

19.Doküman

Ağ Komutları

Konu Etiketleri

ağ komutları , network , ifconfig , ping , route , traceroute , whois , host , dig , arp , tcpdump , DNS , hosts

Ağ(Network) Komutları

Ağ ve komutları başlı başına bir kitap konusu o yüzden burada sadece sıkça veya gerekli durumda işimize yarayacak belli başlı komutları ele alacağız. Eğer detaylı bilgi edinmek istiyorsanız internet aracılığı ile network(ağ) hakkında bir çok güncel kaynağa ulaşabilirsiniz. Lafı daha fazla uzatmadan anlatıma **ifconfig** komutu ile başlayalım.

ifconfig

Sistemde bulunan ağ bağlantı ve IP yapılandırması ayarlarını inceleyip değiştirmemize olanak tanır.

Ağ bağlantı kartlarını listelemek için **ifconfig** komutu kullanılır.(Bu çıktıda ip, mac ve broadcast adresi gibi bilgiler yer alıyor.)

```
root@taylan:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe95:8c5e prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:95:8c:5e txqueuelen 1000 (Ethernet)
    RX packets 11647 bytes 17053858 (16.2 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2899 bytes 175710 (171.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 20 bytes 1116 (1.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 20 bytes 1116 (1.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 5e:ca:33:83:cd:e4 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Çıktıda yer alan **eth0 ethernet** kartımızı ifade ediyor. Başka **ethernet** kartları olması durumunda diğer kartlar da **eth1 ..eth2** şeklinde belirtilir.

lo bilgisayarın kendisini yani **localhost**'u ifade ediyor.

wlan0 ise **kablosuz ağ kartını** ifade eder. Yine başka kartlar olması durumunda **wlan1 ..wlan2** şekilde belirtilir.

Ayrıca bütün kart bilgilerini listelemek yerine teker teker de listelemek mümkün. Bunun için komutumuzu

ifconfig kart_adı şeklinde girmemiz yeterli.

Örneğin yalnızca **kablosuz kart bilgilerini listelemek** istersem **ifconfig wlan0** komutunu vermem yeterli olacaktır.

```
root@taylan:~# ifconfig wlan0
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.9 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::6763:cbea:ca38:fcbf prefixlen 64 scopeid 0x20<link>
    ether d4:6e:0e:02:0e:0d txqueuelen 1000 (Ethernet)
    RX packets 281 bytes 49087 (47.9 KiB)
    RX errors 0 dropped 8 overruns 0 frame 0
    TX packets 23 bytes 3324 (3.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Yukarıdaki çıktıda yer alan ayarlarda değişiklik yapmamız mümkün. Örneğin kablosuz ağ bağlantısının **yerel ip adresini** değiştirmek istersek komutumuzu `ifconfig wlan0 yeni_ip_adresi` şeklinde kullanabiliriz.

Ben kablosuz bağlantımın **192.168.1.9** olan adresi **192.168.1.10** olarak değiştirmek istiyorum bunun için `ifconfig wlan0 192.168.1.10` komutunu vermem yeterli.

Gelin bu durumu çıktıları karşılaştırarak test edelim.

```
root@taylan:~# ifconfig wlan0
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.9 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::6763:cbea:ca38:fcbf prefixlen 64 scopeid 0x20<link>
    ether d4:6e:0e:02:0e:0d txqueuelen 1000 (Ethernet)
    RX packets 384 bytes 66137 (64.5 KiB)
    RX errors 0 dropped 8 overruns 0 frame 0
    TX packets 24 bytes 3442 (3.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@taylan:~# ifconfig wlan0 192.168.1.10
root@taylan:~# ifconfig wlan0
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.10 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::6763:cbea:ca38:fcbf prefixlen 64 scopeid 0x20<link>
    ether d4:6e:0e:02:0e:0d txqueuelen 1000 (Ethernet)
    RX packets 406 bytes 70032 (68.3 KiB)
    RX errors 0 dropped 8 overruns 0 frame 0
    TX packets 24 bytes 3442 (3.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Ve son duruma bakarak ip adresimizin istediğimiz şekilde değiştiğini görmüş olduk. Aynı şekilde **ağ maskesini(netmask)** ve **broadcast** adreslerini dilediğimiz gibi düzenleyebiliriz. Hatta bu işlemi hepsi birlikte olacak şekilde bile yapabiliriz. Bunun için komutumuzu `ifconfig wlan0 yeni_ip_adresi netmask yeni_ağ_maskesi broadcast yeni_broadcast_adresi` şeklinde kullanmamız yeterli olacaktır.

Bu durumu bir örnek ile açıklayalım. Değişiklik yapmak üzere konsola `ifconfig wlan0 192.168.1.15 netmask 255.255.255.255 broadcast 192.168.2.255` komutunu girerek bir önceki durumu ile karşılaştıralım.

