Commands cheat-sheet

Ch.1 UFW

sudo ufw default deny incoming
sudo ufw default allow outgoing
sudo iptables -L
sudo ufw enable
sudo iptables -L

sudo ufw allow ssh
sudo ufw deny http
sudo ufw allow http/s

Port ranges:
sudo ufw allow 6000:6007/tcp
Specific Ips:
sudo ufw allow from <ip>/<mask>

Ch.2 Directory and file security basics

Creating users and groups:

sudo groupadd <name>
getent group | grep <name>

sudo useradd -m -u <uid> -g <group> -G <groups,with,commas> angelos
sudo passwd angelos

sudo useradd -e YYYY – MM – DD username (to set exp date for a user's account)
sudo chage -l username (to verify the exp date)

sudo usermod -a -G group username (Add a user to a secondary group)
sudo usermod -G group username (Change user's primary group)
sudo usermod -e YYYY – MM – DD username

Basic UNIX permissions, users and groups:

The owner of a file, or group the file belongs to can be seen with the ls -l command.
To change one of the two, use the command:

chown <user> <file>
chown :<group> <file

Three permission groups: User, Group, Other
Three permission types: Read, Write, Execute
Count in 3-bit binary to assign permissions to each one of the three types:
r w x → 1 1 0 = 6
Ex.

600, the user can read and write on the file but not execute it.
660, both the user and the group can read and write but not execute.
777, everyone can read, write and execute the file.

To modify a file’s permissions use:

chmod <3-digit-number[0-7]> <file>
Alternatively:
chmod u+rwx <file>
chmod g+rwx <file>

Working with ACLs:

sudo apt install acl

Create a demo file to put ACLs on:
touch acl_demo.txt
chmod 460 acl_demo.txt
getfacl acl_demo.txt

To modify ACL policies use
setfacl -m (modify) u:user:rwx <file>
setfacl -m (modify) g:group:rwx <file>

To create an ACL for a directroy:
setfacl -m d:u:frank:r directroy

Removing ACLs:
setfacl -x u:maggie acl_demo.txt
Deletes the entire thing, if we have multiple permissions it is not recommended

setfacl -m m::r acl_demo.txt
Creates a mask and adds different “effective” permissions to users affected by the ACL. This works on top of the normal ACL. It effectively does a bitwise AND on the permission bits.

Ch.3 The lazy sysadmin’s tools

Creating systemd services:

touch /etc/systemd/system/<name>.service #Write the unitfile

[Unit]
Description=Example Service
After=network.target
StartLimitIntervalSec=0

[Service]
Type=simple
Restart=always
RestartSec=1
User=serviceuser
ExecStartPre=
ExecStart=/path/to/executable [options]
ExecStop=
ExecReload=

[Install]
WantedBy=multi-user.target
sudo systemctl daemon-reload
sudo systemctl enable <name>
sudo service <name> start

Executable file content:

#!/bin/sh

echo “Hello from test service!”

Managing tasks with crontab

crontab -e (to modify the cron table for a user)
https://crontab.guru/ to visualize the format in plain english

Viewing audit logs, making scripts to parse data from log files.

Ch.4 Introduction to snort, and auditing/scanning with Lynis and rkhunter

Lynis:

```
cd /opt/
wget https://downloads.cisofy.com/lynis/lynis-2.6.6.tar.gz
tar xvzf lynis-2.6.6.tar.gz
mv lynis /usr/local/
ln -s /usr/local/lynis/lynis /usr/local/bin/lynis
lynis audit system
```

RootkitHunter

```
sudo apt install rkhunter
rkhunter -c
```

Snort:

```
sudo apt install snort
snort --version
sudo vim /etc/snort/snort.conf ipvar HOME_NET <ip addr output on eth0>

wget https://www.snort.org/downloads/community/community-rules.tar.gz
sudo tar -xvzf community-rules.tar.gz -C /etc/snort/rules

sudo ip link set enp0s3 promisc on

sudo snort -d -l /var/log/snort/ -h 192.168.1.0/24 -A console -c /etc/snort/snort.conf
```