

Assignment 3

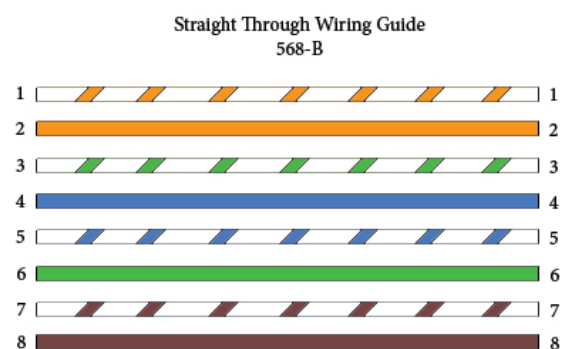
A) CAT-5/CAT-6 cable preparation with RJ-45 connector; both straight and cross cabling.

Requirements:

- Ethernet cable
- RJ45 Crimpable Connectors for CAT-5e or RJ45 Crimpable Connectors for CAT-6
- Wire stripper
- RJ-45 Crimping tool
- A tester

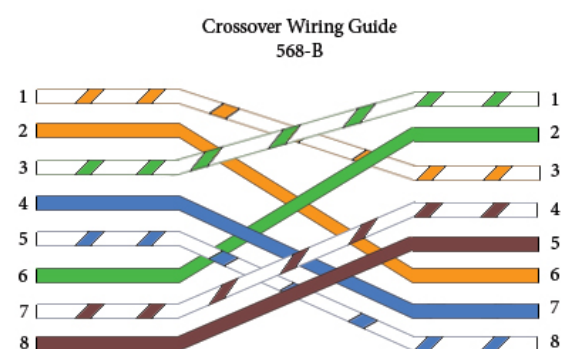
For making a straight-through wired cable:

The wiring configuration at one end is identical to the wiring configuration at the other end. This means that pin 1 on one end is connected to pin 1 on the other end, pin 2 to pin 2, and so on.



For making a crossover wired cable:

The wiring configuration is crossed over at one end, meaning that some of the wires are reversed compared to the other end. Specifically, the transmit (TX) pins on one end are connected to the receive (RX) pins on the other end, and vice versa.



Steps to create an ethernet cable:

- Prepare Cable: Strip outer insulation, untwist wires, and arrange in a required sequence.

- Trim and Align: Cut excess wire, and align wires in the correct order.
- Insert Wires: Place wires into the RJ45 connector in sequence.
- Crimp Connector: Use a crimping tool to attach the connector securely.
- Repeat: Repeat steps for the other end.
- Test Cable: Use the cable tester to verify connectivity.

B) IP address configuration (both Static and DHCP) on Linux and Windows systems.

Static IP address configuration in Linux:

- Open a terminal window
- Edit the network configuration file using a text editor such as nano or vi. For example:

```
sudo nano /etc/network/interfaces
```

- Locate the network interface you want to configure (e.g., eth0).
- Change or add the configuration to specify a static IP address, netmask, gateway, and DNS servers. For example:

```
auto eth0
iface eth0 inet static
    address 192.168.1.100
    netmask 255.255.255.0
    gateway 192.168.1.1
    dns-nameservers 8.8.8.8 8.8.4.4
```

- Save the file and exit the text editor
- Restart the networking service to apply the changes

```
sudo systemctl restart networking
```

DHCP IP address configuration in Linux:

- Open a terminal window
- Edit the network configuration file using a text editor such as nano or vi. For example:

```
sudo nano /etc/network/interfaces
```

- Locate the network interface you want to configure (e.g., eth0).
- Change or add the configuration to specify a static IP address, netmask, gateway, and DNS servers. For example:

```
auto eth0
```

```
iface eth0 inet dhcp
```

- Save the file and exit the text editor
- Restart the networking service to apply the changes

```
sudo systemctl restart networking
```

Static IP address configuration in Windows:

- Open Control Panel.
- Go to Network and Sharing Center.
- Click on Change adapter settings.
- Right-click on the network adapter you want to configure and select Properties.
- Select Internet Protocol Version 4 (TCP/IPv4) and click Properties.
- Select "Use the following IP address" and enter the IP address, subnet mask, default gateway, and DNS server addresses.
- Click OK to save the settings.