```
root@taylan:~# ifconfig wlan0
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.10 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::6763:cbea:ca38:fcbf prefixlen 64 scopeid 0x20<link>
    ether d4:6e:0e:02:0e:0d txqueuelen 1000 (Ethernet)
    RX packets 580 bytes 100062 (97.7 KiB)
    RX errors 0 dropped 8 overruns 0 frame 0
    TX packets 25 bytes 3560 (3.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@taylan:~# ifconfig wlan0 192.168.1.15 netmask 255.255.255.255 broadcast 192.168.2.255
root@taylan:~# ifconfig wlan0
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.15 netmask 255.255.255.255 broadcast 192.168.2.255
    inet6 fe80::6763:cbea:ca38:fcbf prefixlen 64 scopeid 0x20<link>
    ether d4:6e:0e:02:0e:0d txqueuelen 1000 (Ethernet)
    RX packets 588 bytes 101526 (99.1 KiB)
    RX errors 0 dropped 8 overruns 0 frame 0
    TX packets 25 bytes 3560 (3.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Çıktıları incelediğimizde istediğimiz doğrultuda değişikliklerin yapılmış olduğunu gördük.

ifconfig komutu kullanımı ile ilgili son olarak mevcut kartları açma ve kapatma işlemi görelim.

Örnek olarak kablosuz kartı ele alalım:

Kablosuz kartı kapatmak istersek; **ifconfig wlan0 down** komutunu kullanırız.

```
root@taylan:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.67.177  netmask 255.255.255.0  broadcast 192.168.67.255
    inet6 fe80::1fe9:db5f:79b4:fb30  prefixlen 64  scopeid 0x20<link>
    ether 00:0c:29:9e:e4:77  txqueuelen 1000  (Ethernet)
    RX packets 392463  bytes 572155231 (545.6 MiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 181910  bytes 11157336 (10.6 MiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0x10<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 104  bytes 5876 (5.7 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 104  bytes 5876 (5.7 KiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlan0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
    ether 9e:4b:32:0d:f7:1b  txqueuelen 1000  (Ethernet)
    RX packets 0  bytes 0 (0.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 0  bytes 0 (0.0 B)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@taylan:~# ifconfig wlan0 down
root@taylan:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.67.177  netmask 255.255.255.0  broadcast 192.168.67.255
    inet6 fe80::1fe9:db5f:79b4:fb30  prefixlen 64  scopeid 0x20<link>
    ether 00:0c:29:9e:e4:77  txqueuelen 1000  (Ethernet)
    RX packets 392482  bytes 572156401 (545.6 MiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 181913  bytes 11157546 (10.6 MiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0x10<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 104  bytes 5876 (5.7 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 104  bytes 5876 (5.7 KiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@taylan:~#
```

Kapalı kablosuz kartı açmak istersek ise; `ifconfig wlan0 up` komutunu kullanırız.


```
root@taylan:~# ifconfig wlan0 up
root@taylan:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.67.177  netmask 255.255.255.0  broadcast 192.168.67.255
    inet6 fe80::1fe9:db5f:79b4:fb30  prefixlen 64  scopeid 0x20<link>
    ether 00:0c:29:9e:e4:77  txqueuelen 1000  (Ethernet)
    RX packets 392499  bytes 572157451 (545.6 MiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 181914  bytes 11157636 (10.6 MiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0x10<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 104  bytes 5876 (5.7 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 104  bytes 5876 (5.7 KiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlan0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
    ether de:d7:d1:6a:22:76  txqueuelen 1000  (Ethernet)
    RX packets 0  bytes 0 (0.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 0  bytes 0 (0.0 B)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@taylan:~#
```

