Archive and transfer files

Linux #redhat #files

Compress and un-Compress Files

The `tar` command

- `tar` takes a bunch of files together and put it in one container, or one file you could say
- It is just like in Windows, you have a bunch of files and you zip them together.
- So most likely the `tar` files does not compress as much as the actual `gzip` command.

How do you put your entire home directory into one file?
Example using `tar` command:

```
[user@localhost ~]$ tar cvf macc.tar .
```

- `tar` is an archiving utility
- The `c` option creates a new archive.
- The `v` option specifies to verbosely list files processed.
- The `f` option specifies to use archive file or device ARCHIVE, If this option is not given, `tar` will first examine the environment variable TAPE'. If it is set, its value will be used as the archive name. Otherwise, `tar` will assume the compiled-in default.
- In this example the name of the file that everything will come into will be "macc.tar". This name can be whatever the name you want to give it, but we have to put `.tar` at the end because that will identify that this is a `tar` file.
- The entire `/home/user` directory is being tared, because we are using the relative path symbol, which is a dot (`.`). `tar cvf macc.tar /home/user` would have the same effect.
    - The English translation for this line would be: "Put everything that you see right now into a file called macc.tar"

Output:

```
...
./uwu/
./findpath
./listings
./errfile
./.viminfo
./myfirstfile
./.myfirstfile.swp
./permissiontestfile
./.ssh/
./.ssh/id_rsa
./.ssh/id_rsa.pub
tar: ./macc.tar: file is the archive; not dumped
```

- When you run it, a lot of messages will come up on the screen, some of the messages that you'll see, tar file is archived.
- The last message is because is trying to tar itself.

Output from `ls -ltr`:

```
...
-rw-r--r--. 1 mmarin mmarin       103 Jun  6 14:14 myfirstfile
-rw-r--r--. 1 root   root          0 Jun 12 00:50 permissiontestfile
-rw-r--r--. 1 mmarin mmarin 480419840 Jun 18 14:40 macc.tar
```

- A new 'macc.tar' file appeared in the users home directory.
- Note it is highlighted in red.

How to untar a `tar` file:

- Meaning taking out all the contents from this file and putting them into a directory.
- It is recommended to create create a new blank directory and move this file to that directory where we will be putting all the contents from the `tar` file, so it does not blend with other files.

Example using `tar` command:

```
[mmarin@linuxtest blah]$ tar xvf macc.tar
```

- The `x` option of the `tar` command stands for "extract" and specifies to Extract files from an archive.
- Note we are located in a blank directory called 'blah'.

Output:

```
...
./myfirstfile
./.myfirstfile.swp
./permissiontestfile
./.ssh/
./.ssh/id_rsa
./.ssh/id_rsa.pub
```

- It'll come through all these untarring files.
- Now you should see the same stuff that is in your home directory but now in the 'blah' directory.

The `gzip` command

- It compresses a file
- For this example, the size of the user home directory tared is the following:
  - `-rw-r--r--. 1 mmarin mmarin 480419840 Jun 18 14:40 macc.tar`
    - 480419840 bytes.

Example using `gzip` command:

```
[mmarin@linuxtest blah]$ gzip macc.tar
```

Output from `ls -l macc.tar.gz

```
-rw-r--r--. 1 mmarin mmarin 263804317 Jun 18 14:40 macc.tar.gz
```

- Note the size of the file is a lot lighter.
- Note that a '.gz' was appended to the name of the file.
- Now this file is compressed.

The `gzip -d` OR `gunzip` command

- These two commands are used to un-compress a file.

