

Linux Inspector Komutları

Seyitahmet Genç

Augustos, 2021

İçindekiler

1	Giriş	11
2	Komutlar	12
2.1	dstat Komutu	12
2.1.1	dstat Komutu Yardım Sayfası	13
2.1.2	Başlıca OS Bileşenlerinin İstatistiklerini Görüntülemek	14
2.1.3	Process ve Memory İstatistiklerini Görüntülemek	15
2.1.4	İşlemleri CPU Kullanımına Göre Sıralı Bir Şekilde Görüntülemek	16
2.1.5	İşlemleri Memory Kullanımına Göre Sıralı Bir Şekilde Görüntülemek	17
2.1.6	İşlemlere Ait Tüm İstatistikleri Listelemek	18
2.1.7	Eklentileri Listelemek	19
2.1.8	Renksiz Çıktı Almak	19
2.2	mpstat Komutu	20
2.2.1	mpstat Komutu Yardım Sayfası	20
2.2.2	CPU İstatistiklerini Görüntüleme	20
2.2.3	İstatistikleri CPU Numaralarıyla Birlikte Görüntüleme	21
2.2.4	İstatistikleri Spesifik Bir CPU'dan Görüntülemek	21
2.2.5	İstatistikleri Belirli Bir Zaman Aralığıyla Görüntülemek	22
2.2.6	Tüm İstatistikleri Görüntülemek	22
2.3	numastat Komutu	23

2.3.1	numastat Komutu Yardım Sayfası	23
2.3.2	Genel İstatistikleri Görüntülemek	24
2.3.3	İstatistikleri Minimize Edilmiş Sütunlarla Görüntülemek	25
2.3.4	İstatistikleri Sıralı Bir Şekilde Görüntülemek	25
2.3.5	meminfo Tarzı Sistem Geneli RAM Kullanım İstatistiklerini Görüntülemek	26
2.3.6	Verilen Bir İşleme Ait İstatistikleri Görüntülemek	27
2.3.7	Birden Çok İşleme Ait İstatistikleri Görüntülemek	27
2.4	perf Komutu	28
2.4.1	perf Komutu Yardım Sayfası	28
2.4.2	Sembolik Eventlerin Listelenmesi	29
2.4.3	Performans İstatistiklerinin Görüntülenmesi	30
2.4.4	Gerçek Zamanlı Sistem Profilinin Görüntülenmesi	31
2.5	pidstat Komutu	32
2.5.1	pidstat Komutu Yardım Sayfası	32
2.5.2	Genel İstatistiklerin Görüntülenmesi	33
2.5.3	I/O İstatistiklerinin Görüntülenmesi	34
2.5.4	Sayfa Hataları ve Hafıza Kullanımının Görüntülenmesi	35
2.6	/proc/cpuinfo Dosyası	36
2.7	rdmsr Komutu	37
2.8	sar Komutu	38
2.8.1	sar Komutu Yardım Sayfası	38

2.8.2	CPU İstatistiklerinin Görüntülenmesi	39
2.8.3	İstatistiklerin Dosyaya Yazılması	40
2.8.4	İstatistiklerin Dosyadan Okunması	40
2.8.5	Hafıza Kullanım İstatistiklerinin Görüntülenmesi	41
2.8.6	Sayfalama İstatistiklerinin Görüntülenmesi	41
2.8.7	Blok Cihaz İstatistiklerinin Görüntülenmesi	42
2.8.8	Ağ İstatistiklerinin Görüntülenmesi	43
2.9	<code>tiptop</code> Komutu	44
2.9.1	<code>tiptop</code> Komutu - Komut Satırı Seçenekleri	45
2.9.2	<code>tiptop</code> Komutu - Live-Mode Modu Komutları	46
2.10	<code>top</code> Komutu	47
2.10.1	Spesifik Bir Kullanıcıya Ait İşlemlerin Görüntülenmesi	49
2.10.2	<code>top</code> Komutu Seçenekleri	50
2.11	<code>vmstat</code> Komutu	51
2.11.1	<code>vmstat</code> Komutu Yardım Sayfası	52
2.11.2	Hafızaya Ait Tüm İstatistikleri Görüntülemek	53
2.11.3	Fork Sayısını Görüntüleme	53
2.11.4	Event Sayaçlarına Ait Verileri Görüntüleme	54
2.11.5	Disk İstatistiklerini Görüntüleme	55
2.12	<code>dmesg</code> Komutu	56
2.12.1	<code>dmesg</code> Komutu Kullanım Seçenekleri	58

2.13 lsmod Komutu	60
2.14 lsof Komutu	62
2.14.1 lsof Komutu Yardım Sayfası	63
2.14.2 Spesifik Bir Kullanıcıya Ait Dosyaların Görüntülenmesi	64
2.14.3 Spesifik Bir İşleme Ait Dosyaların Görüntülenmesi	65
2.14.4 lsof Komutu Kullanım Seçenekleri	65
2.15 lspci Komutu	66
2.15.1 lspci Komutu Kullanım Seçenekleri	66
2.16 nvidia-smi Komutu	67
2.16.1 nvidia-smi Komutu Kullanım Seçenekleri	67
2.17 /proc/devices Dosyası	68
2.18 /proc/softirqs Dosyası	70
2.19 ltrace Komutu	71
2.19.1 ltrace Komutu Yardım Sayfası	72
2.19.2 ltrace Komutu Kullanım Seçenekleri	73
2.20 lttng Komutu	74
2.20.1 lttng Komutu Kullanım Seçenekleri	75
2.21 ps Komutu	76
2.21.1 ps Komutu Kullanım Seçenekleri	77
2.22 stap Komutu	78
2.22.1 stap Komutu Kullanım Seçenekleri	79

2.23 strace Komutu	80
2.23.1 strace Komutu Yardım Sayfası	82
2.23.2 Yapılan Sistem Çağrısı Sayısının Görüntülenmesi	83
2.23.3 Spesifik Sistem Çağrılarının Görüntülenmesi	83
2.23.4 strace Komutu Kullanım Seçenekleri	84
2.24 sysdig Komutu	85
2.24.1 Spesifik İşlemlerin Filtrelenmesi	86
2.24.2 Sistem İşlemlerinin Listelenmesi	87
2.24.3 Ağ Bağlantılarının Listelenmesi	88
2.24.4 İşlemlerin CPU Kullanım Oranlarına Göre Sıralı Gösterilmesi	88
2.25 blktrace Komutu	89
2.25.1 blktrace Komutu Kullanım Seçenekleri	89
2.25.2 blktrace Komutu Örnek Kullanımlar	90
2.26 df Komutu	91
2.26.1 df Komutu Yardım Sayfası	91
2.26.2 Boyutların 1024'ün Katı Olarak Gösterilmesi	92
2.26.3 Boyutların 1000'in Katı Olarak Gösterilmesi	92
2.26.4 df Komutu Kullanım Seçenekleri	93
2.27 du Komutu	94
2.27.1 Çıktının İnsan Okunabilir Formatta Üretilmesi	94
2.27.2 Çıkıyla Dosya ve Dizinlerin Dahil Edilmesi	95

2.27.3 Çıktıya Toplam Kullanımın Dahil Edilmesi	96
2.27.4 Sadece Toplam Kullanımın Çıktı Olarak Üretilmesi	96
2.28 find Komutu	97
2.28.1 find Komutu Yardım Sayfası	97
2.28.2 İsim ile Arama Yapma	98
2.28.3 Boş Dosya ve Dizinlerin Aranması	98
2.28.4 İzinler ile Arama Yapılması	99
2.29 iostat Komutu	100
2.29.1 iostat Komutu Kullanım Seçenekleri	102
2.30 iotop Komutu	103
2.30.1 iotop Komutu Yardım Sayfası	104
2.30.2 I/O İşlemi Yapan İşlemlerin Görüntülenmesi	105
2.30.3 Spesifik Bir İşleme Ait Bilgilerin Görüntülenmesi	105
2.30.4 iotop Komutu Kullanım Seçenekleri	105
2.31 swapon Komutu	106
2.31.1 swapon Komutu Kullanım Seçenekleri	106
2.32 jcmand Komutu	107
2.33 jdb Komutu	108
2.33.1 jdb İçerisinde Kullandılabile Komutlar	109
2.34 jinfo Komutu	111
2.34.1 jinfo Komutu Kullanım Seçenekleri	111

2.35 jmap Komutu	112
2.35.1 jmap Komutu Kullanım Seçenekleri	112
2.36 jps Komutu	113
2.36.1 jps Komutu Kullanım Seçenekleri	113
2.37 jrunscript Komutu	114
2.37.1 jrunscript Komutu Kullanım Seçenekleri	114
2.38 jstack Komutu	115
2.38.1 jstack Komutu Kullanım Seçenekleri	115
2.39 jstat Komutu	116
2.40 ethtool Komutu	117
2.40.1 ethtool Komutu Yardım Sayfası	118
2.40.2 NIC Tarafından Kullanılan Sürücünün Görüntülenmesi	119
2.40.3 NIC İstatistiklerinin Görüntülenmesi	119
2.41 ip Komutu	120
2.41.1 ip Komutu Yardım Sayfası	120
2.41.2 Tüm Ağ Cihazlarıyla İlişkilendirilmiş Tüm IP Adreslerinin Görüntülenmesi	121
2.41.3 Link Layer Bilgilerinin Görüntülenmesi	122
2.41.4 Route Bilgilerinin Görüntülenmesi	123
2.41.5 ip Komutuna Ait Kullanım Seçenekleri	123
2.42 iperf Komutu	124
2.42.1 iperf Komutu Kullanım Seçenekleri	124

2.43 iptraf Komutu	125
2.43.1 iptraf Komutu Kullanım Seçenekleri	125
2.44 lldptool Komutu	126
2.44.1 lldptool Komutu Kullanım Seçenekleri	126
2.45 netcat Komutu	127
2.45.1 Port Dinleme İşlemleri	128
2.45.2 Dosya Kopyalama İşlemleri	129
2.45.3 Bağlantı Modunun UDP Olarak Değiştirilmesi	129
2.46 netstat Komutu	130
2.46.1 netstat Komutunun Kullanım Seçenekleri	130
2.47 nmap Komutu	135
2.47.1 nmap Komutu Yardım Sayfası	136
2.47.2 Hostname ya da IP Adresi ile Tarama Yapılması	137
2.47.3 nmap Komutu Kullanım Seçenekleri	138
2.48 /proc/net/bonding Dizini	139
2.49 snmpget Komutu	139
2.49.1 snmpget Komutu Örnek Kullanımları	139
2.50 tcpdump Komutu	140
2.50.1 tcpdump Komutu Yardım Sayfası	141
2.50.2 tcpdump Komutu Kullanım Seçenekleri	141
2.51 telnet Komutu	142

2.51.1 telnet Komutu Kullanım Seçenekleri	142
2.52 free Komutu	143
2.52.1 free Komutuna ait Kullanım Seçenekleri	144
2.53 /proc/meminfo Dosyası	145

1 Giriş

Bu belgede linux sistemlerde sistem yöneticilerinin sıklıkla kullandığı, sistemin CPU, RAM gibi donanımlarının performanslarını ve istatistiklerini görüntülemek, ağ ile ilgili ayarlarını yönetmek gibi işlemleri gerçekleştirebilmek için kullandıkları komutlar örnek kullanımları ile birlikte anlatılmıştır.

Belgede geçen ekran görüntüleri sanal makine üzerinde çalışan bir linux sisteme aittir. Dolayısıyla bazı komutların çıktıları fiziksel makine üzerinde vereceği çıktılara göre farklılık gösterebilir.

2 Komutlar

2.1 dstat Komutu

dstat, ağ bağlantıları, I/O cihazları ve CPU gibi sistem bileşenlerinden bilgi veya istatistik almak için kullanılan bir komuttur. Genellikle sistem yöneticileri tarafından yukarıda bahsedilen konular hakkında bilgi sahibi olmak için kullanılır. vmstat, netstat, iostat gibi komutların ayrı ayrı üreteceği çıktılar dstat komutu ile üretilebilir. dstat komutunun sahip olduğu birkaç özellik şu şekilde sıralanabilir:

- vmstat, netstat, iostat, ifstat ve mpstat araçlarının ürettiği bilgileri birleştirir.
- İstatistikleri aynı anda görüntüleyebilir.
- Gruplandırılmış blok/ağ araçlarını özetleyebilir.
- Farklı ünitelerin farklı renklerde gözükmeyi sağlayan renkli çıktı özelliğini destekler.
- Cihaz başına kesintileri görüntüleyebilir.
- Kesin birimleri gösterir ve dönüştürme hatalarını minimuma indirmeye çalışır.
- Sistem stresli olduğunda zaman kayması yaşatmaz.

2.1.1 dstat Komutu Yardım Sayfası

Bu işlem için komut "**dstat -help**" şeklinde kullanılmalıdır. Bu komut, dstat komutunun kullanımı ile alakalı bilgileri göstermektedir. Komuta ait örnek bir çıktı şekil 1 üzerinde gösterilmektedir.

```
[~] dstat --help
Usage: dstat [-afv] [options..] [delay [count]]
Versatile tool for generating system resource statistics

Dstat options:
  -c, --cpu          enable cpu stats
  -C 0,3,total      include cpu0, cpu3 and total
  -d, --disk         enable disk stats
  -D total,hda      include hda and total
  -g, --page         enable page stats
  -i, --int          enable interrupt stats
  -I 5,eth2         include int5 and interrupt used by eth2
  -l, --load         enable load stats
  -m, --mem          enable memory stats
  -n, --net          enable network stats
  -N eth1,total     include eth1 and total
  -p, --proc         enable process stats
  -r, --io           enable io stats (I/O requests completed)
  -s, --swap         enable swap stats
  -S swap1,total    include swap1 and total
  -t, --time         enable time/date output
  -T, --epoch        enable time counter (seconds since epoch)
  -y, --sys          enable system stats

  --aio             enable aio stats
  --fs, --filesystem enable fs stats
  --ipc             enable ipc stats
  --lock            enable lock stats
  --raw              enable raw stats
  --socket          enable socket stats
  --tcp              enable tcp stats
  --udp              enable udp stats
  --unix             enable unix stats
  --vm               enable vm stats
```

Şekil 1: dstat -help Komutu

2.1.2 Başlıca OS Bileşenlerinin İstatistiklerini Görüntülemek

Bu işlem için komut "dstat" şeklinde kullanılmalıdır. Bu komut, sisteme ait CPU, disk, ağ, sayfalama ve sistem istatistiklerini göstermektedir. Komuta ait örnek bir çıktı şe²kil 2 üzerinde gösterilmektedir.

[~] dstat													
----total-cpu-usage----					-dsk/total-		-net/total-		---paging--		---system--		
usr	sys	idl	wai	hiq	siq	read	writ	recv	send	in	out	int	csw
2	3	87	7	0	0	990k	62k	0	0	0	0	339	421
1	0	99	0	0	0	0	0	0	0	0	0	86	134
1	1	99	0	0	0	0	132k	60B	0	0	0	152	244
1	1	99	0	0	0	0	0	60B	0	0	0	98	172
0	0	100	0	0	0	0	0	120B	42B	0	0	88	149
1	1	98	0	0	1	0	0	60B	0	0	0	353	652
2	1	98	0	0	0	0	0	60B	0	0	0	459	831
1	0	99	0	0	0	0	0	60B	0	0	0	144	187
1	1	99	0	0	0	0	0	60B	0	0	0	111	158
1	1	98	0	0	0	0	0	60B	0	0	0	541	1001
1	1	98	0	0	0	0	0	60B	0	0	0	271	515
0	0	100	0	0	0	0	0	60B	0	0	0	66	107
1	0	99	0	0	0	0	36k	150B	90B	0	0	87	147
0	1	99	0	0	0	0	0	60B	0	0	0	92	140
1	1	97	0	1	0	0	48k	60B	0	0	0	170	258
0	0	100	0	0	0	0	0	60B	0	0	0	87	136
1	0	99	0	0	0	0	0	60B	0	0	0	94	145
0	0	100	0	0	0	0	0	60B	0	0	0	76	120
0	0	100	0	0	0	0	0	60B	0	0	0	91	150
1	1	99	0	0	0	0	28k	60B	0	0	0	102	143
0	0	100	0	0	0	0	0	60B	0	0	0	83	132
1	0	99	0	0	0	0	0	60B	0	0	0	77	126
0	0	100	0	0	0	0	0	60B	0	0	0	85	150
0	1	99	0	0	0	0	0	60B	0	0	0	84	121
0	0	100	0	0	0	0	0	60B	0	0	0	117	149
1	0	99	0	0	0	0	0	60B	0	0	0	82	125
0	0	100	0	0	0	0	0	0	0	0	0	86	133
1	1	99	0	0	0	0	0	0	0	0	0	86	132
1	1	99	0	0	0	0	0	0	0	0	0	161	305
2	1	98	0	0	0	0	0	0	0	0	0	617	1172
0	1	99	0	0	0	0	0	0	0	0	0	149	259
1	0	99	0	0	0	0	0	0	0	0	0	81	127
0	0	100	0	0	0	0	0	0	0	0	0	87	139

Sekil 2: dstat Komutu

2.1.3 Process ve Memory İstatistiklerini Görüntülemek

Bu işlem için komut "**dstat --vmstat**" şeklinde kullanılmalıdır. Bu komut, sisteme ait işlem ve hafıza istatistiklerini göstermektedir. Komuta ait örnek bir çıktı şe^{kil 3} üzerinde verilmiştir.

[~] dstat --vmstat		-----memory-usage----- -----paging-- -dsk/total- ---system--- -----total-cpu-usage-----																
run	blk	new	used	buff	cach	free	in	out	read	writ	int	csw	usr	sys	idl	wai	hiq	siq
0	0	1.4	357M	42.3M	459M	3103M	0	0	197k	14k	130	182	1	1	97	1	0	0
0	0	0	357M	42.3M	459M	3103M	0	0	0	0	90	137	1	1	99	0	0	0
0	0	0	357M	42.3M	459M	3103M	0	0	0	0	585	1013	3	1	96	0	1	0
0	0	0	357M	42.3M	459M	3103M	0	0	0	0	437	783	2	1	98	0	0	0
0	0	0	357M	42.3M	459M	3103M	0	0	0	0	106	148	1	1	99	0	0	0
0	0	0	357M	42.3M	459M	3103M	0	0	0	0	88	124	1	0	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	44k	102	156	0	1	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	189	264	2	1	97	0	1	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	97	131	1	1	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	87	122	1	0	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	48k	116	164	1	0	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	82	114	1	1	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	106	148	1	1	98	0	0	1
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	99	132	1	0	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	100	134	1	1	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	89	127	1	0	99	0	0	0
2.0	0	0	357M	42.4M	459M	3103M	0	0	0	0	112	174	1	0	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	153	257	1	1	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	111	151	1	1	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	86	129	1	0	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	111	139	1	1	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	0	87	127	1	0	99	0	0	0
0	0	0	357M	42.4M	459M	3103M	0	0	0	12k	109	161	0	1	99	0	0	0

Şe^{kil 3}: dstat --vmstat Komutu

2.1.4 İşlemleri CPU Kullanımına Göre Sıralı Bir Şekilde Görüntülemek

Bu işlem için komut "**dstat -c --top-cpu**" şeklinde kullanılmalıdır. Bu komut, sistemde çalışan işlemleri CPU'yu en çok kullanandan en az kullanana doğru sıralı bir şekilde listelemektedir. Komuta ait örnek bir çıktı şekil 4 üzerinde görülebilir.

[~] dstat -c --top-cpu						
----total-cpu-usage----						-most-expensive-
usr	sys	idl	wai	hiq	siq	cpu process
0	0	99	1	0	0	knotify4 0.4
1	7	92	0	1	0	vmtoolsd 0.5
3	2	96	0	0	0	konsole 1.5
1	0	99	0	0	0	Xorg 1.0
2	1	97	0	0	0	vmtoolsd 0.5
1	1	98	0	0	0	konsole 0.5
1	1	99	0	0	0	knotify4 0.5
1	1	98	0	0	0	vmtoolsd 1.0
2	1	97	0	0	1	konsole 0.5
3	2	95	0	0	0	Xorg 5.0
2	1	97	0	0	0	konsole 1.0
3	2	96	0	0	0	Xorg 5.0
1	1	98	0	0	0	vmtoolsd 1.0
1	1	98	0	0	0	vmtoolsd 0.5
1	1	98	0	0	0	konsole 0.5
4	2	93	0	1	0	Xorg 8.5
2	1	97	0	0	0	Xorg 2.0
2	1	98	0	0	0	knotify4 0.5
1	1	98	0	0	0	konsole 1.0
1	1	98	0	0	0	knotify4 0.5

Şekil 4: dstat -c --top-cpu Komutu

2.1.5 İşlemleri Memory Kullanımına Göre Sıralı Bir Şekilde Görüntülemek

Bu işlem için komut "**dstat -d --top-mem**" şeklinde kullanılmalıdır. Bu komut, sistemde çalışan işlemleri kullandıkları RAM miktarına göre en çok kullanandan en az kullanana doğru sıralı bir şekilde listelemektedir. Komuta ait örnek bir çıktı şekil 5 üzerinde görülebilir.

<u>read</u>	<u>writ</u>	<u>memory</u>	<u>process</u>
67k	3673B	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	64k	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	0	Xorg	62.0M
0	24k	Xorg	62.0M
0	0	Xorg	62.0M

Şekil 5: dstat -d --top-mem Komutu

2.1.6 İşlemlere Ait Tüm İstatistikleri Listelemek

Bu işlem için komut "**dstat -a**" şeklinde kullanılmalıdır. Bu komut, sistemde çalışan işlemlere ait tüm istatistikleri listelemektedir. Komuta ait örnek bir çıktı şekilde 6 üzerinde görülebilir.

[~] dstat -a													
----total-cpu-usage----				-dsk/total-		-net/total-		---paging--		---system---			
usr	sys	idl	wai	hiq	siq	read	writ	recv	send	in	out	int	csw
0	0	99	0	0	0	29k	2866B	0	0	0	0	83	132
0	1	99	0	0	0	0	0	0	0	0	0	78	127
2	0	98	0	0	0	0	0	0	0	0	0	331	583
0	0	100	0	0	0	0	0	0	0	0	0	92	176
0	1	99	0	0	0	0	0	0	0	0	0	84	140
1	0	99	0	0	0	0	0	0	0	0	0	79	116
0	0	100	0	0	0	0	60k	0	0	0	0	91	156
0	0	100	0	0	0	0	0	0	0	0	0	178	328
1	1	99	0	0	0	0	0	0	0	0	0	119	162
0	0	100	0	0	0	0	0	0	0	0	0	69	126
0	0	100	0	0	0	0	0	0	0	0	0	91	146
1	1	98	0	0	0	0	0	0	0	0	0	365	706
9	2	88	0	0	1	0	0	0	0	0	0	267	515
15	11	69	3	1	0	472k	0	0	0	0	0	501	1018

Şekil 6: dstat -a Komutu

2.1.7 Eklentileri Listelemek

Bu işlem için komut "**dstat -list**" şeklinde kullanılmalıdır. Bu komut, sistemde yüklü ve dstat komutunun kullanımına hazır olan eklentileri listelemektedir. Komuta ait örnek bir çıktı Şekil 7 üzerinde görülebilir.

```
[~] dstat --list
internal:
    aio, cpu, cpu24, disk, disk24, disk24old, epoch, fs, int, int24, io, ipc, load, lock, mem,
    net, page, page24, proc, raw, socket, swap, swapold, sys, tcp, time, udp, unix, vm
/usr/share/dstat:
    battery, battery-remain, cpufreq, dbus, disk-util, fan, freespace, gpfs, gpfs-ops,
    helloworld, innodb-buffer, innodb-io, innodb-ops, lustre, memcache-hits, mysql-io,
    mysql-keys, mysql5-cmds, mysql5-conn, mysql5-io, mysql5-keys, net-packets, nfs3, nfs3-ops,
    nfsd3, nfsd3-ops, ntp, postfix, power, proc-count, rpc, rpcd, sendmail, snooze, thermal,
    top-bio, top-cpu, top-cputime, top-cputime-avg, top-io, top-latency, top-latency-avg,
    top-mem, top-oom, utmp, vm-memctl, vmk-hba, vmk-int, vmk-nic, vz-cpu, vz-io, vz-ubc, wifi
```

Şekil 7: dstat -list Komutu

2.1.8 Renksiz Çıktı Almak

Bu işlem için komut "**dstat --nocolor**" şeklinde kullanılmalıdır. Bu komut ile dstat komutunun ürettiği çıktıının renksiz olması sağlanır. Komuta ait örnek bir çıktı Şekil 8 üzerinde görülebilir.

```
[~] dstat --nocolor
----total-cpu-usage---- -dsk/total- -net/total- ---paging-- ---system--
usr sys idl wai hiq siq| read  writ| recv  send|  in   out | int   csw
 0  0 99  0  0  0| 37k 3141B|  0     0 |  0     0 |  0     0 |  87   136
 1  1 99  0  0  0|  0     0 |  0     0 |  0     0 |  0     0 | 156   236
 1  1 98  0  0  0|  0     0 |  0     0 |  0     0 |  0     0 | 274   443
 1  1 99  0  0  0|  0     0 |  0     0 |  0     0 |  0     0 | 111   185
 0  0 100  0  0  0|  0     0 |  0     0 |  0     0 |  0     0 |  74   123
 1  0 99  0  0  0|  0     0 |  0     0 |  0     0 |  0     0 |  92   150
 0  0 100  0  0  0|  0     0 | 60k   0 |  0     0 |  0     0 |  93   144
 0  0 100  0  0  0|  0     0 |  0     0 |  0     0 |  0     0 |  78   132
 0  1 99  0  0  0|  0     0 |  0     0 |  0     0 |  0     0 |  76   131
 1  0 99  0  0  0|  0     0 |  0     0 |  0     0 |  0     0 | 107   157
 0  0 100  0  0  0|  0     0 |  0     0 |  0     0 |  0     0 |  84   191
```

Şekil 8: dstat --nocolor Komutu

2.2 mpstat Komutu

mpstat, işlemci ile alakalı istatistikleri görüntülemeye yarayan bir komuttur. Sistemin CPU kullanımı istatistiklerini doğru bir şekilde görüntüler. CPU kullanımı ve performansı hakkında bilgilerin görüntülenebilmesini sağlar. Komut, sistemdeki CPU'ları CPU0-CPUX şeklinde numaralandırır ve çıktısını buna göre üretir.

2.2.1 mpstat Komutu Yardım Sayfası

Bu işlem için komut "**mpstat -h**" şeklinde kullanılmalıdır. Bu komut, mpstat komutunun nasıl kullanılabileceğini göstermektedir. Komuta ait örnek bir çıktı şekil 9 üzerinde görülebilir.

```
[~] mpstat -h
Usage: mpstat [ options ] [ <interval> [ <count> ] ]
Options are:
[ -A ] [ -I { SUM | CPU | ALL } ] [ -u ]
[ -P { <cpu> [,...] | ON | ALL } ] [ -V ]
[~] █
```

Şekil 9: mpstat -h Komutu

2.2.2 CPU İstatistiklerini Görüntüleme

Bu işlem için komut "**mpstat**" şeklinde kullanılmalıdır. Bu komut ile tüm CPU istatistikleri ekrana yazdırılır. Komuta ait örnek bir çıktı şekil 10 üzerinde verilmiştir.

```
[~] mpstat
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain) 08/12/2021 _x86_64_ (2 CPU)
05:55:40 PM CPU %usr %nice %sys %iowait %irq %soft %steal %guest %idle
05:55:40 PM all 1.45 0.00 1.71 2.13 0.09 0.06 0.00 0.00 94.56
[~] █
```

Şekil 10: mpstat Komutu

2.2.3 İstatistikleri CPU Numaralarıyla Birlikte Görüntüleme

Bu işlem için komut "**mpstat -P ALL**" şeklinde kullanılmalıdır. Bu komut ile üretilen çıktıının sadece tüm CPU'lar ile sınırlı kalmaması sağlanıp CPU numaralarına göre istatistikler ayrı olarak yazdırılır. Komuta ait örnek bir çıktı şekil 11 üzerinde görülebilir.

```
[~] mpstat -P ALL
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain) 08/12/2021 _x86_64_ (2 CPU)

06:00:14 PM CPU %usr %nice %sys %iowait %irq %soft %steal %guest %idle
06:00:14 PM all 1.17 0.00 1.36 1.61 0.08 0.06 0.00 0.00 95.73
06:00:14 PM 0 1.34 0.00 1.67 2.29 0.15 0.05 0.00 0.00 94.51
06:00:14 PM 1 1.00 0.00 1.05 0.94 0.00 0.06 0.00 0.00 96.95
```

Şekil 11: mpstat -P ALL Komutu

2.2.4 İstatistikleri Spesifik Bir CPU'dan Görüntülemek

Bu işlem için komut "**mpstat -P X**" şeklinde kullanılmalıdır. Komuttaki X, istenilen CPU numarasıdır. Bu komut ile istenilen numaralı CPU'nun istatistikleri ekrana yazdırılır. Komuta ait örnek bir çıktı şekil 12 üzerinde verilmiştir.

```
[~] mpstat -P 0
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain) 08/12/2021 _x86_64_ (2 CPU)

06:04:14 PM CPU %usr %nice %sys %iowait %irq %soft %steal %guest %idle
06:04:14 PM 0 1.15 0.00 1.41 1.88 0.13 0.05 0.00 0.00 95.38
[~] █
```

Şekil 12: mpstat -P X Komutu

2.2.5 İstatistikleri Belirli Bir Zaman Aralığıyla Görüntülemek

Bu işlem için komut "**mpstat X Y**" şeklinde kullanılmalıdır. Burada X başlangıç zamanını, Y ise bitiş zamanını ifade etmektedir. Bu komut ile istatistiklerin verilen zaman aralığında görüntülenmesi sağlanmaktadır. Komuta ait örnek bir çıktı şekil 13 üzerinde gösterilmektedir.

```
[~] mpstat 1 5
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain) 08/12/2021 _x86_64_ (2 CPU)

06:17:20 PM CPU %usr %nice %sys %iowait %irq %soft %steal %guest %idle
06:17:21 PM all 0.51 0.00 0.00 0.00 0.00 0.00 0.00 0.00 99.49
06:17:22 PM all 3.78 0.00 1.62 0.00 0.54 0.00 0.00 0.00 94.05
06:17:23 PM all 4.40 0.00 1.65 0.00 0.55 0.55 0.00 0.00 92.86
06:17:24 PM all 0.51 0.00 1.02 0.00 0.00 0.00 0.00 0.00 98.48
06:17:25 PM all 0.51 0.00 0.00 0.00 0.00 0.00 0.00 0.00 99.49
Average: all 1.88 0.00 0.84 0.00 0.21 0.10 0.00 0.00 96.97
[~] █
```

Şekil 13: mpstat X Y Komutu

2.2.6 Tüm İstatistikleri Görüntülemek

Bu işlem için komut "**mpstat -A**" şeklinde kullanılmalıdır. Bu komut ile mpstat komutunun toplayabileceği tüm istatistikler görüntülenmektedir. Komuta ait örnek bir çıktı şekil 14 üzerinde görülebilir.

```
[~] mpstat -A
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain) 08/12/2021 _x86_64_ (2 CPU)

06:21:48 PM CPU %usr %nice %sys %iowait %irq %soft %steal %idle
06:21:48 PM all 0.73 0.00 0.76 0.75 0.06 0.05 0.00 0.00 97.66
06:21:48 PM 0 0.82 0.00 0.93 1.06 0.11 0.04 0.00 0.00 97.05
06:21:48 PM 1 0.64 0.00 0.59 0.44 0.00 0.06 0.00 0.00 98.27

06:21:48 PM CPU intr/s
06:21:48 PM all 131.32
06:21:48 PM 0 7.39
06:21:48 PM 1 1.02

06:21:48 PM CPU 0/s 1/s 4/s 8/s 9/s 12/s 14/s 15/s 16/s 17/s 18/s 19/s 24/s 25/s 26/s 2 44/
7/s 28/s 29/s 30/s 31/s 32/s 33/s 34/s 35/s 36/s 37/s 38/s 39/s 40/s 41/s 42/s 43/s
5/s 45/s 46/s 47/s 48/s 49/s 50/s 51/s 52/s 53/s 54/s 55/s 56/s 57/s NMI/s LOC/s SPU/s PMI/s
INI/s RES/s CAL/s TLB/s TRM/s THR/s MCE/s MCP/s ERR/s MIS/s
06:21:48 PM 0 0.15 0.18 0.00 0.00 0.00 0.00 0.71 0.00 2.54 0.00 3.61 0.02 0.02 0.00 0.00 0.00 0
.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
0 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
0 0.00 9.99 0.03 0.39 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
06:21:48 PM 1 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
0 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
0 0.00 11.61 0.07 0.51 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
[~] █
```

Şekil 14: mpstat -A Komutu

2.3 numastat Komutu

numastat, linux sistemlerde işlemelere ait hafıza kullanımını görüntülemeye yarayan popüler araçlardan biridir. NUMA, "Non-uniform memory access" kelimelerinin birleşiminden, stat ise "statistics" kelimesinin kısaltımından gelmektedir. Dolayısıyla numastat komutu, düğüm başına bellek istatistiklerinin görüntülenmesini sağlayan bir komuttur.