Ayrıca diğer kartlar için de açama-kapama işleminde aynı şekilde komutumuzu `ifconfig kart_adı down` ve `ifconfig kart_adı up` şeklinde kullanabiliriz.

ping

Hedef ile bizim sistemimiz arasında iletişimin sağlanıp sağlanmadığını kontrol ederek hedef sunucunun çalışıp çalışmadığını veya aktarım hızının ne kadar olduğunu öğrenmemizi sağlar. Bir tür kontrol mekanizması da diyebiliriz. Komutun kullanımı `ping hedef_adresi` şeklindedir.

Örneğin biz `www.google.com` adresi ile aramızdaki iletişimin hızını sorgulayalım. Bunun için komut satırına `ping www.google.com` yazıyorum. Ancak burada önemli bir nokta var, o da bu işlemin biz `Ctrl + C` tuş kombinasyonu ile durdurana kadar devam edecek olmasıdır.

```
root@taylan:~#
```

Eğer direk olarak sınırlama getirmek isterseniz komutu `ping -c 4 www.google.com` şeklinde kullanabilirsiniz. Bu sayede `www.google.com` adresine yalnızca **4 sorgu paketi** gönderilir. Elbette buradaki sayı sizin belirlemeniz ile değişebilir.

```
root@taylan:~# ping -c 4 www.google.com
PING www.google.com (172.217.22.100) 56(84) bytes of data.
64 bytes from www.google.com (172.217.22.100): icmp_seq=1 ttl=54 time=60.1 ms
64 bytes from www.google.com (172.217.22.100): icmp_seq=2 ttl=54 time=58.7 ms
64 bytes from www.google.com (172.217.22.100): icmp_seq=3 ttl=54 time=59.1 ms
64 bytes from www.google.com (172.217.22.100): icmp_seq=4 ttl=54 time=58.10 ms

--- www.google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 36ms
rtt min/avg/max/mdev = 58.650/59.212/60.072/0.579 ms
root@taylan:~#
```

route

Sistemimizdeki IP yönlendirme tablosunun içeriğini görmemizi sağlar. IP yönlendirme tablosunun içeriğini görmek için konsola `route -n` komutunu yazalım.

```
root@taylan:~# route -n
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0          192.168.67.2   0.0.0.0         UG    100    0      0 eth0
192.168.67.0     0.0.0.0        255.255.255.0   U     0      0      0 eth0
192.168.67.0     0.0.0.0        255.255.255.0   U     100    0      0 eth0
```

Burada yer alan gateway(ağ geçidi adresi) yerel ağdan internet ağına geçerken kullanılan yönlendirme adresidir.

traceroute

Bir önceki kısımda `route` komutu ile gördüğümüz yerel ağda geçerli olan yönlendirme takibini, belirli bir hedef adrese yapabilmemize olanak sağlayan komut `traceroute` komutudur. Yani komutumuz belirli bir hedefe gönderilen paketin hangi host'lardan geçtiğini bizlere gösterir. Bir nevi izlediği yolu yani adımlarını takip etmemizi

sağlar. Komutun kullanımı **traceroute hedef_adresi** şeklindedir. Bu sefer hedef adres olarak **www.offensive-security.com** adresini örnek verelim. Bunun için konsola **traceroute www.offensive-security.com** yazıyorum.

```
root@taylan:~# traceroute www.offensive-security.com
traceroute to www.offensive-security.com (192.124.249.5), 30 hops max, 60 byte packets
 1  _gateway (192.168.67.2)  0.121 ms  0.123 ms  0.055 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  * * *
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  * * *
22  * * *
23  * * *
24  * * *
25  * * *
26  * * *
27  * * *
28  * * *
29  * * *
30  * * *
root@taylan:~#
```

Çıktı ile birlikte adresin yönlendirme rotasını görmüş olduk.

whois

Whois kavramını bilmeyenler için **whois**, genel olarak domain bilgilerini içeren bir mekanizmadır. Yani **whois**; domain ne zaman kurulmuş, ne zamana kadar geçerli, kimin üzerine kayıtlı ve bunun gibi diğer tüm bilgileri tutar. Bizler de bu bilgileri konsol ekranından **whois hedef_adresi** şeklinde kullandığımız komut bütünü ile sorgulayabiliriz.

