctl linger=yes //为未登录用户保持服务

[tammy@red user]$ crontab -e //设置开机自启动的用户任务
@reboot systemctl --user start container-logserver.service
```

## 15. 为容器配置持久存储

扩展上一个任务的 logserver 容器服务：

- 1) 配置主机的系统日志，以在系统重启后能保留其数据
- 2) 将主机中/var/log/journal/目录及任何子目录中包含的 \*.journal 文件复制到目录/home/tammy/container\_logserver 下
- 3) 将服务配置为启动时自动将主机中的/home/tammy/container\_logserver 目录挂载到容器中的/usr/share/nginx/html/目录

### 解题参考：

```
[root@red ~]# mkdir -p /var/log/journal //确认日志目录
[root@red ~]# systemd-tmpfiles --create --prefix /var/log/journal/ //初始化日志目录
[root@red ~]# systemctl restart systemd-journald //重启日志服务
[root@red ~]# find /var/log/journal -name "*.journal" -exec cp {}
/home/tammy/container_logserver/ \; //复制日志文件

[root@red ~]# chown tammy:tammy /home/tammy/container_logserver/*.journal
//调整文件归属（若复制文件后容器无法重启，可以补做此项）
```



## 二、虚拟机 blue

### 16. 设置 root 密码

获得系统 blue 的管理权限，并将 blue 的 root 密码设置为 redhat。

解题参考：

```
//重启 blue 系统，按 e 键打断启动过程
//修改 linux 行 (ro 改 rw, 末尾添加 rd.break) 然后按 ctrl+x 启动
switch_root:/# chroot /sysroot/ //切换到根系统
sh-4.2# echo redhat | passwd --stdin root //修改 root 口令为指定的字符串
sh-4.2# touch /.autorelabel //标记下一次启动重做 SELinux 标记
sh-4.2# exit //退出恢复模式
switch_root:/# reboot //重启系统
```

### 17. 配置网络地址

为此虚拟机配置以下网络参数： 1)

主机名：blue.net0.example.com 2)

IP 地址：172.25.0.26/24

3) 默认网关：172.25.0.254

4) DNS 服务器：172.25.0.254

解题参考：

```
[root@blue ~]# hostnamectl set-hostname blue.net0.example.com //设置固定主机名
[root@blue ~]# nmcli connection show //找出网卡对应的连接名
[root@blue ~]# nmcli connection modify "网卡名" ipv4.method manual ipv4.addresses
172.25.0.26/24" ipv4.gateway 172.25.0.254 ipv4.dns 172.25.0.254
[root@blue ~]# nmcli connection modify "网卡名" connection.autoconnect yes //配置地址，设置自启
[root@blue ~]# nmcli connection up "网卡名" //激活连接
```



## 18. 配置默认软件仓库

YUM 软件仓库已可从以下地址使用：

<http://server1.net0.example.com/rhel8/BaseOS>

<http://server1.net0.example.com/rhel8/AppStream>

请配置你的虚拟机，以将这些位置作为默认软件源。

**解题参考：**

```
[root@blue ~]# vi /etc/yum.repos.d/el8.repo
[BaseOS]
name = BaseOS
baseurl = http://server1.net0.example.com/rhel8/BaseOS
gpgcheck = 0

[AppStream]
name = AppStream
baseurl = http://server1.net0.example.com/rhel8/AppStream
gpgcheck = 0

[root@blue ~]# dnf repolist //确认仓库列表
[root@blue ~]# dnf -y install vim net-tools bind-utils //安装常用工具，顺便测试源
```

## 19. 调整逻辑卷大小

将逻辑卷 vo 及其文件系统大小调整到 300MiB。确保文件系统内容保持不变。

**解题参考：**

```
[root@blue ~]# lvscan //找出要扩展的逻辑卷
[root@blue ~]# lvextend -L 300MiB /dev/test/vo //扩展逻辑卷
[root@blue ~]# blkid /dev/test/vo //检查文件系统格式
[root@blue ~]# xfs_growfs 逻辑卷对应的挂载点 //适用于 XFS 文件系统
或者
[root@blue ~]# resize2fs 逻辑卷对应的挂载点 //适用于 EXT2/3/4 文件系统
```

## 20. 添加交换分区

为你的系统额外添加一个 512MiB 的交换分区，此交换分区应在系统启动时自动挂载，不要删除或以任何方式改动系统上原有的交换分区。

**解题参考：**