2.3.1 numastat Komutu Yardım Sayfası

Bu işlem için komut "**numastat -help**" şeklinde kullanılmalıdır. Bu komut ile numastat komutunun kullanımına dair bilgiler görüntülenebilmektedir. Komuta ait örnek bir çıktı [Şekil 15](#) üzerinde verilmiştir.

```
[~] numastat --help
Usage: numastat [-c] [-m] [-n] [-p <PID>|<pattern>] [-s[<node>]] [-v] [-V] [-z] [ <
PID>|<pattern>... ]
-c to minimize column widths
-m to show meminfo-like system-wide memory usage
-n to show the numastat statistics info
-p <PID>|<pattern> to show process info
-s[<node>] to sort data by total column or <node>
-v to make some reports more verbose
-V to show the numastat code version
-z to skip rows and columns of zeros
[~] █
```

Şekil 15: numastat -help Komutu

2.3.2 Genel İstatistikleri Görüntülemek

Bu işlem için komut "**numastat**" şeklinde kullanılmalıdır. Bu komut ile düğüm başına NUMA bilgileri gibi bilgiler görüntülenebilir. Komuta ait örnek bir çıktı şe^{kil 16} üzerinde görülebilir.

[~] numastat	
node0	769183
numa_hit	0
numa_miss	0
numa_foreign	0
interleave_hit	22858
local_node	769183
other_node	0
[~]	█

Şe^{kil 16}: numastat Komutu

Şe^{kil 16} üzerinde görülen bilgilerin incelenmesi gereklidir:

- **numa_hit**: Başarıyla ayrılan hafızayı göstermektedir.
- **numa_miss**: Başka bir düğüm için ayrılması planlanan fakat bu düğüme ayrılan hafızayı göstermektedir.
- **numa_foreign**: Bu düğüm için ayrılması planlanan fakat başka bir düğüme ayrılan hafızayı göstermektedir.
- **interleave_hit**: Düğümde, plana uygun şekilde başarıyla ayrılan aralıklı belleği göstermektedir.
- **local_node**: İşlem çalıştığı sırada ayrılan hafızayı göstermektedir.
- **other_node**: İşlem farklı bir düğümde çalışırken bu düğüme atanan hafızayı göstermektedir.

2.3.3 İstatistikleri Minimize Edilmiş Sütunlarla Görüntülemek

Bu işlem için komut "**numastat -c**" şeklinde kullanılmalıdır. Bu komut ile sütun genişliklerinin veri içeriğine göre dinamik olarak daraltılması sağlanmaktadır. Komut bu şekilde kullanıldığında bellek miktarları en yakın megabayta yuvarlanmaktadır. Komuta ait örnek bir çıktı şekil 17 üzerinde gösterilmektedir.

```
[~] numastat -c

Per-node numastat info (in MBs):
      Node 0  Total
-----
Numa_Hit        3136  3136
Numa_Miss        0     0
Numa_Foreign      0     0
Interleave_Hit     89    89
Local_Node       3136  3136
Other_Node         0     0
[~] █
```

Sekil 17: numastat -c Komutu

2.3.4 İstatistikleri Sıralı Bir Şekilde Görüntülemek

Bu işlem için komut "**numastat -s**" şeklinde kullanılmalıdır. Bu komut ile çıktı değeri daha yüksek olandan daha düşük olana doğru sıralı bir şekilde gösterilmektedir. Komuta ait örnek bir çıktı şekil 18 üzerinde verilmiştir.

```
[~] numastat -s

Per-node numastat info (in MBs):
      Node 0  Total
-----
Numa_Hit        3197.05  3197.05
Local_Node       3197.05  3197.05
Interleave_Hit     89.29   89.29
Numa_Foreign      0.00    0.00
Numa_Miss         0.00    0.00
Other_Node         0.00    0.00
[~] █
```

Sekil 18: numastat -s Komutu

2.3.5 meminfo Tarzı Sistem Geneli RAM Kullanım İstatistiklerini Görüntülemek

Bu işlem için komut "**numastat -m**" şeklinde kullanılmalıdır. Bu komut ile sistem geneli RAM istatistikleri görüntülenebilmektedir. Komuta ait örnek bir çıktı şekil 19 üzerinde görülebilir.

```
[~] numastat -m
```

Per-node system memory usage (in MBs):		
	Node 0	Total
MemTotal	4095.43	4095.43
MemFree	3311.38	3311.38
MemUsed	784.05	784.05
Active	233.84	233.84
Inactive	238.86	238.86
Active(anon)	176.69	176.69
Inactive(anon)	7.80	7.80
Active(file)	57.14	57.14
Inactive(file)	231.06	231.06
Unevictable	0.00	0.00
Mlocked	0.00	0.00
Dirty	0.08	0.08
Writeback	0.00	0.00
FilePages	296.33	296.33
Mapped	90.38	90.38
AnonPages	100.29	100.29
Shmem	8.13	8.13
KernelStack	4.34	4.34
PageTables	19.77	19.77
NFS_Unstable	0.00	0.00
Bounce	0.00	0.00
WritebackTmp	0.00	0.00
Slab	88.99	88.99
SReclaimable	25.42	25.42
SUnreclaim	63.57	63.57
HugePages_Total	0.00	0.00
HugePages_Free	0.00	0.00
HugePages_Surp	0.00	0.00

Şekil 19: numastat -m Komutu

2.3.6 Verilen Bir İşleme Ait İstatistikleri Görüntülemek

Bu işlem için komut "**numastat -p PID**" şeklinde kullanılmalıdır. Buradaki PID, istenilen process id'sini temsil etmektedir. Komuta ait örnek bir çıktı şekil 20 üzerinde gösterilmektedir.

Per-node process memory usage (in MBs) for PID 4579 (vim)		
	Node 0	Total
Huge	0.00	0.00
Heap	2.99	2.99
Stack	0.09	0.09
Private	2.55	2.55
Total	5.62	5.62
[~] █		

Şekil 20: numastat -p PID Komutu

2.3.7 Birden Çok İşleme Ait İstatistikleri Görüntülemek

Bu işlem için komut "**numastat -v PIDs**" şeklinde kullanılmalıdır. Aslında -v parametresi, numastat komutunun daha çok bilgi vermesini sağlamaktadır fakat birden çok işlem id'si verildiği durumda istenilen işlemelere ait istatistiklerin görüntülenebilmesi sağlanabilmektedir. Komuta ait örnek bir çıktı şekil 21 üzerinde verilmiştir.

[~] numastat -v 4579 4662		
Per-node process memory usage (in MBs) for PID 4579 (vim)		
	Node 0	Total
Huge	0.00	0.00
Heap	2.99	2.99
Stack	0.09	0.09
Private	2.55	2.55
Total	5.62	5.62
[~] █		

Per-node process memory usage (in MBs) for PID 4662 (htop)		
	Node 0	Total
Huge	0.00	0.00
Heap	0.47	0.47
Stack	0.04	0.04
Private	1.29	1.29
Total	1.80	1.80
[~] █		

Şekil 21: numastat -v PIDs Komutu

2.4 perf Komutu

Linux sistemlerde performans denetleme işlemi yapılabilmesi için pek çok sayıda araç bulunmaktadır. perf komutu da bu araçlardan biri olup, hafif CPU profili oluşturulmasını sağlamaktadır. perf komutu ile CPU performans sayaçları, izleme noktaları, uprobe ve kprobe'lerin kontrolü yapılabilmekte, program eventleri izlenilebilmekte ve bu bilgiler raporlanabilmektedir.

2.4.1 perf Komutu Yardım Sayfası

Bu işlem için komut "**perf help**" şeklinde kullanılmalıdır. Bu komut ile perf komutunun kullanımına dair bilgiler gösterilmektedir. Komuta ait örnek bir çıktı şekil 22 üzerinde görülebilir.

```
[~] perf help

usage: perf [--version] [--help] [OPTIONS] COMMAND [ARGS]

The most commonly used perf commands are:
annotate      Read perf.data (created by perf record) and display annotated code
archive       Create archive with object files with build-ids found in perf.data file
bench         General framework for benchmark suites
buildid-cache Manage build-id cache.
buildid-list  List the buildids in a perf.data file
data          Data file related processing
diff          Read perf.data files and display the differential profile
evlist        List the event names in a perf.data file
inject        Filter to augment the events stream with additional information
kmem          Tool to trace/measure kernel memory properties
kvm           Tool to trace/measure kvm guest os
list          List all symbolic event types
lock          Analyze lock events
mem           Profile memory accesses
record        Run a command and record its profile into perf.data
report        Read perf.data (created by perf record) and display the profile
sched         Tool to trace/measure scheduler properties (latencies)
script        Read perf.data (created by perf record) and display trace output
stat          Run a command and gather performance counter statistics
test          Runs sanity tests.
timechart     Tool to visualize total system behavior during a workload
top           System profiling tool.
probe         Define new dynamic tracepoints
trace         strace inspired tool

See 'perf help COMMAND' for more information on a specific command.
```

Şekil 22: perf help Komutu

2.4.2 Sembolik Eventlerin Listelenmesi

Bu işlem için komut "**perf list**" şeklinde kullanılmalıdır. Bu komut ile donanım, yazılım ve kernel dahil tüm sembolik eventler görüntülenebilir. Komuta ait örnek bir çıktı şekil 23 üzerinde gösterilmektedir.

List of pre-defined events (to be used in -e):	
ref-cycles	[Hardware event]
alignment-faults	[Software event]
context-switches OR cs	[Software event]
cpu-clock	[Software event]
cpu-migrations OR migrations	[Software event]
dummy	[Software event]
emulation-faults	[Software event]
major-faults	[Software event]
minor-faults	[Software event]
page-faults OR faults	[Software event]
task-clock	[Software event]
L1-dcache-load-misses	[Hardware cache event]
L1-dcache-loads	[Hardware cache event]
L1-dcache-stores	[Hardware cache event]
L1-icache-load-misses	[Hardware cache event]
branch-load-misses	[Hardware cache event]
branch-loads	[Hardware cache event]
dTLB-load-misses	[Hardware cache event]
dTLB-loads	[Hardware cache event]
dTLB-store-misses	[Hardware cache event]
dTLB-stores	[Hardware cache event]
iTLB-load-misses	[Hardware cache event]
iTLB-loads	[Hardware cache event]
cycles-ct OR cpu/cycles-ct/	[Kernel PMU event]
cycles-t OR cpu/cycles-t/	[Kernel PMU event]
el-abort OR cpu/el-abort/	[Kernel PMU event]
el-capacity OR cpu/el-capacity/	[Kernel PMU event]
el-commit OR cpu/el-commit/	[Kernel PMU event]

Şekil 23: perf list Komutu

Bu komut ile birlikte ek parametre verilmesi durumunda spesifik bir eventin çıktısı görüntülenebilir. Örneğin komut "**perf list sw**" şeklinde çalıştırılırsa, sadece yazılım eventleri listelenecektir. Aynı şekilde "**perf list hw**", sadece donanım eventlerini listeleyecektir.

2.4.3 Performans İstatistiklerinin Görüntülenmesi

Bu işlem için komut "**perf stat cmd**" şeklinde kullanılmalıdır. Burada cmd, komut satırı üzerinde çalışan bir komutu ifade etmektedir. Komuta ait örnek bir çıktı şe^{kil 24} üzerinde verilmiştir.

```
[~] perf stat ls
Desktop Documents dotfiles Downloads Music Pictures Public ss Templates Videos

Performance counter stats for 'ls':

      1.653078    task-clock (msec)          #    0.714 CPUs utilized
              0    context-switches           #    0.000 K/sec
              0    cpu-migrations            #    0.000 K/sec
            249    page-faults              #    0.151 M/sec
<not supported>    cycles
<not supported>    stalled-cycles-frontend
<not supported>    stalled-cycles-backend
<not supported>    instructions
<not supported>    branches
<not supported>    branch-misses

 0.002315976 seconds time elapsed
```

Şe^{kil 24: perf stat cmd Komutu}

2.4.4 Gerçek Zamanlı Sistem Profilinin Görüntülenmesi

Bu işlem için komut "**perf top -a**" şeklinde kullanılmalıdır. Komuttaki -a parametresi ile tüm event tiplerinin gösterilmesi sağlanmaktadır. Bu komut ile CPU'nun gerçek zamanlı sistem profili görüntülenebilir. Komuta ait örnek bir çıktı şekil 25 üzerinde görülebilmektedir.

Samples: 638 of event 'cpu-clock', Event count (approx.): 142171849		
Overhead	Shared Object	Symbol
8.97%	[kernel]	[k] avtab_search_node
7.89%	[kernel]	[k] _spin_unlock_irqrestore
4.22%	vmmouse_drv.so	[.] 0x0000000000002529
2.22%	[kernel]	[k] finish_task_switch
2.16%	perf	[.] rb_next
2.16%	[kernel]	[k] system_call_after_swapgs
2.06%	[kernel]	[k] vsnprintf
2.04%	[kernel]	[k] handle_IRQ_event
1.71%	perf	[.] symbols_insert
1.68%	[kernel]	[k] format_decode
1.48%	[kernel]	[k] module_get_kallsym
1.47%	[kernel]	[k] __do_softirq
1.46%	libc-2.12.so	[.] __int_malloc
1.21%	[kernel]	[k] kallsyms_expand_symbol
1.01%	libpixman-1.so.0.32.8	[.] 0x0000000000075e0d
0.98%	libc-2.12.so	[.] memcpy
0.98%	[kernel]	[k] number
0.94%	[kernel]	[k] strcmp
0.94%	[kernel]	[k] string
0.93%	[kernel]	[k] __do_page_fault
0.91%	[kernel]	[k] clear_page
0.84%	libpixman-1.so.0.32.8	[.] 0x0000000000075e07
0.83%	[kernel]	[k] strnlen
0.81%	libpixman-1.so.0.32.8	[.] 0x0000000000075df9
0.81%	[kernel]	[k] task_has_capability
0.81%	libc-2.12.so	[.] __strcmp_sse42
0.75%	libelf-0.164.so	[.] gelf_getsym
0.70%	[kernel]	[k] sock_poll
0.70%	libpixman-1.so.0.32.8	[.] 0x000000000000075e19
0.68%	libpixman-1.so.0.32.8	[.] 0x000000000000075dfe
0.67%	perf	[.] hex2u64

Sekil 25: perf top -a Komutu

2.5 pidstat Komutu

pidstat komutu, anlık olarak linux çekirdeği tarafından yönetilen görevleri izlemek amacıyla kullanılmaktadır. Bu komut ile linux çekirdeği tarafından yönetilen tüm görevin aktiviteleri standart output'a yazdırılmaktadır. Ayrıca pidstat komutu ile seçilen bir process'e ait child process'ler de izlenebilmektedir.

2.5.1 pidstat Komutu Yardım Sayfası

Bu işlem için komut "**pidstat help**" şeklinde kullanılmalıdır. Bu komut ile pidstat komutunun kullanımına dair bilgiler görülebilmektedir. Komuta ait örnek bir çıktı şekil 26 üzerinde verilmiştir.

```
[~] pidstat help
Usage: pidstat [ options ] [ <interval> [ <count> ] ]
Options are:
[ -C <command> ] [ -d ] [ -h ] [ -I ] [ -l ] [ -r ] [ -t ] [ -u ] [ -V ] [ -w ]
[ -p { <pid> [,...] | SELF | ALL } ] [ -T { TASK | CHILD | ALL } ]
[~] █
```

Şekil 26: pidstat help Komutu

2.5.2 Genel İstatistiklerin Görüntülenmesi

Bu işlem için komut "pidstat" şeklinde kullanılmalıdır. Bu komut ile aktif olan görevler görüntülenebilmektedir. Komuta ait örnek bir çıktı şekil 27 üzerinde görülebilmektedir.

[~] pidstat							
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain)					08/12/2021	_x86_64_(2 CPU)	
08:48:52 PM	PID	%usr	%system	%guest	%CPU	CPU	Command
08:48:52 PM	1	0.00	0.15	0.00	0.15	0	init
08:48:52 PM	2	0.00	0.00	0.00	0.00	0	kthreadd
08:48:52 PM	3	0.00	0.00	0.00	0.00	0	migration/0
08:48:52 PM	4	0.00	0.00	0.00	0.00	0	ksoftirqd/0
08:48:52 PM	7	0.00	0.00	0.00	0.00	1	migration/1
08:48:52 PM	9	0.00	0.00	0.00	0.00	1	ksoftirqd/1
08:48:52 PM	11	0.00	0.02	0.00	0.02	0	events/0
08:48:52 PM	12	0.00	0.03	0.00	0.03	1	events/1
08:48:52 PM	24	0.00	0.00	0.00	0.00	0	sync_supers
08:48:52 PM	28	0.00	0.01	0.00	0.01	0	kblockd/0
08:48:52 PM	29	0.00	0.00	0.00	0.00	1	kblockd/1
08:48:52 PM	34	0.00	0.07	0.00	0.07	0	ata_sff/0
08:48:52 PM	35	0.00	0.01	0.00	0.01	1	ata_sff/1
08:48:52 PM	38	0.00	0.00	0.00	0.00	1	kseriod
08:48:52 PM	51	0.00	0.04	0.00	0.04	0	khugepaged
08:48:52 PM	267	0.00	0.01	0.00	0.01	1	mpt_poll_0
08:48:52 PM	274	0.00	0.00	0.00	0.00	1	scsi_eh_1
08:48:52 PM	468	0.00	0.00	0.00	0.00	0	jbd2/dm-0-8
08:48:52 PM	574	0.01	0.01	0.00	0.02	1	udevd
08:48:52 PM	1209	0.00	0.00	0.00	0.00	1	flush-253:0
08:48:52 PM	1750	0.00	0.00	0.00	0.00	0	rsyslogd
08:48:52 PM	1873	0.00	0.01	0.00	0.01	1	irqbalance
08:48:52 PM	1930	0.00	0.00	0.00	0.00	1	rpcbind
08:48:52 PM	1983	0.01	0.01	0.00	0.02	0	lldpad
08:48:52 PM	1993	0.16	0.09	0.00	0.25	0	vmtoolsd
08:48:52 PM	2068	0.00	0.00	0.00	0.00	0	fcoemon
08:48:52 PM	2097	0.00	0.00	0.00	0.00	0	VGAuthService

Sekil 27: pidstat Komutu

2.5.3 I/O İstatistiklerinin Görüntülenmesi

Bu işlem için komut "**pidstat -d**" şeklinde kullanılmalıdır. Bu komut ile sistemde çalışan işlemlerin I/O istatistikleri görüntülenebilmektedir. Komuta ait örnek bir çıktı şe^{kil 28} üzerinde verilmiştir.

[~] pidstat -d	Linux 2.6.32-754.el6.x86_64 (localhost.localdomain)	08/12/2021	_x86_64_
(2 CPU)			
08:50:38 PM	PID	kB_rd/s	kB_wr/s kB_ccwr/s Command
08:50:38 PM	2968	0.13	0.00 0.00 gnome-keyring-d
08:50:38 PM	2978	9.37	0.02 0.01 startkde
08:50:38 PM	3139	5.45	0.00 0.00 kdeinit4
08:50:38 PM	3140	1.12	0.00 0.00 klauncher
08:50:38 PM	3142	4.08	0.05 0.02 kded4
08:50:38 PM	3144	0.05	0.00 0.00 gam_server
08:50:38 PM	3168	0.18	0.08 0.03 kglobalaccel
08:50:38 PM	3171	0.01	0.00 0.00 kwrapper4
08:50:38 PM	3172	2.01	0.02 0.02 ksmserver
08:50:38 PM	3174	4.23	0.01 0.01 kwin
08:50:38 PM	3177	2.78	0.02 0.02 knotify4
08:50:38 PM	3178	6.85	0.65 0.25 plasma-desktop
08:50:38 PM	3182	0.05	0.00 0.00 kio_file
08:50:38 PM	3184	0.00	0.02 0.02 kaccess
08:50:38 PM	3192	14.37	0.00 0.00 vmtoolsd
08:50:38 PM	3193	1.53	0.00 0.00 nm-applet
08:50:38 PM	3216	0.05	0.02 0.02 im-settings-dae
08:50:38 PM	3221	3.40	0.05 0.05 krunner
08:50:38 PM	3225	0.22	0.01 0.01 pulseaudio
08:50:38 PM	3226	0.47	0.00 0.00 restorecond
08:50:38 PM	3233	0.05	0.02 0.01 klipper
08:50:38 PM	3241	0.02	0.01 0.00 gconfd-2
08:50:38 PM	3242	0.02	0.00 0.00 gconf-helper
08:50:38 PM	3264	0.06	0.03 0.03 kmix
08:50:38 PM	3265	0.02	0.00 0.00 xsettings-kde
08:50:38 PM	3411	0.01	0.00 0.00 gconf-im-settin
08:50:38 PM	3428	3.43	0.09 0.02 konsole
08:50:38 PM	3430	2.16	0.01 0.00 zsh

Sekil 28: pidstat -d Komutu

2.5.4 Sayfa Hataları ve Hafıza Kullanımının Görüntülenmesi

Bu işlem için komut "**pidstat -r**" şeklinde kullanılmalıdır. Bu komut ile sistemde çalışan işlemler için sayfa hataları ve hafıza kullanımı istatistikleri görüntülenebilmektedir. Komuta ait örnek bir çıktı şəkil 29 üzerinde gösterilmektedir.

[~] pidstat -r	Linux 2.6.32-754.el6.x86_64 (localhost.localdomain)	08/12/2021	_x86_64_(2 CPU)				
<hr/>							
09:34:06 PM	PID	minflt/s	majflt/s	VSZ	RSS	%MEM	Command
09:34:06 PM	1	0.81	0.00	19356	1568	0.04	init
09:34:06 PM	574	0.30	0.00	11400	1536	0.04	udevd
09:34:06 PM	1619	0.05	0.00	29764	876	0.02	auditd
09:34:06 PM	1750	0.09	0.00	249152	1652	0.04	rsyslogd
09:34:06 PM	1873	1.19	0.00	18308	808	0.02	irqbalance
09:34:06 PM	1924	0.04	0.00	184864	1164	0.03	vmware-vmblock-
09:34:06 PM	1930	0.05	0.00	18980	876	0.02	rpcbind
09:34:06 PM	1983	0.04	0.00	13392	716	0.02	lldpad
09:34:06 PM	1993	1.55	0.00	249092	4544	0.11	vmtoolsd
09:34:06 PM	2068	0.02	0.00	8360	404	0.01	fcoemon
09:34:06 PM	2097	0.42	0.00	54044	6268	0.15	VGAuthService
09:34:06 PM	2158	0.41	0.00	32572	1900	0.05	dbus-daemon
09:34:06 PM	2194	0.38	0.01	175276	4780	0.12	NetworkManager
09:34:06 PM	2201	0.22	0.00	58132	2448	0.06	modem-manager
09:34:06 PM	2217	0.08	0.00	23352	1380	0.03	rpc.statd
09:34:06 PM	2242	0.14	0.00	9116	1556	0.04	dhclient
09:34:06 PM	2267	0.02	0.00	45000	696	0.02	wpa_supplicant
09:34:06 PM	2268	0.23	0.00	189144	3404	0.08	cupsd
09:34:06 PM	2330	0.04	0.00	4076	656	0.02	acpid
09:34:06 PM	2370	0.47	0.00	39424	5660	0.14	hal
09:34:06 PM	2371	0.18	0.00	20396	1204	0.03	hal-runner
09:34:06 PM	2415	0.08	0.00	22516	1108	0.03	hal-addon-inpu
09:34:06 PM	2418	0.08	0.00	22512	1128	0.03	hal-addon-stor
09:34:06 PM	2420	0.08	0.00	18004	1048	0.03	hal-addon-acpi
09:34:06 PM	2448	0.09	0.00	91272	1592	0.04	pcscd
09:34:06 PM	2466	0.37	0.00	386156	1940	0.05	automount
09:34:06 PM	2505	0.12	0.00	30740	2144	0.05	ntpd

Şekil 29: pidstat -r Komutu

Bu istatistikler görüntülenirken -p parametresi ile istenilen bir process'e ait istatistiklerin görüntülenmesi sağlanabilmektedir. Örneğin "**pidstat -r -p PID**" ile id'si verilen process'e ait sayfa hataları ve hafıza kullanımı görüntülenebilmektedir.

2.6 /proc/cpuinfo Dosyası

Bahsedilen diğer başlıkların aksine bir dosya olan /proc/cpuinfo, sistemin sahip olduğu işlemciye ait bilgileri içinde barındırmaktadır. Örnek bir dosya şekil 30 üzerinde verilmiştir.

```
[~] cat /proc/cpuinfo
processor       : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 60
model name     : Intel(R) Core(TM) i5-4460 CPU @ 3.20GHz
stepping        : 3
microcode      : 4294967295
cpu MHz         : 3199.997
cache size      : 6144 KB
physical id    : 0
siblings        : 2
core id         : 0
cpu cores       : 2
apicid          : 0
initial apicid : 0
fpu              : yes
fpu_exception   : yes
cpuid level    : 13
wp               : yes
flags            : fpu vme de pse tsc msr pae mce sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm constant_
tsc arch_perfmon xtopology tsc_reliable nonstop_tsc unfair_spinlock pni pclmulqdq
ssse3 fma cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer aes x
save avx f16c rdrand hypervisor lahf_lm abm arat xsaveopt invpcid_single pti ret
poline fsgsbase bmi1 avx2 smep bmi2 invpcid ibpb ibrs stibp arch_capabilities ss
bd
bogomips        : 6399.99
clflush size    : 64
cache_alignment : 64
address sizes   : 45 bits physical, 48 bits virtual
power management:
```

Şekil 30: /proc/cpuinfo Dosyası

2.7 rdmsr Komutu

Bu komut, CPU'nun MSR (machine spesific registers)'ların okunabilmesini sağlamaktadır.

Komuta ait kullanımlar şekil 31 üzerinde gösterilmiştir.

```
--help, -h
    Print a list of available options

--version, -V
    Print current version

--hexadecimal, -x
    Display output in hexadecimal (lower case)

--capital-hex, -X
    Display output in hexadecimal (upper case)

--decimal, -d
    Display output in signed decimal

--unsigned, -u
    Display output in unsigned decimal

--octal, -o
    Display output in octal

--c-language, -c
    Format output as a C language constant

--zero-pad, -0
    Output leading zeroes

--zero-pad, -0
    Output leading zeroes

--raw, -r
    Output raw binary

--all, -a
    All processors

--processor <#>, -p
    Select processor number (default: 0)

--bitfield h:l, -f
    Output bits [h:l] only
```

Şekil 31: rdmsr Komutu

2.8 sar Komutu

İsmi System Activity Report kelimelerinin kısaltımından alan sar komutu ile sistemin CPU, hafıza ve I/O kullanımını ile ilgili bilgiler görüntülenebilmekte, rapor edilebilmekte ve kaydedilebilmektedir.

