

## Sinyallerin bir listesi ve ne anlama geldikleri

- Bunların hepsi bir Linux i386 sisteminden kaydedilmiştir. Sayılar platformlar arasında değişiklik gösterebilir.
- Linux, bizim ilgilenmediğimiz gerçek zamanlı sistemi için 34-64 sinyallerini kullanır.
- "man 7 signal" sinyaller hakkındaki resmi kılavuz sayfasını verir.
- Bu oldukça kapsamlı bir sinyal listesidir. Bunların sadece bir kısmı aşağıdaki bağlamlarda ortaya çıkacaktır  
Program yap.

Hayır.	Kısa isim	Ne anlama geliyor
1	SIGHUP	Eğer bir süreç terminalden çalıştırılıyorsa ve terminal aniden ortadan kaybolursa, süreç bu sinyali alır. "HUP" "hang <b>up</b> " kelimesinin kısaltmasıdır ve telefon modemlerinin kullanıldığı günlerde telefonun kapatılması anlamına gelir.
2	SIGINT	İşlem "yarıda kesildi". Bu, kontrol terminalinde Control+C tuşlarına bastığınızda gerçekleşir.
3	SIGQUIT	
4	SIGILL	Yasadışı komut. Program, CPU'nun anlayamayacağı bazı makine kodları içeriyordu.
5	SIGTRAP	Bu sinyal esas olarak hata ayıklayıcılar ve program izleyicileri içinde kullanılır.
6	SIGABRT	Program <code>abort()</code> fonksiyonunu çağırdı. Bu bir acil durdurmadır.
7	SIGBUS	Belleğe yanlış bir şekilde erişilmeye çalışıldı. Bunun nedeni bellek erişimindeki hizalama hataları vb. olabilir.
8	SIGFPE	Programda bir kayan nokta istisnası meydana geldi.
9	SIGKILL	Süreç, <code>kill</code> komutunu kullanan biri tarafından açıkça öldürüldü. Program.
10	SIGUSR1	Programcılara istediklerini yapmaları için bırakıldı.
11	SIGSEGV	Süreç tahsis edilmemiş belleğe erişilmeye çalışıldı. Bu genellikle dizilerin sonunun okunmasından vb. kaynaklanır.
12	SIGUSR2	Programcılara istediklerini yapmaları için bırakıldı.
13	SIGPIPE	Bir süreç, bir <b>boru</b> ("üretici   tüketici") aracılığıyla onu tüketen başka bir sürece beslenen bir çıktı üretiyorsa ve tüketici ölürse, üreticiye bu sinyal gönderilir.
14	SIGALRM	Bir süreç, <code>alarm()</code> fonksiyonunu çağırarak gelecekte bir zamanda işletim sisteminden bir "uyandırma çağrısı" talep edebilir. O zaman geldiğinde uyandırma çağrısı bu sinyalden oluşur.
15	SIGTERM	Süreç, <code>kill</code> komutunu kullanan biri tarafından açıkça öldürüldü. Program.
16	kullanılmayan	
17	SIGCHLD	Süreç daha önce bir veya daha fazla alt süreç oluşturmuştu. <code>fork()</code> işlevi. Bu süreçlerden biri veya daha fazlası o zamandan beri ölmüştür.

18	SIGCONT	(SIGSTOP ile birlikte okunmalıdır.) Bir süreç SIGSTOP gönderilerek duraklatılmışsa, sürece SIGCONT gönderilmesi onu tekrar uyandırır ("devam ettirir").
19	SIGSTOP	(SIGCONT ile birlikte okunmalıdır.) Bir süreç SIGSTOP gönderilirse işletim sistemi tarafından duraklatılır. Tüm
<b>Hayır.</b>	<b>Kısa isim</b>	<b><i>Ne anlama geliyor</i></b>
		durumu yeniden başlatılmaya hazır olarak korunur (SIGCONT tarafından) ancak o zamana kadar daha fazla CPU döngüsü almaz.
20	SIGTSTP	Esasen SIGSTOP ile aynıdır. Bu, kullanıcı terminalde Control+Z tuşlarına bastığında gönderilen sinyaldir. (SIGTSTP "terminal durdurma"nın kısaltmasıdır) SIGTSTP ve SIGSTOP arasındaki tek fark duraklamanın sadece SIGTSTP için varsayılan eylemdir ancak SIGSTOP için gerekli eylemdir. Süreç SIGTSTP'yi farklı şekilde ele almayı seçebilir ancak SIGSTOP ile ilgili herhangi bir seçeneği yoktur.
21	SIGTTIN	İşletim sistemi, terminalinden girdi okumaya çalıştığında arka planda çalışan bir sürece bu sinyali gönderir. Tipik yanıt duraklatmak (SIGSTOP ve SIFTSTP'ye göre) ve süreç ön plana geri getirildiğinde gelen SIGCONT'u beklemektir.
22	SIGTTOU	İşletim sistemi, terminaline çıktı yazmaya çalıştığında arka planda çalışan bir sürece bu sinyali gönderir. Tipik yanıt SIGTTIN'e göredir.
23	SIGURG	İşletim sistemi, ağ bağlantısı kullanan bir sürece "acil" bant dışı veri gönderildiğinde bu sinyali gönderir.
24	SIGXCPU	İşletim sistemi bu sinyali CPU limitini aşan bir sürece gönderir. Make 'i çalıştırmadan önce "ulimit -t unlimited" kabuk komutu ile herhangi bir CPU limitini iptal edebilirsiniz, ancak make 'te CPU limitine ulaşırsanız bir şeylerin yanlış gitmiş olması daha olasıdır.
25	SIGXFSZ	İşletim sistemi, dosya boyutu sınırının üzerinde bir dosya oluşturmaya çalışan bir işleme bu sinyali gönderir. make komutunu çalıştırmadan önce "ulimit -f unlimited" kabuk komutu ile herhangi bir dosya boyutu sınırını iptal edebilirsiniz, ancak make 'te dosya boyutu sınırına ulaşırsanız bir şeylerin yanlış gitmiş olması daha olasıdır.
26	SIGVTALRM	Bu SIGALRM'ye çok benzer, ancak SIGALRM belirli bir gerçek zaman geçtikten sonra gönderilirken, SIGVTALRM süreci çalıştırmak için belirli bir süre harcandıktan sonra gönderilir.
27	SIGPROF	Bu da SIGALRM ve SIGVTALRM'e çok benzer, ancak SIGALRM belirli bir miktar gerçek zaman geçtikten sonra gönderilirken, SIGPROF süreci çalıştırmak ve süreç adına sistem kodunu çalıştırmak için belirli bir süre harcandıktan sonra gönderilir.
28	SIGWINCH	(Bugünlerde çoğunlukla kullanılmıyor.) Bir süreç, pencerelerinden biri yeniden boyutlandırıldığında bu sinyali gönderirdi.
29	SIGIO	(SIGPOLL olarak da bilinir.) Bir süreç, işlemesi için hazır bir girdi olduğunda veya bir çıkış kanalı yazılmaya hazır hale geldiğinde bu sinyalin kendisine gönderilmesini ayarlayabilir.

30	SIGPWR	Gücün kısa süreli bir acil durum güç kaynağına geçtiğini belirtmek için bir güç yönetimi hizmeti tarafından süreçlere gönderilen bir sinyal. Süreç (özellikle uzun süre çalışan daemonlar) acil durum gücü kesilmeden önce cleanlt'i kapatmaya özen gösterebilir.
31	SIGSYS	Kullanılmadı.