Whois sorgusuna örnek olarak yine **www.offensive-security.com** adresini hedef alalım.

```
root@taylan:~# whois offensive-security.com
Domain Name: OFFENSIVE-SECURITY.COM
Registry Domain ID: 606288052_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.gandi.net
Registrar URL: http://www.gandi.net
Updated Date: 2019-08-15T00:27:04Z
Creation Date: 2006-09-24T20:27:59Z
Registry Expiry Date: 2020-09-24T20:27:59Z
Registrar: Gandi SAS
Registrar IANA ID: 81
Registrar Abuse Contact Email: abuse@support.gandi.net
Registrar Abuse Contact Phone: +33.170377661
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Name Server: NS-182-B.GANDI.NET
Name Server: NS-185-C.GANDI.NET
Name Server: NS-34-A.GANDI.NET
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2020-01-15T10:20:47Z <<<
```

host

Hedef adres hakkında bilgi almanızı sağlar. **host** komutu ile IP adresinden alan adı(domain name) ve alan adından(domain name) IP adresine ulaşabiliriz. Bu komutun alabildiği farklı parametreleri bulunmaktadır. Ancak ben bu kısımda bunlara değinmeden yalnızca temel kullanımına örnek veriyorum. Kullanımı **host adres_adı** şeklindedir.

```
root@taylan:~# host www.yandex.com
www.yandex.com has address 213.180.204.62
www.yandex.com has IPv6 address 2a02:6b8::11:11
root@taylan:~# host 213.180.204.62
62.204.180.213.in-addr.arpa domain name pointer yandex.com.
root@taylan:~#
```

Bu komut hakkında ufak bir araştırma ile çok fazla Türkçe de dahil olmak üzere kaynağa ulaşabilirsiniz.

dig

dig(domain information groper/domain bilgi çukuru) DNS kayıtlarına bakmak için kullanımı oldukça kolay olduğundan yaygın olarak kullanılmaktadır.

Bu komutumuz da parametreler alabilmektedir, ancak ben burada sizlere yine temel işlevinden bahsederek geriye kalan parametrelerini araştırmayı sizlere bırakıyorum. DNS sorgulaması yapmak istediğimiz adresi konsoldan **dig hedef_adresi** şeklinde belirtiyoruz.

Örnek olması açısından ben tekrar **www.offensive-security.com** adresini hedef alıyorum.


```

root@taylan:~# dig www.offensive-security.com

; <<>> DiG 9.11.4-P2-3-Debian <<>> www.offensive-security.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 36295
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 3

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.offensive-security.com.      IN      A

;; ANSWER SECTION:
www.offensive-security.com. 300 IN      A      192.124.249.5

;; AUTHORITY SECTION:
offensive-security.com. 12930 IN      NS      ns-34-a.gandi.net.
offensive-security.com. 12930 IN      NS      ns-185-c.gandi.net.
offensive-security.com. 12930 IN      NS      ns-182-b.gandi.net.

;; ADDITIONAL SECTION:
ns-182-b.gandi.net.      94557 IN      A      213.167.230.183
ns-182-b.gandi.net.      94557 IN      AAAA    2001:4b98:aaab::b7

;; Query time: 84 msec
;; SERVER: 192.168.1.1#53(192.168.1.1)
;; WHEN: Çrş 0ca 15 05:24:33 EST 2020
;; MSG SIZE rcvd: 192

root@taylan:~# █

```

arp

IP-MAC Adresi eşleştirmelerinin tutulduğu tablolardır. Kullanımı **arp** şeklindedir.

```

root@taylan:~# arp
Address                  HWtype  HWaddress           Flags Mask            Iface
_gateway                ether    52:54:00:12:35:02    C                     eth0
root@taylan:~#