Example using `gzip -d` command:

```
[mmarin@linuxtest blah]$ gzip -d macc.tar.gz
```

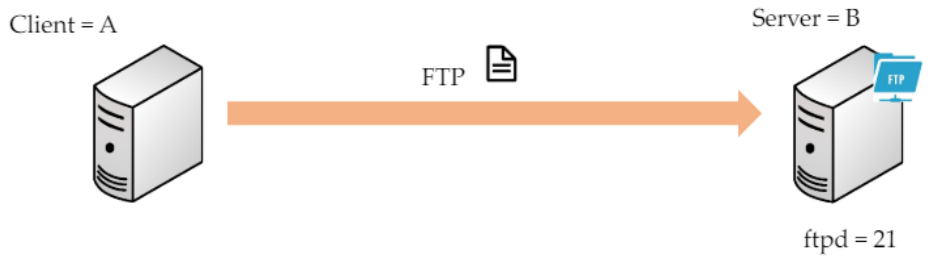
Output from `ls -l macc.tar`

```
-rw-r--r--. 1 mmarin mmarin 480419840 Jun 18 14:40 macc.tar
```

- The file has ben un-compressed.
- Note the name extension '.gz' has disappeared.

FTP - File Transfer Protocol

- The File Transfer Protocol is a standard network protocol used for the transfer of computer files between a client and server on a computer network. FTP is built on a client-server model architecture using separate control and data connections between the client and the server. (Wikipedia)
- Protocol = Set of rules used by computers to communicate
- Default FTP Port = 21



- If the recipient software is not running any kind of listening service to receive that protocol, then its gonna reject that transfer.
  - Make sure it is running the FTP daemon, service, or application.

Install and Configure FTP on the remote server

- Become root

- `rpm –qa | grep ftp`
- `ping www.google.com`
- `yum install vsftpd`
- `vi /etc/vsftpd/vsftpd.conf` (make a copy first)
- **Find the following lines and make the changes as shown below:**
- • ## Disable anonymous login ##
  - `anonymous_enable=NO`
- • ## Uncomment ##
  - `ascii_upload_enable=YES`
  - `ascii_download_enable=YES`
- • ## Uncomment - Enter your Welcome message - This is optional ##
  - `ftpd_banner=Welcome to UNIXMEN FTP service.`
- • ## Add at the end of this file ##
  - `use_localtime=YES`
- `systemctl start vsftpd`
- `systemctl enable vsftpd`
- `systemctl stop firewalld`
- `systemctl disable firewalld`
- `useradd mmarin` (if the user does not exist)

Determine if the package has been installed previously:
Example using `rpm` command:

```
[root@localhost ~]# rpm –qa | grep vsftpd
```

- Of there is no output, the package has not been isntalled.

Install vsftpd
Example using `yum install` command:

```
[root@localhost ~]# yum install vsftpd
```

- This command is going to all the repositories and it is check if, hey do you have the 'vsftpd' package that I need to download and then install?
- You usually have to confirm download and installation by typing "y".

Copy the `/etc/vsftpd/vsftpd.conf` file
Example using `cp` command:

```
[root@localhost vsftpd]# cp vsftpd.conf vsftpd.conf.orig
```

- Creates a backup file of the original `/etc/vsftpd/vsftpd.conf` file.

Inside the `/etc/vsftpd/vsftpd.conf` file
Example using `vi` command:

```
[root@localhost vsftpd]# vi vsftpd.conf
```

File editor:

```
...
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=NO
...
# ASCII mangling is a horrible feature of the protocol.
ascii_upload_enable=YES
ascii_download_enable=YES
...
# You may fully customise the login banner string:
ftpd_banner=Welcome to blah FTP service.
...
# Local Time
use_localtime=YES
...
```

- Make the changes previously instructed.
- Remember you can search word in `vi` by typing `/` while not in INSERT mode.
- Now you have to start the service 'vsftpd' using the `systemctl start` command. Then you would probably want to enable the service as well and stop `firewalld` just for training purposes.
  - If you do not want to stop the `firewalld` service you will need to add the rule to allow port 21 as an incoming traffic for the firewall.
- Remember you need a user account for this.

Install and Configure FTP on the client

Install FTP client on the client server

- Become root
- `yum install ftp`
- `su - mmarin`
- `touch kruger` (This is a regular file, you could pick any name)

Commands to transfer file to the FTP server:

- `ftp 192.168.1.x`
- Enter username and password
- bi
- hash
- put kruger
- bye.