2.8.1 sar Komutu Yardım Sayfası

Bu işlem için komut "sar help" şeklinde kullanılmalıdır. Bu komut ile sar komutunun kullanımına dair bilgiler ekranda gösterilebilmektedir. Komuta ait örnek bir çıktı şekil 32 üzerinde görülebilmektedir.

```
[~] sar help
Usage: sar [ options ] [ <interval> [ <count> ] ]
Options are:
[ -A ] [ -b ] [ -B ] [ -C ] [ -d ] [ -h ] [ -m ] [ -p ] [ -q ] [ -r ] [ -R ]
[ -S ] [ -t ] [ -u [ ALL ] ] [ -v ] [ -V ] [ -w ] [ -W ] [ -y ]
[ -I { <int> [,...] | SUM | ALL | XALL } ] [ -P { <cpu> [,...] | ALL } ]
[ -j { ID | LABEL | PATH | UUID | ... } ] [ -n { <keyword> [,...] | ALL } ]
[ -o [ <filename> ] | -f [ <filename> ] ] [ --legacy ]
[ -i <interval> ] [ -s [ <hh:mm:ss> ] ] [ -e [ <hh:mm:ss> ] ]
[~] █
```

Şekil 32: sar help Komutu

2.8.2 CPU İstatistiklerinin Görüntülenmesi

Bu işlem için komut "sar" şeklinde kullanılmalıdır. Bu komut ile CPU'ya ait istatistikler ekranda görüntülenebilir. Komuta ait örnek bir çıktı şekil 33 üzerinde verilmiştir.

```
[~] sar
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain)      08/13/2021      _x86_64_
(2 CPU)

12:00:01 AM    CPU    %user    %nice    %system    %iowait    %steal    %idle
12:10:01 AM    all     0.18     0.00     0.22       0.01     0.00   99.59
12:20:01 AM    all     0.33     0.00     0.25       0.00     0.00   99.42
12:30:01 AM    all     0.24     0.00     0.22       0.00     0.00   99.54
12:40:01 AM    all     0.19     0.00     0.17       0.00     0.00   99.64
12:50:01 AM    all     0.22     0.00     0.20       0.00     0.00   99.57
01:00:02 AM    all     0.19     0.00     0.17       0.00     0.00   99.64
Average:      all     0.23     0.00     0.20       0.00     0.00   99.57

06:47:11 AM      LINUX RESTART

06:50:01 AM    CPU    %user    %nice    %system    %iowait    %steal    %idle
07:00:01 AM    all     0.69     0.00     0.54       0.53     0.00   98.24
Average:      all     0.69     0.00     0.54       0.53     0.00   98.24
[~] █
```

Şekil 33: sar Komutu

Bu komuta ekstra parametreler verilerek çıktıının farklı şekillerde üretilmesi sağlanabilecektir. Örneğin komut "sar 2 5" şeklinde kullanılarak istatistiklerin 2 saniyede bir olmak üzere 5 kez ekrana yazdırılması sağlanabilir.

2.8.3 İstatistiklerin Dosyaya Yazılması

Bu işlem için komut "-o" parametresi ile kullanılmalıdır. Parametreden sonra istenilen dosya adının da yazılması unutulmamalıdır. Komuta ait örnek bir çıktı şekil 34 üzerinde görülebilir.

```
[~] ls
Desktop  dotfiles  Music  Public  Templates
Documents  Downloads  Pictures  ss  Videos
[~] sar 1 3 -o data
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain)  08/13/2021  _x86_64_
(2 CPU)

07:12:19 AM    CPU    %user    %nice    %system    %iowait    %steal    %idle
07:12:20 AM    all    0.00    0.00    0.00    0.51    0.00    99.49
07:12:21 AM    all    0.51    0.00    0.51    0.00    0.00    98.99
07:12:22 AM    all    0.51    0.00    0.51    0.00    0.00    98.98
Average:      all    0.34    0.00    0.34    0.17    0.00    99.16
[~] ls
data  Documents  Downloads  Pictures  ss      Videos
Desktop  dotfiles  Music  Public  Templates
[~] █
```

Şekil 34: sar -o Komutu

2.8.4 İstatistiklerin Dosyadan Okunması

Bu işlem için komut "-f" parametresi ile kullanılmalıdır. Parametreden sonra okunacak dosya adının da yazılması unutulmamalıdır. Komuta ait örnek bir çıktı şekil 35 üzerinde görülebilir.

```
[~] ls
data  Documents  Downloads  Pictures  ss      Videos
Desktop  dotfiles  Music  Public  Templates
[~] sar -f data
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain)  08/13/2021  _x86_64_
(2 CPU)

07:12:19 AM    CPU    %user    %nice    %system    %iowait    %steal    %idle
07:12:20 AM    all    0.00    0.00    0.00    0.51    0.00    99.49
07:12:21 AM    all    0.51    0.00    0.51    0.00    0.00    98.99
07:12:22 AM    all    0.51    0.00    0.51    0.00    0.00    98.98
Average:      all    0.34    0.00    0.34    0.17    0.00    99.16
[~] █
```

Şekil 35: sar -f Komutu

2.8.5 Hafıza Kullanım İstatistiklerinin Görüntülenmesi

Bu işlem için komut "-r" parametresi ile kullanılmalıdır. Komuta ait örnek bir çıktı şekil 36 üzerinde verilmiştir.

```
[~] sar -r 1 3
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain) 08/13/2021 _x86_64_
(2 CPU)

07:20:35 AM kbmemfree kbmemused %memused kbbuffers kbcached kbcommit %commi
t
07:20:36 AM 3190732 865592 21.34 44452 471784 574920 7.9
8
07:20:37 AM 3190732 865592 21.34 44452 471784 574920 7.9
8
07:20:38 AM 3190668 865656 21.34 44452 471784 574920 7.9
8
Average: 3190711 865613 21.34 44452 471784 574920 7.9
8
[~] █
```

Şekil 36: sar -r Komutu

2.8.6 Sayfalama İstatistiklerinin Görüntülenmesi

Bu işlem için komut "-B" parametresi ile kullanılmalıdır. Komuta ait örnek bir çıktı şekil 37 üzerinde görülebilmektedir.

```
[~] sar -B 1 3
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain) 08/13/2021 _x86_64_
(2 CPU)

07:22:08 AM pgpgin/s pgpgout/s fault/s majflt/s pgfree/s pgscank/s pgscand/
s pgsteal/s %vmeff
0 0.00 0.00 39.58 0.00 185.42 0.00 0.0
0 0.00 0.00 42.27 0.00 254.64 0.00 0.0
0 0.00 0.00 31.31 0.00 110.10 0.00 0.0
0 0.00 0.00 37.67 0.00 182.88 0.00 0.0
Average: 0 0.00 0.00 0.00 0.00 0.00 0.00 0.0
[~] █
```

Şekil 37: sar -b Komutu

2.8.7 Blok Cihaz İstatistiklerinin Görüntülenmesi

Bu işlem için komut "-d" parametresi ile kullanılmalıdır. Komuta ait örnek bir çıktı şekil 38 üzerinde gösterilmektedir.

```
[~] sar -d 1 1
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain) 08/13/2021 _x86_64_
(2 CPU)

07:27:15 AM      DEV      tps  rd_sec/s  wr_sec/s  avgrrq-sz  avgqu-sz  awai
t  svctm %util
07:27:16 AM  dev8-0    5.05    0.00   145.45    28.80    0.01    2.4
0  0.60  0.30
07:27:16 AM  dev11-0   0.00    0.00    0.00    0.00    0.00    0.0
0  0.00  0.00
07:27:16 AM dev253-0   18.18    0.00   145.45    8.00    0.05    2.8
3  0.17  0.30
07:27:16 AM dev253-1   0.00    0.00    0.00    0.00    0.00    0.0
0  0.00  0.00

Average:      DEV      tps  rd_sec/s  wr_sec/s  avgrrq-sz  avgqu-sz  awai
t  svctm %util
Average:  dev8-0    5.05    0.00   145.45    28.80    0.01    2.4
0  0.60  0.30
Average:  dev11-0   0.00    0.00    0.00    0.00    0.00    0.0
0  0.00  0.00
Average: dev253-0   18.18    0.00   145.45    8.00    0.05    2.8
3  0.17  0.30
Average: dev253-1   0.00    0.00    0.00    0.00    0.00    0.0
0  0.00  0.00
[~] █
```

Şekil 38: sar -d Komutu

2.8.8 Ağ İstatistiklerinin Görüntülenmesi

Bu işlem için komut "-n" parametresi ile kullanılmalıdır. Komuta ait örnek bir çıktı şekil 39 üzerinde verilmiştir.

[~] sar -n ALL		Linux 2.6.32-754.el6.x86_64 (localhost.localdomain)		08/13/2021		_x86_64_(2 CPU)	
12:00:01 AM	IFACE	rxpck/s	txpck/s	rxkB/s	txkB/s	rxcmp/s	txcmp/s
s rxmcst/s							
12:10:01 AM	lo	0.00	0.00	0.00	0.00	0.00	0.0
0 0.00							
12:10:01 AM	eth0	0.27	0.08	0.02	0.01	0.00	0.0
0 0.00							
12:20:01 AM	lo	0.00	0.00	0.00	0.00	0.00	0.0
0 0.00							
12:20:01 AM	eth0	0.22	0.08	0.01	0.01	0.00	0.0
0 0.00							
12:30:01 AM	lo	0.00	0.00	0.00	0.00	0.00	0.0
0 0.00							
12:30:01 AM	eth0	0.22	0.07	0.01	0.01	0.00	0.0
0 0.00							
12:40:01 AM	lo	0.00	0.00	0.00	0.00	0.00	0.0
0 0.00							
12:40:01 AM	eth0	0.23	0.05	0.01	0.00	0.00	0.0
0 0.00							
12:50:01 AM	lo	0.00	0.00	0.00	0.00	0.00	0.0
0 0.00							
12:50:01 AM	eth0	0.19	0.05	0.01	0.00	0.00	0.0
0 0.00							
01:00:02 AM	lo	0.00	0.00	0.00	0.00	0.00	0.0
0 0.00							
01:00:02 AM	eth0	0.20	0.05	0.01	0.00	0.00	0.0
0 0.00							
Average:	lo	0.00	0.00	0.00	0.00	0.00	0.0
0 0.00							
Average:	eth0	0.22	0.06	0.01	0.00	0.00	0.0
0 0.00							

Şekil 39: sar -n Komutu

2.9 tiptop Komutu

Bu komut ile linux görevleri için sistemin donanım performans sayaçları görüntülenebilmektedir. tiptop Komutu, sistemde çalışan görevler için dinamik bir gerçek-zamanlı görüntüleme olanağı sağlamaktadır. Komut, top komutuna çok benzemektedir fakat tiptop komutu gösterdiği bilgileri donanım sayaçlarından almaktadır.

tiptop komutunun iki adet çalışma modu bulunmaktadır. Bunlar live-mode ve batch-mode olarak isimlendirilmektedir. İki modda da sistem donanım sayaçlarındaki veriler için periyodik olarak sıraya alınır ve çeşitli oranlarla ekrana yazdırılır. Live-mode modunda çıktı belirli zaman aralıklarında güncellenmektedir, ayrıca programa farklı girdiler verilerek işlemler gerçekleştirilebilir. Batch-mode modunda ise bilgiler standart output'a yollanır, dolayısıyla dosyaya yazma işlemi için ideal bir moddur. Batch-mode modunda program ile etkileşime girilememektedir.

Komut root yetkileri ile çalıştırılmadığı sürece kullanıcı sadece kendisine ait görevlerin istatistiklerini görüntüleyebilmektedir.

2.9.1 tiptop Komutu - Komut Satırı Seçenekleri

- **-b:** tiptop'u batch-mode modunda başlatır. Eğer -n parametresi ile iterasyon sayısı belirtildiğinde komut sonsuza kadar çalışır.
- **-cpu-min VALUE:** CPU aktivitesi % olarak verilen eşinin altında olan görevlerin boşta (idle) kabul edilmesini sağlar ve çıktıda bu görevlere yer vermez.
- **-d VALUE:** Yenileme sıklığının değiştirilmesini sağlar. Değer 0.01'den büyük olmalıdır.
- **-E FILENAME:** Hataların kaydının tutulacağı dosyanın seçilmesini sağlar. Varsayılan olarak batch-mode modunda stderr, live-mode modunda ise geçici bir dosyadır.
- **-epoch:** Her yenilemede epoch değerini de yazdırır.
- **-h –help:** Komutun kullanımına dair bilgileri yazdırır.
- **-H:** Thread'ları da gösterir.
- **-i:** Boştaki görevleri de gösterir.
- **-K –kernel:** Gösterilen değerlere çekirdek aktivitelerini de ekler. Bu parametrenin kullanımı için root yetkilerine sahip olunması gerekmektedir.
- **-n VALUE:** Komutun çalışacağı iterasyon sayısını belirler.
- **-o FILENAME:** Batch-mode modunda komutun çıktısını yazacağı dosyanın belirlenmesini sağlar.
- **-p –pid VALUE:** İşlemlerin verilen değere göre filtrelenmesini sağlar. Bu değer nümerik bir değer olan process id'si ya da bir string olabilir.
- **-u USER:** Sadece belirtilen kullanıcıya ait görevlerin listelenmesini sağlar. Kullanıcı adıyla ya da id'si ile belirtilebilir.
- **-U:** Görevlerin sahiplerinin gösterilmesini sağlar.

2.9.2 tiptop Komutu - Live-Mode Modu Komutları

- **c:** Görev isimleri ve komut satırları arasında geçiş yapılmasını sağlar.
- **d:** Yenileme aralığının değiştirilebilmesini sağlar.
- **e:** O ana kadar karşılaşılan hataların gösterilmesini sağlar.
- **H:** İşlemlerin toplam değerleri ile işleme ait thread'lerin değerlerinin gösterimi arasında geçiş yapılmasını sağlar.
- **i:** Sadece aktif görevlerin gösterilmesi ile boşta olan değerlerin de gösterilmesi arasında geçiş yapılmasını sağlar.
- **K:** Sadece kullanıcı aktivitelerinin gösterimi ile çekirdek aktivitelerinin de gösterilmesi arasında geçiş yapılmasını sağlar. Kullanılması için superuser yetkileri gerekmektedir.
- **k:** Bir işlemin kill edilmesini sağlar. Kullanıcıya istenilen işlemin id'si ve gönderilmek istenen sinyal sorulur.
- **p:** Görevlerin işlem adına ya da id'sine göre filtrelenmesini sağlar.
- **q:** Programdan çıkış yapar.
- **R:** Sıralamayı değiştirir: büyükten küçüğe ya da küçükten büyüğe şeklinde.
- **u:** İşlemlerin kullanıcıya göre filtrelenmesini sağlar.

2.10 top Komutu

Bu komut, sistem üzerinde çalışan işlemleri görüntülemek için kullanılmaktadır. Programa ait örnek bir çıktı [Şekil 40](#) üzerinde görülebilir.

```
top - 10:05:08 up 3:18, 3 users, load average: 0.58, 0.13, 0.04
Tasks: 181 total, 1 running, 180 sleeping, 0 stopped, 0 zombie
Cpu(s): 6.0%us, 0.9%sy, 0.0%ni, 92.8%id, 0.0%wa, 0.4%hi, 0.0%si, 0.0%st
Mem: 4056324k total, 1004908k used, 3051416k free, 63772k buffers
Swap: 3145724k total, 0k used, 3145724k free, 540456k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2813	root	20	0	221m	46m	12m	S	21.0	1.2	0:22.81	Xorg
3438	seyitahm	20	0	460m	26m	17m	S	13.1	0.7	0:12.78	konsole
3188	seyitahm	20	0	795m	51m	23m	S	12.4	1.3	0:07.58	plasma-desktop
3184	seyitahm	20	0	523m	20m	14m	S	6.9	0.5	0:03.62	kwin
5798	seyitahm	20	0	15028	1296	944	R	1.6	0.0	0:01.55	top
3187	seyitahm	20	0	821m	23m	14m	S	1.0	0.6	1:55.10	knotify4
2478	root	20	0	22512	1124	952	S	0.3	0.0	0:28.99	haldd-addon-stor
3201	seyitahm	20	0	344m	17m	14m	S	0.3	0.4	0:35.23	vmtoolsd
1	root	20	0	19356	1576	1244	S	0.0	0.0	0:02.78	init
2	root	20	0	0	0	0	S	0.0	0.0	0:00.02	kthreadd
3	root	RT	0	0	0	0	S	0.0	0.0	0:00.04	migration/0
4	root	20	0	0	0	0	S	0.0	0.0	0:00.21	ksoftirqd/0
5	root	RT	0	0	0	0	S	0.0	0.0	0:00.00	stopper/0
6	root	RT	0	0	0	0	S	0.0	0.0	0:00.09	watchdog/0
7	root	RT	0	0	0	0	S	0.0	0.0	0:00.03	migration/1
8	root	RT	0	0	0	0	S	0.0	0.0	0:00.00	stopper/1
9	root	20	0	0	0	0	S	0.0	0.0	0:00.39	ksoftirqd/1
10	root	RT	0	0	0	0	S	0.0	0.0	0:00.08	watchdog/1
11	root	20	0	0	0	0	S	0.0	0.0	0:01.76	events/0
12	root	20	0	0	0	0	S	0.0	0.0	0:02.55	events/1
13	root	20	0	0	0	0	S	0.0	0.0	0:00.00	events/0
14	root	20	0	0	0	0	S	0.0	0.0	0:00.00	events/1
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	events_long/0
16	root	20	0	0	0	0	S	0.0	0.0	0:00.00	events_long/1
17	root	20	0	0	0	0	S	0.0	0.0	0:00.00	events_power_ef
18	root	20	0	0	0	0	S	0.0	0.0	0:00.00	events_power_ef
19	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cgroup
20	root	20	0	0	0	0	S	0.0	0.0	0:00.00	khelper

Şekil 40: top Komutu

Şekil 40 üzerinde görülen siyah arkaplanlı alandaki başlıklar açıklamak gerekirse:

- **PID**: Görevin eşsiz id'sini gösterir.
- **USER**: Görevin sahibi olan kullanıcıyı gösterir.
- **PR**: Görevin önceliğini (priority) gösterir.
- **NI**: Görevin nice value denilen değerini gösterir. Bu değerin negatif olması görevin önceliğinin yüksek olduğu, pozitif olması önceliğinin düşük olduğu anlamına gelmektedir.
- **VIRT**: Görev tarafından kullanılan sanal bellek miktarını gösterir.
- **SHR**: Görev tarafından kullanılan paylaşımı bellek miktarını gösterir.
- **%CPU**: Görevin CPU kullanım yüzdesini gösterir.
- **%MEM**: Görevin hafıza kullanım yüzdesini gösterir.

2.10.1 Spesifik Bir Kullanıcıya Ait İşlemlerin Görüntülenmesi

Bu işlem için komut "**top -u USER**" şeklinde kullanılmalıdır. Burada user, istenilen kullanıcının giriş adını temsil etmektedir. Bu komut ile istenilen kullanıcıya ait işlemlerin görüntülenmesi sağlanır. Komuta ait örnek bir çıktı şekil 41 üzerinde verilmiştir.

```
top - 10:14:55 up 3:27, 3 users, load average: 0.00, 0.01, 0.00
Tasks: 181 total, 2 running, 179 sleeping, 0 stopped, 0 zombie
Cpu(s): 3.0%us, 1.1%sy, 0.0%ni, 95.6%id, 0.0%wa, 0.2%hi, 0.2%si, 0.0%st
Mem: 4056324k total, 1005188k used, 3051136k free, 63972k buffers
Swap: 3145724k total, 0k used, 3145724k free, 540496k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3438	seyitahm	20	0	461m	26m	17m	S	4.0	0.7	0:14.46	konsole
3201	seyitahm	20	0	344m	17m	14m	S	1.3	0.4	0:36.91	vmtoolsd
3187	seyitahm	20	0	821m	23m	14m	S	1.0	0.6	2:00.77	knotify4
3188	seyitahm	20	0	795m	51m	23m	R	1.0	1.3	0:08.59	plasma-desktop
2972	seyitahm	20	0	65184	2664	2144	S	0.0	0.1	0:00.01	gnome-keyring-d
2982	seyitahm	20	0	103m	1440	1168	S	0.0	0.0	0:00.16	startkde
2990	seyitahm	20	0	20064	684	428	S	0.0	0.0	0:00.00	dbus-launch
2991	seyitahm	20	0	32072	1504	868	S	0.0	0.0	0:00.39	dbus-daemon
3051	seyitahm	20	0	57336	756	124	S	0.0	0.0	0:00.15	ssh-agent
3087	seyitahm	20	0	109m	508	316	S	0.0	0.0	0:00.67	gpg-agent
3149	seyitahm	20	0	262m	4564	2976	S	0.0	0.1	0:00.06	kdeinit4
3150	seyitahm	20	0	263m	7016	5700	S	0.0	0.2	0:00.26	klauncher
3152	seyitahm	20	0	539m	17m	14m	S	0.0	0.4	0:01.48	kded4
3154	seyitahm	20	0	13680	1244	1084	S	0.0	0.0	0:00.01	gam_server
3178	seyitahm	20	0	285m	11m	9060	S	0.0	0.3	0:00.37	kglobalaccel
3181	seyitahm	20	0	4056	448	376	S	0.0	0.0	0:00.00	kwrapper4
3182	seyitahm	20	0	400m	10m	9152	S	0.0	0.3	0:00.53	ksmserver
3184	seyitahm	20	0	523m	20m	14m	S	0.0	0.5	0:03.65	kwin
3192	seyitahm	20	0	265m	5924	4144	S	0.0	0.1	0:00.00	kio_file
3194	seyitahm	20	0	287m	10m	9140	S	0.0	0.3	0:00.60	kaccess
3222	seyitahm	20	0	537m	11m	8888	S	0.0	0.3	0:00.69	nm-applet
3231	seyitahm	20	0	201m	4568	3660	S	0.0	0.1	0:00.03	polkit-gnome-au
3236	seyitahm	20	0	439m	5188	3984	S	0.0	0.1	0:08.27	pulseaudio
3238	seyitahm	20	0	112m	3992	3320	S	0.0	0.1	0:00.02	im-settings-dae
3240	seyitahm	20	0	30404	5396	1520	S	0.0	0.1	0:00.75	restorecond
3243	seyitahm	20	0	134m	2024	1720	S	0.0	0.0	0:00.00	gvfsd

Şekil 41: top -u USER Komutu

2.10.2 top Komutu Seçenekleri

top komutuna ait kullanım seçenekleri aşağıda listelenmiştir. Listede başında "-" bulunan elemanlar, komutun çalıştırılması sırasında kullanılabilecek seçenekleri gösterirken bulunmayan elemanlar komutun çalışması sırasında yapılabilecek seçenekleri göstermektedir.

- **k:** Komut içerisinde istenilen processin id'si bulunduktan sonra o process'in kill edilmesini sağlar.
- **c:** İşlemlerin absolute path'leri ile gösterilmesini sağlar.
- **u:** İşlemlerin istenilen kullanıcıya göre filtrelenmesini sağlar.
- **P:** İşlemlerin CPU kullanımlarına göre sıralanmasını sağlar.
- **M:** İşlemlerin hafiza kullanımlarına göre sıralanmasını sağlar.
- **-b:** Komut bu parametre ile başlatıldığında çıktısının başka bir dosya ya da programa gön-derilebilmesini sağlar.
- **-s:** Komutun güvenli modda çalıştırılmasını sağlar.
- **-d:** Yenileme süresinin değiştirilebilmesini sağlar.

2.11 vmstat Komutu

vmstat komutu, sistemdeki işlemler, hafıza, sayfalama, blok I/O, disk ve CPU zamanlama gibi konularda bilgiler elde edinilmesini sağladığı için bir performans monitör komutudur. Komuta ait örnek bir çıktı şe^{kil 42} üzerinde görülebilir.

```
[~] vmstat
procs -----memory----- -swap-- -----io----- system-- -----cpu-----
 r b    swpd   free   buff   cache   si    so    bi    bo   in   cs us sy id wa st
 1 0      0 3041884  67132 542028   0    0    17    2   44   67  0  0 99  0  0
[~] █
```

Şe^{kil 42}: vmstat Komutu

Şe^{kil 42} üzerinde görülen başlıklardan önemli olanlardan bahsetmek gereklidir:

- **Free:** Kullanımda olmayan hafıza miktarını gösterir.
- **si:** Her saniye diskten alınan (swapped in) hafıza miktarını kilobyte cinsinden yazdırır.
- **so:** Her saniye diske verilen (swapped out) hafıza miktarını kilobyte cinsinden yazdırır.

Ayrıca yine şe^{kil 42} üzerinde görüldüğü gibi vmstat komutu hafızanın yanı sıra I/O, sistem ve CPU'ya ait bazı istatistikleri de göstermektedir.

2.11.1 vmstat Komutu Yardım Sayfası

Bu işlem için komut "vmstat help" şeklinde kullanılmamıştır. Bu komut ile vmstat komutunun kullanımına dair bilgiler ekrana yazdırılmaktadır. Komuta ait örnek bir çıktı şe⁴³kil 43 üzerinde verilmiştir.

```
[~] vmstat help
usage: vmstat [-V] [-n] [delay [count]]
               -V prints version.
               -n causes the headers not to be reprinted regularly.
               -a print inactive/active page stats.
               -d prints disk statistics
               -D prints disk table
               -p prints disk partition statistics
               -s prints vm table
               -m prints slabinfo
               -t add timestamp to output
               -S unit size
               delay is the delay between updates in seconds.
               unit size k:1000 K:1024 m:1000000 M:1048576 (default is K)
               count is the number of updates.
[~] █
```

Sekil 43: vmstat help Komutu

2.11.2 Hafızaya Ait Tüm İstatistikleri Görüntülemek

Bu işlem için komut "**vmstat -a**" şeklinde kullanılmalıdır. Bu komut ile aktif ve inaktif tüm hafıza gösterilmektedir. Komuta ait örnek bir çıktı şe^{kil 44} üzerinde verilmiştir.

```
[~] vmstat -a
procs -----memory----- swap-----io---- system-----cpu-----
 r b    swpd   free  inact active   si   so    bi    bo   in   cs us sy id wa st
 0 0      0 3041404 326940 450516    0    0    17     2   44   67  0  0 99  0  0
[~] █
```

Şe^{kil 44:} vmstat -a Komutu

2.11.3 Fork Sayısını Görüntüleme

Bu işlem için komut "**vmstat -f**" şeklinde kullanılmalıdır. Bu komut ile sistemdeki fork sayısı ekrana yazdırılmaktadır. Komuta ait örnek bir çıktı şe^{kil 45} üzerinde görülebilir.

```
[~] vmstat -f
          6634 forks
[~] █
```

Şe^{kil 45:} vmstat -f Komutu

2.11.4 Event Sayaçlarına Ait Verileri Görüntüleme

Bu işlem için komut "**vmstat -s**" şeklinde kullanılmalıdır. Bu komut ile çeşitli event sayaçları ve hafıza istatistikleri ekrana bir tablo biçiminde yazdırılır. Komuta ait örnek bir çıktı Şekil 46 üzerinde gösterilmektedir.

```
[~] vmstat -s
 4056324  total memory
 1015020  used memory
 450600   active memory
 327008   inactive memory
 3041304  free memory
    67324  buffer memory
 542052   swap cache
 3145724  total swap
      0    used swap
 3145724  free swap
 10941 non-nice user cpu ticks
    176 nice user cpu ticks
    9074 system cpu ticks
 3532588 idle cpu ticks
    6088 IO-wait cpu ticks
     725 IRQ cpu ticks
     619 softirq cpu ticks
      0 stolen cpu ticks
 597988 pages paged in
 74509  pages paged out
      0 pages swapped in
      0 pages swapped out
1572294 interrupts
2397980 CPU context switches
1628826418 boot time
    6668 forks
[~] █
```

Şekil 46: vmstat -s Komutu

2.11.5 Disk İstatistiklerini Görüntüleme

Bu işlem için komut "**vmstat -d**" şeklinde kullanılmalıdır. Bu komut ile sistemdeki tüm disklere ait istatistikler görüntülenebilmektedir. Komuta ait örnek bir çıktı şekil 47 üzerinde verilmiştir.

[~] vmstat -d										
disk-	reads-----			writes-----			IO-----			
	total	merged	sectors	ms	total	merged	sectors	ms	cur	sec
ram0	0	0	0	0	0	0	0	0	0	0
ram1	0	0	0	0	0	0	0	0	0	0
ram2	0	0	0	0	0	0	0	0	0	0
ram3	0	0	0	0	0	0	0	0	0	0
ram4	0	0	0	0	0	0	0	0	0	0
ram5	0	0	0	0	0	0	0	0	0	0
ram6	0	0	0	0	0	0	0	0	0	0
ram7	0	0	0	0	0	0	0	0	0	0
ram8	0	0	0	0	0	0	0	0	0	0
ram9	0	0	0	0	0	0	0	0	0	0
ram10	0	0	0	0	0	0	0	0	0	0
ram11	0	0	0	0	0	0	0	0	0	0
ram12	0	0	0	0	0	0	0	0	0	0
ram13	0	0	0	0	0	0	0	0	0	0
ram14	0	0	0	0	0	0	0	0	0	0
ram15	0	0	0	0	0	0	0	0	0	0
loop0	0	0	0	0	0	0	0	0	0	0
loop1	0	0	0	0	0	0	0	0	0	0
loop2	0	0	0	0	0	0	0	0	0	0
loop3	0	0	0	0	0	0	0	0	0	0
loop4	0	0	0	0	0	0	0	0	0	0
loop5	0	0	0	0	0	0	0	0	0	0
loop6	0	0	0	0	0	0	0	0	0	0
loop7	0	0	0	0	0	0	0	0	0	0
sda	17922	10824	1191776	108361	7899	10774	149378	22826	0	75
sr0	45	0	360	59	0	0	0	0	0	0
dm-0	27550	0	1181890	271495	18666	0	149328	77237	0	75
dm-1	325	0	2600	690	0	0	0	0	0	0

Sekil 47: vmstat -d Komutu

2.12 dmesg Komutu

Linux ve Unix benzeri sistemlerde önyükleme ve sistemi başlatma, bilgisayar açıldığında gerçekleşen eventler dizisinin iki farklı aşamasıdır.

Önyükleme işlemleri sistemin başlatılmasını, çekirdeğin hafızaya yüklendiği ve ilk ram-disk'e (initrd veya initramfs) bağlandığı, systemd'nin başladığı noktaya götürür. Buradan sonra başlatma işlemleri geri kalan işlemleri halleder ve işletim sisteminin başlatılmasını tamamlar. Bu olaylar dizinin erken aşamalarında, syslogd veya rsyslogd gibi logging daemon'ları çalışır durumda değildir. Bu başlatma aşamasında sistem önemli hata mesajları ve uyarıları kaybetmemek için çekirdek içerisinde mesaj deposu olarak kullanılan bir ring buffer yer almaktadır.

