

```
etc/mtab
...output omitted...
[root@host etcbackup]# tar -xf /root/etc.tar
```

默认情况下，从存档中提取文件时，将从存档内容的权限中去除 umask。要保留存档文件的权限，可在提取存档时使用 **p** 选项。

在本例中，将存档 **/root/myscripts.tar** 提取到 **/root/scripts** 目录，同时保留所提取文件的权限：

```
[root@host ~]# mkdir /root/scripts
[root@host ~]# cd /root/scripts
[root@host scripts]# tar -xpf /root/myscripts.tar
```

创建压缩存档

tar 命令支持三种压缩方式。**tar** 命令支持三种不同的压缩方式。**gzip** 压缩速度最快，历史最久，使用也最为广泛，能够跨发行版甚至跨平台使用。**bzip2** 压缩创建的存档文件通常比 **gzip** 创建的文件小，但可用性不如 **gzip** 广泛；而 **xz** 压缩方式相对较新，但通常提供可用方式中最佳的压缩率。



注意

任何压缩算法的效用都取决于被压缩数据的类型。已经压缩过的数据文件，如压缩图片格式或 RPM 文件，通常导致较低的压缩率。

最好是使用单个顶级目录，其中可包含其他的目录和文件，以通过有序的方式来简化文件提取。

使用下列选项之一来创建压缩的 **tar** 存档：

- **-z** 或 **--gzip** 进行 gzip 压缩 (**filename.tar.gz** 或 **filename.tgz**)
- **-j** 或 **--bzip2** 进行 bzip2 压缩 (**filename.tar.bz2**)
- **-J** 或 **-xz** 进行 xz 压缩 (**filename.tar.xz**)

创建 **gzip** 压缩存档 **/root/etcbackup.tar.gz**，其内容来自 host 上的 **/etc** 目录：

```
[root@host ~]# tar -czf /root/etcbackup.tar.gz /etc
tar: Removing leading `/' from member names
```

创建 **bzip2** 压缩存档 **/root/logbackup.tar.bz2**，其内容来自 host 上的 **/var/log** 目录：

```
[root@host ~]$ tar -cjf /root/logbackup.tar.bz2 /var/log
tar: Removing leading `/' from member names
```

创建 **xz** 压缩存档 **/root/sshconfig.tar.xz**，其内容来自 host 上的 **/etc/ssh** 目录：

```
[root@host ~]$ tar -cJf /root/sshconfig.tar.xz /etc/ssh
tar: Removing leading `/' from member names
```

创建存档后，使用 **tf** 选项来验证存档的内容。在列出压缩存档文件的内容时，不必强制使用压缩代理选项。例如，要列出采用 **gzip** 压缩方式的 **/root/etcbackup.tar.gz** 文件中存档的内容，可使用以下命令：

```
[root@host ~]# tar -tf /root/etcbackup.tar.gz /etc
etc/
etc/fstab
etc/crypttab
etc/mtab
...output omitted...
```

提取压缩的存档

提取压缩的 **tar** 存档时，要执行的第一步是决定存档文件应提取到的位置，然后创建并更改到目标目录。**tar** 命令会判断之前使用的压缩方式，因此通常不需要使用在创建存档时所用的同一压缩选项。可以在 **tar** 命令中添加解压缩方式。如果选择这样做，则必须使用正确的解压缩类型选项；否则，**tar** 会生成错误来指出选项中指定的解压缩类型与文件的解压缩类型不匹配。

要将 **gzip** 压缩的存档 **/root/etcbackup.tar.gz** 的内容提取到 **/tmp/etcbackup** 目录：

```
[root@host ~]# mkdir /tmp/etcbackup
[root@host ~]# cd /tmp/etcbackup
[root@host etcbackup]# tar -tf /root/etcbackup.tar.gz
etc/
etc/fstab
etc/crypttab
etc/mtab
...output omitted...
[root@host etcbackup]# tar -xzf /root/etcbackup.tar.gz
```

要将 **bzip2** 压缩的存档 **/root/logbackup.tar.bz2** 的内容提取到 **/tmp/logbackup** 目录：

```
[root@host ~]# mkdir /tmp/logbackup
[root@host ~]# cd /tmp/logbackup
[root@host logbackup]# tar -tf /root/logbackup.tar.bz2
var/log/
var/log/lastlog
var/log/README
var/log/private/
var/log/wtmp
var/log/btmp
...output omitted...
[root@host logbackup]# tar -xjf /root/logbackup.tar.bz2
```

要将 **xz** 压缩的存档 **/root/sshbackup.tar.xz** 的内容提取到 **/tmp/sshbackup** 目录：

```
[root@host ~]$ mkdir /tmp/sshbackup
[root@host ~]# cd /tmp/sshbackup
[root@host logbackup]# tar -tf /root/sshbackup.tar.xz
etc/ssh/
etc/ssh/moduli
etc/ssh/ssh_config
```

```
etc/ssh/ssh_config.d/  
etc/ssh/ssh_config.d/05-redhat.conf  
etc/ssh/sshd_config  
...output omitted...  
[root@host sshbackup]# tar -xJf /root/sshbackup.tar.xz
```

