ÖN BİLGİ

Bu belge

- https://www.syslogs.org/docs/Surecler.pdf

resmi adresindeki veya linkin kırık olması ihtimaline karşın alternatif olarak

- https://www.includekarabuk.com/kitaplik/indirmeDeposu/Siber_Guvenlik_Teknik_Makaleler/Teori/BaskalarinaAitMakaleler/Syslogs.org_5_Surecler.pdf

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

1)

Bir kullanıcıya ait süreçlerin listesini görebilmek için ps komutunu **(u)** parametresi ile kullanıyoruz.

> ps -u hefese

Process'lerle beraber işlemci ve bellek kullanımlarını da görmek istiyorsak:

> ps -u hefese -u

(Page 2)

2)

top komutu ile ps komutunun verdiği çıktıları anlık olarak değişimli izleyebiliriz.

> top

Sonuçları güncelleme süresi varsayılan olarak 3 saniyedir. Bunu -d parametresi ile değiştirebiliriz.

> top -d 10

Sadece belirli bir kullanıcının process'lerini görüntülemek için:

> top -u hefese

(Page 5)

3)

disown komutu bir kabuğa (terminale) bağlı olan bir process'in o kabuğa (terminale) bağımlı olmaktan kurtarır. Böylelikle kabuk (terminal) ölse dahi process çalışmaya devam eder.

(page 8)

4)

Her an çalışan süreçlerden biri veya birkaçı beklenmedik döngüye girebilir. Bunun sonucu olarak sistemin kaynaklarını, özellikle hafızayı tüketici bir duruma gelebilir. Bu tür kısır döngüye giren süreçleri bulup eğer hayati önem taşımıyorlarsa `öldürmek' gerekir. Süreci öldürmekten kasıt programı tamamen durdurarak sistemle ilişkisini kesmektir. Bu sayede programın hafızada kapladığı bölge serbest kalacak, çekirdek de hafıza düzenlemesini tekrar yaparak başka süreçlere daha fazla yer ayıracaktır. Bir süreci öldürmek için kill komutu kullanılır.

```
> kill 67                    // 67 nolu pid'ye sahip process öldürülür.
```

Eğer yukarıdaki komut işe yaramazsa daha kökten çözüm -9 parametresidir.

```
> kill -9 67
```

Process'leri pid numaralarına gerek duymadan isimleriyle öldürebilmek için killall komutu kullanılır.

```
> killall httpd
```