Emrecan Atlıhan

Ödev-1)

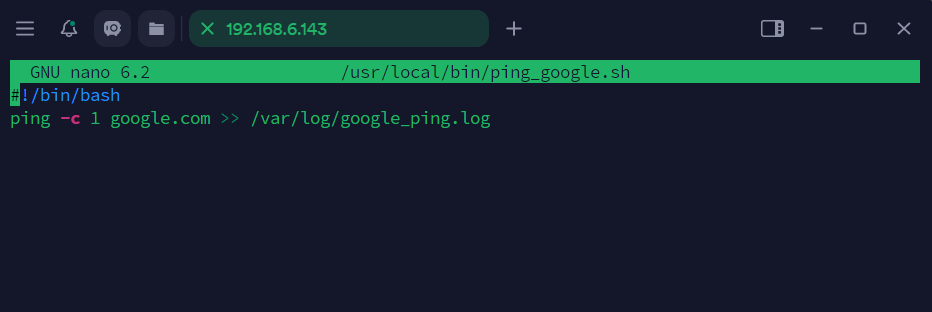
Crontab nedir?

**Crontab**, Linux ve Unix tabanlı sistemlerde belirli zamanlarda veya belirli aralıklarla otomatik olarak çalıştırılacak görevleri yönetmek için kullanılan bir araçtır. "Cron" adı verilen bir arka plan hizmeti tarafından desteklenir ve sistem yöneticileri tarafından çeşitli zamanlanmış görevleri otomatikleştirmek amacıyla kullanılır.

Başlangıç için öncelikle google’a ping atan bir bash script kodu yazalım. Bu koda çalışma izni verelim.

sudo nano /usr/local/bin/ping\_google.sh

sudo chmod +x /usr/local/bin/ping\_google.sh

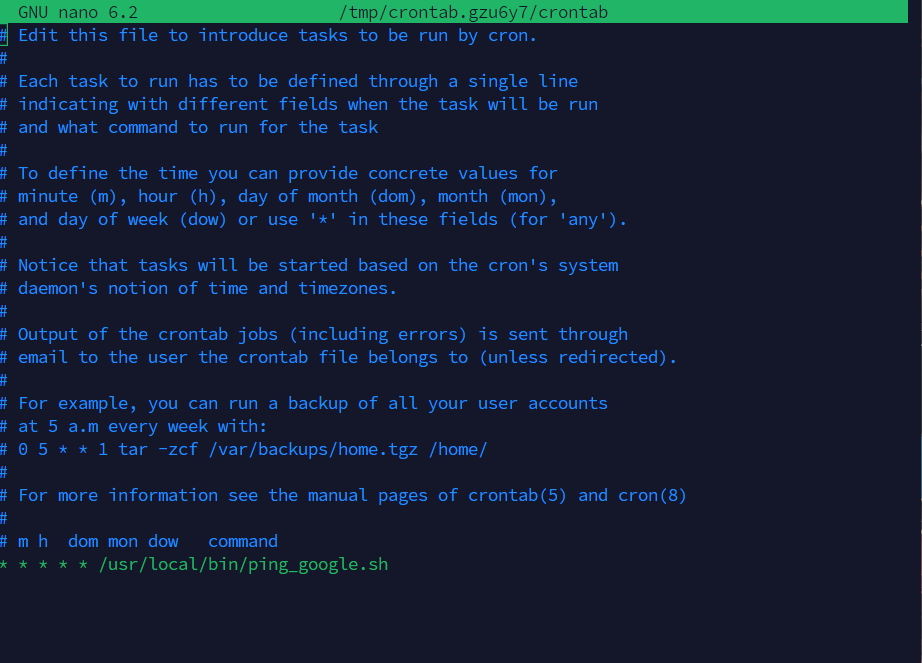


Daha sonra log dosyalarının kayıt olacağı log dosyasını oluşturmak lazım.

sudo touch /var/log/google\_ping.log >> /var/log/google\_ping.log son kısım dosyaya yazdırmak için