列出压缩的 tar 存档的工作方式与列出未压缩的 tar 存档相同。



注意

此外，**gzip**、**bzip2** 和 **xz** 也可单独用于压缩单个文件。例如，**gzip etc.tar** 命令将生成 **etc.tar.gz** 压缩文件，**bzip2 abc.tar** 命令将生成 **abc.tar.bz2** 压缩文件，而 **xz myarchive.tar** 命令则生成 **myarchive.tar.xz** 压缩文件。

对应的解压缩命令为 **gunzip**、**bunzip2** 和 **unxz**。例如，**gunzip /tmp/etc.tar.gz** 命令将生成 **etc.tar** 解压缩 tar 文件，**bunzip2 abc.tar.bz2** 命令将生成 **abc.tar** 解压缩 tar 文件，而 **unxz myarchive.tar.xz** 命令则生成 **myarchive.tar** 解压缩 tar 文件。



参考文献

tar(1)、**gzip(1)**、**gunzip(1)**、**bzip2(1)**、**bunzip2(1)**、**xz(1)**、**unxz(1)** man page

► 指导练习

管理压缩的 TAR 存档

在本练习中，您将使用 tar 命令创建存档文件并提取其内容。

成果

您应能够归档目录树，并将存档内容提取到其他位置。

在你开始之前

在 workstation 上，以 student 用户身份并使用密码 student 进行登录。

从 workstation，运行 **lab archive-manage start** 命令。该命令将运行一个起始脚本，它将确定主机 servera 是否可从网络访问。该脚本还确保 servera 上不存在将要在练习中创建的文件和目录。

```
[student@workstation ~]$ lab archive-manage start
```

- 1. 使用 ssh 命令，以 student 用户身份登录 servera。

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. 切换为 root 用户，因为只有 root 用户才能访问 /etc 目录的所有内容。

```
[student@servera ~]$ su -  
Password: redhat  
[root@servera ~]#
```

- 3. 使用 tar 命令及 **-czf** 选项，以利用 gzip 压缩方式创建 /etc 目录的存档。将存档文件另存为 /tmp/etc.tar.gz。

```
[root@servera ~]# tar -czf /tmp/etc.tar.gz /etc  
tar: Removing leading `/' from member names  
[root@servera ~]#
```

- 4. 使用 tar 命令及 **-tzf** 选项，以验证 **etc.tar.gz** 存档包含来自 /etc 目录的文件。

```
[root@servera ~]# tar -tzf /tmp/etc.tar.gz
etc/
etc/mtab
etc/fstab
etc/crypttab
etc/resolv.conf
...output omitted...
```

- 5. 在 servera 上，创建名为 **/backuptest** 的目录。通过将 **etc.tar.gz** 备份文件解压缩到 **/backuptest** 目录，验证该文件是否为有效的存档。

5.1. 创建 **/backuptest** 目录。

```
[root@servera ~]# mkdir /backuptest
```

5.2. 更改为 **/backuptest** 目录。

```
[root@servera ~]# cd /backuptest
[root@servera backuptest]#
```

5.3. 在提取之前，列出 **etc.tar.gz** 存档的内容。

```
[root@servera backuptest]# tar -tzf /tmp/etc.tar.gz
etc/
etc/mtab
etc/fstab
etc/crypttab
etc/resolv.conf
...output omitted...
```

5.4. 将 **/tmp/etc.tar.gz** 存档提取到 **/backuptest** 目录。

```
[root@servera backuptest]# tar -xzf /tmp/etc.tar.gz
[root@servera backuptest]#
```

5.5. 列出 **/backuptest** 目录的内容。验证该目录是否包含来自 **/etc** 目录的文件。

```
[root@servera backuptest]# ls -l
total 12
drwxr-xr-x. 95 root root 8192 Feb  8 10:16 etc
[root@servera backuptest]# cd etc
[root@servera etc]# ls -l
total 1204
-rw-r--r--.  1 root root      16 Jan 16 23:41 adjtime
-rw-r--r--.  1 root root    1518 Sep 10 17:21 aliases
drwxr-xr-x.  2 root root     169 Feb   4 21:58 alternatives
-rw-r--r--.  1 root root     541 Oct   2 21:01 anacrontab
...output omitted...
```

- 6. 从 servera 退出。

```
[root@servera backuptest]# exit  
logout  
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation]$
```

完成

在 workstation 上，运行 **lab archive-manage finish** 脚本来完成本练习。

```
[student@workstation ~]$ lab archive-manage finish
```

本引导式练习到此结束。

在系统之间安全地传输文件

培训目标

学完本节后，您应能够通过 SSH 与远程系统安全地来回传输文件。

使用 SECURE COPY 传输文件

OpenSSH 命令对于确保在远程系统上安全运行 shell 命令非常有用。Secure Copy 命令 **scp** 是 OpenSSH 套件的一部分，可将文件从远程系统复制到本地系统或从本地系统复制到远程系统。此命令利用 SSH 服务器进行身份验证，并在数据传输之前对其进行加密。

