

ÖN BİLGİ

Bu belge

- https://www.syslogs.org/docs/Zamanlanmis_Gorverler.doc

resmi adresindeki veya linkin kırık olması ihtimaline karşın alternatif olarak

- https://www.includekarabuk.com/kitaplik/indirmeDeposu/Siber_Guvenlik_Teknik_Makaleler/Teori/BaskalarinaAitMakaleler/Syslogs.org_10_Zamanlanmis_Gorevler.pdf

adresindeki makaleye çalışılarak elde edilen notlarımı kapsamaktadır. Bu çıkarılan notlar belgemde alıntılar ve/veya kişisel ilavelerim mevcuttur.

1)

Linux'ta belirli aralıklarla yapılması gereken görevler, zamanı gelince otomatik olarak çalıştırılacak şekilde yapılandırılabilir. RedHat linux sistem ile ilgili görevleri yönetebilmek için bazı komutları zamanı gelince otomatik olarak çalıştıracak şekilde hazırlamıştır. Redhat Linux 3 adet zamanlanmış görev aracı ile gelir.

- at
- cron
- batch

Cron

cron belirli bir zamanda, ayın belirli gününde, haftanın belirli gününde veya haftada bir çeşitli görevleri çalıştırmak için tasarlanmış bir **Daemon** dur.

cron, sistemin daima açık olduğunu farzeder. Eğer bir görevin çalıştırılacağı zamanda sistem açık değilse, sistem açılınca zamanı geçmiş komut çalıştırılmaz.

Cron ile zamanlanan görevler yeri gelince periyodik olarak çalıştırılır. Bir görev belirtilen zamanda sadece 1 defalığına çalıştırılsın isteniyorsa at veya batch komutları kullanılmalıdır. Cron servisinin çalışabilmesi için vixie-cron RPM paketi kurulmuş olmalıdır. Bu paketin kurulu olup olmadığını denetlemek için

```
> rpm -q vixie-cron
```

komutunu kullanabilirsiniz. Cron RPM paketi kurulu ise sistemde çalışıyor olmalıdır. Servisin durumunu kontrol etmek için aşağıdaki komut kullanılabilir.

```
> service cron status
```

Output:

```
[ ok ] cron is running.
```

Aşağıda /etc/crontab dosyası görülmektedir.

```
SHELL=/bin/bash  
PATH=/sbin:/bin:/usr/sbin:/usr/bin  
MAILTO=root
```

```
HOME=/
# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

İlk dört satır zamanlanmış görevlerin kullanacağı kabuk değişkenlerini tanımlamaktadır.

- SHELL değişkeni zamanlanmış görevin hangi komut yorumlayıcı ile çalıştırılacağını gösterir.
- PATH değişkeni ise zamanlanmış görevdeki komutların hangi klasörlerde bulunduğunu gösterir.
- MAILTO değişkeni cron görevlerinin çıktısının hangi kullanıcıya mail olarak gönderileceğini belirtir. MAILTO="" şeklinde yazılırsa hiçbir kullanıcıya mail gitmez.
- HOME değişkeni ise zamanlanmış komutun ev dizini olarak nereyi algılaması gerektiği gösterilir.

Bundan sonraki satırların tümünün başında yer alan sayısal ifadeler ve devamı aşağıdaki tarih tanımlamalarını belirtir.

```
minute hour day month dayofweek command
```

```
minute    — 0 ile 59 arası tamsayı
hour      — 0 ile 23 arası tamsayı
day       — 1 ile 31 arası tamsayı (seçilen aya uygun olarak yazılmalıdır, örneğin bazı aylarda
          31 gün yoktur)
month     — 1 den 12 ye kadar tamsayı
dayofweek — 0 dan 7 ye kadar tamsayı (0 ya da 7 Pazar gününü belirtir.)
command   — çalıştırılacak olan komut
```

Yukarıdaki her bir alan için (*) işareti o alanın alabileceği tüm değerleri simgeler. Örneğin ay için * konmuşsa bu tüm aylarda çalıştırılacak anlamına gelir.

Redhat Linux

- Saatlik /etc/cron.hourly,
- Günlük /etc/cron.daily,
- Haftalık /etc/cron.weekly
- Aylık /etc/cron.monthly

görevler için hazır görev klasörleri oluşturmuştur. Örneğin hergün çalıştırılmasını istediğimiz bir komut **/etc/cron.daily/** klasörüne kabuk program olarak yerleştirilebilir.

Örnek cron içerikleri:

i)

```
# her pazartesi sabah 3:30 da sistemin bellek kullanımını  
# /tmp/meminfo adlı bir metin dosyasına yazan program  
30 3 * * 1 root cat /proc/meminfo >> /tmp/meminfo
```

ii)

```
# Her ayın ilk günü sabah 4:10 da yedek alan script  
10 4 1 * * root /root/scripts/backup.sh
```

(page 2-4)

2)

At ve Batch

cron komutu periyodik olarak çalıştırılacak komutlar için kullanılırken at ve batch komutları ise istenen bir zamanda bir kereliğine çalıştırılacak komutlar içindir. At ya da batch komutunu kullanabilmek için at RPM paketi yüklü olmalıdır.

```
> rpm -q at // at paketinin yüklü olup olmadığını sorgular. ( -q | --query )
```

rpm paketi yüklü ise at DAEMON programının çalışıyor olduğu kontrol edilmelidir.

```
> service atd status
```

Output:

```
[ ok ] atd is running.
```

At Komutu Kullanımı

at komutu belirli bir zamanda bir defalığına belirli bir komutu veya komutları çalıştırmakta kullanılır. Aşağıdaki komut saat 1:05 de ev dizininin içindekilerini liste.txt dosyasına yazdırıyor.

```
root@kali:# > at 1:05
```

```
At > ls -l > /home/root/liste.txt
```

```
At > // (Boş bir satır enter'lyoruz ve CTRL+D (EOF) ile komut satırından çıkıyoruz)
```

```
Job 1 at 2015-11-10 11:03
```

Bu örnekte oluşturulan at işlemi 1 numaralı iş olarak iş listesine kaydedilmiştir. İş listesini görmek için atq komutu kullanılır.

```
> atq
```

Output:

```
1      Sun Dec 20 01:05:00 2015 a root
```

Bir at işinin zamanı geldiğinde çalıştırılır ve kuyruktan silinir. Oluşturduğumuz bir at işini zamanı gelmeden önce silmek için atrm komutu kullanılır.

```
> atrm 1
```

(page 5-6)