Ring buffer, mesajlar için ayrılmış bir hafıza alanıdır. Tasarımı basittir ve sabit bir alana sahiptir. Yeni mesajlar için yer kalmadığında eski mesajların üzerinde yazılma yapılır (overwrite). İşte çekirdekte bulunan bu ring buffer, aygit sürücülerinin başlatma mesajları, donanımdan gelen mesajlar ve çekirdek modüllerinden gelen mesajlar gibi bilgileri depolar. Bu düşük düzeyli başlangıç mesajlarını depoladığından dolayı donanım hataları ve diğer başlatma sorunlarıyla ilgili bir araştırmaya başlamak için oldukça iyi bir yerdır.

dmesg komutu da tam olarak bu amaçla kullanılan bir programdır. dmesg komutu ile ring buffer içerisinde depollanmış mesajlar görüntülenebilmektedir. Komuta ait örnek bir çıktı [şekil 48](#) üzerinde verilmiştir. Komutun çıktısı yaklaşık 4 sayfa uzunluğunda olduğundan ekran görüntüsüne sağlığı kadar gösterilmektedir.

```
[~] dmesg
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Linux version 2.6.32-754.el6.x86_64 (mockbuild@x86-01.bsys.centos.org) (gcc version 4.4.7
20120313 (Red Hat 4.4.7-23) (GCC) ) #1 SMP Tue Jun 19 21:26:04 UTC 2018
Command line: ro root=/dev/mapper/vg_centos-lv_root rd_NO_LUKS KEYBOARDTYPE=pc KEYTABLE=
trq LANG=en_US.UTF-8 rd_LVM_LV=vg_centos/lv_swap rd_NO_MD SYSFONT=latarcyrheb-sun16 rd_LV
M_LV=vg_centos/lv_root rd_NO_DM rhgb quiet
KERNEL supported cpus:
    Intel GenuineIntel
    AMD AuthenticAMD
    Centaur CentaurHauls
Disabled fast string operations
BIOS-provided physical RAM map:
BIOS-e820: 0000000000000000 - 000000000009f800 (usable)
BIOS-e820: 000000000009f800 - 00000000000a0000 (reserved)
BIOS-e820: 0000000000dc000 - 000000000100000 (reserved)
BIOS-e820: 000000000100000 - 000000000bfee0000 (usable)
BIOS-e820: 000000000bfee0000 - 000000000bfeff000 (ACPI data)
BIOS-e820: 000000000bfeff000 - 000000000bff00000 (ACPI NVS)
BIOS-e820: 000000000bff00000 - 000000000c000000 (usable)
BIOS-e820: 000000000f000000 - 000000000f800000 (reserved)
BIOS-e820: 000000000fec0000 - 000000000fec10000 (reserved)
BIOS-e820: 000000000fee0000 - 000000000fee10000 (reserved)
BIOS-e820: 000000000fffe0000 - 00000000100000000 (reserved)
BIOS-e820: 00000000100000000 - 00000000140000000 (usable)
SMBIOS version 2.7 @ 0xF69B0
SMBIOS 2.7 present.
DMI: VMware, Inc. VMware Virtual Platform/440BX Desktop Reference Platform, BIOS 6.00 07/
22/2020
Phoenix BIOS detected: BIOS may corrupt low RAM, working around it.
e820 update range: 0000000000000000 - 0000000000010000 (usable) ==> (reserved)
Hypervisor detected: VMware
e820 update range: 0000000000000000 - 00000000000010000 (usable) ==> (reserved)
e820 remove range: 00000000000a0000 - 0000000000100000 (usable)
last_pfn = 0x140000 max_arch_pfn = 0x400000000
```

Sekil 48: dmesg Komutu

2.12.1 dmesg Komutu Kullanım Seçenekleri

- **-H:** Bu parametre, dmesg komutunun çıktısını otomatik olarak "less" ile çalıştırır. Ayrıca komut çıktısındaki timestamp'lar daha okunaklı hale getirilir.
- **-T:** Bu parametre timestamp'ların standart tarih ve zaman olarak gösterilmesini sağlamaktadır.
- **-follow:** Bu parametre dmesg komutunun çalışmaya devam etmesini sağlar. Böylece yeni bir event gerçekleştiğinde (Örneğin sisteme bir USB bellek takılması) buna dair event mesajı görüntülenebilir.

Bu kullanımlar dışında komutun çıktısı piping işlemi ile grep komutuna yollandan istenilen mesajlar görüntülenebilmektedir. Örneğin "**dmesg | grep -i usb**" komutu ile çıktı içerisinde usb geçen satırlar taranabilir.

Çekirdekteki ring buffer'a yazılan mesajların tümü, kendisiyle ilişkilendirilmiş bir seviyeye sahiptir. Bu seviyeler ile mesajın önemi ölçülmektedir. Bu seviyeler:

- **emerg:** Sistem kullanılamaz durumdadır.
- **alert:** Acilen önlemler alınmalıdır.
- **crit:** Kritik kondisyonları belirtir.
- **err:** Hata durumlarını belirtir.
- **warn:** Uyarıları gösterir.
- **notice:** Normal fakat dikkate alınması gereken durumları gösterir.
- **info:** Bilgi veren mesajlardır.
- **debug:** Debugging mesajlarıdır.

Komut çıktısında spesifik bir seviyeye ait mesajların görüntülenmesi istendiğinde komut "**-l**" parametresi ile beraber kullanılmalıdır. Örneğin info seviyesi mesajlar görüntülenmek istenirse, "**dmesg -l info**" komutu kullanılmalıdır.

dmesg komutu ile mesajlar ayrıca "**facility**" denilen kategorilere ayrılmıştır. Bu kategoriler:

- **kern**: Kernel mesajlarını,
- **user**: Kullanıcı seviyesi mesajlarını,
- **mail**: Mail sistemini,
- **daemon**: Sistem daemonlarını,
- **auth**: Güvenlik / yetki mesajlarını,
- **syslog**: Dahili syslogd mesajlarını,
- **lpr**: Line printer alt sistemini,
- **news**: Ağ haber alt sistemini ifade etmektedir.

dmesg komutu "**-f**" parametresi ile kullanılarak istenilen facility'e ait mesajların filtrelenmesi sağlanabilir. Örneğin "**dmesg -f daemon**" komutu ile daemon mesajları ekrana yazdırılabilmektedir.

2.13 lsmod Komutu

Linux sistemlerde işletim sisteminin kalbi denilebilecek çekirdek, modüler bir tasarıma sahiptir. Driver da denilen çekirdek modülleri, çekirdeğin işlevsellliğini artıracak kodlardan ibarettir. Bu modüller çekirdek üzerinde iki şekilde çalışabilir, ya çekirdeğe gömülü olarak varlardır ya da çalışma esnasında çekirdeğe yüklenebilirler.

lsmod komutu, çekirdek üzerinde çalışan modülleri listelemeye yarayan bir komuttur.

Hiçbir seçenek ya da argüman kabul etmeyen bu komut, sadece **/proc/modules** dosyasını okur ve güzelce formatlayarak içeriğini ekrana yazdırır. Komuta ait örnek bir çıktı şkil 49 üzerinde görülebilir.

Module	Size	Used by
autofs4	27000	3
bnx2fc	92192	0
cnic	56026	1 bnx2fc
uio	10462	1 cnic
fcoe	23362	0
libfcoe	57623	2 bnx2fc, fcoe
libfc	111170	3 bnx2fc, fcoe, libfcoe
scsi_transport_fc	55395	3 bnx2fc, fcoe, libfc
scsi_tgt	12141	1 scsi_transport_fc
8021q	20507	0
garp	7184	1 8021q
stp	2218	1 garp
llc	5450	2 garp, stp
fuse	80180	2
vhgfs	50614	0
vsock	43273	4
ipt_REJECT	2383	2
nf_conntrack_ipv4	9218	2
nf_defrag_ipv4	1483	1 nf_conntrack_ipv4
iptable_filter	2793	1
ip_tables	17895	1 iptable_filter
ip6t_REJECT	4372	2
nf_conntrack_ipv6	7985	3
nf_defrag_ipv6	26468	1 nf_conntrack_ipv6
xt_state	1492	5
nf_conntrack	79601	3 nf_conntrack_ipv4, nf_conntrack_ipv6, xt_state
ip6table_filter	2889	1
ip6_tables	18828	1 ip6table_filter
ib_ipoib	81191	0
rdma_ucm	15739	0
ib_ucm	12360	0
ib_uverbs	40532	2 rdma_ucm, ib_ucm
ib_umad	13519	0

Şekil 49: lsmod Komutu

Şekil 49 üzerinde görülen başlıklarını açıklamak gerekirse:

- **Module:** Modüllerin isimlerinin bulunduğu sütundur.
- **Size:** Modüllerin boyutlarının (byte olarak) bulunduğu sütundur.
- **Used by:** Modüllerin kullanım sayısını göstermektedir.

Ayrıca diğer komutlarda da olduğu gibi komutun çıktısı grep ile süzülerek istenilen bir modül aranabilir. Örneğin kvm modülünün durumu sorgulanmak istenirse, "lsmod | grep kvm" komutu ile bu istek gerçekleştirilebilir.

2.14 lsof Komutu

Linux sistemlerde her şey bir dosya olarak kullanılmaktadır. İşmini "List Of Open File" kelimelerinin kısaltımından alan lsop komutu, açık olan dosyaların listesinin görüntülenmesini sağlamaktadır. Sadece sıradan dosyaları değil, özel blok dosyaları, paylaşımlı kütüphaneler, özel karakter dosyaları, pipe'lar, soketler gibi pek çok dosyanın da görüntülenebilmesini sağladığından grep gibi programlarla birlikte kullanıldığında bu dosyaların aranması ve listelenmesi işlemlerinde oldukça kullanışlı bir komuttur. Komuta ait örnek bir çıktı şekil 50 üzerinde görülebilir.

kdeinit4	3079	seyitahmet	mem	REG	253,0	146592	51	/lib64/libpt
hread-2.12.so								
kdeinit4	3079	seyitahmet	mem	REG	253,0	47760	52	/lib64/librt
-2.12.so								
kdeinit4	3079	seyitahmet	mem	REG	253,0	600048	70	/lib64/libm-
2.12.so								
kdeinit4	3079	seyitahmet	mem	REG	253,0	91096	67	/lib64/libz.
so.1.2.3								
kdeinit4	3079	seyitahmet	mem	REG	253,0	1142944	68	/lib64/libgl
ib-2.0.so.0.2800.8								
kdeinit4	3079	seyitahmet	mem	REG	253,0	124640	60	/lib64/libse
linux.so.1								
kdeinit4	3079	seyitahmet	mem	REG	253,0	114496	59	/lib64/libre
solv-2.12.so								
kdeinit4	3079	seyitahmet	mem	REG	253,0	20016	69	/lib64/libgt
hread-2.0.so.0.2800.8								
kdeinit4	3079	seyitahmet	mem	REG	253,0	311432	86	/lib64/libgo
bject-2.0.so.0.2800.8								
kdeinit4	3079	seyitahmet	mem	REG	253,0	1300408	92	/usr/lib64/l
ibX11.so.6.3.0								
kdeinit4	3079	seyitahmet	mem	REG	253,0	153576	91	/usr/lib64/l
ibxcb.so.1.1.0								
kdeinit4	3079	seyitahmet	mem	REG	253,0	13168	90	/usr/lib64/l
ibXau.so.6.0.0								
kdeinit4	3079	seyitahmet	mem	REG	253,0	76848	93	/usr/lib64/l
ibXext.so.6.4.0								
kdeinit4	3079	seyitahmet	mem	REG	253,0	167424	100	/lib64/libex
pat.so.1.5.2								
kdeinit4	3079	seyitahmet	mem	REG	253,0	646384	85	/usr/lib64/l
ibfreetype.so.6.3.22								
kdeinit4	3079	seyitahmet	mem	REG	253,0	157960	84	/usr/lib64/l
ibpng12.so.0.49.0								
kdeinit4	3079	seyitahmet	mem	REG	253,0	223040	101	/usr/lib64/l
ibfontconfig.so.1.4.4								
kdeinit4	3079	seyitahmet	mem	REG	253,0	40776	95	/usr/lib64/l

Sekil 50: lsof Komutu

Sekil 50 üzerinde görülebildiği gibi, açık olan dosyalar, dosyayı çalıştırılan kullanıcı, dosyanın tipi, boyutu gibi bilgiler komutun çıktısında yer almaktadır.

2.14.1 lsof Komutu Yardım Sayfası

Bu işlem için komut "lsof -h" şeklinde kullanılmalıdır. Bu komut ile lsof komutunun kullanımına dair bilgiler ekranda gösterilebilmektedir. Komuta ait örnek bir çıktı şekil 51 üzerinde verilmiştir.

```
lsof 4.82
latest revision: ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/
latest FAQ: ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/FAQ
latest man page: ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/lsof_man
usage: [-?abhlnNoOPRtUvVX] [+|-c c] [+|-d s] [+D D] [+|-f[gG]] [+|-e s]
[-F [f]] [-g [s]] [-i [i]] [+|-L [l]] [+m [m]] [+|-M] [-o [o]] [-p s]
[+|-r [t]] [-s [p:s]] [-S [t]] [-T [t]] [-u s] [+|-w] [-x [fl]] [-Z [Z]] [--] [names]
Defaults in parentheses; comma-separated set (s) items; dash-separated ranges.
-?-h list help      -a AND selections (OR)      -b avoid kernel blocks
-c c cmd c ^c /c/[bix] +c w COMMAND width (9)
+d s dir s files    -d s select by FD set      +D D dir D tree *SLOW?*
                     +|-e s exempt s *RISKY*   -i select IPv[46] files
                     -n no host names       -N select NFS files
                     -O avoid overhead *RISKY* -P no port names
                     -s list file size       -t terse listing
                     -U select Unix socket   -v list version info
                     +|-w Warnings (+)       -X skip TCP&UDP* files
-Z Z context [Z]
-- end option scan
+f|-f +filesystem or -file names      +|-f[gG] flaGs
-F [f] select fields; -F? for help
+|-L [l] list (+) suppress (-) link counts < l (0 = all; default = 0)
                     +m [m] use|create mount supplement
+|-M portMap registration (-)        -o o o 0t offset digits (8)
-p s exclude(^)|select PIDs         -S [t] t second stat timeout (15)
-T qs TCP/TPI Q,St (s) info
-g [s] exclude(^)|select and print process group IDs
-i i select by IPv[46] address: [46][proto][@host|addr][:svc_list|port_list]
+|-r [t[m<fmt>]] repeat every t seconds (15); + until no files, - forever.
                     An optional suffix to t is m<fmt>; m must separate t from <fmt> and
                     <fmt> is an strftime(3) format for the marker line.
-s p:s exclude(^)|select protocol (p = TCP|UDP) states by name(s).
-u s exclude(^)|select login|UID set s
-x [fl] cross over +d|+D File systems or symbolic Links
```

Şekil 51: lsof -h Komutu

2.14.2 Spesifik Bir Kullanıcıya Ait Dosyaların Görüntülenmesi

Bu işlem için komut "**lsof -u USER**" şeklinde kullanılmalıdır. Burada USER istenilen kullanıcının giriş adını belirtmektedir. Bu komut ile spesifik bir kullanıcı ait dosyaların görüntülenmesi sağlanmaktadır. Komuta ait örnek bir çıktı şekil 52 üzerinde verilmiştir.

[~] lsof -u seyitahmet						
COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF NODE NAME
gnome-key	2908	seyitahmet	cwd	DIR	253,0	4096 172705 /var/gdm
gnome-key	2908	seyitahmet	rtd	DIR	253,0	4096 2 /
gnome-key	2908	seyitahmet	txt	REG	253,0	702056 72575 /usr/bin/gnome-keyring-daemon
gnome-key	2908	seyitahmet	mem	REG	253,0	161776 24 /lib64/ld-2.12.so
gnome-key	2908	seyitahmet	mem	REG	253,0	23088 49 /lib64/libdl-2.12.so
gnome-key	2908	seyitahmet	mem	REG	253,0	1930416 25 /lib64/libc-2.12.so
gnome-key	2908	seyitahmet	mem	REG	253,0	146592 51 /lib64/libpthread-2.12.so
gnome-key	2908	seyitahmet	mem	REG	253,0	47760 52 /lib64/librt-2.12.so
gnome-key	2908	seyitahmet	mem	REG	253,0	91096 67 /lib64/libz.so.1.2.3
gnome-key	2908	seyitahmet	mem	REG	253,0	1142944 68 /lib64/libglib-2.0.so.0.2800.8
gnome-key	2908	seyitahmet	mem	REG	253,0	124640 60 /lib64/libse_linux.so.1
gnome-key	2908	seyitahmet	mem	REG	253,0	114496 59 /lib64/libresolv-2.12.so
gnome-key	2908	seyitahmet	mem	REG	253,0	20016 69 /lib64/libgthread-2.0.so.0.2800.8
gnome-key	2908	seyitahmet	mem	REG	253,0	311432 86 /lib64/libgo_bject-2.0.so.0.2800.8
gnome-key	2908	seyitahmet	mem	REG	253,0	102808 40047 /usr/lib64/libgvfscommon.so.0.0.0
gnome-key	2908	seyitahmet	mem	REG	253,0	268240 79 /lib64/libdbus-1.so.3.4.0
gnome-key	2908	seyitahmet	mem	REG	253,0	117968 41198 /usr/lib64/libgp11.so.0.0.0
gnome-key	2908	seyitahmet	mem	REG	253,0	14280 142 /lib64/libgsm

Şekil 52: lsof -u USER Komutu

Bu işlemin tersinin yapılması için yani spesifik kullanıcı hariç tüm kullanıcılar ait dosyaları görüntülemek için komut "**lsof -u ^USER**" şeklinde kullanılmalıdır.

2.14.3 Spesifik Bir İşleme Ait Dosyaların Görüntülenmesi

Bu işlem için komut "**lsof -c PROCESS**" şeklinde kullanılmalıdır. Burada PROCESS istenilen işlemin adını belirtmektedir. Bu komut ile spesifik bir işleme ait dosyaların görüntülenmesi sağlanmaktadır. Komuta ait örnek bir çıktı şe^{kil 53} üzerinde verilmiştir.

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
kdeinit4	3079	seyitahmet	cwd	DIR		253,0	4096	391695 /home/sevitahmet
kdeinit4	3079	seyitahmet	rtd	DIR		253,0	4096	2 /
kdeinit4	3079	seyitahmet	txt	REG		253,0	74504	23337 /usr/bin/kdeinit4
kdeinit4	3079	seyitahmet	mem	REG		253,0	161776	24 /lib64/ld-2.12.so
kdeinit4	3079	seyitahmet	mem	REG		253,0	23088	49 /lib64/libdl-2.12.so
kdeinit4	3079	seyitahmet	mem	REG		253,0	1930416	25 /lib64/libc-2.12.so
kdeinit4	3079	seyitahmet	mem	REG		253,0	146592	51 /lib64/libpthread-2.12.so
kdeinit4	3079	seyitahmet	mem	REG		253,0	47760	52 /lib64/librt-2.12.so
kdeinit4	3079	seyitahmet	mem	REG		253,0	600048	70 /lib64/libm-2.12.so
kdeinit4	3079	seyitahmet	mem	REG		253,0	91096	67 /lib64/libz.so.1.2.3
kdeinit4	3079	seyitahmet	mem	REG		253,0	1142944	68 /lib64/libglib-2.0.so.0.2800.8
kdeinit4	3079	seyitahmet	mem	REG		253,0	124640	60 /lib64/libselinux.so.1
kdeinit4	3079	seyitahmet	mem	REG		253,0	114496	59 /lib64/libresolv-2.12.so
kdeinit4	3079	seyitahmet	mem	REG		253,0	20016	69 /lib64/libgthread-2.0.so.0.2800.8
kdeinit4	3079	seyitahmet	mem	REG		253,0	311432	86 /lib64/libgobject-2.0.so.0.2800.8
kdeinit4	3079	seyitahmet	mem	REG		253,0	1300408	92 /usr/lib64/libX11.so.6.3.0
kdeinit4	3079	seyitahmet	mem	REG		253,0	153576	91 /usr/lib64/libxcb.so.1.1.0
kdeinit4	3079	seyitahmet	mem	REG		253,0	13168	90 /usr/lib64/libXau.so.6.0.0

Şe^{kil 53}: lsof -c PROCESS Komutu

2.14.4 lsof Komutu Kullanım Seçenekleri

Diger seçenekler dışında kullanılabilceek birkaç seçenek şunlardır:

- **-p PID**: id'si verilen işlem tarafından açılmış dosyaların görüntülenmesini sağlar.
- **-p ^PID**: id'si verilen işlem tarafından açılmış dosyaların dışındaki tüm görüntülenmesini sağlar.
- **-D path**: Yolu verilen dizin tarafından açılan tüm dosyaların listelenmesini sağlar.
- **-i**: Ağ bağlantıları tarafından açılan tüm dosyaların listelenmesini sağlar.

2.15 lspci Komutu

PCI (Peripheral Component Interconnect), bilgisayar sistemine ekstra donanımlar eklenmesi için tanımlanmış bir arayüzdür. Örneğin, sisteme bir ethernet card takılmak istensin. Kartın sistemin geri kalıyla iletişim kurabilmesi için bir protokol gerekmektedir. PCI, bu protokol olarak kullanılan standart bir arayüzdür.

lspci komutu, linux sistemlerde PCI bus'ları ve PCI alt sistemine bağlı cihazlar ile alakalı bilgiler edinebilmek için kullanılan bir komuttur. Komuta ait örnek bir çıktı şekil 54 üzerinde görülebilir.

```
[~] lspci
00:00.0 Host bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX Host bridge (rev 01)
00:01.0 PCI bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX AGP bridge (rev 01)
00:07.0 ISA bridge: Intel Corporation 82371AB/EB/MB PIIX4 ISA (rev 08)
00:07.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01)
00:07.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 08)
00:07.7 System peripheral: VMware Virtual Machine Communication Interface (rev 10)
00:0f.0 VGA compatible controller: VMware SVGA II Adapter
00:10.0 SCSI storage controller: LSI Logic / Symbios Logic 53c1030 PCI-X Fusion-MPT Dual
Ultra320 SCSI (rev 01)
00:11.0 PCI bridge: VMware PCI bridge (rev 02)
00:15.0 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.1 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.2 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.3 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.4 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.5 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.6 PCI bridge: VMware PCI Express Root Port (rev 01)
```

Şekil 54: lspci Komutu

2.15.1 lspci Komutu Kullanım Seçenekleri

- **-v:** Komutun daha ayrıntılı bir çıktı vermesini sağlar.
- **-mm:** Komutun daha okunaklı bir çıktı vermesini sağlar. -v parametresi ile birleştirilerek komutun hem ayrıntılı hem okunaklı bir çıktı vermesi sağlanabilir.
- **-k:** Hangi kernel driver'ının kullanıldığı gösterilmesini sağlar.

2.16 nvidia-smi Komutu

nvidia-smi, NVIDIA'nın Fermi ve daha yüksek mimarilerinden Tesla, QUadro, GRID ve GeForce için izleme ve yönetim kabiliyetleri sağlayan bir programdır.

2.16.1 nvidia-smi Komutu Kullanım Seçenekleri

- : Komut parametresiz kullanıldığında sisteme bağlı GPU'ların özetini ekrana yazdırır.
- **-h, --help**: Komutun kullanımıyla alakalı bilgilerin ekrana yazdırılmasını sağlar.
- **-i, --id**: Spesifik bir GPU'nun seçilmesini sağlar.
- **-f, --filename**: Çıktının belirtilen dosyaya yazılmasını sağlar.
- **-l, --loop**: Komutun "Ctrl+C" kombinasyonu girilene kadar çalışmasını sağlar.
- **-x, --xml-format**: XML formatında çıktı üretilemesini sağlar.
- **-u, --unit**: Çıktının GPU özelliklerinden ziyade ünite özellikleri şeklinde üretilmesini sağlar.
- **-d, --display**: Sadece seçilen özelliklerin gösterilmesini sağlar. Bu seçenekler MEMORY, UTILIZATION, ECC, TEMPERATURE, POWER, CLOCK, COMPUTE, PIDS, PERFORMANCE seçenekleridir. İstenilen seçenekler aralarında virgül olacak şekilde birlikte kullanılabilir. -u ve -x parametreleri ile birlikte çalışmaz.

2.17 /proc/devices Dosyası

/proc/cpuinfo gibi bir dosya olan /proc/devices dosyası, ayarlanmış karakter ve blok cihazları hakkında bilgi içeren bir dosyadır. Modülleri yüklenmemiş cihazlar bu dosyada bulunmaz.

Komuta ait örnek bir çıktı şekil 55 üzerinde görüntülenebilir.

```
[~] cat /proc/devices
Character devices:
  1 mem
  4 /dev/vc/0
  4 tty
  4 ttys
  5 /dev/tty
  5 /dev/console
  5 /dev/ptmx
  7 vcs
 10 misc
 13 input
 14 sound
 21 sg
 29 fb
 116 alsa
 128 ptm
 136 pts
 162 raw
 180 usb
 189 usb_device
 202 cpu/msr
 203 cpu/cpuid
 226 drm
 231 infiniband_mad
 231 infiniband_verbs
 231 infiniband_cm
 248 hidraw
 249 usbmon
 250 bsg
 251 pcmcia
 252 watchdog
 253 rtc
 254 tpm

Block devices:
  1 ramdisk
 259 blkext
  7 loop
  8 sd
  9 md
 11 sr
 65 sd
 66 sd
 67 sd
```

Şekil 55: /proc/devices Dosyası

Şekil 55 üzerinde görülebildiği gibi dosyanın çıktısı içerisinde major number ve cihazın ismi bulunmaktadır ve çıktı iki parçaya ayrılmıştır: Karakter cihazları ve blok cihazları.

Karakter cihazları blok cihazlarına benzemektedir fakat aralarında iki basit fark bulunmaktadır:

- **1.** Karakter cihazlarının buffering'e ihtiyacı yoktur. Blok cihazları kendilerine gelen istekleri adreslemeden sıralayabilmek için kendilerine atanan buffer'i kullanabilirler. Bu özellikle bilgi saklamak için tasarlanan cihazlar üzerinde kullanışlıdır (hard diskler gibi) çünkü gelen bilginin yazılımadan önce sıralanabilmesi daha verimli bir şekilde saklanması sağlanabilir.
- **2.** Karakter cihazları tanımlı olmayan uzunlukta veriler gönderirler. Blok cihazları ise cihaz başına önceden tanımlanmış uzunluktaki veriler üzerinden iletişim kurarlar.

2.18 /proc/softirqs Dosyası

Linux çekirdeği, önemli olmayan kesilebilir çekirdek fonksiyonlarını ikiye ayırmıştır. Bu-
lar softirqs ve taskletsler ile work queue ile çalıştırılanlardır.

Softirqs ve tasklets birbirleriyle ilişkilidir çünkü tasklets softirqs kullanılarak implement
edilmiştir. Linux çekirdeğinin kodunda geçen softirq kelimesi de genellikle iki türlü ertelenenbilir
fonksiyonu da ifade eden bir şekilde kullanılmıştır.

/proc/softirqs dosyası da bu softirq'lerin görüntülenebilmesini sağlar. Komuta ait örnek
bir çıktı şekil 56 üzerinde verilmiştir.

	CPU0	CPU1
HI:	0	0
TIMER:	41285	29232
NET_TX:	16	111
NET_RX:	25	107
BLOCK:	8838	0
BLOCK_IOPOLL:	0	0
TASKLET:	122	9
SCHED:	2695	3840
HRTIMER:	206	277
RCU:	48165	33790

Şekil 56: /proc/softirqs Dosyası

Şekil 56 üzerinde görülen başlıklardan birkaçını açıklamak gerekirse:

- **HI:** Yüksek öncelikli taskletler ile ilgilenir.
- **TIMER:** Timer kesmeleri ile ilgili tasklet sayısını gösterir.
- **NET_TX:** Paketleri ağ kartlarına iletir.
- **NET_RX:** Ağ kartlarından paketleri alır.
- **TASKLET:** Sıradan taskletlerle ilgilenir.

Başlıklar, öncelik sırasına göre sıralanmıştır.

2.19 ltrace Komutu

ltrace komutu, kendisine parametre olarak verilen komutu parametre olarak verilen komut çalışmayı durdurana kadar çalıştırın bir programdır. Bu çalışma esnasında yapılan dinamik kütüphane çağrılarını ve bu komuta gelen sinyalleri yakalar ve kaydeder. Bunun yanında parametre olarak çalıştırdığı komut tarafından yapılan sistem çağrılarını durdurabilmekte ve yazdırabilmektedir. Komuta ait örnek bir çıktı şekil 57 üzerinde gösterilmektedir.

```
[~] cat hello.c
#include <stdio.h>

int main()
{
    printf("Hello World!\n");
    return 0;
}
[~] ltrace ./hello
(0, 0, 692736, -1, 0x1f25bc2)          = 0x3a6d422160
__libc_start_main(0x4004c4, 1, 0x7ffc1dea8098, 0x4004f0, 0x4004e0 <unfinished ...>
puts("Hello World!"Hello World!
)                                     = 13
+++ exited (status 0) +++
[~] █
```

Şekil 57: ltrace Komutu

2.19.1 ltrace Komutu Yardım Sayfası

Bu işlem için komut "**ltrace -h**" şeklinde kullanılmalıdır. Bu komut ile ltrace komutunun kullanımı ile alakalı bilgiler ekrana yazdırılabilmektedir. Komuta ait örnek bir çıktı şekil 58 üzerinde verilmiştir.

```
[~] ltrace -h
Usage: ltrace [option ...] [command [arg ...]]
Trace library calls of a given program.

-á, --align=COLUMN  align return values in a secific column.
-c                  count time and calls, and report a summary on exit.
-C, --demangle      decode low-level symbol names into user-level names.
-d, --debug         print debugging info.
      --dl          show calls to symbols in dlopened libraries.
-e expr             modify which events to trace.
-f                  follow forks.
-h, --help          display this help and exit.
-i                  print instruction pointer at time of library call.
-l, --library=FILE  print library calls from this library only.
-L                  do NOT display library calls.
-n, --indent=NR     indent output by NR spaces for each call level nesting.
-o, --output=FILE   write the trace output to that file.
-p PID              attach to the process with the process ID pid.
-r                  print relative timestamps.
-s STRLEN           specify the maximum string size to print.
-S                  display system calls.
-t, -tt, -ttt       print absolute timestamps.
-T                  show the time spent inside each call.
-u USERNAME         run command with the userid, groupid of username.
-V, --version       output version information and exit.
-x NAME             treat the global NAME like a library subroutine.

Report bugs to ltrace-devel@lists.alioth.debian.org
[~] █
```

Şekil 58: ltrace -h Komutu

2.19.2 ltrace Komutu Kullanım Seçenekleri

- **-a:** Belirli bir sütundaki dönüş değerlerinin hizalanmasını sağlar. Varsayılan sütun ekran genişliğinin 5/8'i kadardır.
- **-A:** Geri kalanı üç nokta (...) ile yazdırmadan önce yazdırılacak maksimum öğe sayısının belirtilmesini sağlar. Aynı zamanda recursive açılımların sayısını da sınırlar.
- **-b:** İzlenen işlem tarafından alınan sinyallerin yazdırılmasını devre dışı bırakır.
- **-c:** Her kütüphane çağrısi için zaman ve çağrıların sayılmasını sağlar ve programın çıkışında bu sayımların özetini rapor eder.
- **-C:** Düşük seviyeli simbol adlarının daha anlaşılır bir şekilde yazdırılmasını sağlar. Sistem tarafından kullanılan '_' prefix'in kaldırmasının yanı sıra C++ fonksiyon isimlerinin daha okunabilir hale getirilmesini de sağlamaktadır.
- **-D:** ltrace komutunun debug çıktısını gösterir. Genellikle 77 mask'ı ile kullanılır, böylece tüm debug mesajları görüntülenebilir.
- **-F:** Alternatif bir config dosyası ile çalışılabilmesini sağlar. Varsayılan olaran /etc/ltrace.conf ya da /.ltrace.conf dosyaları kullanılmaktadır.
- **-i:** Kütüphane çağrısi sırasında instruction pointer'in yazdırılmasını sağlar.
- **-n:** Komutun çıktısına girdi verilebilmesini sağlar. Bu seçenek ile programın akışının görselleştirilmesi kolaylaştırılabilir.
- **-o:** Çıktının stderr yerine bir dosyaya gönderilmesini sağlar.
- **-s:** Yazdırılacak maksimum string uzunluğunu belirtir. Varsayılan olarak 32'dir.
- **-S:** Kütüphane çağrılarının yanında sistem çağrılarının da gösterilmesini sağlar.
- **-T:** Çağrıların harcadığı sürenin (Bitiş Süresi - Başlangıç Süresi) gösterilmesini sağlar.