您可以为所要复制的文件的源或目标指定一个远程位置。远程位置的格式应为 **[user@]host:/path**。该参数的 **user@** 部分是可选的。如果不指定该部分，则使用您当前的本地用户名。运行此命令时，**scp** 客户端将使用基于密钥的身份验证或以提示您输入密码的方式向远程 SSH 服务器进行身份验证，就像 **ssh** 一样。

以下示例演示了如何将 **host** 上的本地文件 **/etc/yum.conf** 和 **/etc/hosts** 复制到 **remotehost** 远程系统上 **remoteuser** 的主目录：

```
[user@host ~]$ scp /etc/yum.conf /etc/hosts remoteuser@remotehost:/home/remoteuser
remoteuser@remotehost's password: password
      yum.conf          100%   813      0.8KB/s  00:00
      hosts            100%   227      0.2KB/s  00:00
```

您还可以沿另一个方向复制文件，即从远程系统复制到本地文件系统。在本例中，**remotehost** 上的文件 **/etc/hostname** 将被复制到本地目录 **/home/user**。**scp** 命令以用户 **remoteuser** 的身份向 **remotehost** 进行身份验证。

```
[user@host ~]$ scp remoteuser@remotehost:/etc/hostname /home/user
remoteuser@remotehost's password: password
      hostname        100%    22      0.0KB/s  00:00
```

要以递归方式复制整个目录树，可使用 **-r** 选项。在下例中，**remotehost** 上的远程目录 **/var/log** 以递归方式复制到 **host** 上的本地目录 **/tmp/**。您必须以 **root** 身份连接到远程系统，以确保能读取远程 **/var/log** 目录中的所有文件。

```
[user@host ~]$ scp -r root@remoteuser:/var/log /tmp
root@remotehost's password: password
...output omitted...
```

使用安全文件传输程序传输文件

要以交互方式从 SSH 服务器上传或下载文件，请使用安全文件传输程序 **sftp**。**sftp** 命令的会话使用安全身份验证机制，并将数据加密后再与 SSH 服务器来回传输。

与 **scp** 命令一样，**sftp** 使用 **[user@]host** 来标识目标系统和用户名。如果未指定用户，该命令将尝试使用本地用户名作为远程用户名进行登录。随后会显示 **sftp>** 提示符。

```
[user@host ~]$ sftp remoteuser@remotehost  
remoteuser@remotehost's password: password  
Connected to remotehost.  
sftp>
```

交互式 **sftp** 会话接受各种命令，这些命令在远程文件系统上运行的方式与在本地文件系统上相同，如 **ls**、**cd**、**mkdir**、**rmdir** 和 **pwd**。**put** 命令将文件上传到远程系统。**get** 命令从远程系统下载文件。**exit** 命令可退出 **sftp** 会话。

将本地系统上的 **/etc/hosts** 文件上传到 **remotehost** 上新建的目录 **/home/remoteuser/hostbackup**。**sftp** 会话始终假设 **put** 命令后跟的是本地文件系统上的文件，并且首先连接用户的主目录（此例中为 **/home/remoteuser**）：

```
sftp> mkdir hostbackup  
sftp> cd hostbackup  
sftp> put /etc/hosts  
Uploading /etc/hosts to /home/remoteuser/hostbackup/hosts  
/etc/hosts                                         100%   227      0.2KB/s   00:00  
sftp>
```

要从远程主机下载 **/etc/yum.conf** 到本地系统上的当前目录，可执行 **get /etc/yum.conf** 命令，然后使用 **exit** 命令退出 **sftp** 会话。

```
sftp> get /etc/yum.conf  
Fetching /etc/yum.conf to yum.conf  
/etc/yum.conf                                         100%   813      0.8KB/s   00:00  
sftp> exit  
[user@host ~]$
```



参考文献

scp(1) 和 **sftp(1)** man page

► 指导练习

在系统之间安全地传输文件

在本练习中，您将使用 **scp** 命令将文件从远程系统复制到本地目录。

成果

您应能够将文件从远程主机复制到本地计算机上的一个目录。

在你开始之前

在 **workstation** 上，以 **student** 用户身份并使用密码 **student** 进行登录。

从 **workstation**，运行 **lab archive-transfer start** 命令。该命令将运行一个起始脚本，它将确定主机 **servera** 和 **serverb** 是否可从网络访问。该脚本还确保 **servera** 上不存在将要在练习中创建的文件和目录。

```
[student@workstation ~]$ lab archive-transfer start
```

- 1. 使用 **ssh** 命令，以 **student** 用户身份登录 **servera**。

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. 使用 **scp** 命令，将 **serverb** 上的 **/etc/ssh** 目录复制到 **servera** 上的 **/home/student/serverbackup** 目录。

2.1. 在 **servera** 上，创建名为 **/home/student/serverbackup** 的目录。

```
[student@servera ~]$ mkdir ~/serverbackup
```

