**Question**

**Create a service with the name showtime , after starting the service, every minute it should print the current time in a file in the user home directory.**

**Ex:-**

**sudo service showtime start -> It should start writing in file.**

**sudo service showtime stop -> It should stop writing in file.**

**sudo service showtime status -> It should show status.**

**My Approach**

To achieve this, I create a simple bash script that writes the current time to a file every minute, and then create a systemd service to manage the execution of that script.

**Step 1 : Become root**

**Command = sudo -i**

**Step 2 : Go to path /usr/local/bin/**

**Command = cd /usr/local/bin/**

**Step 3 : Create a file named showtime.sh and write following**

**script**

**Command = nano Showtime.sh**

**Script**

**#!/bin/bash**

**while true; do**

**date >> /home/sigmoid/time\_service**

**sleep 60**

**Done**

**Step 4 : Give execute permission to execute this file**

**Command = chmod a+x showtime.sh**

**Step 5 : Go to path /root directory and then go to path /etc/systemd/system and create a file named showtime.service**

**Command = cd ..**

**cd /etc/systemd/system**

**nano showtime.services**

**Step 6 : Write this things in showtime.service and save it.**

**[Unit]**

**Description=Showtime Service**

**After=network.target**

**[Service]**

**Type=simple**

**ExecStart=/usr/local/bin/showtime.sh**

**Restart=always**

**[Install]**

**WantedBy=multi-user.target**

Note - Here we can see that we have passed the path of our script showtime.sh in ExecStart container.

**Step 7 : Now reload systemd**

**Command = sudo systemctl daemon-reload**

**Step 8 : Now start showtime.service**

**Command = sudo systemctl start showtime.service**

**Step 9 : Now you can see that in our desired given path (here it is /home/sigmoid/time\_service) after starting the service, every 1 minute it append/print the current time.**

**Step 10 : To stop the service otherwise it will run forever.**

**command = sudo systemctl stop showtime.service .**