秋明分享平台之

RHCE-el7-EX300真题详解

撰稿人：秋明

撰稿人邮箱：tyumen@zhoufengjie.cn

QQ群：90122290

本文url： https://www.zhoufengjie.cn/?p=283

本文撰稿日期： 20170725

本站口号：开源、分享、共进

注解：任何想通过本平台分享文档的，可以随时联系，文档里面注明出稿人和邮箱，方便文档使用者同撰稿的兄弟技术咨询沟通和交流；

目录

[1 开考说明 3](#_Toc488874245)

[2 考试环境说明 3](#_Toc488874246)

[3 考试了解 3](#_Toc488874247)

[4 考试题目 4](#_Toc488874248)

[4.1 配置 SELinux 4](#_Toc488874249)

[4.2 配置 SSH访问 4](#_Toc488874250)

[4.3 自定义用户环境(别名设置) 5](#_Toc488874251)

[4.4 配置防火墙端口转发 5](#_Toc488874252)

[4.5 配置链路聚合 6](#_Toc488874253)

[4.6 配置 IPv6 地址 7](#_Toc488874254)

[4.7 配置本地邮件服务 8](#_Toc488874255)

[4.8 通过 Samba 发布共享目录 9](#_Toc488874256)

[4.9 配置多用户 Samba 挂载 11](#_Toc488874257)

[4.10 配置 NFS 共享服务 13](#_Toc488874258)

[4.11 挂载 NFS共享 14](#_Toc488874259)

[4.12 实现一个 web 服务器 15](#_Toc488874260)

[4.13 实现一个 web 服务器 16](#_Toc488874261)

[4.14 配置虚拟主机 18](#_Toc488874262)

[4.15 配置 web 内容的访问 19](#_Toc488874263)

[4.16 实现动态 WEB 内容 20](#_Toc488874264)

[4.17 创建一个脚本 21](#_Toc488874265)

[4.18 创建一个添加用户的脚本 23](#_Toc488874266)

[4.19 配置 iSCSI 服务端 24](#_Toc488874267)

[4.20 配置 iSCSI 客户端 26](#_Toc488874268)

[4.21 配置一个数据库 27](#_Toc488874269)

[4.22 数据库查询(填空) 29](#_Toc488874270)

# 开考说明

开始考试的时候，会需要先填写基本的信息：

第一页：

姓名：这个RHCSA、RHCE的考试，姓名一定要填写一致，有空格的时候都要有空格；

邮箱：用来收成绩的，现在只有电子证书了，邮箱错了收不到成绩

地址：随便写

补考信息：第一次考试，不用写；

第二页和第三页选择，下一步就ok了；

大家一定不能把样例作为考题，考题是一定会在不断变化的，只是变化大小而已；所以RHCSA和RHCE虽然简单，不过大家在做练习的时候还是一定要理解题目；

另：个人发现在考试的时候，考官会随机的抽取参与考试的学员进行挖坑，比如：在RHCE上把yum的repo文件给一个隐藏权限不可改、默认已经给安装好的软件配置文件不知名的行改错配置默认起不来等；

# 考试环境说明

真实机(无 root 权限):foundation.groupX.example.com

虚拟机 1(有 root 权限):system1.groupX.example.com

虚拟机 2(有 root 权限):system2.groupX.example.com

考试服务器(提供 DNS/YUM/认证/素材.. ..):

server1.groupX.example.com、host.groupX.example.com

# 考试了解

这里文字很多，大概的意思就是告诉你两台虚拟机的ip地址信息，以及一些安全限制，用户密码，域等信息。

# 考试题目

下面的内容是考试题目解答，注：一定要理解考题，考试的时候题目可能会有一些变化，比如路径、名称、字符串、版本号等；

* 1. 配置 SELinux

试题概述:

确保 SELinux 处于强制启用模式。

解题参考:

[root@system1 ~]# vim /etc/selinux/config

SELINUX=enforcing

[root@system1 ~]# setenforce 1

[root@system1 ~]# getenforce

Enforcing

* 1. 配置 SSH访问

试题概述:

按以下要求配置 SSH 访问:

 用户能够从域 groupX.example.com 内的客户端 SSH 远程访问您的两个虚拟机系统

 在域 my133t.org 内的客户端不能访问您的两个虚拟机系统

解题参考:

[root@system1 ~]# vim /etc/ssh/sshd\_config

.. ..