2.20 lttng Komutu

İsmi "Linux Trace Toolkit: next generation" kelimelerinin kısaltımından alan lttng komutu, linux çekirdeği, kullanıcı uygulamaları ve kullanıcı kütüphanelerinin izlenebilmesini sağlayan bir komuttur.

lttng komutu, linux çekirdeğini izleyebilmek için linux çekirdek modülleri ile kullanıcı uygulamaları ve kütüphanelerini izleyebilmek için dinamik olarak yüklenilebilen kütüphanelerden oluşmaktadır.

Komut ile kullanılabilen 5 adet tracing domain bulunmaktadır. Komuta ait bazı seçeneklerde bu domainlerin belirtilmesi gerekmektedir. Bu domainler:

- **-j, -jul:** Komutun java.util.logging (JUL) domainine uygulanmasını sağlar.
- **-k, -kernel:** Komutun linux çekirdek domainine uygulanmasını sağlar.
- **-l, -log4j:** Komutun apache log4j (Java) domainine uygulanmasını sağlar.
- **-p, -python:** Komutun python domainine uygulanmasını sağlar.
- **-u, -userspace:** Komutun kullanıcı alan domainine uygulanmasını sağlar.

2.20.1 lttng Komutu Kullanım Seçenekleri

- **-g, --group:** Unix izleme grubunun değiştirilmesini sağlar. Varsayılan grup tracing grubudur.
- **-m, --mi:** Komutun çıktısının insan-okunabilir şekilde olması yerine makine arayüz tipinde olmasını sağlar. Komut, xml'i desteklemektedir. MI modu ile geleneksel formattı yazdırma yerine makine çıktısı sözdizimi kullanılır.
- **-n, --no-sessionid:** Otomatik olarak yeni bir oturun daemonu oluşturulmasını engeller.
- **-q, --quiet:** Uyarı ve hatalar da dahil olmak üzere tüm mesajların yazdırılmasını engeller.
- **-v, --verbose:** Komutun daha çok bilgi vermesini sağlar. Üç seviyesi bulunmaktadır: -v, -vv, -vvv.
- **-h, --help:** Komutun yardım sayfasının görüntülenmesini sağlar.
- **--list-commands:** Komut ile birlikte kullanılabilen komutların listelenmesini sağlar.
- **--list-options:** Komut ile birlikte kullanılabilen genel seçeneklerin listelenmesini sağlar.
- **-V, --version:** lttng komutunun versiyonunun ekrana yazdırılmasını sağlar.

2.21 ps Komutu

Linux, multitasking ve multi-user bir sistemdir. Dolayısıyla birden çok işlemin birbirlerine müdahale etmeden aynı anda çalışmasına izin verir. İşlem, linux sistemlerde önemli bir konsepttir. İşlem, bir programın çalışan bir parçasıdır ve işletim sistemi içerisinde kendisine verilen işlerin yapılmasını sağlamaktadır.

ps komutu, linux sistemlerde çalışan işlemlerin görüntülenebilmesini sağlayan bir komuttur. İsmi "Process Status" kelimelerinin baş harflerinden almaktadır. ps komutu ile sistemde çalışan işlemler, bu işlemlerin id'leri ile beraber kendisine verilen seçeneklerle değişkenlik gösteren farklı bilgilerin listelenmesini sağlamaktadır. Verdiği bilgileri /proc dosya sistemindeki dosyaları okuyarak elde etmektedir. Komuta ait örnek bir çıktı şekil 59 üzerinde görülebilir.

```
[~] ps
  PID TTY          TIME CMD
 4184 pts/1        00:00:00 zsh
 4222 pts/1        00:00:00 ps
[~] █
```

Şekil 59: ps Komutu

Şekil 59 üzerinde görülen başlıklarını açıklamak gerekirse:

- **PID:** Process'in ID'sini gösterir.
- **TTY:** Kullanıcının giriş yapmış olduğu terminal tipini gösterir.
- **TIME:** İşlemenin dakika ve saniye olarak çalışma süresini gösterir.
- **CMD:** İşlemi çalıştıran komutun adını gösterir.

Bazen ps komutu çalıştırıldığında TIME başlığı 00:00:00 olarak gösterilmektedir (Şekil 59 üzerinde olduğu gibi). Bu, o işleme çekirdek tarafından hiç CPU zamanı verilmediğini göstermektedir. Örneğin zsh komutunun TIME başlığında değeri 00:00:00 olarak gösterilmektedir çünkü zsh'in kendisi CPU zamanı kullanmamıştır, CPU'yu kullanan işlemlerin parent process'i olarak çalışmıştır.

2.21.1 ps Komutu Kullanım Seçenekleri

- **-A, -e:** Çalışan tüm işlemlerin görüntülenmesini sağlar.
- **-a:** Bir terminal ile ilişkilendirilmemiş tüm işlemlerin görüntülenmesini sağlar.
- **-d:** Oturum liderleri hariç tüm işlemlerin listelenmesini sağlar.
- **-N, -deselect:** Negate operatörü olarak kullanılır. Kendisinden önceki parametrelerin tersi işleminin yapılmasını sağlar.
- **-T:** Açık olan terminalle ilişkilendirilmiş tüm işlemlerin listelenmesini sağlar.
- **-x:** Komutu çalıştırılan kullanıcıya ait tüm işlemlerin listelenmesini sağlar.
- **-C:** Verilen komut adı ile eşleşen işlemlerin listelenmesini sağlar.
- **-G:** Verilen grup id'si ya da adı ile eşleşen işlemlerin listelenmesini sağlar.
- **-p:** Verilen process id'si ile eşleşen işlemlerin listelenmesini sağlar.
- **-s:** Verilen session id'si ile eşleşen işlemlerin listelenmesini sağlar.
- **-u:** Verilen kullanıcı adı ya da id'si ile eşleşen işlemlerin listelenmesini sağlar.

2.22 stap Komutu

stap komutu, systemtap aracının front-end komutudur. Basit bir scripting dilinde yazılmış inceleme talimatlarını alır, bu talimatları C koduna çevirir, kodu derler ve istenilen sistem izleme / araştırma işlevlerini gerçekleştirmek için ortaya çıkan çekirdek modülünü çalışan bir linux çekirdeğine yükler. Program kullanıcı tarafından kesilene, script içerisinde exit() fonksiyonu çalıştırılana ya da yeterli sayıda soft error alınana kadar çalışır.

Scripting dili kesinlikle yazılmıştır (strictly typed), bildirimsizdir (declaration free), prosedürelidir (procedural) ve awk'dan esinlenmiştir. Çekirdekteki kaynak kod noktalarının veya eventlerin, eşzamanlı olarak yürütülen alt rutinler (subroutines) olan işleyicilerle ilişkilendirilmesine izin verir. Kavramsal olarak gdb programındaki kesme noktası komut listelerine benzemektedir.

2.22.1 stap Komutu Kullanım Seçenekleri

- **-h, -help:** Komutun yardım sayfasını gösterir.
- **-V, -version:** Komutun versiyonunu gösterir.
- **-p NUM:** NUM kadar pass aşamasından sonra programı durdurur. Toplamda 5 adet pass aşaması bulunmaktadır. Bunlar parse, elaborate, translate, compile ve run aşamalarıdır.
- **-v:** Komutun çıktısında daha fazla bilgi vermesini sağlar.
- **-k:** Tüm işlemlerden sonra geçici olarak oluşturulan klasörün silinmemesini sağlar. İşlem sırasında oluşturulan C kodunun incelenmesi ya da oluşturulan çekirdek modülünün tekrar kullanılabilmesi gibi işlemler için idealdir.
- **-g:** Guru modunu aktif hale getirir. Gömülü C kodu gibi güvenli olmayan expert-level yapıların kullanılabilmesini sağlar.
- **-P:** Prologue-searching modu aktif hale getirir. Bu mod ile \$target değişkenleri için hatalı debugging bilgilerini düzeltebilmek için heuristikler aktif edilir.
- **-u:** Optimizasyonu devre dışı bırakır. Elaboration aşamasında kullanılmayan kodun seçilmesi devre dışı bırakılır.
- **-w:** Uyarı mesajlarını devre dışı bırakır.

Komut ile birlikte kullanılabilecek diğer seçenekler ve bahsedilen scripting dili ile ilgili bilgiler için [bu link](#) ziyaret edilebilir.

2.23 strace Komutu

strace, linux sistemler üzerindeki en güclü süreç izleme (monitoring), teşhis (diagnostic) ve öğretim (instructional) araçlarından biridir. Bu işlevlerinin yanında sorunların giderilmesinde kullanılan bir hata ayıklama aracı olarak da görev yapmaktadır. strace komutunun sıkılıkla kullanıldığı alanlar şunlardır:

- Programların hata ayıklama (debugging) işlemleri
- Programların hata giderme (troubleshooting) işlemleri
- Bir işlem tarafından yapılan sistem çağrılarına müdahale edilmesi
- Bir işlem tarafından yapılan sistem çağrılarının kaydedilmesi
- Bir işlemin aldığı sinyallerin görüntülenmesi
- Çalışan işlemlerin takip edilmesi.

Kaynak kodun kullanılabilir olmadığı durumlarda strace komutu programın sistemle nasıl bir etkileşim içerisinde olduğunu analiz etmek için çalışan programı debug eder. Yapılan tüm sistem çağrılarının adını, argümanlarını ve döndürdüğü değerleri stderr'e yazdırır. Komuta ait örnek bir çıktı şekil 60 üzerinde verilmiştir.

Şekil 60: strace Komutu

2.23.1 strace Komutu Yardım Sayfası

Bu işlem için komut "**strace -help**" şeklinde kullanılmalıdır. Bu komut ile strace komutunun kullanımı ile alakalı bilgiler ekrana yazdırılmaktadır. Komuta ait örnek bir çıktı şe⁶¹kil 61 üzerinde gösterilmektedir.

```
[~] strace -help
usage: strace [-CdfhiqrttTvVxxy] [-I n] [-e expr]...
              [-a column] [-o file] [-s strsize] [-P path]...
              [-p pid... / [-D] [-E var=val]... [-u username] PROG [ARGS]
or: strace -c[df] [-I n] [-e expr]... [-O overhead] [-S sortby]
              [-p pid... / [-D] [-E var=val]... [-u username] PROG [ARGS]
-c -- count time, calls, and errors for each syscall and report summary
-C -- like -c but also print regular output
-d -- enable debug output to stderr
-D -- run tracer process as a detached grandchild, not as parent
-f -- follow forks, -ff -- with output into separate files
-i -- print instruction pointer at time of syscall
-q -- suppress messages about attaching, detaching, etc.
-r -- print relative timestamp, -t -- absolute timestamp, -tt -- with usecs
-T -- print time spent in each syscall
-v -- verbose mode: print unabbreviated argv, stat, termios, etc. args
-x -- print non-ascii strings in hex, -xx -- print all strings in hex
-y -- print paths associated with file descriptor arguments
-h -- print help message, -V -- print version
-a column -- alignment COLUMN for printing syscall results (default 40)
-b execve -- detach on this syscall
-e expr -- a qualifying expression: option=[!]all or option=[!]val1[,val2]...
          options: trace, abbrev, verbose, raw, signal, read, write
-I interruptible --
  1: no signals are blocked
  2: fatal signals are blocked while decoding syscall (default)
  3: fatal signals are always blocked (default if '-o FILE PROG')
  4: fatal signals and SIGTSTP (^Z) are always blocked
    (useful to make 'strace -o FILE PROG' not stop on ^Z)
-o file -- send trace output to FILE instead of stderr
-O overhead -- set overhead for tracing syscalls to OVERHEAD usecs
-p pid -- trace process with process id PID, may be repeated
-s strsize -- limit length of print strings to STRSIZE chars (default 32)
-S sortby -- sort syscall counts by: time, calls, name, nothing (default time)
-u username -- run command as username handling setuid and/or setgid
-E var=val -- put var=val in the environment for command
-E var -- remove var from the environment for command
-P path -- trace accesses to path
[~] █
```

Şe⁶¹kil 61: strace -help Komutu

2.23.2 Yapılan Sistem Çağrısı Sayısının Görüntülenmesi

Bu işlem için komut "**strace -c CMD**" şeklinde kullanılmalıdır. Burada CMD, strace ile birlikte çalıştırılan programı belirtmektedir. Bu komut ile çalıştırılan programın yaptığı sistem çağrıları ve çağrı sayısı görüntülenebilmektedir. Komuta ait örnek bir çıktı şekil 62 üzerinde görülebilmektedir.

```
[~] strace -c pwd
/home/seyyitahmet
% time      seconds   usecs/call     calls    errors syscall
-----
100.00  0.000183       61          3      open
      0.00  0.000000       0          1      read
      0.00  0.000000       0          1      write
      0.00  0.000000       0          5      close
      0.00  0.000000       0          4      fstat
      0.00  0.000000       0         10      mmap
      0.00  0.000000       0          3      mprotect
      0.00  0.000000       0          2      munmap
      0.00  0.000000       0          3      brk
      0.00  0.000000       0          1      1 access
      0.00  0.000000       0          1      execve
      0.00  0.000000       0          1      getcwd
      0.00  0.000000       0          1      arch_prctl
-----
100.00  0.000183           36        1 total
[~] █
```

Şekil 62: strace -c CMD Komutu

2.23.3 Spesifik Sistem Çağrılarının Görüntülenmesi

Bu işlem için komut "**strace -e trace=X CMD**" şeklinde kullanılmalıdır. Burada X, istenilen sistem çağrısının adını; CMD ise çalıştırılmak istenen programı temsil etmektedir. Komuta ait örnek bir çıktı şekil 63 üzerinde verilmiştir.

```
[~] strace -e trace=open pwd
open("/etc/ld.so.cache", O_RDONLY)      = 3
open("/lib64/libc.so.6", O_RDONLY)      = 3
open("/usr/lib/locale/locale-archive", O_RDONLY) = 3
/home/seyyitahmet
+++ exited with 0 ***
[~] █
```

Şekil 63: strace -e trace=X CMD Komutu

2.23.4 strace Komutu Kullanım Seçenekleri

Bahsedilen başlıklar dışında kullanılabilecek birkaç seçenekin listelenmesi gereklidir:

- **-r:** Her çağrıda timestamp'in yazdırılması sağlanır.
- **-T:** Her çağrı sırasında geçen zamanın yazdırılması sağlanır.
- **-t:** Her çağrıda wall clock time'in yazdırılması sağlanır.
- **-i:** Instruction pointer'in yazdırılması sağlanır.
- **-o:** Komutun çıktısının bir dosyaya yazdırılması sağlanır.

2.24 sysdig Komutu

sysdig komutu, sistem yöneticilerinin ve geliştiricilerin işine oldukça yarayabilecek bir biçimde sistemin davranışları ile ilgili faydalı bilgiler veren bir komuttur. Programın geliştirici ekibi; depolama, iletim, ağ ve bellek alt sistemlerinin merkezi, tutarlı ve ayrıntılı bir görünümünü sağlayarak sistem seviyesi izleme ve sorun giderme işlemlerinin kolaylaştırılmasını amaçlayarak bu programı ortaya çıkarmışlardır. Ayrıca bu işlemler için bir filtreleme dili de oluşturmuşlardır. Bu dil ile ele alınan verilerin doğal ve interaktif bir yol ile işlenmesi sağlanmıştır. Bunun yanında sıkılıkla karşılaşılan sorunları giderebilmek amacıyla lua scriptlerinden oluşan ve "chisel" denilen büyük de bir kütüphane oluşturmuştur. sysdig komutu; strace, tcpdump ve lsop komutlarının oldukça güçlü bir versiyonu olarak görülebilir. Özette, bir sistemin durumunu ve faaliyetlerini analiz etmek amacıyla kullanılan sağlam bir performans analiz programıdır.

sysdig komutu parametresiz çalıştırıldığında sistemde çalışan eventleri listelemeye devam etmektedir. Komuta ait örnek bir çıktı şekil 64 üzerinde verilmiştir. Komuta dair detaylı kullanım seçeneklerini incelemek için [bu link](#) ziyaret edilebilir.

```
[~] sudo sysdig
[sudo] password for seyithahmet:
7 06:43:33.256173356 1 sysdig (19204) > switch next=0 pgft_maj=0 pgft_min=2735 vm_size=6
9704 vm_rss=5712 vm_swap=0
8 06:43:33.257387799 1 <NA> (0) > switch next=19204(sysdig) pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
23 06:43:33.258905248 1 sysdig (19204) > switch next=0 pgft_maj=0 pgft_min=2780 vm_size=69708 vm_rss=5840 vm_swap=0
24 06:43:33.260033089 1 <NA> (0) > switch next=12 pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
25 06:43:33.260069534 1 <NA> (12) > switch next=19204(sysdig) pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
40 06:43:33.260148740 1 sysdig (19204) > switch next=0 pgft_maj=0 pgft_min=2785 vm_size=69708 vm_rss=5860 vm_swap=0
41 06:43:33.260167590 0 <NA> (0) > switch next=3374(konsole) pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
42 06:43:33.260181405 0 konsole (3374) > switch next=11 pgft_maj=44 pgft_min=8711 vm_size=529820 vm_rss=29356 vm_swap=0
43 06:43:33.260224671 0 <NA> (11) > switch next=0 pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
44 06:43:33.260308804 1 <NA> (0) > switch next=3374(konsole) pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
45 06:43:33.260315948 1 konsole (3374) < poll res=1 fds=10:f1
46 06:43:33.260330147 1 konsole (3374) > recvmsg fd=8(<u>)
47 06:43:33.260333462 1 konsole (3374) < recvmsg res=-11(EAGAIN) size=4096 data=tuple=NULL
48 06:43:33.260345639 1 konsole (3374) > ioctl fd=10(<f>/dev/ptmx) request=541B argument=7FFDE335560C
49 06:43:33.260348332 1 konsole (3374) < ioctl res=0
50 06:43:33.260349480 1 konsole (3374) > read fd=10(<f>/dev/ptmx) size=566
51 06:43:33.260352685 1 konsole (3374) < read res=566 data=7 06:43:33.256173356 1 sysdig (19204) > switch next=0 pgft_maj=0 pgft_min=2735 v
52 06:43:33.260412554 1 konsole (3374) > recvmsg fd=8(<u>)
53 06:43:33.260413688 1 konsole (3374) < recvmsg res=-11(EAGAIN) size=4096 data=tuple=NULL
54 06:43:33.260415874 1 konsole (3374) > poll fds=3:p1 9:u1 8:u1 5:u1 10:f1 23:p1 timeout=9
55 06:43:33.260421608 1 konsole (3374) > switch next=0 pgft_maj=44 pgft_min=8711 vm_size=529820 vm_rss=29356 vm_swap=0
```

Şekil 64: sysdig Komutu

2.24.1 Spesifik İşlemlerin Filtrelenmesi

Bu işlem için komut "sysdig proc.name=X" şeklinde kullanılmalıdır. Burada X, istenilen işlemin adını belirtmektedir. Bu komut ile istenilen işlem ya da işlemlere ait sonuçlar filtrelenebilmektedir. Komuta ait örnek bir çıktı şekil 65 üzerinde gösterilmektedir.

```
[~] sudo sysdig proc.name=cat [60/61]
[sudo] password for seyitahmet:
72722 06:52:37.238926312 0 cat (19545) < execve res=0 exe=cat args=hello.c. tid=19545(cat) pid=19545(cat) ptid=19506(zsh) cwd= fdlimit=1024 pgft_maj=0 pgft_min=81 vm_size=288 vm_rss=20 vm_swap=0 comm=cat cgroups=cpuset/.ns=/cpu_cgroup/.cpuacct/.mem_cgroup/.devices/.freezer=.net_cls... env=SSH_AGENT_PID=2995.KDE_MULTIHEAD=false.HOSTNAME=localhost.localdomain.IMSETTI... tty=34819 pgid=19545(cat) loginuid=500
72723 06:52:37.238945197 0 cat (19545) > brk addr=0
72724 06:52:37.238946162 0 cat (19545) < brk res=1787000 vm_size=288 vm_rss=48 vm_swap=0
72725 06:52:37.238964884 0 cat (19545) > mmap addr=0 length=4096 prot=3(PROT_READ|PROT_WRITE) flags=10(MAP_PRIVATE|MAP_ANONYMOUS) fd=4294967295 offset=0
72726 06:52:37.238966712 0 cat (19545) < mmap res=7F619432A000 vm_size=292 vm_rss=100 vm_swap=0
72727 06:52:37.238973946 0 cat (19545) > access mode=4(R_OK)
72728 06:52:37.238979506 0 cat (19545) < access res=-2(ENOENT) name=/etc/ld.so.preload
72729 06:52:37.238987077 0 cat (19545) > open
72730 06:52:37.238990539 0 cat (19545) < open fd=3(<f>/etc/ld.so.cache) name=/etc/ld.so.cache flags=1(0_RDONLY) mode=0 dev=FD00
72731 06:52:37.238991648 0 cat (19545) > fstat fd=3(<f>/etc/ld.so.cache)
72732 06:52:37.238992443 0 cat (19545) < fstat res=0
72733 06:52:37.238992950 0 cat (19545) > mmap addr=0 length=98096 prot=1(PROT_READ) flags=2(MAP_PRIVATE) fd=3(<f>/etc/ld.so.cache) offset=0
72734 06:52:37.238994305 0 cat (19545) < mmap res=7F6194312000 vm_size=388 vm_rss=136 vm_swap=0
72735 06:52:37.238994883 0 cat (19545) > close fd=3(<f>/etc/ld.so.cache)
72736 06:52:37.238995274 0 cat (19545) < close res=0
72737 06:52:37.239033441 0 cat (19545) > open
72738 06:52:37.239039734 0 cat (19545) < open fd=3(<f>/lib64/libc.so.6) name=/lib64/libc.so.6 flags=1(0_RDONLY) mode=0 dev=FD00
72739 06:52:37.239040826 0 cat (19545) > read fd=3(<f>/lib64/libc.so.6) size=832
72740 06:52:37.239042220 0 cat (19545) < read res=832 data=.ELF.....@.....0a.....@.8...@.N.M.....@. .....
72741 06:52:37.239044979 0 cat (19545) > fstat fd=3(<f>/lib64/libc.so.6)
72742 06:52:37.239045721 0 cat (19545) < fstat res=0
72743 06:52:37.239047143 0 cat (19545) > mmap addr=3A6DA00000 length=3750184 prot=5(PROT_READ|PROT_EXEC) flags=1026(MAP_PRIVATE|MAP_DENYWRITE) fd=3(<f>/lib64/libc.so.6) offset=0
72744 06:52:37.239049451 0 cat (19545) < mmap res=3A6DA00000 vm_size=4052 vm_rss=180 vm_swap=0
72745 06:52:37.239050133 0 cat (19545) > mprotect
72746 06:52:37.239139969 0 cat (19545) > switch next=11 pgft_maj=0 pgft_min=122 vm_size=4052 vm_rss=180 vm_swap=0
72748 06:52:37.239165865 0 cat (19545) < mprotect
72749 06:52:37.239167386 0 cat (19545) > mmap addr=3A6DD8A000 length=24576 prot=3(PROT_READ|PROT_WRITE) flags=1030(MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE) fd=3(<f>/lib64/libc.so.6) offset=1613824
72750 06:52:37.239173930 0 cat (19545) < mmap res=3A6DD8A000 vm_size=4052 vm_rss=180 vm_swap=0
72751 06:52:37.239179629 0 cat (19545) > mmap addr=3A6DD90000 length=14632 prot=3(PROT_READ|PROT_WRITE) flags=14(MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS) fd=4294967295 offset=0
72752 06:52:37.239181366 0 cat (19545) < mmap res=3A6DD90000 vm_size=4052 vm_rss=184 vm_swap=0
```

Şekil 65: sysdig proc.name=X Komutu

Bu şekilde kullanımda, işlemler mantıksal operatörler kullanılarak birleştirilebilir. Örneğin komut "sysdig proc.name=cat or proc.name=vim" şeklinde kullanıldığında cat veya vim programlarına ait işlemler filtrelenebilir.

2.24.2 Sistem İşlemlerinin Listelenmesi

Bu işlem için komut "sysdig -c ps" şeklinde kullanılmalıdır. Bu komut ile sistem üzerinde çalışan işlemlerin listelenmesi sağlanmaktadır. Komuta ait örnek bir çıktı şe^{kil 66} üzerinde verilmiştir.

TID	PID	USER	VIRT	RES	FDLIMIT	CMD
1	1	root	18.90M	1.54M	429496729	init
574	574	root	11.02M	1.38M	429496729	udevd
1507	1507	root	29.07M	872.00KB	429496729	auditd
1508	1507	root	29.07M	872.00KB	429496729	auditd
1541	1541	root	243.32M	1.66M	429496729	rsyslogd
1542	1541	root	243.32M	1.66M	429496729	rs:main
1543	1541	root	243.32M	1.66M	429496729	rsyslogd
1544	1541	root	243.32M	1.66M	429496729	rsyslogd
1592	1592	root	17.95M	728.00KB	429496729	irqbalance
1618	1618	rpc	18.54M	892.00KB	429496729	rpcbind
1796	1796	root	13.08M	716.00KB	429496729	lldpad
1975	1975	root	8.16M	408.00KB	429496729	fcoemon
1982	1982	root	180.53M	1.14M	429496729	vmware-vmblock-
1985	1982	root	180.53M	1.14M	429496729	vmware-vmblock-
1986	1982	root	180.53M	1.14M	429496729	vmware-vmblock-
2029	2029	root	243.22M	4.44M	429496729	vmtoolsd
2092	2092	root	52.81M	6.11M	429496729	VGAuthService
2129	2129	dbus	31.81M	1.87M	429496729	dbus-daemon
2131	2129	dbus	31.81M	1.87M	429496729	dbus-daemon
2161	2029	root	243.22M	4.44M	429496729	vmtoolsd
2193	2193	root	171.12M	4.65M	429496729	NetworkManager
2200	2200	root	56.77M	2.39M	429496729	modem-manager
2216	2216	rpcuser	22.80M	1.35M	429496729	rpc.statd
2263	2263	root	8.90M	1.52M	429496729	dhclient
2264	2193	root	171.12M	4.65M	429496729	NetworkManager
2266	2266	root	184.71M	3.32M	429496729	cupsd
2267	2267	root	43.95M	700.00KB	429496729	wpa_supplicant
2317	2317	root	3.98M	656.00KB	429496729	acpid
2369	2369	haldaemon	38.50M	5.54M	429496729	hald
2370	2370	root	19.92M	1.18M	429496729	hald-runner
2371	2369	haldaemon	38.50M	5.54M	429496729	hald
2415	2415	root	21.99M	1.09M	429496729	hald-addon-inpu
2416	2416	root	21.98M	1.10M	429496729	hald-addon-stor
2421	2421	haldaemon	17.58M	1.02M	429496729	hald-addon-acpi
2448	2448	root	89.13M	1.55M	429496729	pcscd
2454	2448	root	89.13M	1.55M	429496729	pcscd
2468	2468	root	377.11M	1.90M	429496729	automount
2469	2468	root	377.11M	1.90M	429496729	automount
2470	2468	root	377.11M	1.90M	429496729	automount
2485	2468	root	377.11M	1.90M	429496729	automount
2488	2468	root	377.11M	1.90M	429496729	automount
2506	2506	ntp	30.02M	2.09M	429496729	ntpd

Şekil 66: sysdig -c ps Komutu

2.24.3 Ağ Bağlantılarının Listelenmesi

Bu işlem için komut "sysdig -c netstat" şeklinde kullanılmalıdır. Bu komut ile sistem üzerinde yapılan ağ bağlantıları listelenebilmektedir. Komuta ait örnek bir çıktı şkil 67 üzerinde görülebilir.

```
[~] sudo sysdig -c netstat
Proto Server Address          Client Address          State      TID/PID/Program Name
tcp   127.0.0.1:25            0.0.0.0:*           LISTEN    2592/2592/master
udp   0.0.0.0:111             0.0.0.0:*           LISTEN    1618/1618/rpcbind
udp   0.0.0.0:944             0.0.0.0:*           LISTEN    1618/1618/rpcbind
tcp   0.0.0.0:111             0.0.0.0:*           LISTEN    1618/1618/rpcbind
udp   :::111                  0.0.0.0:*           LISTEN    1618/1618/rpcbind
udp   :::944                  0.0.0.0:*           LISTEN    1618/1618/rpcbind
tcp   :::111                  0.0.0.0:*           LISTEN    1618/1618/rpcbind
tcp   127.0.0.1:631           0.0.0.0:*           LISTEN    2266/2266/cupsd
udp   0.0.0.0:631             0.0.0.0:*           LISTEN    2266/2266/cupsd
tcp   :::631                  0.0.0.0:*           LISTEN    2266/2266/cupsd
udp   0.0.0.0:39006            0.0.0.0:*           LISTEN    2216/2216/rpc.statd
tcp   0.0.0.0:52784            0.0.0.0:*           LISTEN    2216/2216/rpc.statd
udp   :::54124                0.0.0.0:*           LISTEN    2216/2216/rpc.statd
tcp   :::47112                0.0.0.0:*           LISTEN    2216/2216/rpc.statd
udp   127.0.0.1:703            0.0.0.0:*           LISTEN    2216/2216/rpc.statd
udp   fe80::20c:29ff:fee5:37dc 0.0.0.0:*           LISTEN    2506/2506/ntpd
udp   :::123                  0.0.0.0:*           LISTEN    2506/2506/ntpd
udp   127.0.0.1:123            0.0.0.0:*           LISTEN    2506/2506/ntpd
udp   :::123                  0.0.0.0:*           LISTEN    2506/2506/ntpd
udp   0.0.0.0:123              0.0.0.0:*           LISTEN    2506/2506/ntpd
udp   192.168.213.137:123     0.0.0.0:*           LISTEN    2506/2506/ntpd
udp   0.0.0.0:68                0.0.0.0:*           LISTEN    2263/2263/dhclient
[~]
```

Şekil 67: sysdig -c netstat Komutu

2.24.4 İşlemlerin CPU Kullanım Oranlarına Göre Sıralı Gösterilmesi

Bu işlem için komut "sysdig -c topprocs_cpu" şeklinde kullanılmalıdır. Bu komut ile sistem üzerinde çalışan işlemlerden CPU kullanım oranı en fazla olan işlemler listelenebilmektedir. Komuta ait örnek bir çıktı şkil 68 üzerinde görülebilir.