```

tcpdump

Sistemimizin yaptığı bağlantıları ve sistemimize yapılan bağlantıları anlık olarak görüntülememize olanak sağlar. Kullanımı **tcpdump** şeklindedir.

```

root@taylan:~# tcpdump
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
05:26:22.198789 IP taylan.57042 > Tenda.Home.domain: 9992+ A? taylan.Home. (29)
05:26:22.212646 IP taylan.47359 > Tenda.Home.domain: 19069+ PTR? 1.1.168.192.in-addr.arpa. (42)
05:26:22.213904 IP Tenda.Home.domain > taylan.47359: 19069- 1/0/0 PTR Tenda.Home. (66)
05:26:22.214035 IP taylan.49504 > Tenda.Home.domain: 56199+ PTR? 15.2.0.10.in-addr.arpa. (40)
05:26:27.203227 IP taylan.57042 > Tenda.Home.domain: 9992+ A? taylan.Home. (29)
05:26:27.218169 IP taylan.49504 > Tenda.Home.domain: 56199+ PTR? 15.2.0.10.in-addr.arpa. (40)
█

```

Ayrıca adres çözümlemesi yapmadan direk olarak bağlantıları takip etmek istersek **tcpdump -n** komutunu kullanabiliriz.

```
root@taylan:~# tcpdump -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
05:27:22.240186 IP 10.0.2.15.47102 > 192.168.1.1.53: 51142+ A? taylan.Home. (29)
05:27:27.246432 IP 10.0.2.15.47102 > 192.168.1.1.53: 51142+ A? taylan.Home. (29)
05:27:32.248972 IP 10.0.2.15.51175 > 192.168.1.1.53: 18888+ A? taylan. (24)
05:27:32.254034 IP 192.168.1.1.53 > 10.0.2.15.51175: 18888- 1/0/0 A 192.168.1.47 (40)
05:27:32.257834 IP 10.0.2.15.40971 > 192.168.1.1.53: 14699+ A? taylan.Home. (29)
05:27:37.264461 IP 10.0.2.15.40971 > 192.168.1.1.53: 14699+ A? taylan.Home. (29)
05:27:42.269939 IP 10.0.2.15.35362 > 192.168.1.1.53: 6507+ A? taylan. (24)
05:27:42.274814 IP 192.168.1.1.53 > 10.0.2.15.35362: 6507- 1/0/0 A 192.168.1.47 (40)
05:27:42.905370 IP 10.0.2.15.52338 > 192.168.1.1.53: 46295+ A? detectportal.firefox.com. (42)
05:27:42.905543 IP 10.0.2.15.52338 > 192.168.1.1.53: 41183+ AAAA? detectportal.firefox.com. (42)
```

DNS Ayarları

Komut satırından DNS ayarlarımızı değiştirmek istersek DNS bilgilerinin tutulduğu `/etc/resolv.conf` dosyasında değişiklik yapmamız gerekiyor. İşlemeleri adım adım açıklayarak ilerleyelim.

İlk olarak DNS ayarlarının bulunduğu dosya içeriğine göz atıyorum. Çünkü daha sonra değişiklik yaptığımızda ilk hali ile kıyaslamamız gerekecek. Bu işlemi `cat` komutu yardımı ile gerçekleştireceğiz.

```
root@taylan:~# cat /etc/resolv.conf
# Generated by NetworkManager
search localdomain
nameserver 192.168.67.2
nameserver 192.168.1.2
```

Şimdi eski DNS(nameserver) yerine bizim eklemek istediğimiz adresleri `echo` komutu yardımı ile girelim.

```
root@taylan:~# echo "nameserver 8.8.8.8" > /etc/resolv.conf
root@taylan:~# echo "nameserver 8.8.4.4" >> /etc/resolv.conf
root@taylan:~# cat /etc/resolv.conf
nameserver 8.8.8.8
nameserver 8.8.4.4
root@taylan:~#
```

Komutları kısaca açıklayacak olursak ilk olarak `echo "nameserver 8.8.8.8" > resolv.conf` komut bütününde `>` karakteri ile birlikte dosya içerisinde var olan ifadeleri sildik ve dosya içerisine `nameserver 8.8.8.8` ifadesini ekledik.