DenyUsers \*@\*.my133t.org \*@172.34.0.\* //此题也可由防火墙解决，或者tcp\_wrapper（hosts.allow/hosts.deny）解决

[root@system1 ~]# systemctl restart sshd

* 1. 自定义用户环境(别名设置)

试题概述:

在系统 system1 和 system2 上创建自定义命令为 qstat,此自定义命令将执行以下命令: /bin/ps -Ao pid,tt,user,fname,rsz

此命令对系统中所有用户有效。

解题参考:

[root@system1 ~]# vim /etc/bashrc

.. ..

alias qstat='/bin/ps -Ao pid,tt,user,fname,rsz'

[root@system1 ~]# source /etc/bashrc //或重登录后生效

[root@system1 ~]# qstat //确认别名可用

此题也可以编辑vi /bin/qstat输入内容，然后chmod +x /bin/qstat

* 1. 配置防火墙端口转发

试题概述:

在系统 system1 配置端口转发,要求如下:

 在 172.25.0.0/24 网络中的系统,访问 system1 的本地端口 5423 将被转发到 80

 此设置必须永久有效

解题参考:

[root@system1 ~]# systemctl restart firewalld

[root@system1 ~]# systemctl enable firewalld

//调整防火墙信任区域,简化对后续各种服务的防护

[root@system1 ~]# firewall-cmd --set-default-zone=trusted //将默认区域设置为信任

[root@system1 ~]# firewall-cmd --permanent --add-source=172.34.0.0/24 --zone=block

//阻止未授权网络 my133t.org，此要求是拒绝访问的要求一起在这做了配置

[root@system1 ~]# firewall-cmd --permanent --zone=trusted --add-forward-

port=port=5423:proto=tcp:toport=80

[root@system1 ~]# firewall-cmd --reload

* 1. 配置链路聚合

试题概述:

在 system1.groupX.example.com 和 system2.groupX.example.com 之间按以下要求配置

一个链路:

 此链路使用接口 eth1 和 eth2

 此链路在一个接口失效时仍然能工作;

 此链路在 system1 使用下面的地址 172.16.X.20/255.255.255.0  此链路在 system2 使用下面的地址 172.16.X.25/255.255.255.0  此链路在系统重启之后依然保持正常状态

解题参考【第一台，第二台做法一样】:

[root@system1~]# nmcli connection add con-name team0 type team ifname team0 config '{ "runner":{"name":"activebackup" } }' //建立新的聚合连接

 [root@system1~]# nmcli connection add con-name team0-eth1 type team-slave ifname eth1 master team0 //指定成员网卡 eth1

[root@system1~]# nmcli connection add con-name team0-eth2 type team-slave ifname eth2 master team0 //指定成员网卡 eth2

[root@system1~]# nmcli con modify team0 ipv4.method manual ipv4.addresses "172.16.X.20/24" connection.autoconnect yes //为聚合连接配置 IP 地址

[root@system1 ~]# nmcli connection up team0 //激活聚合连接

[root@system1 ~]# nmcli con up team0-eth1 //激活成员连接eth1

[root@system1 ~]# nmcli con up team0-eth2 //激活成员连接eth2

[root@system1 ~]#teamdctl team0 state //确认连接状态

验证方法就是两侧互ping，然后down端口测试

* 1. 配置 IPv6 地址

试题概述:

在您的考试系统上配置接口 eth0 使用下列 IPv6 地址:

 system1 上的地址应该是 2017:7:22::305/64

 system2 上的地址应该是 2017:7:22::306/64

 两个系统必须能与网络 2017:7:22/64 内的系统通信

 地址必须在重启后依旧生效

 两个系统必须保持当前的 IPv4 地址并能通信

解题参考【配置第一台，第二台一样】:

[root@system1 ~]# nmcli connection show //获知连接名称

System eth0 5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03802-3-ethernet eth0

[root@system1 ~]# nmcli connection modify "System eth0" ipv6.method manual \ ipv6.addresses 2017:7:22::305/64

[root@system1 ~]# nmcli connection up "System eth0"

测试验证方法，相互通过ipv6的ping测试，ping6 2017:7:22::305

* 1. 配置本地邮件服务

试题概述:

在系统 system1 和 system2 上配置邮件服务,满足以下要求:

 这些系统不接收外部发送来的邮件

 在这些系统上本地发送的任何邮件都会自动路由到 server1.groupX.example.com

 从这些系统上发送的邮件显示来自于 groupX.example.com 您可以通过发送邮件到本地用户 arthur 来测试您的配置,系统

server1.groupX.example.com 已经配置把此用户的邮件转到下列 URL: http://server1.groupX.example.com/received\_mail/3

解题参考[两台配置一样]:

[root@system1 ~]# vim /etc/postfix/main.cf

relayhost = [server1.groupX.example.com] //后端邮件服务器

inet\_interfaces = loopback-only //仅本机

myorigin = groupX.example.com //发件来源域

mynetworks = 127.0.0.0/8 [::1]/128 //信任网络

mydestination = //此行的值设为空

local\_transport = error:local delivery disabled //此项提供拒绝报错,若无要求可不设

[root@system1 ~]# systemctl restart postfix

[root@system1 ~]# systemctl enable postfix

[root@system1 ~]# echo 'Zhou FengJie' | mail -s 'Tyumen Mail Test' arthur //系统 1 发信测试

测试就访问web的url：http://server1.groupX.example.com/received\_mail/3查看内容

不更改配置文件的话，也可以通过postconf命令来直接修改配置

* 1. 通过 Samba 发布共享目录

试题概述:

在 system1 上通过 SMB 共享/common 目录:

 您的 SMB 服务器必须是 STAFF 工作组的一个成员

 共享名必须为 common

 只有 groupX.example.com 域内的客户端可以访问 common 共享

 common 必须是可以浏览的

 用户 harry 必须能够读取共享中的内容,如果需要的话,验证的密码是 migwhisk

解题参考:

[root@system1 ~]# yum -y install samba

[root@system1 ~]# mkdir /common

[root@system1 ~]# setsebool -P samba\_export\_all\_rw=on //取消 SELinux 限制

 [root@system1 ~]# useradd harry ; pdbedit -a harry

[root@system1 ~]# vim /etc/samba/smb.conf

[global]

 workgroup = STAFF

[common]

 path = /common

hosts allow = 172.24.0.0/24

[root@system1 ~]# systemctl restart smb

[root@system1 ~]# systemctl enable smb

* 1. 配置多用户 Samba 挂载

试题概述:

在 system1 通过 SMB 共享目录/devops,并满足以下要求:

 共享名为 devops

 共享目录 devops 只能被 groupX.example.com 域中的客户端使用

 共享目录 devops 必须可以被浏览

 用户 kenji 必须能以读的方式访问此共享,该问密码是 atenorth

 用户 chihiro 必须能以读写的方式访问此共享,访问密码是 atenorth

 此共享永久挂载在 system2.groupX.example.com 上的/mnt/dev 目录,并使用用户

kenji 作为认证,任何用户可以通过用户 chihiro 来临时获取写的权限

解题参考:

在system1上：

[root@system1 ~]# mkdir /devops

[root@system1 ~]# useradd kenji ; pdbedit -a kenji

[root@system1 ~]# useradd chihiro ; pdbedit -a chihiro

[root@system1 ~]# setfacl -m u:chihiro:rwx /devops/

[root@system1 ~]# vim /etc/samba/smb.conf .. ..

[devops]

 path = /devops

 write list = chihiro

 hosts allow = 172.24.0.0/24

[root@system1 ~]# systemctl restart smb

在system2上：

[root@system2 ~]# yum -y install samba-client cifs-utils

[root@system2 ~]# smbclient -L serverX

.. ..

[root@system2 ~]# mkdir /mnt/dev

[root@system2 ~]# vim /etc/fstab

//serverX.example.com/devops username=kenji,password=atenorth,multiuser,sec=ntlmssp,\_netdev 0 0

[root@system2 ~]# mount -a

在system2上验证权限：

[root@system2 ~]# su - student //切换到普通用户

[student@system2 ~]$ touche /mnt/dev/a.txt //会被deny

[student@system2 ~]$ cifscreds add -u chihiro serverX //向服务器提交用户认证凭据 Password: //提供 Samba 用户 chihiro 的密码

[student@system2 ~]$ touch /mnt/dev/b.txt //会成功

* 1. 配置 NFS 共享服务

试题概述:

在 system1 配置 NFS 服务,要求如下:

 以只读的方式共享目录/public,同时只能被 groupX.example.com 域中的系统访问

 以读写的方式共享目录/protected,能被 groupX.example.com 域中的系统访问

 访问/protected 需要通过 Kerberos 安全加密,您可以使用下面 URL 提供的密钥:

http://host.groupX.example.com/material/nfs\_server.keytab

 目录/protected 应该包含名为 project 拥有人为 krishna 的子目录

 用户 krishna 能以读写方式访问/protected/project

解题参考：

[root@system1 ~]# mkdir -p /public /protected/project

[root@system1 ~]# chown krishna /protected/project/

[root@system1 ~]# wget -O /etc/krb5.keytab http://host.groupX.example.com/material/nfs\_server.keytab

[root@system1 ~]# vim /etc/exports

/public 172.24.X.0/24(ro)

/protected 172.24.X.0/24(rw,sec=krb5p)

[root@system1 ~]# vim /etc/sysconfig/nfs //若未明确要求版本,此操作可不做,如果考试的时候要求版本号为4的话，配置如下

RPCNFSDARGS="-V 4"

[root@system1 ~]# systemctl start nfs-secure-server nfs-server //启用两个系统服务

[root@system1 ~]# systemctl enable nfs-secure-server nfs-server

[root@system1 ~]# exportfs -rv

* 1. 挂载 NFS共享

试题概述:

在 system2 上挂载一个来自 system1.goup3.exmaple.com 的共享,并符合下列要求:

 /public 挂载在下面的目录上/mnt/nfsmount

 /protected挂载在下面的目录上/mnt/nfssecure 并使用安全的方式,密钥下载 URL:

 http://host.groupX.example.com/nfs\_client.keytab

 用户 krishna 能够在/mnt/nfssecure/project 上创建文件

 这些文件系统在系统启动时自动挂载

解题参考:

[root@system2 ~]# mkdir -p /mnt/nfsmount /mnt/nfssecure

[root@system2 ~]# wget -O /etc/krb5.keytab http://host.groupX.example.com/nfs\_client.keytab

[root@system2 ~]# systemctl start nfs-secure

[root@system2 ~]# systemctl enable nfs-secure

[root@system2 ~]# showmount -e system1

Export list for system1:

/protected 172.25.X.0/24

/public 172.25.X.0/24

//启用安全 NFS 的客户端服务 //查看对方提供了哪些共享

 [root@system2 ~]# vim /etc/fstab

.. ..

serverX.example.com:/public /mnt/nfsmount nfs \_netdev 0 0 serverX.example.com:/protected /mnt/nfssecure nfs sec=krb5p,\_netdev 0 0

[root@system2 ~]# mount –a

[root@system2 ~]# ssh krishna@system2

[krishna@system2 ~]$ touch /mnt/nfssecure/project/a.txt //可以成功创建

* 1. 实现一个 web 服务器

试题概述:

为 http://system1.groupX.example.com 配置 Web 服务器:

 从http://server1.groupX.example.com/materials/station.html 下载一个主页文

件,并将该文件重命名为 index.html

 将文件 index.html 拷贝到您的 web 服务器的 DocumentRoot 目录下

 不要对文件 index.html 的内容进行任何修改

 来自于 groupX.example.com 域的客户端可以访问此 Web 服务

 来自于 my133t.org 域的客户端拒绝访问此 Web 服务

解题参考:

[root@serverX ~]# yum -y install httpd

[root@serverX ~]# vim /etc/httpd/conf.d/00-default.conf

<VirtualHost \*:80>

ServerName system1.groupX.example.com

DocumentRoot /var/www/html

</VirtualHost>

//添加第一个(默认)虚拟主机

[root@serverX ~]# cd /var/www/html/

[root@serverX html]# wget http://classroom/pub/materials/station.html -O index.html

[root@serverX html]# systemctl restart httpd

[root@serverX html]# systemctl enable httpd

* 1. 实现一个 web 服务器

试题概述:

为站点 http://system1.groupX.example.com 配置 TLS 加密:

 一个已签名证书从 http://host.groupX.example.com/materials/system1.crt 获取

 此证书的密钥从 http://host.groupX.example.com/materials/system1.key 获取

 此证书的签名授权信息从http://host.groupX.example.com/materials/groupX.crt获取

解题参考:

[root@serverX ~]# yum -y install mod\_ssl

[root@serverX ~]# cd /etc/pki/tls/certs/

[root@serverX certs]# wget http://host.groupX.example.com/materials/system1.crt

[root@serverX certs]# wget http://host.groupX.example.com/materials/system1.key

[root@serverX certs]# cd /etc/pki/tls/private/

[root@serverX private]# wget http://host.groupX.example.com/materials/groupX.crt

[root@serverX private]# vim /etc/httpd/conf.d/ssl.conf

<VirtualHost \_default\_:443>

DocumentRoot "/var/www/html" ServerName system1.groupX.example.com:443

.. ..