```
[root@blue ~]# fdisk /dev/vdb //修改磁盘 vdb
```



```
... ..
Command (m for help): n //添加新分区
Partition number (2-128, default 2): //直接回车（默认）
First sector (4194304-20971486, default 4194304): //直接回车（默认）
Last sector, *sectors or +size{K,M,G,T,P} (4194304-20971486, default 20971486): +512M
Created a new partition 2 of type 'Linux filesystem' and of size 512 MiB.
Command (m for help): w //保存分区表，并退出
The partition table has been altered.
Syncing disks.
[root@blue ~]# partprobe /dev/vdb //刷新分区表

[root@blue ~]# mkswap /dev/vdb2 //格式化自建分区 vdb2
[root@blue ~]# vim /etc/fstab

/dev/vdb2 swap swap defaults 0 0

[root@blue ~]# swapon -a //启用 fstab 中的交换设备
[root@blue ~]# swapon -s //查看交换分区信息
```

## 21. 创建逻辑卷

根据以下要求，创建新的逻辑卷：

- 1) 逻辑卷的名字为 mylv，属于 myvg 卷组，大小为 50 个扩展单元
- 2) 卷组 myvg 中的逻辑卷的扩展块大小应当为 16MiB
- 3) 使用 vfat 文件系统将逻辑卷 mylv 格式化
- 4) 此逻辑卷应当在系统启动时自动挂载到 /mnt/mydata 目录下

解题参考：

```
[root@blue ~]# fdisk /dev/vdb //修改磁盘 vdb
... ..
Command (m for help): n //添加新分区
Partition number (3-128, default 3): //直接回车（默认）
First sector (5242880-20971486, default 5242880): //直接回车（默认）
Last sector, *sectors or +size{K,M,G,T,P} (5242880-20971486, default 20971486): +1000M
Created a new partition 3 of type 'Linux filesystem' and of size 1000 MiB.
Command (m for help): w //保存分区表，并退出
The partition table has been altered.
Syncing disks.
[root@blue ~]# partprobe /dev/vdb //刷新分区表

[root@blue ~]# vgcreate -s 16MiB myvg /dev/vdb3 //建卷组（使用分区 vdb3）
[root@blue ~]# lvcreate -l 50 -n mylv myvg //建逻辑卷

[root@blue ~]# mkfs.vfat /dev/myvg/mylv //格式化
[root@blue ~]# mkdir /mnt/mydata //创建挂载点目录
[root@blue ~]# vim /etc/fstab //设置开机挂载

/dev/myvg/mylv /mnt/mydata vfat defaults 0 0
[root@blue ~]# mount -a //启用&测试开机挂载
```

## 22. 创建 VDO 卷

根据如下要求，创建新的 VDO 卷：

- 1) 使用未分区的磁盘 (/dev/vdc)



- 2) 此 VDO 卷的名称为 myvdo 3)
- 此 VDO 卷的逻辑大小为 50G
- 4) 此 VDO 卷使用 xfs 文件系统格式化
- 5) 此 VDO 卷在系统启动时自动挂载到/vblock 目录下

#### 解题参考：

```
[root@blue ~]# yum install vdo //装包
[root@blue ~]# systemctl enable --now vdo //起服务
[root@blue ~]# vdo create --name=myvdo --device=/dev/vdc --vdoLogicalSize=50G //新建 VDO 卷
[root@blue ~]# mkfs.xfs -K /dev/mapper/myvdo //格式化
//或者 mkfs.ext4 -E nodiscard /dev/mapper/myvdo
[root@blue ~]# mkdir /vblock //创建挂载点目录

[root@blue ~]# vim /etc/fstab
...
/dev/mapper/myvdo /vblock xfs _netdev 0 0
[root@blue ~]# mount -a //启用&测试开机挂载
[root@blue ~]# reboot //如果重启系统失败，需验证管理密码，并修复 fstab 错误后再重启
```

## 23. 配置系统调优

为你的系统选择建议的 tuned 配置集并将它设为默认设置。

#### 解题参考：

```
[root@blue ~]# yum -y install tuned //装包（如果没装的话）
[root@blue ~]# systemctl restart tuned //起服务
[root@blue ~]# systemctl enable tuned //设置开机自启

[root@blue ~]# tuned-adm recommend //查看推荐方案（比如 virtual-guest）
[root@blue ~]# tuned-adm profile virtual-guest //切换为指定优化方案
[root@blue ~]# tuned-adm active //确认当前活动方案
```