CPU%	Process	PID
9.96%	Xorg	2750
4.98%	konsole	3374
3.99%	plasma-desktop	3126
1.99%	sysdig	19726
1.00%	tmux	19464
0.00%	rs:main	1541
0.00%	rsyslogd	1541
0.00%	gvfsd	3178
0.00%	klipper	3182
0.00%	gdm-binary	2708

Şekil 68: sysdig -c topprocs_cpu Komutu

2.25 blktrace Komutu

blktrace komutu, istek kuyruğu (request queue) işlemleri hakkında kullanıcı alanına kadar ayrıntılı bilgi sağlayabilen bir blok katmanı I/O izleme komutudur. Bu komut ile belirli bir blok cihazı için I/O eventleri izlenebilir ve bir dosyaya kaydedilebilir. blktrace komutu, event izlerini çekirdekten çıkartmak (extract) amacıyla kullanılabilen bir programdır.

2.25.1 blktrace Komutu Kullanım Seçenekleri

- **-A:** Filtreleme maskesinin değiştirilebilmesini sağlar.
- **-a:** Ayarlı maskenin filtreye uygulanmasını sağlar.
- **-b:** Event çıkarımı işlemi için buffer boyutunun ayarlanması sağlar. Varsayılan değeri 512KiB'dır.
- **-d:** Verilen cihazın izlenmesini sağlar.
- **-I:** Verilen dosya içerisinde bulunan cihazların izlenmesini sağlar.
- **-n:** Kullanılacak buffer sayısının değiştirilmesini sağlar. Varsayılan değer 4'tür.
- **-l:** Komutu ağ dinleme modunda çalıştırır.
- **-h:** Komutu ağ istemci modunda çalıştırır. Verilen host'a ile bağlantı kurulur.
- **-p:** Kullanılacak ağ portunun değiştirilmesini sağlar. Varsayılan port 8462'dir.
- **-D:** Komutun çıktısının bir dosyaya yazılmasını sağlar.
- **-v:** Komutun versiyon bilgisini gösterir.

Yukarıda bahsedilen seçeneklerden filtreleme maskeleri için kullanılan -a parametresi ile birlikte kullanılabilecek maskeler ise şunlardır:

- **barrier**: barrier özelliği
- **complete**: sürücü tarafından tamamlanmış (completed by driver)
- **fs**: istekler (requests)
- **issue**: sürücü tarafından verilen (issued by driver)
- **queue**: kuyruk operasyonları (queue operations)
- **read**: okuma izleri (read traces)
- **requeue**: tekrar kuyruk operasyonları (requeue operations)
- **sync**: senkronize özelliği (synchronous attribute)
- **write**: yazma izleri (write traces)
- **notify**: iz mesajları (trace messages)
- **drv_data**: sürücü özel ek izler (additional driver specific trace)

2.25.2 blktrace Komutu Örnek Kullanımlar

Örneğin blktrace komutu "**blktrace /dev/sda /dev/sdb**" şeklinde kullanırsa, komut /dev/sda ve /dev/sdb disklerini izleyecek ve kaydettiği bilgileri çalıştırıldığı dizinde sda ve sdb dosyaları içerisine kaydedecektir. Elde edilen bilgiler daha sonra blkparse komutu kullanılarak formatlanıp daha rahat bir şekilde incelenebilir.

Başka bir örnek olarak komut "**blktrace -d /dev/sda -o - | blkparse -i -**" şeklinde kullanıldığında, blktrace komutu /dev/sda diskü üzerindeki I/O işlemlerini izleyecek, bu komuttan elde edilen çıktı blkparse komutuna gönderilerek insan okunabilir formatta (human readable format) çıktı verecektir. Böylece disk üzerindeki I/O'lar izlenebilecektir.

2.26 df Komutu

İsmi "Disk Free" kelimelerinin kısaltımından alan df komutu, dosya sistemlerinin sahip olduğu toplam alan ve kullanılabilir alan hakkında bilgi almak için kullanılan bir komuttur. Komuta ait örnek bir çıktı şe^{kil 69} üzerinde verilmiştir.

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
/dev/mapper/vg_centos-lv_root	27228028	6846396	20103708	26%	/
tmpfs	2028160	712	2027448	1%	/dev/shm
/dev/sda1	487652	41728	420324	10%	/boot

Şe^{kil 69}: df Komutu

2.26.1 df Komutu Yardım Sayfası

Bu işlem için komut "**df –help**" şeklinde kullanılmalıdır. Bu komut ile df komutunun kullanımına dair bilgiler ekrana yazdırılmaktadır. Komuta ait örnek bir çıktı şe^{kil 70} üzerinde gösterilmektedir.

```
[~] df --help
Usage: df [OPTION]... [FILE]...
Show information about the file system on which each FILE resides,
or all file systems by default.

Mandatory arguments to long options are mandatory for short options too.
-ä, --all           include dummy file systems
-B, --block-size=SIZE use SIZE-byte blocks
--direct           show statistics for a file instead of mount point
--total            produce a grand total
-h, --human-readable print sizes in human readable format (e.g., 1K 234M 2G)
-H, --si             likewise, but use powers of 1000 not 1024
-i, --inodes         list inode information instead of block usage
-k                 like --block-size=1K
-l, --local          limit listing to local file systems
--no-sync           do not invoke sync before getting usage info (default)
-P, --portability    use the POSIX output format
--sync              invoke sync before getting usage info
-t, --type=TYPE      limit listing to file systems of type TYPE
-T, --print-type     print file system type
-x, --exclude-type=TYPE limit listing to file systems not of type TYPE
-v                  (ignored)
--help              display this help and exit
--version           output version information and exit

Display values are in units of the first available SIZE from --block-size,
and the DF_BLOCK_SIZE, BLOCK_SIZE and BLOCKSIZE environment variables.
Otherwise, units default to 1024 bytes (or 512 if POSIXLY_CORRECT is set).

SIZE may be (or may be an integer optionally followed by) one of following:
KB 1000, K 1024, MB 1000*1000, M 1024*1024, and so on for G, T, P, E, Z, Y.

Report df bugs to bug-coreutils@gnu.org
GNU coreutils home page: <http://www.gnu.org/software/coreutils/>
General help using GNU software: <http://www.gnu.org/gethelp/>
For complete documentation, run: info coreutils 'df invocation'
[~] █
```

Şe^{kil 70}: df –help Komutu

2.26.2 Boyutların 1024'ün Katı Olarak Gösterilmesi

Bu işlem için komut "**df -h**" şeklinde kullanılmalıdır. Bu komut ile df komutunun çıktısında gösterilen alan boyutları 1024'ün katı olarak formatlanır, böylece boyutlar insanlar tarafından daha rahatlıkla anlaşılabılır olmaktadır. Komuta ait örnek bir çıktı şekil [71](#) üzerinde verilmiştir.

```
[~] df -h
Filesystem           Size  Used  Avail Use% Mounted on
/dev/mapper/vg_centos-lv_root
                      26G   6.6G   20G  26% /
tmpfs                 2.0G  712K  2.0G   1% /dev/shm
/dev/sda1              477M   41M  411M  10% /boot
[~] █
```

Şekil 71: df -h Komutu

2.26.3 Boyutların 1000'in Katı Olarak Gösterilmesi

Bu işlem için komut "**df -H**" şeklinde kullanılmalıdır. Bu komut ile df komutunun çıktısında gösterilen alan boyutları 1000'in katı olarak formatlanır, böylece boyutlar insanlar tarafından daha rahatlıkla anlaşılabılır olmaktadır. Komuta ait örnek bir çıktı şekil [72](#) üzerinde görülebilmektir.

```
[~] df -H
Filesystem           Size  Used  Avail Use% Mounted on
/dev/mapper/vg_centos-lv_root
                      28G   7.1G   21G  26% /
tmpfs                 2.1G  730k  2.1G   1% /dev/shm
/dev/sda1              500M   43M  431M  10% /boot
[~] █
```

Şekil 72: df -H Komutu

2.26.4 df Komutu Kullanım Seçenekleri

Bahsedilen başlıklardan farklı olarak df komutu ile kullanılabilcek bazı seçenekler şunlardır:

- **-a, -all:** Sahte (psuedo), kopyalı (duplicate) ve erişilemez (inaccessible) dosya sistemleri de dahil tüm sistemlere dair bilgilerin gösterilmesini sağlar.
- **-B, -block-size:** Yazdırılmadan önce blok boyutunun ayarlanılmasını sağlar.
- **-i, -inodes:** Blok kullanımı yerine inode bilgisinin gösterilmesini sağlar.
- **-l, -local:** Sadece yerel dosya sistemlerinin listelenmesini sağlar.
- **-P, -portability:** POSIX çıktı formatının kullanılmasını sağlar.
- **-sync:** Kullanım bilgilerinin alınmasından önce sync'yi çalıştırır.
- **-total:** Kullanılabilir alanın hesaplanması işleminde önemsiz olan tüm girdileri eleyerek genel bir toplam değer üretmesini sağlar.
- **-t, -type:** Listeleme işleminin verilen dosya tipi ile sınırlanmasını sağlar.
- **-T, -print-type:** Dosya sistemi tiplerinin de çıktıya dahil edilmesini sağlar.

2.27 du Komutu

İsmini "Disk Usage" kelimelerinin kısaltımından alan du komutu, dosya alanı kullanımının ölçülmesi işleminde kullanılmaktadır. du komutu ile dosya sistemi üzerinde aşırı yer kaplayan dosya ve dizinlerin tespiti gerçekleştirilebilmektedir. Komuta ait örnek bir çıktı şekil 73 üzerinde verilmiştir.

```
[staj] du
312      ./hafta3/linux-dosya-sistemi
8        ./hafta3/linux-packets
12      ./hafta3/linux-kullanicilar
336      ./hafta3
64        ./hafta4
16      ./hafta2
8        ./hafta1
428      .
[staj]
```

Şekil 73: du Komutu

2.27.1 Çıktının İnsan Okunabilir Formatta Üretilmesi

Bu işlem için komut "**du -h**" şeklinde kullanılmalıdır. Bu komut ile üretilen çıktıların çoğu insan için anlamsız sayılar yerine anlamlı birimler cinsinden üretilmesi sağlanmaktadır. Komuta ait örnek bir çıktı şekil 74 üzerinde gösterilmektedir.

```
[staj] du -h
312K      ./hafta3/linux-dosya-sistemi
8.0K      ./hafta3/linux-packets
12K      ./hafta3/linux-kullanicilar
336K      ./hafta3
64K      ./hafta4
16K      ./hafta2
8.0K      ./hafta1
428K      .
[staj]
```

Şekil 74: du -h Komutu

2.27.2 Çıktıya Dosya ve Dizinlerin Dahil Edilmesi

Bu işlem için komut "**du -a**" şeklinde kullanılmalıdır. Bu komut ile üretilen çıktıya dizin içerisindeki tüm dosya ve dizinlerin dahil edilmesi sağlanmaktadır. Komuta ait örnek bir çıktı şekil [75](#) üzerinde verilmiştir.

```
[staj] du -a
4      ./hafta3/linux-dosya-sistemi/hafta3-6.txt
4      ./hafta3/linux-dosya-sistemi/hafta3-5.txt
4      ./hafta3/linux-dosya-sistemi/hafta3-2.txt
4      ./hafta3/linux-dosya-sistemi/hafta3-4.txt
4      ./hafta3/linux-dosya-sistemi/hafta3-7.txt
256     ./hafta3/linux-dosya-sistemi/linux-dosya-sistemi-hfta3.pdf
20      ./hafta3/linux-dosya-sistemi/toword.txt
4      ./hafta3/linux-dosya-sistemi/hafta3-3.txt
8      ./hafta3/linux-dosya-sistemi/hafta3-1.txt
312     ./hafta3/linux-dosya-sistemi
4      ./hafta3/linux-packets/1.txt
8      ./hafta3/linux-packets
8      ./hafta3/linux-kullanicilar/1.txt
12      ./hafta3/linux-kullanicilar
336     ./hafta3
12      ./hafta4/4.txt
28      ./hafta4/1.txt
16      ./hafta4/2.txt
4      ./hafta4/3.txt
64      ./hafta4
8      ./hafta2/hafta2.txt
4      ./hafta2/hafta2helper.txt
16      ./hafta2
4      ./hafta1/hafta1.txt
8      ./hafta1
428     .
[staj] █
```

Şekil 75: du -a Komutu

2.27.3 Çıktıya Toplam Kullanımın Dahil Edilmesi

Bu işlem için komut "**du -c**" şeklinde kullanılmalıdır. Bu komut ile çıktıının son satırına komutun çağrıldığı dizinin toplam boyutu da eklenecektir. Komuta ait örnek bir çıktı şe^{kil 76} überinde görülebilmektedir.

```
[staj] du -c -h
312K    ./hafta3/linux-dosya-sistemi
8.0K     ./hafta3/linux-packets
12K     ./hafta3/linux-kullanicilar
336K    ./hafta3
64K     ./hafta4
16K     ./hafta2
8.0K    ./hafta1
428K    .
428K    total
[staj] █
```

Şe^{kil 76:} du -c Komutu

2.27.4 Sadece Toplam Kullanımım Çıktı Olarak Üretilmesi

Bu işlem için komut "**du -s**" şeklinde kullanılmalıdır. Komuta ait örnek bir çıktı şe^{kil 77} überinde gösterilmektedir.

```
[staj] du -s -h
428K .
[staj] █
```

Şe^{kil 77:} du -s Komutu

2.28 find Komutu

Adını yaptığı işten alan find komutu, dosya ve dizinlerin aranmasında kullanılan bir komuttur. Dosya ve dizinleri isim, oluşturulma tarihi, düzenlenme tarihi, sahiplik ve yetkiler gibi pek çok özelliklerini kullanarak arayabilmektedir.

2.28.1 find Komutu Yardım Sayfası

Bu işlem için komut "**find -help**" şeklinde kullanılmalıdır. Bu komut ile find komutunun kullanımına dair bilgiler ekrana yazdırılmaktadır. Komuta ait örnek bir çıktı şekil 78 üzerinde görülebilmektedir.

```
[staj] find --help
Usage: find [-H] [-L] [-P] [-Olevel] [-D help|tree|search|stat|rates|opt|exec] [path...] [expression]

default path is the current directory; default expression is -print
expression may consist of: operators, options, tests, and actions:

operators (decreasing precedence; -and is implicit where no others are given):
  ( EXPR )    ! EXPR    -not EXPR    EXPR1 -a EXPR2    EXPR1 -and EXPR2
  EXPR1 -o EXPR2    EXPR1 _or EXPR2    EXPR1 , EXPR2

positional options (always true): -daystart -follow -regextype

normal options (always true, specified before other expressions):
  -depth --help -maxdepth LEVELS -mindepth LEVELS -mount -noleaf
  --version -xdev -ignore_readdir_race -noignore_readdir_race

tests (N can be +N or -N or N): -amin N -anewer FILE -atime N -cmin N
  -cnewer FILE -ctime N -empty -false -fstype TYPE -gid N -group NAME
  -ilname PATTERN -iname PATTERN -inum N -iwholename PATTERN -iregex PATTERN
  -links N -lname PATTERN -min N -mtime N -name PATTERN -newer FILE
  -nouser -nogroup -path PATTERN -perm [+/-]MODE -regex PATTERN
  -readable -writable -executable
  -wholename PATTERN -size N[bckwMG] -true -type [bcdpfslsD] -uid N
  -used N -user NAME -xtype [bcdpfsls]
  -context CONTEXT

actions: -delete -print0 -printf FORMAT -fprintf FILE FORMAT -print
  -fprint0 FILE -fprint FILE -ls -fls FILE -prune -quit
  -exec COMMAND ; -exec COMMAND {} + -ok COMMAND ;
  -execdir COMMAND ; -execdir COMMAND {} + -okdir COMMAND ;

Report (and track progress on fixing) bugs via the findutils bug-reporting
page at http://savannah.gnu.org/ or, if you have no web access, by sending
email to <bug-findutils@gnu.org>.
[staj]
```

Şekil 78: find -help Komutu

2.28.2 İsim ile Arama Yapma

Bu işlem için komut "**find DIR -name NAME**" şeklinde kullanılmalıdır. Burada DIR, arama yapılacak dizini, NAME ise aranacak dosyanın adını ifade etmektedir. Bu komut ile istenilen dizin içerisinde ismi verilen dosyanın aranması sağlanmaktadır. Komuta ait örnek bir çıktı [şekil 79](#) üzerinde verilmiştir.

```
[staj] find . -name hafta2.txt  
./hafta2/hafta2.txt  
[staj]
```

Şekil 79: **find DIR -name NAME** Komutu

Komutun bu şekilde kullanımında dosya ismi tam bilinmiyorsa wildcard karakterleri kullanılarak arama yapılabilir. Örneğin [şekil 79](#) üzerinde görülen örnek değiştirilirse ve dosyanın adının "hafta2.txt" olduğunu bilinmediği varsayılrsa arama "**find . -name "haf*.txt"**" şeklinde de yapılabilir.

2.28.3 Boş Dosya ve Dizinlerin Aranması

Bu işlem için komut "**find DIR -empty**" şeklinde kullanılmalıdır. Burada DIR, arama yapılacak dizini ifade etmektedir. Komuta ait örnek bir çıktı [şekil 80](#) üzerinde gösterilmektedir.

```
[staj] find . -empty  
.emptyfolder  
.empty.txt  
[staj]
```

Şekil 80: **find DIR empty** Komutu

2.28.4 İzinler ile Arama Yapılması

Bu işlem için komut "find DIR -perm PERM şeklinde kullanılmalıdır. Burada DIR, arama yapılacak dizini; PERM ise istenilen izin numarasını ifade etmektedir. Komuta ait örnek bir çıktı şekil 81 üzerinde görülebilir.

```
[staj] find . -perm 664
./hafta3/linux-dosya-sistemi/hafta3-6.txt
./hafta3/linux-dosya-sistemi/hafta3-5.txt
./hafta3/linux-dosya-sistemi/hafta3-2.txt
./hafta3/linux-dosya-sistemi/hafta3-4.txt
./hafta3/linux-dosya-sistemi/hafta3-7.txt
./hafta3/linux-dosya-sistemi/toword.txt
./hafta3/linux-dosya-sistemi/hafta3-3.txt
./hafta3/linux-dosya-sistemi/hafta3-1.txt
./hafta3/linux-packets/1.txt
./hafta3/linux-kullanicilar/1.txt
./hafta4/4.txt
./hafta4/1.txt
./hafta4/2.txt
./hafta4/3.txt
./hafta2/hafta2.txt
./hafta2/hafta2helper.txt
./empty.txt
./hafta1/hafta1.txt
[staj] █
```

Şekil 81: find DIR -perm PERM Komutu

2.29 iostat Komutu

iostat komutu, cihazlar ve disk bölümleri için I/O istatistiklerinin görüntülenmesi işlemelerinde kullanılan bir komuttur. Bu işlemi, cihazların aktif olduğu süre ile ortalama transfer oranlarını inceleyerek gerçekleştirmektedir. iostat komutu tarafından üretilen raporlar kullanılarak fiziksel diskler arasındaki I/O dengesi arttırılabilir. Komuta ait örnek bir çıktı şekil 82 üzerinde verilmiştir.

```
[staj] iostat
Linux 2.6.32-754.el6.x86_64 (localhost.localdomain)      08/16/2021      _x86_64_
2 CPU)

avg-cpu: %user   %nice %system %iowait  %steal   %idle
          0.37    0.00    0.41    0.34    0.00   98.87

Device:         tps   Blk_read/s   Blk_wrtn/s   Blk_read   Blk_wrtn
scd0           0.01     0.06        0.00        360          0
sda            2.60    132.51       17.56    758322     100506
dm-0           7.16    130.75       17.54    748290     100392
dm-1           0.06     0.45        0.00      2600          0

[staj] █
```

Şekil 82: iostat Komutu

Şekil 82 üzerinde görülenleri açıklamak gerekirse:

- **%user**: Kullanıcı seviyesinde çalıştırılırken kullanılan CPU yüzdesini ifade eder.
- **%nice**: Kullanıcı seviyesinde çalıştırılırken nice öncelikli kullanılan CPU yüzdesini ifade eder.
- **%system**: Sistem (çekirdek) seviyesinde çalıştırılırken kullanılan CPU yüzdesini ifade eder.
- **%iowait**: Sistemin olağanüstü bir I/O işlemi ile uğraştığı sırada CPU'nun boşta kalma yüzdesini gösterir.
- **%steal**: Hipervizör (hypervisor) başka bir sanal işlemci tarafından hizmet verirken sanal CPU'nun istem dışı beklemede harcadığı zamanın yüzdesini gösterir.
- **%idle**: Sistem I/O ile uğraşmadığı sırada CPU'nun boşta kalma yüzdesini gösterir.
- **Device**: /dev/directory dizininde listelenen cihaz ya da bölüm adını ifade eder.
- **tps**: Cihaz tarafından yapılan saniye başına transferi ifade eder. Yüksek tps değeri, işlemcinin meşguliyetinin arttığını ifade etmektedir.
- **Blk_read/s**: Bir saniyede cihaz tarafından okunan veriyi temsil etmektedir.
- **Blk_wrt/s**: Bir saniyede cihaz tarafından yazılan veriyi temsil etmektedir.
- **Blk_read**: Toplamda okunan veriyi temsil etmektedir.
- **Blk_wrt**: Toplamda yazılan veriyi temsil etmektedir.

iostat komutu, bu bilgileri şu dizinlerden almaktadır:

- **/proc/stat**: Sistem istatistiklerini içerir.
- **/proc/uptime**: Sistemin çalışma zamanını içerir.
- **/proc/diskstats**: Disk istatistiklerini içerir.
- **/sys**: Blok cihazlarının istatistiklerini içerir.
- **/proc/self/mountstats**: Ağ dosya sistemlerinin istatistiklerini içerir.
- **/dev/disk**: Cihaz isimlerini içerir.

2.29.1 iostat Komutu Kullanım Seçenekleri

- **-x:** Komutun daha detaylı bir çıktı üretmesini sağlar.
- **-c:** Sadece CPU istatistiklerinin gösterilmesini sağlar.
- **-d:** Sadece cihaz raporlarının gösterilmesini sağlar.
- **-k:** İstatistiklerin kilobyte ya da megabyte olarak yakalanmasını sağlar.
- **-p:** Blok cihazları istatistiklerinin gösterilmesini sağlar.
- **-N:** lvm2 istatistiklerinin görüntülenmesini sağlar.

2.30 iotop Komutu

iotop komutu, disk I/O kullanım detaylarının görüntülenmesi ve izlenmesi amacıyla kullanılan bir komuttur. Çalışması için python programına ve çekirdek modüllerine ihtiyaç duymaktadır. Sistem yöneticileri tarafından spesifik bir işlemin sebep olabileceği yüksek I/O okuma/yazma işlemlerinin incelenmesine kullanılmaktadır. top komutuna benzer bir çıktı üreten iotop komutu, çalışmak için genellikle root yetkilerine ihtiyaç duymaktadır. Komuta ait örnek bir çıktı şe⁸³kil 83 üzerinde gösterilmektedir.

Total DISK READ: 0.00 B/s Total DISK WRITE: 0.00 B/s							
TID	PRIo	USER	DISK READ	DISK WRITE	SWAPIN	IO>	COMMAND
1	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	init
2	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kthreadd]
3	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[migration/0]
4	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ksoftirqd/0]
5	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[stopper/0]
6	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[watchdog/0]
7	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[migration/1]
8	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[stopper/1]
9	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ksoftirqd/1]
10	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[watchdog/1]
11	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[events/0]
12	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[events/1]
13	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[events/0]
14	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[events/1]
15	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[events_long/0]
16	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[events_long/1]
17	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[events_power_ef]
18	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[events_power_ef]
19	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[cgroup]
20	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[khelper]
21	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[netns]
22	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[async/mgr]
23	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[pm]
24	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[sync_supers]
25	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[bdi-default]
26	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kintegrityd/0]
27	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kintegrityd/1]
28	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kblockd/0]
29	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kblockd/1]
30	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kacpid]
31	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kacpi_notify]
32	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kacpi_hotplug]
33	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ata_aux]
34	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ata_sff/0]
35	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ata_sff/1]
36	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ksuspend_usbd]
37	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[khubd]
38	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kseriod]

Şekil 83: iotop Komutu

2.30.1 iotop Komutu Yardım Sayfası

Bu işlem için komut "**iotop -h**" şeklinde kullanılmalıdır. Bu komut ile iotop komutunun kullanımına dair bilgiler ekrana yazdırılabilmektedir. Komuta ait örnek bir çıktı şekil 84 üzerinde görülebilir.

```
[staj] iotop -h
Usage: /usr/sbin/iotop [OPTIONS]

DISK READ and DISK WRITE are the block I/O bandwidth used during the sampling
period. SWAPIN and IO are the percentages of time the thread spent respectively
while swapping in and waiting on I/O more generally. PRI0 is the I/O priority at
which the thread is running (set using the ionice command).

Controls: left and right arrows to change the sorting column, r to invert the
sorting order, o to toggle the --only option, p to toggle the --processes
option, a to toggle the --accumulated option, q to quit, any other key to force
a refresh.

Options:
--version          show program's version number and exit
-h, --help         show this help message and exit
-o, --only         only show processes or threads actually doing I/O
-b, --batch        non-interactive mode
-n NUM, --iter=NUM number of iterations before ending [infinite]
-d SEC, --delay=SEC delay between iterations [1 second]
-p PID, --pid=PID processes/threads to monitor [all]
-u USER, --user=USER users to monitor [all]
-P, --processes   only show processes, not all threads
-a, --accumulated show accumulated I/O instead of bandwidth
-k, --kilobytes   use kilobytes instead of a human friendly unit
-t, --time         add a timestamp on each line (implies --batch)
-q, --quiet        suppress some lines of header (implies --batch)
[staj] █
```

Şekil 84: iotop -h Komutu

2.30.2 I/O İşlemi Yapan İşlemlerin Görüntülenmesi

Bu işlem için komut "**iostop -o**" şeklinde kullanılmalıdır. Bu komut ile sadece o an I/O işlemi yapan işlemlerin görüntülenmesi sağlanmaktadır. Komuta ait örnek bir çıktı [Şekil 85](#) üzerinde verilmiştir.

Total DISK READ: 0.00 B/s Total DISK WRITE: 76.68 K/s							
TID	PRIo	USER	DISK READ	DISK WRITE	SWAPIN	IO>	COMMAND
468	be/3	root	0.00 B/s	3.83 K/s	0.00 %	0.04 %	[jbd2/dm-0-8]

Şekil 85: iostop -o Komutu

2.30.3 Spesifik Bir İşleme Ait Bilgilerin Görüntülenmesi

Bu işlem için komut "**iostop -p PID**" şeklinde kullanılmalıdır. Burada PID, istenilen işlemin id'sini ifade etmektedir. Komuta ait örnek bir çıktı [Şekil 86](#) üzerinde görülebilmektedir.

Total DISK READ: 0.00 B/s Total DISK WRITE: 0.00 B/s							
TID	PRIo	USER	DISK READ	DISK WRITE	SWAPIN	IO>	COMMAND
4599	be/4	seyitahm	0.00 B/s	0.00 B/s	0.00 %	0.00 %	zsh

Şekil 86: iostop -p PID Komutu

2.30.4 iostop Komutu Kullanım Seçenekleri

Bahsedilen kullanım seçeneklerinin dışında birkaç seçenekten daha bahsetmek gereklidir:

- **-version:** Komutun versiyonunun gösterilmesini sağlar.
- **-b:** Komutun batch modunda çalıştırılmasını sağlar. Bu modda komut ile etkileşime girmemektedir.
- **-n NUM:** Komutun NUM iterasyonda çalışmasını sağlar.
- **-a:** Çıktının birikmiş bir halde gösterilmesini sağlar (accumulated).
- **-t:** Her satırda timestamp eklenmesini sağlar.
- **-q:** Başlıktaki bazı satırların gizlenmesini sağlar.

2.31 swapon Komutu

swapon komutu, paging ve swapping işlemlerinin yapılmak üzere cihazların seçilmesini sağlar. Seçilen cihaz ya da dosya, "-L LABEL" ya da "-U uuid" şeklinde belirtilebilmektedir.