SSLCertificateFile /etc/pki/tls/certs/system1.crt

SSLCertificateKeyFile /etc/pki/tls/certs/system1.key

SSLCACertificateFile /etc/pki/tls/private/groupX.crt

</VirtualHost>

[root@serverX private]# systemctl restart httpd

* 1. 配置虚拟主机

试题概述:

在system1上扩展您的 web 服务器,为站点 http://www.groupX.example.com 创建一个虚拟主机,然后执行下述步骤:

 设置 DocumentRoot 为/var/www/virtual

 从 http://server1.groupX.example.com/materials/www.html 下载文件并重命名为index.html

 不要对文件 index.html 的内容做任何修改

 将文件 index.html 放到虚拟主机的 DocumentRoot 目录下

 确保 harry 用户能够在/var/www/virtual 目录下创建文件

注意:原始站点 http://system1.groupX.example.com 必须仍然能够访问,名称服务器

groupX.example.com 提供对主机名 www.groupX.example.com 的域名解析。

解题参考:

[root@serverX ~]# mkdir /var/www/virtual

[root@serverX ~]# setfacl -m u:harry:rwx /var/www/virtual/

[root@serverX ~]# cd /var/www/virtual/

[root@serverX virtual]# wget -O index.html http://server1.groupX.example.com/materials/www.html

 [root@serverX virtual]# vim /etc/httpd/conf.d/01-www.conf

<VirtualHost \*:80>

ServerName www.groupX.example.com

DocumentRoot /var/www/virtual

</VirtualHost>

[root@serverX virtual]# systemctl restart httpd

* 1. 配置 web 内容的访问

试题概述:

在您的 system1 上的 web 服务器的 DocumentRoot 目录下创建一个名为private 的目录,要求如下:

 从 http://server1.groupX.example.com/materails/private.html 下载一个文件副本到这个目录,并且得命名为 index.html

 不要对这个文件的内容做任何修改

 从 system1 上,任何人都可以浏览 private 的内容,但是从其他系统不能访问这个目录的内容

解题参考:

[root@serverX ~]# mkdir /var/www/html/private

[root@serverX ~]# cd /var/www/html/private/

[root@serverX private]# wget http://server1.groupX.example.com/materails/private.html -O index.html

[root@serverX private]# vim /etc/httpd/conf.d/00-default.conf

.. ..

<Directory /var/www/html/private>

Require ip 127.0.0.1 ::1 172.24.X.11 //仅允许本机 IP 访问

</Directory>

[root@serverX private]# systemctl restart httpd

* 1. 实现动态 WEB 内容

试题概述:

在 system1 上配置提供动态 Web 内容,要求如下:

 动态内容由名为 alt.groupX.example.com 的虚拟主机提供

 虚拟主机侦听在端口 8909

 从 http://server1.groupX.example.com/materials/webinfo.wsgi 下载一个脚本,然后放在适当的位置,无论如何不要修改此文件的内容

 客户端访问 http://alt.groupX.example.com:8909 可接收到动态生成的 Web 页

 此 http://alt.groupX.example.com:8909/必须能被 groupX.example.com 域内的所有系统访问

解题参考:

[root@serverX ~]# yum -y install mod\_wsgi

[root@serverX ~]# mkdir /var/www/alt

[root@serverX ~]# cd /var/www/alt

[root@serverX alt]# wget http://server1.groupX.example.com/materials/webinfo.wsgi

[root@serverX alt]# vim /etc/httpd/conf.d/02-alt.conf

Listen 8909

<VirtualHost \*:8909>

ServerName alt.groupX.example.com

DocumentRoot /var/www/alt

WSGIScriptAlias / /var/www/alt/webinfo.wsgi

</VirtualHost>

[root@serverX alt]# semanage port -a -t http\_port\_t -p tcp 8909 //开启非标准端口

[root@serverX alt]# systemctl restart httpd

* 1. 创建一个脚本

试题概述:

在 system1 上创建一个名为/root/foo.sh 的脚本,让其提供下列特性:

 当运行/root/foo.sh redhat,输出为 fedora

 当运行/root/foo.sh fedora,输出为 redhat

 当没有任何参数或者参数不是 redhat 或者 fedora 时,其错误输出产生以下的信息:

/root/foo.sh redhat|fedora

解题参考:

[root@serverX ~]# vim /root/foo.sh

#!/bin/bash

if [ $# != 1 ];then

 echo "/root/foo.sh redhat|fedora" >&2

else

 case $1 in

 redhat)

 echo "fedora"

 ;;

 fedora)

 echo "redhat"

 ;;

 \*)

 echo "/root/foo.sh redhat|fedora" >&2

 ;;

fi

[root@serverX ~]# chmod +x /root/foo.sh

* 1. 创建一个添加用户的脚本

试题概述:

在 system1 上创建一个脚本,名为/root/batchusers,此脚本能实现为系统 system1 创建本地用户,并且这些用户的用户名来自一个包含用户名的文件,同时满足下列要求:

 此脚本要求提供一个参数,此参数就是包含用户名列表的文件

 如果没有提供参数,此脚本应该给出下面的提示信息 Usage: /root/batchusers <userfile> 然后退出并返回相应的值

 如果提供一个不存在的文件名,此脚本应该给出下面的提示信息 Input file not found 然后退出并返回相应的值

 创建的用户登陆 Shell 为/bin/false,此脚本不需要为用户设置密码

 您可以从下面的 URL 获取用户名列表作为测试用:

http://server1.groupX.example.com/materials/userlist

解题参考:

[root@serverX ~]# wget -O /root/userlist http://server1.groupX.example.com/materials/userlist

[root@serverX ~]# vim /root/batchusers

#!/bin/bash

if [ $# != 1 ] ; then

echo "Usage: /root/batchusers <userfile>"

exit 1

else

if [ ! -f $1 ] ; then

echo "Input file not found" exit 2

else

 cat $1|while read name

do

 useradd -s /bin/false $name

 done

fi

fi

[root@serverX ~]# chmod +x /root/batchusers

* 1. 配置 iSCSI 服务端

试题概述:

配置 system1 提供 iSCSI服务,磁盘名为 iqn.2017-07.com.example.groupX:system1,并符合下列要求：

 服务端口为3260

 使用 iscsi\_store 作其后端卷,其大小为 3GiB

 此服务只能被 system2.groupX.example.com 访问

解题参考:

1)准备磁盘空间

[root@serverX ~]# fdisk /dev/vda

……

Last sector, +sectors or +size{K,M,G,T,P} (2048-20971486, default 20971486): +3G //结束位置设置 +3G

Command (m for help): w // w 保存分区更改

.. ..

[root@serverX ~]# partprobe /dev/vda

2)安装、配置 iSCSI 磁盘

[root@serverX ~]# yum -y install targetcli

[root@serverX ~]# targetcli

/> backstores/block create iscsi\_store /dev/vda5 //定义后端存储

/> /iscsi create iqn.2017-07.com.example.groupX:system1 //创建 iqn 对象

/>/iscsi/iqn.2017-07.com.example.groupX:system1/tpg1/acls create iqn.2017-07.com.example.groupX:system2 //授权客户机(的 IQN)

/>/iscsi/iqn.2017-07.com.example.groupX:system1/tpg1/luns create /backstores/block/iscsi\_store //绑定存储

/> /iscsi/iqn.2017-07.com.example.groupX:system1/tpg1/portals create 0.0.0.0 3260 //指定监听地址(本机 IP 及端口)

/> saveconfig //保存配置结果(缺省)

/> exit

[root@serverX ~]# systemctl restart target

[root@serverX ~]# systemctl enable target

* 1. 配置 iSCSI 客户端

试题概述:

配置 system2 使其能连接 system1 上提供的 iqn.2017-07.com.example.groupX:system1,

并符合以下要求:

 iSCSI 设备在系统启动的期间自动加载

 块设备 iSCSI 上包含一个大小为 2100MiB 的分区,并格式化为 ext4 文件系统

 此分区挂载在/mnt/data 上,同时在系统启动的期间自动挂载

解题参考:

[root@desktopX ~]# yum -y install iscsi-initiator-utils

[root@desktopX ~]# vim /etc/iscsi/initiatorname.iscsi //设置本机 iqn 名称

InitiatorName=iqn.2017-07.com.example.groupX:system2

[root@desktopX ~]# systemctl restart iscsid //起 iscsid 服务以读取 iqn 名称

[root@desktopX ~]# iscsiadm -m discovery -t st -p system1 //发现磁盘

[root@desktopX ~]# iscsiadm -m node -L all //连接磁盘

[root@desktopX ~]# vim /var/lib/iscsi/nodes/iqn.2017-07.com.example.groupX\:system1/\*/default

node.conn[0].startup = automatic

.. ..

[root@desktopX ~]# systemctl enable iscsi

[root@desktopX ~]# lsblk //查看挂载多出来的磁盘

[root@desktopX ~]# fdisk /dev/sda //格式化磁盘

Command (m for help): n

Partition number (1-128, default 1):

First sector (34-20971486, default 2048):

Last sector, +sectors or +size{K,M,G,T,P} (2048-20971486, default 20971486): +2100M //结束位置设置 +2100M

Command (m for help): w // w 保存分区更改

.. ..

[root@desktopX ~]# partprobe /dev/sda//刷新分区表

[root@desktopX ~]# mkfs.ext4 /dev/sda1 //按要求格式化分区

[root@desktopX ~]# mkdir /mnt/data //创建挂载点

[root@desktopX ~]# blkid /dev/sda1 //找到分区 UUID

/dev/sda1: UUID="6ff20bb3-7543-4fa0-b4fa-bdc99a1e63ce" .. .. [root@desktopX ~]# vim /etc/fstab

.. ..

UUID="6ff20bb3-7543-4fa0-b4fa-bdc99a1e63ce" /mnt/data ext4 \_netdev 0 0

[root@desktopX ~]# mount -a

[root@desktopX ~]# sync ; reboot -f

* 1. 配置一个数据库

试题概述:

在 system1 上创建一个 MariaDB 数据库,名为 Contacts,并符合以下条件:

 数据库应该包含来自数据库复制的内容,复制文件的 URL 为:

 http://server1.groupX.example.com/materials/users.sql

 数据库只能被 localhost 访问

 除了 root 用户,此数据库只能被用户 Raikon 查询,此用户密码为 atenorth

 root 用户的密码为 atenorth,同时不允许空密码登陆。

解题参考:

 1)安装、配置

[root@serverX ~]# yum -y install mariadb-server mariadb [root@serverX ~]# vim /etc/my.cnf

[mysqld]

skip-networking

[root@serverX ~]# systemctl restart mariadb [root@serverX ~]# systemctl enable mariadb

2)设密码、建库

[root@serverX ~]# mysqladmin -u root -p password 'atenorth' //设置密码 [root@serverX ~]# mysql -u root -p

MariaDB [(none)]> CREATE DATABASE Contacts;

MariaDB [(none)]> GRANT select ON Contacts.\* to Raikon@localhost IDENTIFIED BY 'atenorth'; MariaDB [(none)]> DELETE FROM mysql.user WHERE Password=''; //删除空密码账号

MariaDB [(none)]> QUIT

3)导入库

[root@serverX ~]# wget http://server1.groupX.example.com/materials/users.sql

[root@serverX ~]# mysql -u root -p Contacts < users.sql

* 1. 数据库查询(填空)

试题概述:

在系统 system1 上使用数据库 Contacts,并使用相应的 SQL 查询以回答下列问题:

 密码是 solicitous 的人的名字?

 有多少人的姓名是 Barbara 同时居住在 Sunnyvale?

解题参考:

[root@serverX ~]# mysql -u root -p

Enter password:

MariaDB [Contacts]> USE Contacts;

MariaDB [Contacts]> SELECT name FROM base WHERE password='solicitous'; +-------+

| name |

+-------+

| James |

+-------+

MariaDB [Contacts]> SELECT count(\*) FROM base,location WHERE base.name='Barbara' AND location.city='Sunnyvale' AND base.id=location.id ;

1

MariaDB [Contacts]> QUIT