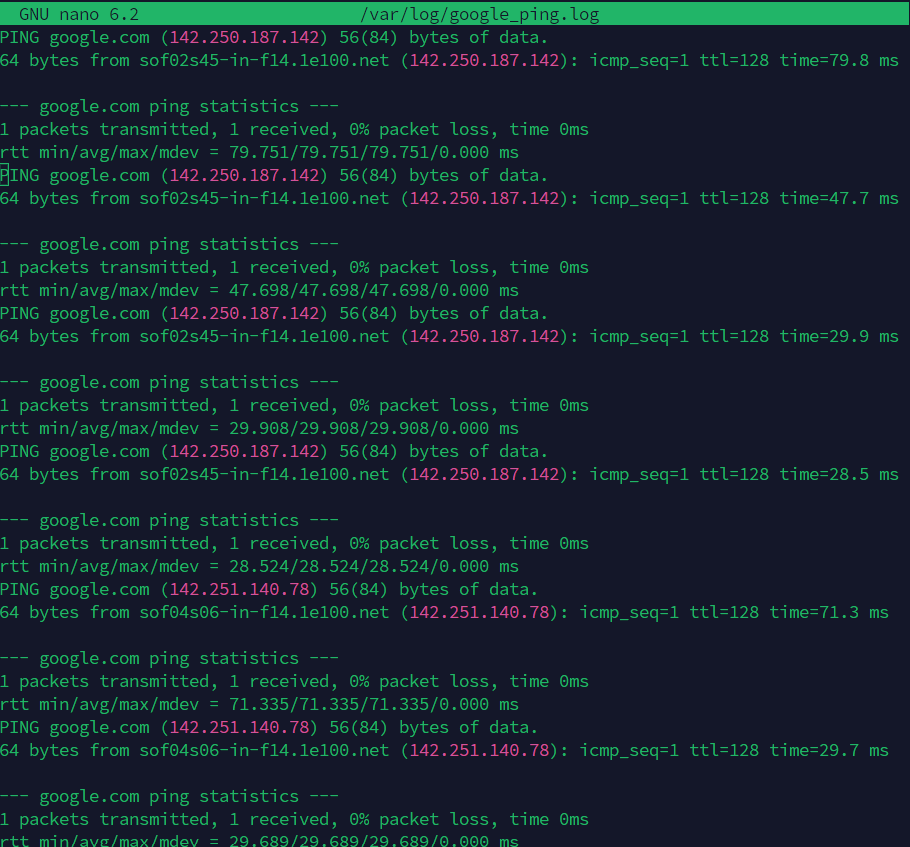
sudo chmod 666 /var/log/google\_ping.log -----> tüm kullanıcılara okuma yetkisi verir (666).

Daha sonra crontabi crontab –e yazrak açalım.



Yorum satırlarında crontabin nasıl çalıştığından bahsedilmektedir. Çalıştıracağımız komutun dosya yolunu belirterek başına zamanlayıcı koyuyoruz. Örnekte sabah 5’te her hafta kullanıcı dosyaları yedeklensin diyor. Bizde zamanlayıcı (\* \* \* \* \*) oludğu için dakika başı ping atıyor.

Son olarak log dosyamızı okuyarak çalışıp çalışmadığını kontrol edelim.



Düzgünce çalışıyor.

Ödev 2)

Öncelikle script dosyasını oluşturalım.

sudo nano /usr/local/bin/system\_monitor.sh

İçine kodumuzu yazalım. Kod için GPT den yardım aldım. Kodun tam olarak ne anlama geldiğini bilmek için GPT’ye yorumlattım.

#!/bin/bash

# CPU kullanım yüzdesi

cpu\_usage=$(top -bn1 | grep "Cpu(s)" | awk '{print $2 + $4}')

* -b (Batch Mode): top komutunu batch modunda çalıştırır. Batch modu, çıktıyı terminal ekranına yazdırmak yerine satır satır verir, bu da komutun çıktılarını bir dosyaya veya başka bir komuta yönlendirmek için daha uygundur. Normalde top komutu interaktif modda çalışır ve sürekli olarak güncellenen bir ekran gösterir. Ancak batch modunda, top tek bir anlık görüntü alır ve çıktı verir, ardından hemen çıkar.
* -n1: Kaç kez çıktının alınacağını belirtir. -n parametresiyle birlikte kullanılır ve 1 sayısı top komutunun yalnızca bir defa çıktı almasını sağlar. Bu, top komutunun tek bir anlık görüntüsünü alıp çıktı olarak döndürmesi anlamına gelir.