2.31.1 swapon Komutu Kullanım Seçenekleri

- **-a, --all:** /etc/fstab dosyası içerisinde "swap" olarak işaretlenen tüm cihazlar kullanılabilir hale getirilir (noauto seçeneğine sahip olanlar hariç). Halihazırda kullanımında olanlar es geçilir.
- **-d, --discard:** Free edilmiş sayfaların tekrar kullanılmadan önce discard edilmesini sağlar. SSD cihazlarda performansı arttıracaktır fakat genellikle arttırmamaktadır.
- **-e, --ifexists:** Var olmayan cihazların es geçilmesini sağlar.
- **-f, --fixpgsz:** Sayfanın boyutu çalışan çekirdek ile uyumuyorsa swap alanının tekrar initialize edilmesini sağlar.
- **-h, --help:** Komutun yardım sayfasının görüntülenmesini sağlar.
- **-L label:** Verilen label'e sahip bölüm kullanılır (/proc/partitions dosyasına erişim gereklidir).
- **-p, --priority p:** Swap cihazının önceliğinin belirtilmesini sağlar. Bu değer -1 ile 32767 arasında olmalıdır. Varsayılan değer -1'dir.
- **-s, --summary:** Cihazın swap kullanımını gösterir. "cat /proc/swaps" komutuna eşdeğerdir. Linux 2.1.25 sürümünden önceki sürümlerde kullanılamaz.
- **-U uuid:** UUID'si verilen bölümün swap olarak kullanılmasını sağlar.
- **-v, --verbose:** Çıktının daha ayrıntılı bilgi içermesini sağlar.
- **-V, --version:** Komutun versiyonu hakkında bilgi yazdırır.

2.32 jcmd Komutu

jcmd komutu, kullanımını destekleyen bir JVM'e tanılama istekleri gönderilmesini sağlayan bir komuttur.

Herhangi bir parametre verilmeden veya -l parametresi ile kullanıldığında çalışan Java işlemlerini, bu işlemlerin id'lerini, main class'larını ve komut satırı argümanlarını yazdırmaktadır.

Komut bir process id'si ile çalıştırılırsa, id'si verilen process üzerinde çalışır.

Komut bir main class ismi ile çalıştırılırsa, o main class'a ait tüm processleri listeler.

Komut PerfCounter.print argümanı ile çalıştırılırsa, Java işlemleri içerisinde kullanılabilir performans sayaçlarını yazdırır.

Komut -f argümanı ile birlikte çalıştırılırsa gönderilecek tanılama istekleri bir dosyada okunabilir.

Komuta ait örnek bir çıktı şekil 87 üzerinde verilmiştir.

```
[~] cat Hello.java
public class Hello {
    public static void main(String[] args) {
        while (true);
    }
}
[~] jcmd
7163 sun.tools.jcmd.JCmd
7036 Hello
[~]
```

Şekil 87: jcmd Komutu

2.33 jdb Komutu

İsmi java debugger kelimelerinin kısaltımından alan jdb komutu, java sınıfları için kullanılabilecek basit bir komut satırı debugger'ıdır. Yerel ya da uzaktaki bir JVM üzerinde kullanılabilir.

jdb komutunu başlatmak için birden çok yol bulunmaktadır. En sık kullanılan yol, derlenmiş java sınıfının jdb komutu ile çalıştırılmasıdır. Bu şekilde çalıştırıldığında, jdb verilen parametrelere ile ikinci bir VM başlatır, belirtilen sınıfı yükler ve sınıfın çalıştırılabilir ilk koduna geldiğinde komutları bekler.

jdb komutunun çalıştırılması için kullanılan bir diğer yol, çalışan bir java VM'ine jdb'yi attach etmektir.

Komuta ait örnek bir çıktı şe^{kil 88} überinde görülebilmektedir.

```
[~] cat Hello.java
public class Hello {
    public static void main(String[] args) {
        while (true);
    }
}
[~] jdb Hello
help
Initializing jdb ...
> ** command list **
connectors          -- list available connectors and transports in this VM
run [class [args]]  -- start execution of application's main class

threads [threadgroup]      -- list threads
thread <thread id>       -- set default thread
suspend [thread id(s)]    -- suspend threads (default: all)
resume [thread id(s)]     -- resume threads (default: all)
where [<thread id> | all] -- dump a thread's stack
wherei [<thread id> | all]-- dump a thread's stack, with pc info
up [n frames]             -- move up a thread's stack
down [n frames]            -- move down a thread's stack
kill <thread id> <expr>   -- kill a thread with the given exception object
interrupt <thread id>    -- interrupt a thread

print <expr>              -- print value of expression
dump <expr>               -- print all object information
eval <expr>                -- evaluate expression (same as print)
set <elvalue> = <expr>     -- assign new value to field/variable/array element
locals                   -- print all local variables in current stack frame

classes                  -- list currently known classes
class <class id>          -- show details of named class
methods <class id>         -- list a class's methods
fields <class id>          -- list a class's fields

threadgroups             -- list threadgroups
threadgroup <name>        -- set current threadgroup

stop in <class id>. <method>[(argument_type,...)]
                           -- set a breakpoint in a method
stop at <class id>:<line> -- set a breakpoint at a line
```

Şe^{kil 88:} jdb Komutu

2.33.1 jdb İçerisinde Kullanılabilen Komutlar

- **cont**: Belirtilen bir breakpoint, exception ya da step'ten sonra programın çalıştırılmasına devam eder.
- **dump**: Primitive değerler için bu komut, print komutu ile eşdeğerdir. Objeler için, objenin içindeki tüm alanların o anki değerlerini yazdırır. Static ve instance alanları da buna dahildir.
- **help, ?:**: Bu komut ile jdb içerisinde kullanılabilecek komutlar ve ne iş yaptıkları ekrana yazdırılır.
- **print**: Primitive değerler için değişkenin değerini yazdırır, objeler ile kullanıldığında ise obje hakkında kısa bir özet yazdırılır.
- **thread**: Çalıştırılacak thread'in seçilmesini sağlar. Thread belirtmek için threads komutunun çıkışlarındaki indexlerden biri kullanılmalıdır.
- **threads**: Çalışan threadlerin listelenmesini sağlar. Tüm threadler için, thread'in adını, anlık durumunu ve diğer komutlar içerisinde kullanılabilecek index değerini yazdırır.
- **run**: jdb komutu çalıştırıldıktan ve gerekli breakpoint'ler set edildikten sonra bu komut ile programın çalıştırılmaya başlaması sağlanabilir. Bu komut sadece program jdb ile başlatıldığında kullanılabilir.
- **where**: Argümansız kullanıldığında, kullanımda olan thread'in stack'ı içerisindeki değerleri yazdırır. all argümanı ile kullanılırsa aynı işlemi tüm threadler için gerçekleştirir.
- **stop at CLASS:LINE**: Bu komut ile istenilen class içerisindeki LINE satırına breakpoint yerleştirilir.
- **step**: Sıradaki satırın çalıştırılmasını sağlar, şimdiki stack frame'de ya da çağrılan metotun içerisinde olsa da çalıştırır.
- **next**: Sadece şimdiki stack frame'deki sıradaki satırın çalıştırılmasını sağlar.

Bahsedilen komutların birkaçının kullanımını içeren örnek bir kullanım durumu [Şekil 89](#) üzerinde gösterilmektedir.

```
[~] cat Hello.java
public class Hello {
    public static void main(String[] args) {
        int bound = 10;
        for (int i = 0; i < bound; i++) {
            System.out.println("current: " + String.valueOf(i));
        }
    }
}
[~] jdb Hello
Initializing jdb ...
> stop at Hello:3
Deferring breakpoint Hello:3.
It will be set after the class is loaded.
> run
run Hello
Set uncaught java.lang.Throwable
Set deferred uncaught java.lang.Throwable
>
VM Started: Set deferred breakpoint Hello:3

Breakpoint hit: "thread=main", Hello.main(), line=3 bci=0
3           int bound = 10;

main[1] step
>
Step completed: "thread=main", Hello.main(), line=4 bci=3
4           for (int i = 0; i < bound; i++) {

main[1] step
>
Step completed: "thread=main", Hello.main(), line=5 bci=10
5           System.out.println("current: " + String.valueOf(i));

main[1] print i
i = 0
main[1] print bound
bound = 10
main[1] step
> current: 0

Step completed: "thread=main", Hello.main(), line=4 bci=38
4           for (int i = 0; i < bound; i++) {

main[1] print i
i = 0
main[1] step
>
Step completed: "thread=main", Hello.main(), line=5 bci=10
5           System.out.println("current: " + String.valueOf(i));

main[1] print i
i = 1
  ■
```

Şekil 89: jdb-usage Komutu

2.34 jinfo Komutu

jinfo komutu, verilen bir Java işlemi ya da core dosyası ya da remote debug server'e ait Java konfigürasyon bilgilerini yazdırır. Bu bilgiler, Java sistem özelliklerini ve JVM komut satırı bayraklarını içermektedir.

Komutun kullanımı takip eden sekillerde olmalıdır:

- jinfo [option] pid
- jinfo [option] executable core
- jinfo [option] [server-id@]remote-hostname-or-IP

2.34.1 jinfo Komutu Kullanım Seçenekleri

- : Komut argümansız çalıştırıldığında, tüm komut satırı bayraklarını ve sistem özelliklerine ait isim, değer çiftlerini ekrana yazdırmaktadır.
- **-flags:** Sadece komut satırı bayraklarını isim, değer çiftleri olarak ekrana yazdırmaktadır.
- **-sysprops:** Sadece Java sistem özelliklerini isim, değer çiftleri olarak ekrana yazdırmaktadır.
- **-h, -help:** Komutun yardım mesajını ekrana yazdırmaktadır.

2.35 jmap Komutu

jmap komutu, verilen işlemin ya da core file'in ya da remote debug server'in paylaşmalı obje hafıza haritası ya da heap hafıza detaylarının yazdırılmasını sağlayan bir komuttur.

Komutun kullanımı takip eden şekillerde olmalıdır:

- jinfo [option] pid
- jinfo [option] executable core
- jinfo [option] [server-id@]remote-hostname-or-IP

2.35.1 jmap Komutu Kullanım Seçenekleri

- : Komut argümansız çalıştırıldığında, jmap paylaşmalı obje haritalarını yazdırmaktadır. Hedefteki VM'e yüklenmiş tüm paylaşmalı obje için başlangıç adresi, mapping boyutu ve full path bilgisi yazdırılır.
- **-heap**: Heap summary bilgisini yazdırır. Hesaplanması GC algoritması kullanılmıştır.
- **-histo**: Heap'a ait histogram yazdırılır. Tüm java sınıfları için, obje sayısı, byte cinsinden hafıza boyutu ve sınıf isimleri yazdırılan bilgiler arasındadır.
- **-permstat**: Java heap'inin kalıcı jenerasyonunun sınıf yükleyici istatistiklerini yazdırır. Tüm sınıf yükleyicileri için, yükleyicinin adı, liveness durumu, adresi, parent class loader'i ve yüklediği sınıf sayısı ekrana yazdırılan bilgiler arasındadır.
- **-h, -help**: Komutun yardım mesajını ekrana yazdırmaktadır.

2.36 jps Komutu

jps komutu, hedef sistemdeki instrumented HotSpot JVM'lerin listelenmesini sağlayan bir komuttur. Komut, sadece erişim izni bulunan JVM'leri listeleyebilmektedir.

Komut, "**jps [options] [hostid]**" şeklinde kullanılabilir.

Eğer komuta argüman olarak bir hostid verilmez ise, komut instrumented JVM'leri local host üzerinde aramaktadır.

Komut, sistemde bulunan tüm instrumented JVM'ler için local VM identifier (lvmid) bilgilerini raporlamaktadır. lvmid genellikle işletim sisteminin JVM işlemi için kullandığı process identifier'dir. Komut parametresiz kullanıldığında, tüm Java uygulamalarının lvmid'leri ile uygunlamanın sınıf adı ya da jar dosyası adını yazdırmaktadır.

2.36.1 jps Komutu Kullanım Seçenekleri

- **-q:** JAR dosyasının adını, sınıf adını ve main metotuna gönderilen argümanların yazdırılmasını engeller, sadece yerel VM identifier'ları çıktı olarak üretir.
- **-m:** Main metotuna gönderilen argümanların yazdırılmasını sağlar.
- **-l:** Uygulamanın main metodu için tam paket adını veya uygulamanın JAR dosyasının tam yolunun çıktısının üretilmesini sağlar.
- **-v:** JVM'e verilen argümanların listelenmesini sağlar.
- **-V:** flags dosyası kullanılarak JVM'e verilen argümanların listelenmesini sağlar.
- **-Joption:** javac tarafından çağrılan java launcher'a seçenek gönderilmesini sağlar.

2.37 jrunscript Komutu

jrunscript komutu, interaktif ve Batch modlarını destekleyen bir komut satırı script kabuğu çalışmaktadır. Dil-bağımsız bir komuttur. Varsayılan olarak kullanılan dil javascript'tir, fakat -l argümanı ile kullanılacak dilin değiştirilmesi sağlanabilmektedir. Dil olarak Java kullanıldığında, jrunscript komutu keşif amaçlı bir programlama stilini desteklemektedir.

2.37.1 jrunscript Komutu Kullanım Seçenekleri

- **-classpath, -cp path:** Script'in kullanımı sırasında ihtiyaç duyabileceği tüm sınıf dosyalarının bulunduğu dizinin belirtilmesini sağlar.
- **-Dname=value:** Java sistem özelliği (Java system property) ayarlanması sağlar.
- **-Jflag:** jrunscript komutunun çalıştığı JVM'e bir flag gönderilmesini sağlar.
- **-I, -l language:** Argüman olarak verilen scripting dilinin kullanılmasını sağlar. Varsayılan olarak JavaScript dili kullanılmaktadır. Farklı dillerin kullanımı için, istenilen script engine'in JAR dosyası -cp argümanı ile verilmelidir.
- **-e script:** Verilen script'in kullanılmasını sağlar.
- **-encoding encoding:** Script dosyalarının okunacağı karakter kodlamasının ayarlanması sağlar.
- **-f script-file:** Verilen script dosyasının kullanılmasını sağlar (batch mode).
- **-f -** Script'in standart input'tan okunmasını sağlar (interactive mode).
- **-help, -?:** Komutun yardım sayfasını yazdırır.
- **-q:** Kullanılabilir tüm script engine'leri yazdırır ve programdan çıkar.

2.38 jstack Komutu

jstack komutu, verilen işlemin ya da core file'in ya da remote debug server'in java threadlerinin java stack trace'lerinin yazdırılmasını sağlayan bir komuttur. Tüm java frame'leri için; frame'in tamamen sınıf adı, metot adı, bci değeri (byte code index) ve satır numarası yazdırılmaktadır.

Komutun kullanımı takip eden sekillerde olmalıdır:

- jstack [option] pid
- jstack [option] executable core
- jstack [option] [server-id@]remote-hostname-or-IP

2.38.1 jstack Komutu Kullanım Seçenekleri

- **-m**: Mixed-mode'u aktif eder. Bu mod (Java ve native C/C++ frames) stack trace'ini yazdırır.
- **-h, -help**: Komutun yardım sayfasını yazdırır.

2.39 jstat Komutu

jstat komutu, bir instrumented HotSpot JVM'e ait performans istatistiklerinin görüntülenmesinde kullanılan bir komuttur. Hedef JVM, kendisinin sanal makine tanımlayıcısı ya da komuta verilen vmid seçeneği ile belirtilmektedir.

Komutun kullanımı şu şekilde yapılmalıdır: "**jstat [generalOption | outputOptions vmid [interval[s|ms] [count]]]**"

Komutta kullanılan vmid parametresinin sözdizimi de şu şekilde olmalıdır: "**[protocol:l://]lvmid[@hostname][:port][/servername]**". Komutta bahsedilen alanların açıklanması gerekirse:

- **protocol:** İletişim protokolünü ifade eder. Eğer hostname verilmediyse ve protokol verilmemişse, varsayılan protokol platform spesifik, optimize edilmiş local protokoldür. Eğer protokol verilmediyse fakat hostname verildiyse, varsayılan protokol rmi olmaktadır.
- **lvmid:** Hedef JVM için yerel sanal makine tanımlayıcısını ifade eder. lvmid, platform spesifik ve sistemdeki JVM'i eşsiz bir şekilde ifade eden bir değerdir. Sanal makine tanımlayıcısının ihtiyacı olan tek bileşendir. lvmid genellikle işletim sisteminin hedef JVM işlemi için atadığı process identifier'dir. lvmid'nin bulunması için jps ya da ps komutları kullanılabilir.
- **hostname:** Hedef host'u belirtmek için kullanılan hostname ya da IP adresini belirtir. Eğer belirtilmezse, hedef host local host olarak seçilir.
- **port:** Uzak sunucu ile iletişim kurulacak portun seçilmesini sağlar. Hostname verilmediyse ya da protokol optimize edilmiş local protokol ise port görmezden gelinir. rmi protokolünde varsayılan rmiregistry portu (1099) kullanılmaktadır.
- **servername:** Bu parametrenin gördüğü muamele implamantasyona dayalıdır. Optimize yerel protokol için bu parametre görmezden gelinir. rmi protokolünde ise remote host üzerindeki rmi uzak objesinin adını belirtir.

Komutun kullanımına dair daha detaylı bilgilere [bu link](#) üzerinden erişilebilir.

2.40 ethtool Komutu

ethtool komutu, linux sistemlerde ethernet cihazlarını ayarlamak ve bağlı ethernet cihazları ile alakalı bilgiler alabilmek için kullanılan bir komuttur. Komuta ait örnek bir çıktı şe^{kil 90} überinde gösterilmektedir. Çıktının elde edilmesinde "eth0" arayüzü kullanılmıştır.

```
[~] sudo ethtool eth0
[sudo] password for seyitahmet:
Settings for eth0:
    Supported ports: [ TP ]
    Supported link modes:  1000baseT/Full
    Supported pause frame use: No
    Supports auto-negotiation: No
    Advertised link modes:  Not reported
    Advertised pause frame use: No
    Advertised auto-negotiation: No
    Speed: 1000Mb/s
    Duplex: Full
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    Auto-negotiation: off
    MDI-X: Unknown
    Link detected: yes
[~] █
```

Şe^{kil 90:} ethtool Komutu

2.40.1 ethtool Komutu Yardım Sayfası

Bu işlem için komut "ethtool -h" şeklinde kullanılmalıdır. Bu komut ile ethtool komutunun kullanımına dair bilgiler ekrana yazdırılabilmiştir. Komuta ait örnek bir çıktı şe⁹¹kil 91 üzerinde verilmiştir.

```
[~] ethtool -h
ethtool version 3.5
Usage:
    ethtool DEVNAME Display standard information about device
    ethtool -s|--change DEVNAME      Change generic options
        [ speed %d ]
        [ duplex half|full ]
        [ port tp|aui|bnc|mii|fibre ]
        [ mdix auto|on|off ]
        [ autoneg on|off ]
        [ advertise %x ]
        [ phyad %d ]
        [ xcvr internal|external ]
        [ wol p|u|m|b|a|g|s|d... ]
        [ sopass %x:%x:%x:%x:%x:%x ]
        [ mslvl %d | mslvl type on|off ... ]
    ethtool -a|--show-pause DEVNAME Show pause options
    ethtool -A|--pause DEVNAME      Set pause options
        [ autoneg on|off ]
        [ rx on|off ]
        [ tx on|off ]
    ethtool -c|--show-coalesce DEVNAME     Show coalesce options
    ethtool -C|--coalesce DEVNAME   Set coalesce options
        [adaptive-rx on|off]
        [adaptive-tx on|off]
        [rx-usecs N]
        [rx-frames N]
        [rx-usecs-irq N]
        [rx-frames-irq N]
        [tx-usecs N]
        [tx-frames N]
        [tx-usecs-irq N]
        [tx-frames-irq N]
        [stats-block-usecs N]
        [pkt-rate-low N]
        [rx-usecs-low N]
        [rx-frames-low N]
        [tx-usecs-low N]
        [tx-frames-low N]
        [pkt-rate-high N]
        [rx-usecs-high N]
        [rx-frames-high N]
```

Şe⁹¹kil 91: ethtool -h Komutu

2.40.2 NIC Tarafından Kullanılan Sürücünün Görüntülenmesi

Bu işlem için komut "ethtool -i NIC" şeklinde kullanılmalıdır. Burada NIC, kullanılacak arayüzü ifade etmektedir. Komuta ait örnek bir çıktı şekil 92 üzerinde gösterilmektedir.

```
[~] sudo ethtool -i eth0
[sudo] password for seyitahmet:
driver: vmxnet
version: 2.1.0.0
firmware-version: N/A
bus-info: 0000:02:01.0
supports-statistics: no
supports-test: no
supports-eeprom-access: no
supports-register-dump: no
supports-priv-flags: no
[~] █
```

Sekil 92: ethtool -i NIC Komutu

2.40.3 NIC İstatistiklerinin Görüntülenmesi

Bu işlem için komut "ethtool -S NIC" şeklinde kullanılmalıdır. Burada NIC, kullanılacak arayüzü ifade etmektedir. Komuta ait örnek bir çıktı şekil 93 üzerinde gösterilmektedir.

```
[~] sudo ethtool -S eth0
no stats available
[~] █
```

Sekil 93: ethtool -S NIC Komutu

Ayrıca ethtool komutu ile arayüzün hızı, duplex-modu gibi bir çok ayarı değiştirilebilmekte, istatistiklerin görüntülenmesine üretilen çıktılar farklı parametreler ile değiştirilebilmektedir.

2.41 ip Komutu

ip komutu, linux sistemlerde birkaç ağ yönetim görevini gerçekleştirmek için kullanılan bir komuttur. Bu komut ile yönlendirmeleri (routes), cihazları (devices) ve tüneleri (tunnels) gösternmek veya manipüle etmek için kullanılmaktadır. ifconfig komutuna çok benzemekle birlikte ifconfig'den çok daha güçlü bir komuttur, çünkü bünyesinde çok daha fazla fonksiyon barındırmaktadır.

ip komutu ile varsayılan ve statik yönlendirme ayarlanabilir yahut değiştirilebilir, tunnel over IP ayarlanabilir, IP adresleri ile ilgili bilgiler görüntülenebilir, arayüzün durumu değiştirilebilir, bu arayüzlere IP adresleri ve yönlendirmeler atanabilir veya silinebilir.

2.41.1 ip Komutu Yardım Sayfası

Bu işlem için komut "**ip help**" şeklinde kullanılmalıdır. Bu komut ile ip komutunun kullanımına ilişkin bilgiler ekrana yazdırılabilmiştir. Komuta ait örnek bir çıktı şekil 94 üzerinde görülebilmektedir.

```
[~] ip help
Usage: ip [ OPTIONS ] OBJECT { COMMAND | help }
      ip [ -force ] -batch filename
where  OBJECT := { link | addr | addrlabel | route | rule | neigh | ntable |
                 tunnel | maddr | mroute | mrule | monitor | xfrm | token }
      OPTIONS := { -V[ersion] | -s[tatistics] | -d[etails] | -r[esolve] |
                  -h[uman-readable] | -iec |
                  -f[amily] { inet | inet6 | ipx | dnet | link } |
                  -o[neline] | -t[imestamp] | -b[atch] [filename] |
                  -rc[vbuf] [size]}  
[~] █
```

Şekil 94: ip help Komutu

2.41.2 Tüm Ağ Cihazlarıyla İlişkilendirilmiş Tüm IP Adreslerinin Görüntülenmesi

Bu işlem için komut "**ip address**" şeklinde kullanılmalıdır. Komuta ait örnek bir çıktı [Şekil 95](#) üzerinde verilmiştir.

```
[~] ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:0c:29:e5:37:dc brd ff:ff:ff:ff:ff:ff
    inet 192.168.213.137/24 brd 192.168.213.255 scope global eth0
        inet6 fe80::20c:29ff:fee5:37dc/64 scope link
            valid_lft forever preferred_lft forever
[~] █
```

Şekil 95: ip address Komutu

Bu komut tüm arayüzlere dair bilgileri listelemektedir. Eğer spesifik bir arayüze dair bilgiler elde edilmek istenirse, address kısmından sonra show ile o arayüzün ismi parametre olarak girilmelidir. Örneğin eth0 arayüzüne ait bilgiler görüntülenmek istenirse, komut "**ip address show eth0**" şeklinde kullanılmalıdır. Bahsedilen komuta ait örnek bir çıktı [Şekil 96](#) üzerinde gösterilmektedir.

```
[~] ip address show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:0c:29:e5:37:dc brd ff:ff:ff:ff:ff:ff
    inet 192.168.213.137/24 brd 192.168.213.255 scope global eth0
        inet6 fe80::20c:29ff:fee5:37dc/64 scope link
            valid_lft forever preferred_lft forever
[~] █
```

Şekil 96: ip address show eth0 Komutu

2.41.3 Link Layer Bilgilerinin Görüntülenmesi

Bu işlem için komut "**ip link**" şeklinde kullanılmalıdır. Komuta ait örnek bir çıktı şekil [97](#) üzerinde verilmiştir.

```
[~] ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:0c:29:e5:37:dc brd ff:ff:ff:ff:ff:ff
[~] █
```

Şekil 97: ip link Komutu

Bu komut -s parametresi ile birlikte kullanılırsa, arayüzlere ait istatistiklerin görüntülenmesi sağlanabilmektedir. Bu işlem için komut "**ip -s link**" şeklinde kullanılmalıdır. Komuta ait örnek bir çıktı şekil [98](#) üzerinde görülebilmektedir.

```
[~] ip -s link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    RX: bytes packets errors dropped overrun mcast
      1240     24      0      0      0
    TX: bytes packets errors dropped carrier collsns
      1240     24      0      0      0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:0c:29:e5:37:dc brd ff:ff:ff:ff:ff:ff
    RX: bytes packets errors dropped overrun mcast
      63752    865      0      0      0
    TX: bytes packets errors dropped carrier collsns
      21310    260      0      0      0
[~] █
```

Şekil 98: ip -s link Komutu

Aynı şekilde komut "**ip link show NIC**" şeklinde kullanılarak verilen arayüze ait bilgilerin görüntülenmesi sağlanabilir. Burada NIC, bilgilerinin görüntülenmesi istenilen arayüzü ifade etmektedir.

2.41.4 Route Bilgilerinin Görüntülenmesi

Bu işlem için komut "**ip route**" şeklinde kullanılmalıdır. Bu komut ile ağa gelen yönlendirme paketlerinin görüntülenebilmesi sağlanmaktadır. Komuta ait örnek bir çıktı şekil 99 üzerinde gösterilmiştir.

```
[~] ip route
192.168.213.0/24 dev eth0  proto kernel  scope link  src 192.168.213.137  metric 1
default via 192.168.213.2 dev eth0  proto static
[~] █
```

Şekil 99: ip route Komutu

2.41.5 ip Komutuna Ait Kullanım Seçenekleri

Bahsedilen kullanım senaryoları dışında ip komutu şu işlemler için de kullanılabilir:

- **add:** Belirtilen arayüze IP adresi eklenmesi için kullanılır. Örnek kullanım: "ip a add 192.168.1.50/24 dev enp3s0"
- **del:** Belirtilen arayüzden IP adresi silinmesi işlemi için kullanılır. Örnek kullanım: "ip a del 192.168.1.50/24 dev enp3s0"
- **up:** Arayüzün aktif edilmesinde kullanılır. Örnek kullanım: "ip link set enp3s0 up"
- **down:** Arayüzün deaktif edilmesinde kullanılır. Örnek kullanım: "ip link set enp3s0 down"

2.42 iperf Komutu

iperf komutu, ağ performansı ölçümleri ve ayarlamasında kullanılan bir komuttur. C dilinde yazılmıştır.

Komut, ya client ya da server modunda çalışmaktadır. Bu ikisi arasındaki bant genişliği, performans ve hız ölçümünü yapmaktadır. Komut, TCP ya da UDP paketi oluşturur ve bunu bir client'ten server'a, bir de server'dan client'e yollayarak bahsedilen ölçümleri gerçekleştirmektedir.

2.42.1 iperf Komutu Kullanım Seçenekleri

- **-s:** Server'i çalıştırmaya yarar.
- **-c:** Client'i çalıştırmaya yarar.
- **-u:** UDP paketi kullanılmasını sağlar.
- **-b:** Bandwidth'i gösterir.
- **-i:** Çıktı aralığının belirtilmesini sağlar (interval).
- **-t:** Zamanın saniye bazından gösterilmesini sağlar.
- **-p:** Port'un ayarlanması sağlar.
- **-w:** Pencere boyutunun ayarlanması sağlar.
- **-d:** Çift yönlü trafiği aktif eder (bi-directional traffic).
- **-l:** Uzunluğun ayarlanması sağlar.
- **-V:** IPv4 yerine IPv6 adresleri kullanıldığındá kullanılır.

2.43 iptraf Komutu

iptraf komutu; ncurses tabanlı, TCP bilgisi, UDP sayıları, ICMP vs OSPF bilgileri, Ethernet yükleme bilgisi, düğüm istatistikleri gibi ağ istatistiklerinin görüntülenmesini sağlayan bir IP LAN komutudur. Komut, "iptraf [-f] [-q] [-u] [-i iface | -g | -d iface | -s iface | -z iface | -l iface [-t timeout] [-B [-L logfile]]] | [-h] " şeklinde kullanılmaktadır.

2.43.1 iptraf Komutu Kullanım Seçenekleri

- **-i iface:** Verilen arayüz ile IP trafik monitörünün başlatılmasını sağlar. Eğer iface yerine "all" parametresi verilmişse, bu komut tüm arayzlere uygulanır.
- **-g:** Genel arayüz istatistiklerinin başlatılmasını sağlar.
- **-d iface:** Verilen arayüz üzerinde komutun detaylı olarak başlatılmasını sağlar.
- **-s iface:** Verilen arayüz üzerindeki TCP ve UDP trafiğinin izlenmesini sağlar.
- **-z iface:** Verilen arayüzdeki paket sayılarını boyutları ile ilişkili olarak gösterir.
- **-l iface:** Verilen arayüz üzerinde LAN istasyonu monitörünün başlatılmasını sağlar. Eğer parametre olarak "all" verilirse, komut tüm arayüzler üzerinde çalışır.
- **-t timeout:** Komutun yalnızca verilen dakika boyunca çalıştırılmasını sağlar.
- **-B:** Standart output'un /dev/null'a yönlendirilmesini sağlar ve standart input'u kapatır.
- **-L logfile:** Log dosyasının isminin değiştirilebilmesini sağlar.
- **-f:** Tüm kilit ve sayaçları temizleyerek çalışan iptraf instance'ının kendisinin çalışan ilk instance olduğunu düşünenmesini sağlar. Sadece komutun anormal bir çıkış yapması ya da sistemin çökmesi gibi durumlarda kullanılmalıdır.
- **-u:** Desteklenmeyen interface'lerin ethernet cihazları olarak kullanılabilmesini sağlar. Böylece bir interface'in ismi değiştirilmiş olsa bile komutun kullanılabilmesi sağlanır.
- **-h:** Komutun yardım sayfasının görüntülenmesini sağlar.

