## **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:

|  |
| --- |
| tar -xvf TarDocs.tar |

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

|  |
| --- |
| sudo tar -cvf Javaless\_Doc.tar --exclude=TarDocs/Documents/Java ~/Projects |

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

|  |
| --- |
| tar -tvf Javaless\_Doc.tar | grep Java |

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

|  |
| --- |
| sudo tar --listed-incremental=snapshot.file -cvzf logs\_backup.tar.gz /var/log |

#### Critical Analysis Question

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

-x extracts the archive and -c creates the archive. Using them back to back would counteract itself.

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

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

|  |
| --- |
| ------0 6 \* \* 3 tar -zcvf auth\_backups.tgz /var/log/auth.log |

### **Step 3: Write Basic Bash Scripts**

1. Brace expansion command to create the four subdirectories:

|  |
| --- |
| #!/bin/bash#  Free memory output to a free\_mem.txt file  free -h > ~backups/freemem/free\_mem.txt  #Disk usage output to a disk\_usage.txt file du -h > ~/backups/disuse/disk\_usage.txt  # List open files to a open\_list.txt file lsof > ~/backups/openlist/open\_list.txt  #Free command to disk space to a free\_disk.txt file df -h > ~/backups/freedisk/free\_disk.txt |

1. Paste your system.sh script edits:

|  |
| --- |
| #!/bin/bash  #Free memory output to a free\_mem.txt file free -h > ~backups/freemem/free\_mem.txt  #Disk usage output to a disk\_usage.txt file du -h > ~/backups/disuse/disk\_usage.txt  # List open files to a open\_list.txt file lsof > ~/backups/openlist/open\_list.txt  #Free command to disk space to a free\_disk.txt file df -h > ~/backups/freedisk/free\_disk.txt |

1. Command to make the system.sh script executable:

|  |
| --- |
| chmod +x system.sh |

#### Optional

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

|  |
| --- |
| sudo ./system.sh cat ~/backups/freedisk/ free\_disk.txt |

1. Command to copy system to system-wide cron directory:

|  |
| --- |
| ---sudo cp system.sh /etc/cron.weekly |

### **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.

* 1. Add your config file edits:

|  |
| --- |
| /var/log/auth.log { rotate 7  missingokweeklynotifemptycompressdelaycompressendscript} |

### **Check for Policy and File Violations**

1. Command to verify `auditd` is active:

|  |
| --- |
| auditctl -w |

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

|  |
| --- |
| Seven retained logs. The maximum log file size is 35. |

Add the edits made to the configuration file:

|  |
| --- |
| sudo nano /etc/audit/rules.d/audit.rules |

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

|  |
| --- |
| auditctl -w |

Add the edits made to the rules file below:

|  |
| --- |
| w /etc/shadow -p wra -k hashpass\_audit  -w /etc/passwd -p wra -k userpass\_audit  -w /var/log/auth.log -p wra -k authlog\_audit |

1. Command to restart auditd:

|  |
| --- |
| sudo systemctl restart auditd |

1. Command to list all auditd rules:

|  |
| --- |
| sudo auditctl -l |

1. Command to produce an audit report:

|  |
| --- |
| sudo aureport -au |

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

|  |
| --- |
| sudo aureport -m |

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

|  |
| --- |
| sudo auditctl -w /var/log/cron |

1. Command to verify auditd rules:

|  |
| --- |
| sudo auditctl -l |

#### 

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

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

|  |
| --- |
| sudo journalctl -priority=0..3 or sudo journalctl -b -p emerg..err |

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

|  |
| --- |
| sudo journalctl -b -u --disk-usage | less or sudo journalctl -b -u system-journald | less |

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

|  |
| --- |
| sudo journalctl --vacuum-time=2d or sudo 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:

|  |
| --- |
| sudo journalctl -p=0..2 > /home/sysadmin/Priority\_High.txt |