# RAM kullanım miktarı (GB)

ram\_usage=$(free -g | awk '/Mem:/ {print $3}')

free komutu, sistemin hafıza (RAM) kullanımını özetlemek için kullanılır ve çeşitli birimlerde çıktılar sağlayabilir. -g gigabyte cinsinden gösterir.

# Disk doluluk yüzdesi

disk\_usage=$(df -h / | awk 'NR==2 {print $5}')

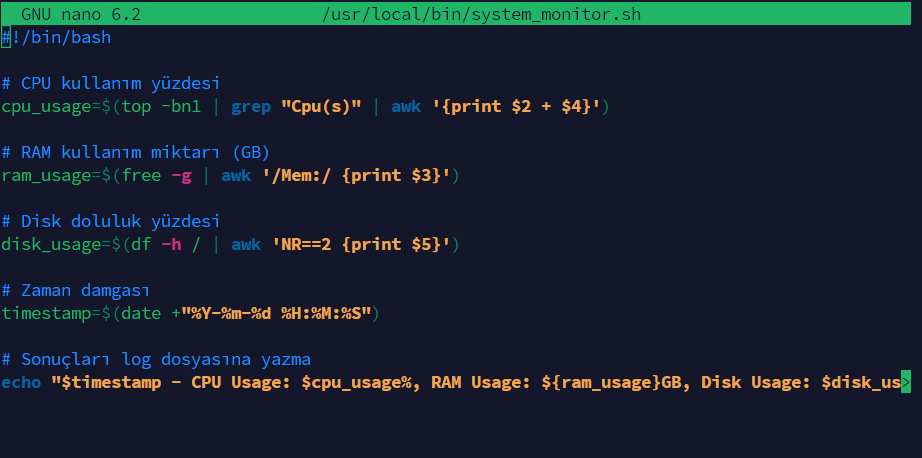
free komutundaki -h seçeneği, hafıza kullanımını "insan tarafından okunabilir" (human-readable) formatta görüntülemek için kullanılır. Bu seçenek, verileri en uygun birimde (KB, MB, GB, vb.) göstererek çıktıyı daha kolay anlaşılır hale getirir.

# Zaman damgası

timestamp=$(date +"%Y-%m-%d %H:%M:%S")

# Sonuçları log dosyasına yazma

echo "$timestamp - CPU Usage: $cpu\_usage%, RAM Usage: ${ram\_usage}GB, Disk Usage: $disk\_usage" >> /var/log/system\_monitoring.log



sudo chmod +x /usr/local/bin/system\_monitor.sh scriti çalıştırılabilir yapmak için komutumuz.

sudo touch /var/log/system\_monitoring.log

sudo chmod 666 /var/log/system\_monitoring.log

Logların girileceği dosya ve okuma izinleri.

Geriye kalan kısmı 2 şekilde yapılabiliyor. Birincisi iki servis ile ikincisi tek servis ile yapılıyor. Tek servis şu şekilde oluyor.

[Unit]

Description=System Monitoring Service

[Service]

ExecStart=/bin/bash -c 'while true; do /usr/local/bin/system\_monitor.sh; sleep 60; done'

[Install]