2.2. 使用 **scp** 命令，将 **serverb** 上的 **/etc/ssh** 目录以递归方式复制到 **servera** 上的 **/home/student/serverbackup** 目录。在提示时，输入 **redhat** 作为密码。请注意，只有 **root** 用户可以读取 **/etc/ssh** 目录中的所有内容。

```
[student@servera ~]$ scp -r root@serverb:/etc/ssh ~/serverbackup
The authenticity of host 'serverb (172.25.250.11)' can't be established.
ECDSA key fingerprint is SHA256:qaS0PToLrqlC02XGk1A0iY7CaP7aPKimerDoaUkv720
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'serverb,172.25.250.11' (ECDSA) to the list of known hosts.
root@serverb's password: redhat
moduli                                         100%   550KB  57.9MB/s  00:00
ssh_config                                     100%   1727    1.4MB/s  00:00
05-redhat.conf                                 100%   690     1.6MB/s  00:00
sshd_config                                    100%  4469    9.5MB/s  00:00
```

```
ssh_host_ed25519_key          100%  387      1.2MB/s  00:00
ssh_host_ed25519_key.pub       100%   82      268.1KB/s  00:00
ssh_host_ecdsa_key             100%  492      1.5MB/s  00:00
ssh_host_ecdsa_key.pub         100%  162      538.7KB/s  00:00
ssh_host_rsa_key               100% 1799      4.9MB/s  00:00
ssh_host_rsa_key.pub           100%  382      1.2MB/s  00:00
```

2.3. 验证 serverb 上的 /etc/ssh 目录是否已复制到 servera 上的 /home/student/serverbackup 目录。

```
[student@servera ~]$ ls -lR ~/serverbackup
/home/student/serverbackup:
total 0
drwxr-xr-x. 3 student student 245 Feb 11 18:35 ssh

/home/student/serverbackup/ssh:
total 588
-rw-r--r--. 1 student student 563386 Feb 11 18:35 moduli
-rw-r--r--. 1 student student 1727 Feb 11 18:35 ssh_config
drwxr-xr-x. 2 student student 28 Feb 11 18:35 ssh_config.d
-rw-----. 1 student student 4469 Feb 11 18:35 sshd_config
-rw-r-----. 1 student student 492 Feb 11 18:35 ssh_host_ecdsa_key
-rw-r--r--. 1 student student 162 Feb 11 18:35 ssh_host_ecdsa_key.pub
-rw-r-----. 1 student student 387 Feb 11 18:35 ssh_host_ed25519_key
-rw-r--r--. 1 student student 82 Feb 11 18:35 ssh_host_ed25519_key.pub
-rw-r-----. 1 student student 1799 Feb 11 18:35 ssh_host_rsa_key
-rw-r--r--. 1 student student 382 Feb 11 18:35 ssh_host_rsa_key.pub

/home/student/serverbackup/ssh/ssh_config.d:
total 4
-rw-r--r--. 1 student student 690 Feb 11 18:35 05-redhat.conf
```

► 3. 从 servera 退出。

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

完成

在 workstation 上，运行 lab archive-transfer finish 脚本来完成本练习。

```
[student@workstation ~]$ lab archive-transfer finish
```

本引导式练习到此结束。

在系统间安全地同步文件

培训目标

学完本节后，您应能够高效而安全地将本地文件或目录的内容与远程服务器上的副本进行同步。

使用 **rsync** 同步文件和目录

rsync 命令是在系统之间安全复制文件的另一种方式。此工具采用的算法可通过仅同步已更改的文件部分来将复制的数据量最小化。它与 **scp** 的区别在于，如果两个服务器间的两个文件或目录相似，**rsync** 将仅复制文件系统间的差异部分，而 **scp** 仍复制所有内容。

rsync 的一个优点是它能够在本地系统和远程系统之间安全而高效地复制文件。虽然首次目录同步的用时与复制操作大致相同，但之后的同步只需通过网络复制差异部分，从而会大幅加快更新的速度。

rsync 的一个重要选项是 **-n** 选项，它用于执行空运行。空运行是对执行命令时所发生情况的模拟。空运行显示了在命令正常运行时 **rsync** 所要进行的更改。请在进行实际 **rsync** 操作前先执行空运行，以确保重要的文件不会被覆盖或删除。

使用 **rsync** 进行同步时，两个常用的选项为 **-v** 和 **-a** 选项。

-v 或 **--verbose** 选项可提供更详细的输出。这对于故障排除和查看实时进度非常有用。

-a 或 **--archive** 选项将启用“存档模式”。这样可实现递归复制并开启很多有用的选项，以保留文件的大部分特征。存档模式与指定以下选项的作用相同：

通过 **rsync -a** 启用的选项（存档模式）

选项	描述
-r、--recursive	以递归方式同步整个目录树
-l、--links	同步符号链接
-p、--perms	保留权限
-t、--times	保留时间戳
-g、--group	保留组所有权
-o、--owner	保留文件所有者
-D、--devices	同步设备文件