2.44 lldptool Komutu

lldptool komutu, llpad'in sorgulanması ve yapılandırılması işlemlerinde kullanılan bir komuttur. Komut, bu işlemleri gerçekleştirebilmek için llpad'in client interface'ine bağlanmaktadır. lldptool komutu, argümansız çalıştırılırsa interactive mode'da çalışmaktadır. Ayrıca llpad'den aldığı event'leri yazdırın bir event listener olarak da görev yapmaktadır.

2.44.1 lldptool Komutu Kullanım Seçenekleri

- **-i [ifname]**: Komutun kullanacağı ağ arayüzünün belirtilmesini sağlar.
- **-V [tlvid]**: TLV tanımlayıcısını belirtmek için kullanılmaktadır. TLVID, spesifik LLDP TLV'lerini belirtmek için kullanılan bir tam sayı değeridir.
- **-n**: Kullanabilen komutlar için "neighbor" seçeneğinin belirtilmesini sağlar.
- **-c <argument list>**: TLV sorguları için "config" seçeneğinin kullanılmasını sağlar.
- **-r**: Ham istemci arayüzü mesajlarının gösterilmesini sağlar.
- **-R**: Sadece ham istemci arayüzü mesajlarının gösterilmesini sağlar.
- **-h**: Komutun yardım sayfasının gösterilmesini sağlar.
- **licence**: Lisans bilgisinin gösterilmesini sağlar.
- **-v**: Komutun versiyon bilgisinin gösterilmesini sağlar.
- **-S**: LLDP istatistiklerinin gösterilmesini sağlar.
- **-t**: TLV bilgisinin gösterilmesini sağlar.
- **-T**: TLV bilgisinin ayarlanması sağlar.
- **-l**: LLDP parametrelerinin gösterilmesini sağlar.
- **-L**: LLDP parametrelerinin ayarlanması sağlar.
- **-q**: Interaktif moddan çıkış yapılmasını sağlar.

2.45 netcat Komutu

netcat komutu, port arama , güvenlik ve ağ izleme aracı olarak kullanılan bir komuttur.

Komut, aşağıda listelenen işlemler için kullanılabilir:

- Port arama
- Dosyaların sunucu üzerinden kopyalanabilmesini sağlama
- Port forwarding
- Proxy sunucusu oluşturma
- Web sunucusu hostlama

netcat komutunun iki adet kullanım modu bulunmaktadır. Bunlar client ve server modlarıdır. Komut, client modunda kullanılırken sözdizimi şu şekildedir: "**nc [-options] [hostname] [ports]**". Komutun server modunda kullanılırkenki sözdizimi ise şu şekildedir: "**nc -l -p port [options] [hostname] [port]**".

2.45.1 Port Dinleme İşlemleri

netcat komutu ile port dinleme işlemi yapılrken genellikle kullanılan üç parametre bulunmaktadır. Bunlar:

- **-z**: Komutun daemonlara hiçbir veri göndermeden onları dinlemesini sağlar.
- **-v**: Komutun verbose modunda çalışmasını sağlar.
- **-w**: Time-out kondisyonu belirtilmesi gerekiğinde kullanılmaktadır.

Komut ile tek bir portun dinlenmesi için bu parametreler ile komut: "**nc -v -w 2 z 127.0.0.1 20**" şeklinde kullanılabilir. Böylece port 20'nin dinlenmesi sağlanabilmektedir.

Komut ile birden çok portun dinlenmesi için bu parametreler ile komut: "**nc -v -w 2 z 127.0.0.1 20 25 30**" şeklinde kullanılabilir. Böylece port 20, 25 ve 30'un dinlenmesi sağlanabilmektedir.

Komut ile bir aralıktaki portların dinlenmesi için bu parametreler ile komut "**nc -v -w 2 z 127.0.0.1 20-25**" şeklinde kullanılabilir. Böylece port 20-25 aralığının dinlenmesi sağlanabilmektedir.

2.45.2 Dosya Kopyalama İşlemleri

netcat komutu ile iki makine arasında dosya gönderme ve alma işlemi yapılmaktadır. Bu, bir makineden gönderilen dosyanın diğer makine üzerinden alınmasıyla gerçekleştirilmektedir.

Örneğin "hello.txt" dosyasının iki makine arasında iletilmesinin istendiği varsayılsın. Dosyanın gönderilmesi istenen makinede komut: "**nc -l port > hello.txt**" şeklinde kullanılmalı iken dosyanın gönderileceği makinede komut "**nc ipaddr port < hello.txt**" şeklinde kullanılmalıdır.

2.45.3 Bağlantı Modunun UDP Olarak Değiştirilmesi

Bu işlem için komut -u parametresi ile kullanılmalıdır. Bu şekilde kullanıldığında komut bağlantı sırasında TCP yerine UDP paketlerini kullanacaktır.

2.46 netstat Komutu

Linux sistemlerde netstat komutu ile ağ bağlantıları, yönlendirme tabloları, arayüz istatistikleri gibi birçok bilgiye ulaşılabilir.

2.46.1 netstat Komutunun Kullanım Seçenekleri

- **-a:** Dinleme yapan-yapmayan tüm soketlerin listelenmesini sağlar.
- **-at:** Tüm TCP portlarının listelenmesini sağlar. Komuta ait örnek bir çıktı şe^{kil 100} üzerinde görülebilir.

```
[staj] netstat -at
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign Address      State
tcp        0      0 *:sunrpc                *:*                  LISTEN
tcp        0      0 *:55600                 *:*                  LISTEN
tcp        0      0 localhost.localdomain:ipp  *:*                  LISTEN
tcp        0      0 localhost.localdomain:smtp *:*                  LISTEN
tcp        0      0 *:39395                 *:*                  LISTEN
tcp        0      0 *:sunrpc                *:*                  LISTEN
tcp        0      0 localhost6.localdomain6:ipp *:*                  LISTEN
[staj] █
```

Şe^{kil 100:} netstat -at Komutu

- **-au:** Tüm UDP portlarının listelenmesini sağlar. Komuta ait örnek bir çıktı şekilde 101 üzerinde verilmiştir.

```
[staj] netstat -au
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         St
ate
udp      0      0 localhost.localdomain:703  *:*
udp      0      0 *:bootpc                  *:*
udp      0      0 *:35283                  *:*
udp      0      0 *:sunrpc                  *:*
udp      0      0 *:ipp                     *:*
udp      0      0 192.168.213.137:ntp     *:*
udp      0      0 localhost.localdomain:ntp  *:*
udp      0      0 *:ntp                     *:*
udp      0      0 *:806                     *:*
udp      0      0 *:33473                  *:*
udp      0      0 *:sunrpc                  *:*
udp      0      0 fe80::20c:29ff:fee5:37dc:ntp *:*
udp      0      0 localhost6.localdomain6:ntp *:*
udp      0      0 *:ntp                     *:*
udp      0      0 *:806                     *:*
[staj] █
```

Şekil 101: netstat -au Komutu

- **-l:** Tüm dinleme yapan soketlerin listelenmesini sağlar.

- **-lt:** Sadece dinleme yapan TCP portlarının listelenmesini sağlar. Komuta ait örnek bir çıktı şékil 101 üzerinde verilmiştir.
- **-lu:** Sadece dinleme yapan UDP portlarının listelenmesini sağlar. Komuta ait örnek bir çıktı şékil 102 üzerinde verilmiştir.

```
[staj] netstat -lt
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp     0      0 *:sunrpc                  *.*                  LISTEN
tcp     0      0 *:55600                   *.*                  LISTEN
tcp     0      0 localhost.localdomain:ipp   *.*                  LISTEN
tcp     0      0 localhost.localdomain:smtp  *.*                  LISTEN
tcp     0      0 *:39395                   *.*                  LISTEN
tcp     0      0 *:sunrpc                  *.*                  LISTEN
tcp     0      0 localhost6.localdomain6:ipp *.*                  LISTEN

[staj] netstat -lu
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp    0      0 localhost.localdomain:703   *.*                  LISTEN
udp    0      0 *:bootpc                 *.*                  LISTEN
udp    0      0 *:35283                  *.*                  LISTEN
udp    0      0 *:sunrpc                 *.*                  LISTEN
udp    0      0 *:ipp                   *.*                  LISTEN
udp    0      0 192.168.213.137:ntp    *.*                  LISTEN
udp    0      0 localhost.localdomain:ntp  *.*                  LISTEN
udp    0      0 *:ntp                   *.*                  LISTEN
udp    0      0 *:806                  *.*                  LISTEN
udp    0      0 *:33473                  *.*                  LISTEN
udp    0      0 *:sunrpc                 *.*                  LISTEN
udp    0      0 fe80::20c:29ff:fee5:37dc:ntp *.*                  LISTEN
udp    0      0 localhost6.localdomain6:ntp *.*                  LISTEN
udp    0      0 *:ntp                   *.*                  LISTEN
udp    0      0 *:806                  *.*                  LISTEN
```

Şékil 102: netstat -lt & netstat -lu Komutu

- **-lx:** Sadece dinleme yapan UNIX portlarının listelenmesini sağlar.

- **-s:** Tüm portların istatistiklerinin listelenmesini sağlar. Komuta ait örnek bir çıktı şe^kil 103 üzerinde gösterilmektedir.

```
[staj] netstat -s
Ip:
    543 total packets received
    3 with invalid addresses
    0 forwarded
    0 incoming packets discarded
    443 incoming packets delivered
    733 requests sent out
Icmp:
    17 ICMP messages received
    0 input ICMP message failed.
    ICMP input histogram:
        destination unreachable: 1
        timeout in transit: 15
        echo requests: 1
    4 ICMP messages sent
    0 ICMP messages failed
    ICMP output histogram:
        destination unreachable: 3
        echo replies: 1
IcmpMsg:
    InType3: 1
    InType8: 1
    InType11: 15
    OutType0: 1
    OutType3: 3
Tcp:
    13 active connections openings
    0 passive connection openings
    12 failed connection attempts
    0 connection resets received
    0 connections established
    29 segments received
    29 segments send out
    0 segments retransmitted
    0 bad segments received.
    12 resets sent
Udp:
    396 packets received
    3 packets to unknown port received.
    0 packet receive errors
    702 packets sent
UdpLite:
TcpExt:
    2 packets directly queued to recvmsg prequeue.
    289 packets directly received from prequeue
    0 packets header predicted
    1 packets header predicted and directly queued to user
    1 acknowledgments not containing data received
    1 predicted acknowledgments
    0 TCP data loss events
IpExt:
    InBcastPkts: 97
    InOctets: 50672
    OutOctets: 52584
    InBcastOctets: 8963
```

Şe^kil 103: netstat -s Komutu

- **-st:** Tüm TCP portlarının istatistiklerinin listelenmesini sağlar. Komuta ait örnek bir çıktı [Şekil 104](#) üzerinde görülebilir.

```
[staj] netstat -st
IcmpMsg:
    InType3: 1
    InType8: 1
    InType11: 15
    OutType0: 1
    OutType3: 3
Tcp:
    13 active connections openings
    0 passive connection openings
    12 failed connection attempts
    0 connection resets received
    0 connections established
    29 segments received
    29 segments send out
    0 segments retransmitted
    0 bad segments received.
    12 resets sent
UdpLite:
TcpExt:
    2 packets directly queued to recvmsg prequeue.
    289 packets directly received from prequeue
    0 packets header predicted
    1 packets header predicted and directly queued to user
    1 acknowledgments not containing data received
    1 predicted acknowledgments
    0 TCP data loss events
IpExt:
    InBcastPkts: 98
    InOctets: 50744
    OutOctets: 52584
    InBcastOctets: 9035
[staj] █
```

Şekil 104: netstat -st Komutu

- **-su:** Tüm UDP portlarının istatistiklerinin listelenmesini sağlar. Komuta ait örnek bir çıktı [Şekil 105](#) üzerinde verilmiştir.

```
[staj] netstat -su
IcmpMsg:
    InType3: 1
    InType8: 1
    InType11: 15
    OutType0: 1
    OutType3: 3
Udp:
    396 packets received
    3 packets to unknown port received.
    0 packet receive errors
    702 packets sent
UdpLite:
IpExt:
    InBcastPkts: 101
    InOctets: 50960
    OutOctets: 52584
    InBcastOctets: 9251
[staj] █
```

Şekil 105: netstat -su Komutu

2.47 nmap Komutu

nmap komutu, ağ keşfi ve güvenlik denetimi işlemlerinde kullanılan bir komut satırı aracıdır. Genellikle bilgisayar korsanları, siber güvenlik uzmanları, ağ ve sistem yöneticileri tarafından kullanılan bu komut, aşağıdaki amaçları gerçekleştirmek için kullanılmaktadır:

- Bir ağa dair gerçek zamanlı bilgilerin elde edilmesi,
- Bir ağda açık olan port sayısının görüntülenmesi,
- Live-host'ların listesinin görüntülenmesi,
- Port, OS ve Host tarama işlemleri,
- Ağda aktif edilmiş IP'lere ait detaylı bilgilerin elde edilmesi.

2.47.1 nmap Komutu Yardım Sayfası

Bu işlem için komut "**nmap -h**" şeklinde kullanılmalıdır. Bu komut ile nmap komutunun kullanımına dair bilgiler ekrana yazdırılabilirmektedir. Komuta ait örnek bir çıktı şekil 106 üzerinde verilmiştir.

```
[~] nmap -h
Nmap 5.51 ( http://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}

TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file

HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[protocol list]: IP Protocol Ping
  -PR: ARP ping - does not need HW address -> IP translation
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host

SCAN TECHNIQUES:
  -sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
  -sU: UDP Scan
  -sN/sF/sX: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -sI <zombie host[:probeport]>: Idle scan
  -sY/sZ: SCTP INIT/COOKIE-ECHO scans
  -sO: IP protocol scan
  -b <FTP relay host>: FTP bounce scan

PORT SPECIFICATION AND SCAN ORDER:
  -p <port ranges>: Only scan specified ports
    Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9
  -F: Fast mode - Scan fewer ports than the default scan
  -r: Scan ports consecutively - don't randomize
  --top-ports <number>: Scan <number> most common ports
  --port-ratio <ratio>: Scan ports more common than <ratio>

SERVICE/VERSION DETECTION:
  -sV: Probe open ports to determine service/version info
  -sR: Check what service uses opened ports using RPC scan
  --version-intensity <level>: Set from 0 (light) to 9 (try all probes)
  --version-light: Limit to most likely probes (intensity 2)
  --version-all: Try every single probe (intensity 9)
  --version-trace: Show detailed version scan activity (for debugging)
```

Sekil 106: nmap -h Komutu

2.47.2 Hostname ya da IP Adresi ile Tarama Yapılması

Bu işlem için komut "**nmap hostname | ip**" şeklinde kullanılmalıdır. Burada hostname, tarama yapılmak istenen hostname'i, aynı şekilde ip de tarama yapılmak istenen ip adresini belirtmektedir. Bu ikisinden herhangi birinin kullanımı yeterli olmaktadır. Komutun hostname ile kullanılmasına ait örnek bir çıktı şe^{kil 107}, ip adresi ile kullanılmasına ait örnek bir çıktı ise şe^{kil 108} üzerinde görülebilmektedir.

```
[~] nmap www.archlinux.org

Starting Nmap 5.51 ( http://nmap.org ) at 2021-08-17 07:28 +03
Nmap scan report for www.archlinux.org (95.217.163.246)
Host is up (0.083s latency).
rDNS record for 95.217.163.246: archlinux.org
Not shown: 997 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 90.49 seconds
```

Şe^{kil 107: nmap hostname Komutu}

```
[~] nmap 95.217.163.246

Starting Nmap 5.51 ( http://nmap.org ) at 2021-08-17 07:30 +03
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.04 seconds
[~]
```

Şe^{kil 108: nmap ip Komutu}

2.47.3 nmap Komutu Kullanım Seçenekleri

- **-v:** Komutun daha ayrıntılı bir çıktı üretmesini sağlar.
- **-sA:** Komutun güvenlik duvarı ayarlarını keşfetmesini sağlar.
- **-sL:** Hostname'lerin tanımlanması işleminde kullanılır.
- **-iL:** Komutun tarama işlemlerini bir dosyadan okumasını sağlar.

Bahsedilen seçenekler dışında, komutun yardım sayfasında da görülebileceği üzere komut ile birlikte kullanılabilcek çok daha fazla parametre ve bu parametreler ile yapılabilecek çok daha detaylı tarama işlemleri bulunmaktadır.

2.48 /proc/net/bonding Dizini

Linux bonding sürücüsü birden çok ağ arayüzünün tek bir mantıksal "bonded" arayüzde birleştirilmesini sağlayan bir metot sunmaktadır. Oluşturulan bu arayüzlerin davranışları moda bağlı olmakla birlikte genellikle hot stanby, yük dengeleme servisleri gibi işlemler için kullanılmaktadır.

Bahsedilen diğer başlıkların aksine bir dizin olan /proc/net/bonding, sistem üzerinde bahsedilen bu bonded arayüzlerin tutulduğu dizindir.

2.49 snmpget Komutu

snmpget komutu, bir ağ varlığı hakkında bilgi sorgulamak için SNMP GET isteğini kullanan bir SNMP uygulamasıdır. Komut, bir ya da daha fazla OID (object identifiers) kabul edebilmektedir.

2.49.1 snmpget Komutu Örnek Kullanımları

Örnek olarak incelenenek komut "**snmpget -c public zeus system.sysDescr.0**" komutu, system.Descr.0 değişkenini zeus hostu üzerinden community string public'i kullanarak alacaktır. Bu işlem esnasında bir hata meydana gelirse, istenilen paket yerine bir hata paketi dönecek ve mesaj ile kullanıcı bilgilendirilecektir.

2.50 tcpdump Komutu

tcpdump komutu, linux sistemlerde bağlantı sorunlarının giderilebilmesi için kullanılan bir paket analiz komutudur. Komut, sistem üzerinden geçen TCP/IP paketleri gibi ağ trafigini yakalamak, filtrelemek ve analiz etmek işlemleri için kullanılabilmektedir. Elde ettiği bilgileri pcap dosyalarında saklar, bu dosyalar daha sonra komutun kendisi ile ya da wireshark gibi bir program ile analiz edilebilmektedir.

Komut parametresiz kullanıldığında varsayılan arayüzü dinlemeye başlamaktadır. Komuta ait örnek bir çıktı şe^{kil 109} üzerinde gösterilmektedir.

```
[~] sudo tcpdump
[sudo] password for seyithahmet: [222/223]
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
08:13:56.836927
08:13:57.167290 ARP, Request who-has 192.168.213.2 tell 192.168.213.1, length 46
08:13:57.167849 IP 192.168.213.137.43031 > 192.168.213.2.domain: 38219+ PTR? 2.213.168.192.in-addr.arpa. (44)
08:13:57.176721 IP 192.168.213.2.domain > 192.168.213.137.43031: 38219 NXDomain 0/1/0 (94)
08:13:57.177089 IP 192.168.213.137.41663 > 192.168.213.2.domain: 49390+ PTR? 1.213.168.192.in-addr.arpa. (44)
08:13:57.185879 IP 192.168.213.2.domain > 192.168.213.137.41663: 49390 NXDomain 0/1/0 (94)
08:13:57.186233 IP 192.168.213.137.50195 > 192.168.213.2.domain: 25514+ PTR? 137.213.168.192.in-addr.arpa. (46)
)
08:13:57.198639 IP 192.168.213.2.domain > 192.168.213.137.50195: 25514 NXDomain 0/1/0 (96)
08:13:58.607357 ARP, Request who-has 192.168.213.2 tell 192.168.213.1, length 46
08:13:59.167444 ARP, Request who-has 192.168.213.2 tell 192.168.213.1, length 46
08:14:00.167040 ARP, Request who-has 192.168.213.2 tell 192.168.213.1, length 46
08:14:02.166798 ARP, Request who-has 192.168.213.2 tell 192.168.213.137, length 28
08:14:02.167069 ARP, Reply 192.168.213.2 is-at 00:50:56:f6:dc:2c (oui Unknown), length 46
08:14:06.110379 ARP, Request who-has 192.168.213.2 tell 192.168.213.1, length 46
08:14:06.666844 ARP, Request who-has 192.168.213.2 tell 192.168.213.1, length 46
08:14:07.6667057 ARP, Request who-has 192.168.213.2 tell 192.168.213.1, length 46
08:14:09.110619 ARP, Request who-has 192.168.213.2 tell 192.168.213.1, length 46
08:14:09.667089 ARP, Request who-has 192.168.213.2 tell 192.168.213.1, length 46
08:14:10.667253 ARP, Request who-has 192.168.213.2 tell 192.168.213.1, length 46
08:14:13.265762 IP 192.168.213.137.32902 > 192.168.213.2.domain: 48297+ A? opensource.apple.com. (38)
08:14:13.284778 IP 192.168.213.137.32902 > 192.168.213.2.domain: 19297+ AAAA? opensource.apple.com. (38)
08:14:13.286531 IP 192.168.213.2.domain > 192.168.213.137.32902: 48297 3/0/0 CNAME world-gen.g.aaplimg.com., A 17.253.73.203, A 17.253.73.202 (107)
08:14:13.286943 IP 192.168.213.2.domain > 192.168.213.137.32902: 19297 1/0/0 CNAME world-gen.g.aaplimg.com. (75)
08:14:13.299319 IP 192.168.213.137.54320 > deber5-vip-bx-003.aaplimg.com.https: Flags [S], seq 3832402086, win 14600, options [mss 1460,sackOK,TS val 8733832 ecr 0,nop,wscale 7], length 0
08:14:13.299632 IP 192.168.213.137.51901 > 192.168.213.2.domain: 58787+ PTR? 203.73.253.17.in-addr.arpa. (44)
08:14:13.308080 IP 192.168.213.2.domain > 192.168.213.137.51901: 58787 1/0/0 PTR deber5-vip-bx-003.aaplimg.com. (87)
08:14:13.389907 IP deber5-vip-bx-003.aaplimg.com.https > 192.168.213.137.54320: Flags [S.], seq 1823872038, ack 3832402087, win 64240, options [mss 1460], length 0
08:14:13.390008 IP 192.168.213.137.54320 > deber5-vip-bx-003.aaplimg.com.https: Flags [.], ack 1, win 14600, length 0
08:14:13.502068 IP 192.168.213.137.54320 > deber5-vip-bx-003.aaplimg.com.https: Flags [P.], seq 1:207, ack 1, win 14600, length 206
```

Şekil 109: tcpdump Komutu

2.50.1 tcpdump Komutu Yardım Sayfası

Bu işlem için komut "**tcpdump -h**" şeklinde kullanılmalıdır. Bu komut ile tcpdump komutunun kullanımına dair bilgiler ekrana yazdırılmaktadır. Komuta ait örnek bir çıktı şekil 110 üzerinde verilmiştir.

```
[~] tcpdump -h
tcpdump version 4.1-PRE-CVS_2017_03_21
libpcap version 1.4.0
Usage: tcpdump [-aAdDefhIJKlLnNOpqRStuUvxX] [ -B size ] [ -c count ]
          [ -C file_size ] [ -E algo:secret ] [ -F file ] [ -G seconds ]
          [ -i interface ] [ -j timestamptype ] [ -M secret ]
          [ -Q|-P in|out|inout ]
          [ -r file ] [ -s snaplen ] [ -T type ] [ -w file ]
          [ -W filecount ] [ -y datalinktype ] [ -z command ]
          [ -Z user ] [ expression ]
[~] █
```

Sekil 110: tcpdump -h Komutu

2.50.2 tcpdump Komutu Kullanım Seçenekleri

- **-i iface:** Spesifik bir ağ arayüzüne gelen paketlerin yakalanmasını sağlar.
- **-c num:** Spesifik sayıda paketin yakalanmasını sağlar.
- **-A:** Yakalanan paketlerin ASCII formatında yazdırılmasını sağlar.
- **-D:** Kullanılabilir tüm arayüzlerin listelenmesini sağlar.
- **-XX:** Yakalanan paketlerin HEX ve ASCII şeklinde gösterilmesini sağlar.
- **-w filename:** Komutun çıktısını .pcap uzantılı bir dosyaya vermesini sağlar.
- **-r filename:** Komutun paketleri argüman olarak verilen dosyadan okumasını sağlar.
- **-n:** Paketlerin ip adresleri ile birlikte yakalanmasını sağlar.

2.51 telnet Komutu

telnet komutu, iki makine arasında TELNET protokülü ile interaktif bir iletişim kurmaya yarayan bir komuttur. Komutun kullanım sözdizimi şu şekildedir: "**telnet [-468ELadr] [-S tos] [-b address] [-e escapechar] [-l user] [-n tracefile] [host [port]]**"

2.51.1 telnet Komutu Kullanım Seçenekleri

- **-4:** IPv4 adres çözümlemesinin zorlanması sağlar.
- **-6:** IPv6 adres çözümlemesinin zorlanması sağlar.
- **-8:** 8-bit operasyonun istenmesini sağlar. Varsayılan olarak telnet 8-bit temiz (8-bit clean) değildir (Unicode gibi 8-bit karakter kodlamalarını tanımaz).
- **-E:** Kaçış karakteri fonksiyonalitesini devre dışı bırakır.
- **-L:** Veri çıkışında 8-bitlik bir veri yolunun belirtilmesini sağlar.
- **-a:** Otomatik giriş yapma denemesi yapılmasını sağlar.
- **-d:** Debug toggle'ının varsayılan değerinin değiştirilebilmesini sağlar.
- **-e escapechar:** Kaçış karakterinin verilen escapechar olarak değiştirilmesini sağlar.
- **host:** Ağ üzerinden iletişim kurulacak host'un belirtilmesini sağlar.
- **port:** Kullanılacak port'un değiştirilebilmesini sağlar. Varsayılan değeri telnet portudur (23).

Komutun kullanımı ve TELNET protokülüne dair daha ayrıntılı bilgiler için [bu link](#) ziyaret edilebilir.

2.52 free Komutu

free komutu, sistem üzerindeki hafıza ve swap'a ait toplam, kullanılan ve boşta olan miktarlarının gösterilmesini sağlayan bir komuttur. Komuta ait örnek bir çıktı şe¹¹¹ kile 111 üzerinde gösterilmektedir.

[-] free						
	total	used	free	shared	buffers	cached
Mem:	4056324	843908	3212416	4504	51412	425728
-/+ buffers/cache:		366768	3689556			
Swap:	3145724	0	3145724			

Şekil 111: free Komutu

Şe¹¹¹kile 111 üzerindeki başlıkların açıklanması gereklidir:

- **total:** Sistemin sahip olduğu toplam hafıza miktarını gösterir (Bu bilgiyi /proc/meminfo dosyası içerisinde bulunan MemTotal ve SwapTotal alanlarından almaktadır).
- **used:** Kullanımda olan hafızayı gösterir.
- **free:** Kullanımda olmayan hafızayı gösterir.
- **shared:** tmpfs tarafından kullanılan hafıza miktarını gösterir.
- **buffers:** Çekirdek tamponları (kernel buffers) tarafından kullanılan hafıza miktarını gösterir.
- **cached:** Page cache ve slabs tarafından kullanılan hafıza miktarını gösterir.
- **buffers/cache:** buffers ve cached alanlarının toplam değerlerini gösterir.

2.52.1 free Komutuna ait Kullanım Seçenekleri

- **-b, --bytes:** Hafızanın byte formatında gösterilmesini sağlar.
- **-k, --kilo:** Hafızanın kilobyte formatında gösterilmesini sağlar (Varsayılan ayar budur).
- **-m, --mega:** Hafızanın megabyte formatında gösterilmesini sağlar.
- **-g, --giga:** Hafızanın gigabyte formatında gösterilmesini sağlar.
- **--tera:** Hafızanın terabyte formatında gösterilmesini sağlar.
- **-h, --human:** Çıktının en yakın üç haneli üniteye yuvarlanılarak gösterilmesini sağlar (İnsan Okunabilir Form - Human Readable Form).
- **-c, --count:** Çıktının verilen sayı kadar gösterilmesini sağlar. -s parametresi ile birlikte kullanılmaktadır.
- **-l, --lohi:** Düşük ve yüksek hafıza istatistiklerinin gösterilmesini sağlar.
- **-s, --seconds:** Komutun verilen saniye kadar aralıklarla çıktı üretmesini sağlar.
- **-t, --total:** Çıktiya ekstra bir satır eklemektedir. Bu satır, önceki satırların toplam değerini gösterir.
- **--help:** Komutun yardım sayfasının gösterilmesini sağlar.
- **-V, --version:** Komutun versiyon bilgisinin gösterilmesini sağlar.

2.53 /proc/meminfo Dosyası

/proc/meminfo dosyası, hem fiziksel hem swap olmak üzere boşta ve kullanılan hafıza miktarı ile birlikte paylaşımı hafıza ile çekirdek tarafından kullanılan buffer miktarları bilgisini içeren bir dosyadır. Dosyaya ait örnek bir çıktı şekil 112 üzerinde verilmiştir.

```
[~] cat /proc/meminfo
MemTotal:       4056324 kB
MemFree:        3211736 kB
Buffers:         51876 kB
Cached:          425744 kB
SwapCached:      0 kB
Active:          352204 kB
Inactive:        297720 kB
Active(anon):   172500 kB
Inactive(anon):  4304 kB
Active(file):   179704 kB
Inactive(file): 293416 kB
Unevictable:     0 kB
Mlocked:         0 kB
SwapTotal:      3145724 kB
SwapFree:        3145724 kB
Dirty:            4 kB
Writeback:        0 kB
AnonPages:       172400 kB
Mapped:           91424 kB
Shmem:            4504 kB
Slab:             103000 kB
SReclaimable:    37016 kB
SUnreclaim:      65984 kB
KernelStack:     4528 kB
PageTables:      20848 kB
NFS_Unstable:    0 kB
Bounce:           0 kB
WritebackTmp:     0 kB
CommitLimit:     5173884 kB
Committed_AS:    589072 kB
VmallocTotal:    34359738367 kB
VmallocUsed:     158480 kB
VmallocChunk:    34359552252 kB
HardwareCorrupted: 0 kB
AnonHugePages:   69632 kB
HugePages_Total:  0
HugePages_Free:   0
HugePages_Rsvd:   0
HugePages_Surp:   0
Hugepagesize:    2048 kB
DirectMap4k:      8192 kB
DirectMap2M:     2088960 kB
DirectMap1G:     2097152 kB
[~] █
```

Sekil 112: meminfo Komutu

Şekil 112 üzerinde görülebildiği gibi /proc/meminfo dosyası; toplam kullanımda olan hafıza, boşta olan hafıza, paylaşımı hafıza ve bunun gibi çok veriyi içeren bir dosyadır.