Daha sonra `echo "nameserver 8.8.4.4" >> resolv.conf` komut bütünü ile de daha önce eklediğimiz ifadeye ek olarak diğer bir DNS adresi olan `8.8.4.4` adresini `>>` karakteri yardımı ile ekledik.

Son olarak da eklediğimiz yeni DNS adreslerinin belgeye eklenme durumunu `cat /etc/resolv.conf` komutu ile teyit ettik.

```
root@taylan:~# cat /etc/resolv.conf
# Generated by NetworkManager
search localdomain
nameserver 192.168.67.2
nameserver 192.168.1.2
root@taylan:~# echo "nameserver 8.8.8.8" > /etc/resolv.conf
root@taylan:~# echo "nameserver 8.8.4.4" >> /etc/resolv.conf
root@taylan:~# cat /etc/resolv.conf
nameserver 8.8.8.8
nameserver 8.8.4.4
root@taylan:~#
```

hosts Dosyası

Yerel bir alan adı sunucusu işlevindedir. Sistemde alan adı çözümlemesi yapılırken bu dosyaya bakılır. Dosyanın konumu `/etc/hosts` şeklindedir. Hemen dosya içeriğine `cat` komutu yardımı ile bir göz atalım.

```
root@taylan:~# cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      kali

# The following lines are desirable for IPv6 capable hosts
::1           localhost ip6-localhost ip6-loopback
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters
root@taylan:~#
```

Böylelikle en sık kullanılan ağ komutları hakkında genel bilgi sahibi olmuş olduk. Artık daha fazla bilgi için network konusunda detaylı araştırma yapmak sizlere kalıyor.

Alıştırmalar Hakkında

Yalnızca okumak yetmez, öğrendiğiniz bilgilerin kalıcı olabilmesi için bolca alıştırmaya yapmalısınız. Doküman içerisindeki bilgileri pekiştirmek için aşağıdaki alıştırmalar ile başlayabilirsiniz. Elbette burada yer alan alıştırmalar faaliyetleri dışında, konuyu öğrendiğinizi hissedene kadar kendiniz de bolca pratik yapmayı da ihmal etmeyin lütfen. Aksi halde öğrendiğiniz bilgiler kısa sürede unutulup gidecektir.

Sistemde bulunan **ağ bağlantı ve IP yapılandırması ayarlarını** konsoldan görüntüleyin.

Konsoldan yalnızca **localhost(lo)** bağlantı bilgilerini görüntüleyin.

Kablolu veya kablosuz kartınızın **ip bağlantı(inet)** adresini konsoldan **değiştirin** ve değişikliği teyit edin.

Kablolu veya kablosuz kartınızın **hem ip adresini hem ağ maskesini(netmask) adresini hem de broadcast adresini tek seferde konsoldan değiştirin** ve değişikliği teyit edin.

Kablolu veya kablosuz kartınızı konsol üzerinden **kapatın ve kapanıp kapanmadığını teyit edin.**

Kapattığınız kartı konsol üzerinden açın ve açılıp açılmadığını da teyit edin.

udemy.com adresi ile aranızdaki **iletişimin sağlanıp sağlanmadığını kontrol ederek hedef sunucunun çalışıp çalışmadığını veya aktarım hızının ne kadar olduğunu ilgili komut ile teyit edin.** Bu işlemi de **10 paket gönderilecek şekilde** sınırlayın.

Konsol üzerinden ***udemy.com*** adresinin **whois** bilgisini sorgulayın.

yandex.com adresinin **alan adından ip adresine ulaşın** ve daha sonra teyit etmek için de **ulaştığınız ip adresinden alan adını sorgulayın.**

Sisteme bağlı cihazların **IP-MAC eşleştirme tablosunu** konsol üzerinden görüntüleyin.

Sisteminizin yaptığı ve sisteminize yapılan bağlantıları konsoldan görüntüleyin.

Geri Bildirimde Bulunun

Sizlere daha verimli bir kaynak sunabilmemiz için, uygulamada veya dokümantasyonlarda yer alan tüm hata ve eksiklerimizi bize bildirebilirsiniz.

Geri Bildirimde Bulunun