存档模式不会保留硬链接，因为这会大幅增加同步时间。如果您还想保留硬链接，请添加 **-H** 选项。

**注意**

如果您使用的是高级权限，则可能还需要另外两个选项：

- **-A** 用于保留 ACL
- **-X** 用于保留 SELinux 上下文

您可以使用 **rsync** 将本地文件或目录的内容与远程计算机上的文件或目录进行同步（使用任一计算机作为源）。您也可以同步两个本地文件或目录的内容。

例如，要让 **/var/log** 目录的内容与 **/tmp** 目录保持同步：

```
[user@host ~]$ su -
Password: password
[root@host ~]# rsync -av /var/log /tmp
receiving incremental file list
log/
log/README
log/boot.log
...output omitted...
log/tuned/tuned.log

sent 11,592,423 bytes received 779 bytes 23,186,404.00 bytes/sec
total size is 11,586,755 speedup is 1.00
[root@host ~]$ ls /tmp
log  ssh-RLjDdarkKiW1
[root@host ~]$
```

通过为源目录加上尾随斜杠，可以同步该目录的内容，而在目标目录中新建子目录。在本例中，**log** 目录不是在 **/tmp** 目录中创建的，仅 **/var/log/** 的内容同步到 **/tmp** 中。

```
[root@host ~]# rsync -av /var/log/ /tmp
sending incremental file list
./
README
boot.log
...output omitted...
tuned/tuned.log

sent 11,592,389 bytes received 778 bytes 23,186,334.00 bytes/sec
total size is 11,586,755 speedup is 1.00
[root@host ~]# ls /tmp
anaconda          dnf.rpm.log-20190318  private
audit             dnf.rpm.log-20190324  qemu-ga
boot.log          dnf.rpm.log-20190331  README
...output omitted...
```

**重要**

为 **rsync** 命令键入源目录时，目录名称中是否存在尾随斜杠至关重要。它将决定同步到目标中的是目录还是仅目录中的内容。

Tab 补全可自动在目录名称中添加尾随斜杠。

跟 **scp** 和 **sftp** 命令一样，**rsync** 使用 **[user@]host:/path** 格式来指定远程位置。远程位置可以是源或目标系统，但两台计算机中的一台必须是本地计算机。

要保留文件所有权，您需要是目标系统上的 **root** 用户。如果目标为远程目标，则以 **root** 身份进行验证。如果目标为本地目标，则必须以 **root** 身份运行 **rsync**。

在本例中，将本地的 **/var/log** 目录同步到 **remotehost** 系统上的 **/tmp** 目录：

```
[root@host ~]# rsync -av /var/log remotehost:/tmp
root@remotehost's password: password
receiving incremental file list
log/
log/README
log/boot.log
...output omitted...
sent 9,783 bytes received 290,576 bytes 85,816.86 bytes/sec
total size is 11,585,690 speedup is 38.57
```

同样，**remotehost** 上的 **/var/log** 远程目录也可同步到 **host** 上的 **/tmp** 本地目录：

```
[root@host ~]# rsync -av remotehost:/var/log /tmp
root@remotehost's password: password
receiving incremental file list
log/boot.log
log/dnf.librepo.log
log/dnf.log
...output omitted...

sent 9,783 bytes received 290,576 bytes 85,816.86 bytes/sec
total size is 11,585,690 speedup is 38.57
```

**参考文献**

rsync(1) man page

► 指导练习

在系统间安全地同步文件

在本练习中，您将通过 SSH 使用 **rsync** 将本地目录的内容与远程服务器上的副本同步。

成果

您应能够使用 **rsync** 命令将本地目录的内容与远程服务器上的副本同步。

在你开始之前

在 **workstation** 上，以 **student** 用户身份并使用密码 **student** 进行登录。

从 **workstation**，运行 **lab archive-sync start** 命令。该命令将运行一个起始脚本，它将确定主机 **servera** 和 **serverb** 是否可从网络访问。该脚本还确保 **servera** 上不存在将要在练习中创建的文件和目录。

```
[student@workstation ~]$ lab archive-sync start
```

- 1. 使用 **ssh** 命令，以 **student** 用户身份登录 **servera**。

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. 在 **servera** 上，创建名为 **/home/student/serverlogs** 的目录。使用 **rsync** 命令，在 **servera** 上的 **/home/student/serverlogs** 目录中创建 **serverb** 上 **/var/log** 目录树的一个初始副本。

- 2.1. 在 **servera** 上，创建名为 **/home/student/serverlogs** 的目标目录，以存储从 **serverb** 同步而来的日志文件。

```
[student@servera ~]$ mkdir ~/serverlogs
```

- 2.2. 使用 **rsync** 命令，将 **serverb** 上的 **/var/log** 目录树同步到 **servera** 上的 **/home/student/serverlogs** 目录。请注意，只有 **root** 用户可以读取 **serverb** 上 **/var/log** 目录中的所有内容。首次同步时将传输所有文件。

```
[student@servera ~]$ rsync -av root@serverb:/var/log ~/serverlogs  
root@serverb's password: redhat  
receiving incremental file list  
log/  
log/README  
log/boot.log  
...output omitted...  
log/tuned/tuned.log
```

```
sent 992 bytes received 13,775,064 bytes 2,119,393.23 bytes/sec
total size is 13,768,109 speedup is 1.00
```