Setting `ftp`:

- if you try to run the `ftp` command it will throw a `bash: ftp: command not found...` because the `ftp` client is not installed
  - To install the client run `yum install ftp` and confirm installation and download
- Now after installing the `ftp` package, you can just type it in the command line and hit Enter, and it will bring you into the `ftp` interactive session.

- You can get out of this session by typing `bye` and you'll get your prompt back

Example using `ftp` command:

```
[root@linuxtest vsftpd]# ftp
ftp> bye
```

Transfer a file into the server:

- you need to know the IP address of the server. In this example the IP address of the server will be 192.168.1.58

Example using `ftp` command:

```
[user@localhost ~]$ ftp 192.168.1.58
```

Output:

```
Connected to 192.168.1.58 (192.168.1.58).
220 Welcome to blah FTP service.
Name (192.168.1.58:mmarin):
```

- By default it picked the username 'mmarin' which is the user we settled the server up with. If you wanted to use a different user name you have to specify it here or you can hit enter to go with the default user.
- You'll have to specify the password to that user on that remote machine

Output:

```
Connected to 192.168.1.58 (192.168.1.58).
220 Welcome to blah FTP service.
Name (192.168.1.58:mmarin):
331 Please specify the password.
Password:
230 Login successful.
Remote system is UNIX.
Using binary mode to transfer files.
ftp>
```

Switch to binary mode
Example using `bi` command on `ftp`:

```
ftp> bi
```

- You want to transfer files always using FTP through the binary mode.

Example using `hash` command on `ftp`:

```
ftp> hash
```

- Hash means that when you are doing the transfer, to show you the hashtags which will tell you that there's a progress going on in hashtag progress.

Put the file
Example using `put` command on `ftp`:

```
ftp> put kruger
```

- When its time to transfer you use the command `put` because your are going from A to B.

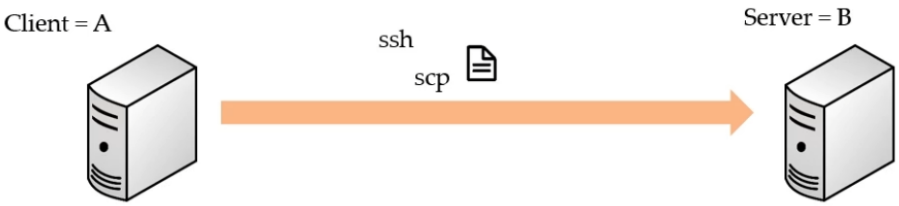
Output:

```
local: kruger remote: kruger
227 Entering Passive Mode (192,168,1,58,149,207).
150 Ok to send data.
I#
226 Transfer complete.
545 bytes sent in 0.000369 secs (1476.96 Kbytes/sec)
```

- Now the file is on the 192.168.1.58 server on the 'mmarin' user home directory.

SCP - Secure Copy Protocol

- The Secure Copy Protocol or “SCP” helps to transfer computer files securely from a local to a remote host. It is somewhat similar to the File Transfer Protocol “FTP”, but it adds security and authentication
  - This protocol is preferred because is more secure.
- Protocol = Set of rules used by computers to communicate
- Default SCP Port = 22 (same as SSH)



- The SCP is transfers files over through the protocol we already have in place. And that is SSH, there is no new protocol for that.
- The service for the SSH daemon should be running under recipient or the remote server in order to accept incoming traffic.

SCP commands to transfer file to the remote server:

- Login as yourself (mmarin)
- `touch jack`
- `scp jack mmarin@192.168.1.x:/home/mmarin`
  - 'mmarin' is the username that actually exists on the remote server at 192.168.1.58 (Example)

- Enter username and password

Example using `scp` command:

```
[user@localhost ~]$  scp jack mmarin@192.168.1.58:/home/mmarin
```

- Transferring the file named 'jack' to the 'mmarin' user home directory at 192.168.1.58 server.

Output:

```
jack                                  100%          22        18.7KB/s        00:00
```