WantedBy=multi-user.target

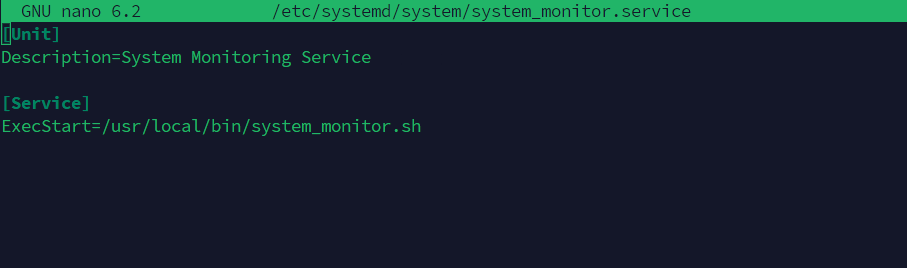
İki servis ile ise yapılışı şu şekilde:

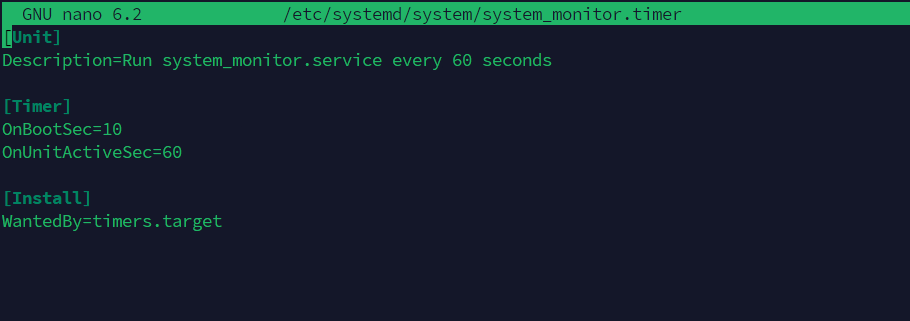
sudo nano /etc/systemd/system/system\_monitor.service

sudo nano /etc/systemd/system/system\_monitor.timer

İlk serviste hangi kodun çalışacağı yazıyor ikinci serviste ise zamanlayıcısı var.

İlk servisin içeriği:

 İkinci servisin içeriği:



Konfigürasyonu yenileyip sistemleri başlatma vakti.

sudo systemctl daemon-reload

sudo systemctl start system\_monitor.timer

sudo systemctl enable system\_monitor.timer

**start**: Timer'ı başlatır.

**enable**: Timer'ı sistem başlangıcında otomatik olarak başlatılacak şekilde ayarlar.

### **Install Bölümünün Alanları**

#### **1. WantedBy**

* **Açıklama**: Bu alan, servisin veya zamanlayıcının hangi hedefe (target) bağlı olduğunu belirtir. Bu hedef, servisin veya zamanlayıcının sistem tarafından otomatik olarak başlatılmasını sağlar.

### **Timer Bölümündeki Alanlar ve Anlamları**

#### **1. OnBootSec**

* **Açıklama**: Sistemin başlangıç anından itibaren geçen süreyi belirtir. Sistemin açılmasından belirli bir süre sonra zamanlayıcıyı başlatır.

#### **2. OnUnitActiveSec**

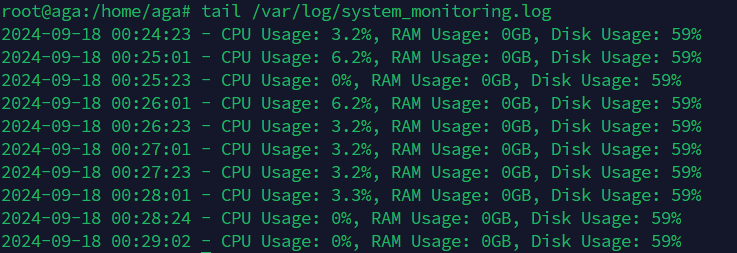
* **Açıklama**: Belirli bir servis veya zamanlayıcının aktif olduğu zamandan itibaren geçen süreyi belirtir. Bu süre tamamlandığında zamanlayıcı tekrar çalışır.

**Unit Bölümünün Alanları**

### **1. Description**

* **Açıklama**: Servisin veya zamanlayıcının kısa bir açıklamasını sağlar. Bu açıklama, systemctl komutuyla servislerin veya zamanlayıcıların listelenmesi sırasında görünür.

Log dosyamıza bakalım.



Çalışıyor.