- 3. 在 serverb 上, 以 root 身份运行 **logger "Log files synchronized"** 命令, 在 /var/log/messages 日志文件中获取一个新条目, 以反映上一次发生同步的时间。

```
[student@servera ~]$ ssh root@serverb 'logger "Log files synchronized"'
Password: redhat
[student@servera ~]$
```

- 4. 使用 **rsync** 命令, 将 serverb 上的 /var/log 目录树安全地同步到 servera 上的 /home/student/serverlogs 目录。请注意, 这一次将仅传输更改过的日志文件。

```
[student@servera ~]$ rsync -av root@serverb:/var/log ~/serverlogs
root@serverb's password: redhat
receiving incremental file list
log/messages
log/secure
log/audit/audit.log

sent 3,496 bytes received 27,243 bytes 8,782.57 bytes/sec
total size is 11,502,695 speedup is 374.21
```

- 5. 从 servera 退出。

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

完成

在 workstation 上, 运行 **lab archive-sync finish** 脚本来完成本练习。

```
[student@workstation ~]$ lab archive-sync finish
```

本引导式练习到此结束。

▶ 开放研究实验

归档和传输文件

任务执行清单

在本实验中，您将使用 **tar**、**rsync** 和 **scp** 归档和备份目录内容。

成果

您应能够：

- 将远程目录同步到本地目录。
- 为同步目录的内容创建存档。
- 将存档安全地复制到远程主机。
- 提取存档。

在你开始之前

在 **workstation** 上，以 **student** 用户身份并使用密码 **student** 进行登录。

从 **workstation**，运行 **lab archive-review start** 命令。该命令将运行一个起始脚本，它将确定主机 **servera** 和 **serverb** 是否可从网络访问。该脚本还确保 **serverb** 和 **workstation** 上不存在将要在实验中创建的文件和目录。

```
[student@workstation ~]$ lab archive-review start
```

1. 在 **serverb** 上，将 **servera** 的 **/etc** 目录树同步到 **/configsync** 目录。
2. 使用 **gzip** 压缩方式，创建包含 **/configsync** 目录的内容的存档 **configfile-backup-servera.tar.gz**。
3. 以 **student** 用户身份并使用密码 **student**，将 **serverb** 上的 **/root/configfile-backup-servera.tar.gz** 存档文件安全地复制到 **workstation** 上的 **/home/student** 目录。
4. 在 **workstation** 上，将 **/home/student/configfile-backup-servera.tar.gz** 存档的内容提取到 **/tmp/savedconfig/** 目录。
5. 在 **workstation** 上，返回到 **student** 主目录。

```
[student@workstation savedconfig]$ cd
```

评估

在 **workstation** 上，运行 **lab archive-review grade** 脚本来确认是否成功完成本实验。

```
[student@workstation ~]$ lab archive-review grade
```

完成

在 workstation 上，运行 **lab archive-review finish** 脚本来完成本练习。

```
[student@workstation ~]$ lab archive-review finish
```

本实验到此结束。

▶ 解决方案

归档和传输文件

任务执行清单

在本实验中，您将使用 **tar**、**rsync** 和 **scp** 归档和备份目录内容。

成果

您应能够：

- 将远程目录同步到本地目录。
- 为同步目录的内容创建存档。
- 将存档安全地复制到远程主机。
- 提取存档。

在你开始之前

在 **workstation** 上，以 **student** 用户身份并使用密码 **student** 进行登录。

从 **workstation**，运行 **lab archive-review start** 命令。该命令将运行一个起始脚本，它将确定主机 **servera** 和 **serverb** 是否可从网络访问。该脚本还确保 **serverb** 和 **workstation** 上不存在将要在实验中创建的文件和目录。

```
[student@workstation ~]$ lab archive-review start
```

1. 在 **serverb** 上，将 **servera** 的 **/etc** 目录树同步到 **/configsync** 目录。

- 1.1. 使用 **ssh** 命令，以 **student** 用户身份登录 **serverb**。

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

- 1.2. 使用 **su** 命令切换到 **root** 用户，因为创建 **/configsync** 目录需要超级用户特权。在后面的步骤中，您将归档 **/etc** 目录树中的文件，这些文件归 **root** 用户所有，这也需要超级用户特权。

```
[student@serverb ~]$ su -  
Password: redhat  
[root@serverb ~]#
```

- 1.3. 创建 **/configsync** 目录，以存储从 **servera** 同步的配置文件。

```
[root@serverb ~]# mkdir /configsync
```

- 1.4. 使用 **rsync** 命令，将 servera 上的 **/etc** 目录树同步到 serverb 上的 **/configsync** 目录。

请注意，只有 **root** 用户才能读取 servera 上 **/etc** 目录中的所有内容。

```
[root@serverb ~]# rsync -av root@servera:/etc /configsync
root@servera's password: redhat
receiving incremental file list
etc/
etc/.pwd.lock
...output omitted...
etc/yum/protected.d -> ../dnf/protected.d
etc/yum/vars -> ../dnf/vars

sent 10,958 bytes received 21,665,987 bytes 3,334,914.62 bytes/sec
total size is 21,615,767 speedup is 1.00
```

