**Week 5 Homework Submission File: Archiving and Logging Data**

Please edit this file by adding the solution commands on the line below the prompt.

Save and submit the completed file for your homework submission.

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

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

*sysadmin@UbuntuDesktop:~/Projects$ sudo tar -xvf TarDocs.tar*

1. Command to **create** the Javaless\_Doc.tar archive from the TarDocs/ directory, while excluding the TarDocs/Documents/Java directory:

*sysadmin@UbuntuDesktop:~/Projects$ sudo tar -cvf Javaless\_Docs.tar --exclude=TarDocs/Documents/Java TarDocs/Documents*

1. Command to ensure Java/ is not in the new Javaless\_Docs.tar archive:

*sysadmin@UbuntuDesktop:~/Projects$ sudo tar -tvf Javaless\_Docs.tar | grep Java*

**Bonus**

* Command to create an incremental archive called logs\_backup\_tar.gz with only changed files to snapshot.file for the /var/log directory:

*sysadmin@UbuntuDesktop:~/Projects$ sudo tar cvvWf logs\_backup.tar.gz --listed-incremental=logs\_backup.snar --level=0 /var/log*

*Q: Not sure how to trigger change and see it in .snar file?*

*tar --list --incremental --verbose --verbose --file logs\_backup.tar.gz*

**Critical Analysis Question**

* Why wouldn't you use the options -x and -c at the same with tar?

*-x is to expand a tar archive and –c is to compress data or create a tar archive. Using both –x and –c will cause an error.*

**Step 2: Create, Manage, and Automate Cron Jobs**

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

*6 \* \* 3 sudo tar -cvfz /auth\_backup.tgz.gzip /var/log/auth.log*

**Step 3: Write Basic Bash Scripts**

1. Brace expansion command to create the four subdirectories:

*sysadmin@UbuntuDesktop:~$ mkdir ~/Backups/{freemem,diskuse,openlist,freedisk}*

1. Paste your system.sh script edits below:
2. #!/bin/bash

[Your solution script contents here]

*Command to make the system.sh#!/bin/bash*

*echo `free -h > Backups/freemem/free\_mem.txt`*

*echo `df -H --output=used > Backups/diskuse/disk\_usage.txt`*

*echo `lsof > Backups/openlist/open\_list.txt`*

*echo `df -H --output=avail > Backups/freedisk/free\_disk.txt`*

1. script executable:

*sysadmin@UbuntuDesktop:~$ bash system.sh*

**Optional**

* Commands to test the script and confirm its execution:

*sysadmin@UbuntuDesktop:~/Backups$ more diskuse/disk\_usage.txt*

*sysadmin@UbuntuDesktop:~/Backups$ more freedisk/free\_disk.txt*

*sysadmin@UbuntuDesktop:~/Backups$ more freemem/free\_mem.txt*

*sysadmin@UbuntuDesktop:~/Backups$ more openlist/open\_list.txt*

**Bonus**

**sysadmin@UbuntuDesktop:~$ sudo chmod 744 system.sh**

* Command to copy system to system-wide cron directory:

*crontab –e (use nano to edit crontab add the following line to my crontab)*

*#weekly cronjobs:*

*\* 0 \* \* 7 sudo ./system.sh*

**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.

* + Add your config file edits below:

*[/var/log/auth.log {*

*missingok*

*rotate 7*

*weekly*

*compress*

*delaycompress*

*notifempty*

*}]*

**Bonus: Check for Policy and File Violations**

1. Command to verify auditd is active:

sysadmin@UbuntuDesktop:~$ systemctl | grep auditd

sysadmin@UbuntuDesktop:~$ sudo auditctl -l

[sudo] password for sysadmin:

sudo: auditctl: command not found

sysadmin@UbuntuDesktop:~$ systemctl status auditd

Unit auditd.service could not be found.

sysadmin@UbuntuDesktop:~$ sudo apt install auditd

Reading package lists... Done

1. Command to set number of retained logs and maximum log file size:
   * Add the edits made to the configuration file below:

*sysadmin@UbuntuDesktop:~$ nano auditd.conf (update the following line)*

*[max\_log\_file = 35*

*num\_logs = 7]*

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

*sysadmin@UbuntuDesktop:/etc$ sudo nano /etc/audit/rules.d/audit.rules*

* + Add the edits made to the rules file below:

*man auditd*

*man audit.rules*

*[w /etc/shadow -p wra -k hashpass\_audit*

*-w /etc/passwd -p w -k userpass\_audit*

*-w /var/log/auth.log -p w -k authlog\_audit]*

1. Command to restart auditd:

*sysadmin@UbuntuDesktop:~$ systemctl restart auditd*

1. Command to list all auditd rules:

*sysadmin@UbuntuDesktop:/etc$ sudo auditctl -l*

*No rules*

1. Command to produce an audit report:

*sysadmin@UbuntuDesktop:/etc$ aureport*

1. Create a user with sudo useradd attacker and produce an audit report that lists account modifications:

*sysadmin@UbuntuDesktop:/etc$ sudo tail /etc/passwd*

*adam:x:1007:1009::/home/adam:/bin/sh*

*billy:x:1008:1010::/home/billy:/bin/sh*

*http:x:1009:1011::/home/http:/bin/sh*

*sally:x:1010:1012::/home/sally:/bin/sh*

*postfix:x:123:127::/var/spool/postfix:/usr/sbin/nologin*

*jane:x:1011:1013::/home/jane:/bin/bash*

*hacker:x:1012:1014:,,,:/home/hacker:/bin/bash*

*badguy:x:1013:1015:,,,:/home/badguy:/bin/bash*

*tripwire:x:124:65534::/home/tripwire:/usr/sbin/nologin*

*attacker:x:1014:1016::/home/attacker:/bin/sh*

*sysadmin@UbuntuDesktop:/etc$ sudo ausearch -ua 1014 -i*

1. Command to use auditd to watch /var/log/cron:

*sysadmin@UbuntuDesktop:/etc$ sudo auditctl -w /var/log/cron -p war -k audit-cron*

1. Command to verify auditd rules:

*sysadmin@UbuntuDesktop:/etc$ sudo auditctl –l (run to verify rule was created)*

*-w /var/log/cron -p rwa -k audit-cron*

**Bonus (Research Activity): Perform Various Log Filtering Techniques**

1. Command to return journalctl messages with priorities from emergency to error:

*journalctl -p 3*

1. Command to check the disk usage of the system journal unit since the most recent boot:

*journalctl --disk-usage*

1. Comand to remove all archived journal files except the most recent two:

*journalctl --vacuum-file=2*

1. Command to filter all log messages with priority levels between zero and two, and save output to /home/sysadmin/Priority\_High.txt:

*journalctl -p > /home/sysadmin/Priority\_High.txt*

1. Command to automate the last command in a daily cronjob. Add the edits made to the crontab file below:

*journalctl -b -p emerg..err*

*sudo journalctl -p 0..2*

*Yes I see what you are saying went back and took a closer look - the ellipses is how you set the range*

*-p, --priority=*

*Filter output by message priorities or priority ranges. Takes*

*either a single numeric or textual log level (i.e. between*

*0/"emerg" and 7/"debug"), or a range of numeric/text log levels in*

*the form FROM..TO. The log levels are the usual syslog log levels*

*as documented in syslog(3), i.e. "emerg" (0), "alert" (1),*

*"crit" (2), "err" (3), "warning" (4), "notice" (5), "info" (6),*

*"debug" (7). If a single log level is specified, all messages with*

*this log level or a lower (hence more important) log level are*

*shown. If a range is specified, all messages within the range are*

*shown, including both the start and the end value of the range.*

*This will add "PRIORITY=" matches for the specified priorities.*

0 18 \* \* \* sudo journalctl -p 0..2 > /home/sysadmin/Priority\_High.txt

*or*

*0 18 \* \* \* sudo journalctl -p emerg..crit*