2. 使用 **gzip** 压缩方式，创建包含 **/configsync** 目录的内容的存档 **configfile-backup-servera.tar.gz**。

- 2.1. 使用 **tar** 命令及 **-czf** 选项，以创建一个 **gzip** 压缩存档。

```
[root@serverb ~]# tar -czf configfile-backup-servera.tar.gz /configsync
tar: Removing leading `/' from member names
[root@serverb ~]#
```

- 2.2. 使用 **tar** 命令及 **-tzf** 选项来列出 **configfile-backup-servera.tar.gz** 存档的内容。

```
[root@serverb ~]# tar -tzf configfile-backup-servera.tar.gz
...output omitted...
configsync/etc/vimrc
configsync/etc/wgetrc
configsync/etc/xattr.conf
```

3. 以 **student** 用户身份并使用密码 **student**，将 serverb 上的 **/root/configfile-backup-servera.tar.gz** 存档文件安全地复制到 workstation 上的 **/home/student** 目录。

```
[root@serverb ~]# scp ~/configfile-backup-servera.tar.gz \
student@workstation:/home/student
...output omitted...
student@workstation's password: student
configfile-backup-servera.tar.gz          100% 5110KB 64.5MB/s   00:00
```

4. 在 workstation 上，将 **/home/student/configfile-backup-servera.tar.gz** 存档的内容提取到 **/tmp/savedconfig/** 目录。

- 4.1. 从 serverb 退出。

```
[root@serverb ~]# exit  
logout  
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation]$
```

4.2. 创建 **/tmp/savedconfig** 目录, **/home/student/configfile-backup-servera.tar.gz** 存档的内容将提取到其中。

```
[student@workstation ~]$ mkdir /tmp/savedconfig
```

4.3. 更改为 **/tmp/savedconfig** 目录。

```
[student@workstation ~]$ cd /tmp/savedconfig  
[student@workstation savedconfig]$
```

4.4. 使用 **tar** 命令及 **-tzf** 选项来列出 **configfile-backup-servera.tar.gz** 存档的内容

```
[student@workstation savedconfig]$ tar -tzf ~/configfile-backup-servera.tar.gz  
...output omitted...  
configsSync/etc/vimrc  
configsSync/etc/wgetrc  
configsSync/etc/xattr.conf
```

4.5. 使用 **tar** 命令及 **-xzf** 选项, 将 **/home/student/configfile-backup-servera.tar.gz** 存档的内容提取到 **/tmp/savedconfig/** 目录。

```
[student@workstation savedconfig]$ tar -xzf ~/configfile-backup-servera.tar.gz  
[student@workstation savedconfig]$
```

4.6. 列出该目录树, 以验证目录中是否包含来自 **/etc** 目录的文件。

```
[student@workstation savedconfig]$ ls -lR  
.:  
total 0  
drwxr-xr-x. 3 student student 17 Feb 13 10:13 configsSync  
  
.configsSync:  
total 12  
drwxr-xr-x. 95 student student 8192 Feb 13 09:41 etc  
  
.configsSync/etc:  
total 1212  
-rw-r--r--. 1 student student 16 Jan 16 23:41 adjtime  
-rw-r--r--. 1 student student 1518 Sep 10 17:21 aliases  
drwxr-xr-x. 2 student student 169 Feb 4 21:58 alternatives  
...output omitted...
```

5. 在 workstation 上，返回到 student 主目录。

```
[student@workstation savedconfig]$ cd
```

评估

在 workstation 上，运行 **lab archive-review grade** 脚本来确认是否成功完成本实验。

```
[student@workstation ~]$ lab archive-review grade
```

完成

在 workstation 上，运行 **lab archive-review finish** 脚本来完成本练习。

```
[student@workstation ~]$ lab archive-review finish
```

总结

在本章中，您学到了：

- **tar** 命令可从一组文件和目录创建存档文件，从存档中提取文件，以及列出存档的内容。
- **tar** 命令提供一组不同的压缩方式，用于缩小存档的大小。
- 除了提供安全远程 shell 外，SSH 服务也提供 **scp** 和 **sftp** 命令，作为与运行 SSH 服务器的远程系统来回传输文件的安全方式。
- **rsync** 命令可安全高效地在两个目录之间同步文件，其中任一目录可以在远程系统上。

章 14

安装和更新软件包

目标

从红帽和 Yum 软件包存储库下载、安装、更新和管理软件包。

培训目标

- 将系统注册到您的红帽帐户，并使用红帽订阅管理为其分配软件更新和支持服务的权利。
- 说明如何以 RPM 软件包形式提供软件，并使用 Yum 和 RPM 调查系统上安装的软件包。
- 使用 **yum** 命令查找、安装和更新软件包。
- 启用和禁用红帽或第三方 YUM 存储库。
- 说明如何借助模块安装特定版本的软件，如何列出、启用和切换模块流，以及如何从模块安装和更新软件包。

章节

- 注册系统以获取红帽支持（及测验）
- 解释和调查 RPM 软件包（及测验）
- 使用 Yum 安装和更新软件包（及引导式练习）
- 启用 Yum 软件存储库（及引导式练习）
- 管理软件包模块流（及引导式练习）

实验

安装和更新软件包

注册系统以获取红帽支持

培训目标

学完本节后，您应能够将系统注册到您的红帽帐户，并使用红帽订阅管理为其分配软件更新和支持服务的权利。

红帽订阅管理

红帽订阅管理提供可用于向计算机授权产品订阅的工具，让管理员能够获取软件包的更新，并且跟踪系统所用支持合同和订阅的相关信息。PackageKit 和 yum 等标准工具可以通过红帽提供的内容分发网络获取软件包和更新。

可以通过红帽订阅管理工具执行下列四项基本任务：

- **注册**系统，将该系统与某一红帽帐户关联。这可以让订阅管理器唯一地清查该系统。不再使用某一系统时，可以取消注册。
- **订阅**系统，授权它获取所选红帽产品的更新。订阅包含特定的支持级别、到期日期和默认存储库。可以通过工具自动附加，或选择具体的授权。随着需求的变化，可以移除订阅。
- **启用存储库**，以提供软件包。默认情况下每一订阅会启用多个存储库，但可以根据需要启用或禁用更新或源代码等其他存储库。
- **审核和跟踪**可用或已用的授权。可以在具体系统上本地查看订阅信息，也可在红帽客户门户的订阅页面或订阅资产管理器 (SAM) 查看具体帐户的订阅信息。

注册系统

将系统注册到红帽客户门户的方法有很多种。您可以使用 GNOME 应用程序或通过 Web 控制台服务访问相应的图形界面，也可以使用命令行工具。

要将系统注册到 GNOME 应用程序，可通过选择 Activities 来启动红帽订阅管理器。在 Type to search... 字段中键入 subscription，再单击红帽订阅管理器。在系统提示身份验证时输入适当的密码。这将显示以下 Subscriptions 窗口：



图 14.1: 红帽订阅管理器主窗口

要注册系统，请单击 Subscriptions 窗口中的 Register 按钮。这将显示以下对话框：

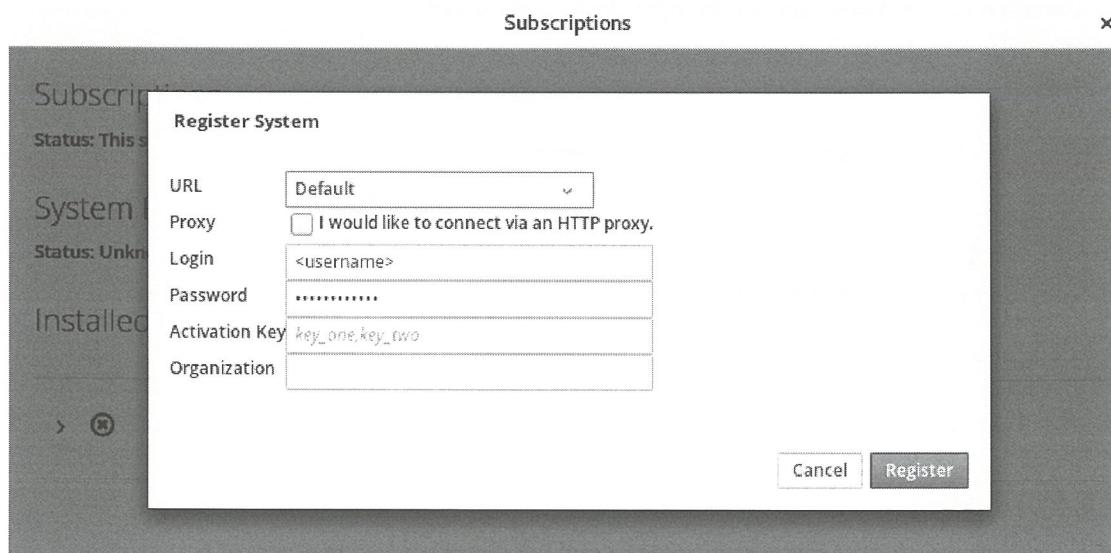


图 14.2: 红帽订阅管理器的服务位置和帐户信息对话框

此对话框将系统注册到订阅服务器。默认情况下，它将服务器注册到红帽客户门户。提供应注册到该系统的红帽客户门户帐户的登录名和密码，再单击 Register 按钮。

注册后，系统会自动附加订阅（如果有）。

完成系统注册并分配了订阅后，关闭 Subscriptions 窗口。系统现在已订阅妥当，准备好从红帽接收更新和/或安装新软件。

从命令行注册

通过 **subscription-manager(8)** 可在不使用图形环境的前提下注册系统。**subscription-manager** 命令可以自动将系统关联到最适合该系统的兼容订阅。

- 注册系统到红帽帐户：