DHCP IP address configuration in Windows:

- Open Control Panel.
- Go to Network and Sharing Center.
- Click on Change adapter settings.
- Right-click on the network adapter you want to configure and select Properties.
- Select Internet Protocol Version 4 (TCP/IPv4) and click Properties.
- Select "Obtain an IP address automatically" and "Obtain DNS server address automatically."
- Click OK to save the settings.

C) Introduction to the following important network-related tools and commands with appropriate examples,

1. ipconfig (Windows)

2. ifconfig (Linux)

This command displays network interface configuration information on Linux systems.

```
> ifconfig
enp2s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 08:8f:c3:ec:53:1e 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

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 6874 bytes 717428 (717.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 6874 bytes 717428 (717.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlo1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.117.108 netmask 255.255.255.0 broadcast 192.168.117.255
    inet6 2409:40e0:1019:69b5:30c2:6105:7f61:51c0 prefixlen 64 scopeid 0x0<global>
    inet6 fe80::30ad:b007:1235:6965 prefixlen 64 scopeid 0x20<link>
    inet6 2409:40e0:1019:69b5:f25d:16b0:1364:6771 prefixlen 64 scopeid 0x0<global>
    ether 74:97:79:85:66:fd txqueuelen 1000 (Ethernet)
    RX packets 367893 bytes 422653142 (422.6 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 125191 bytes 20962757 (20.9 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

3. ip

It is a versatile command-line tool for network configuration on Linux systems. It provides more advanced functionalities compared to ifconfig.

```
> ip
Usage: ip [ OPTIONS ] OBJECT { COMMAND | help }
       ip [ -force ] -batch filename
where  OBJECT := { address | addrlabel | fou | help | ila | ioam | l2tp | link |
                 macsec | maddress | monitor | mptcp | mroute | mrule |
                 neighbor | neighbour | netconf | netns | nexthop | ntable |
                 ntbl | route | rule | sr | tap | tcpmetrics |
                 token | tunnel | tuntap | vrf | xfrm }
       OPTIONS := { -V[ersion] | -s[tatistics] | -d[etails] | -r[esolve] |
                   -h[uman-readable] | -iec | -j[son] | -p[retty] |
                   -f[amily] { inet | inet6 | mpls | bridge | link } |
                   -4 | -6 | -M | -B | -0 |
                   -l[oops] { maximum-addr-flush-attempts } | -br[ief] |
                   -o[neline] | -t[imestamp] | -ts[hort] | -b[atch] [filename] |
                   -rc[vbuf] [size] | -n[etns] name | -N[umeric] | -a[ll] |
                   -c[olor]}

> ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp2s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
    link/ether 08:8f:c3:ec:53:1e brd ff:ff:ff:ff:ff:ff
3: wlo1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 74:97:79:85:66:fd brd ff:ff:ff:ff:ff:ff
    altname wlp3s0
    inet 192.168.117.108/24 brd 192.168.117.255 scope global dynamic noprefixroute wlo1
        valid_lft 2482sec preferred_lft 2482sec
    inet6 2409:40e0:1019:69b5:f25d:16b0:1364:6771/64 scope global temporary dynamic
        valid_lft 7077sec preferred_lft 7077sec
    inet6 2409:40e0:1019:69b5:30c2:6105:7f61:51c0/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 7077sec preferred_lft 7077sec
    inet6 fe80::30ad:b007:1235:6965/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

4. hostname

It displays or sets the hostname of the system.

```
> hostname  
batcomputer
```

5. ping

It tests connectivity between two hosts by sending ICMP echo request packets and waiting for ICMP echo reply packets.

```
> ping google.com  
PING google.com(dell1s04-in-x0e.1e100.net (2404:6800:4002:80f::200e)) 56 data bytes  
64 bytes from dell1s04-in-x0e.1e100.net (2404:6800:4002:80f::200e): icmp_seq=1 ttl=57 time=43.4 ms  
64 bytes from dell1s04-in-x0e.1e100.net (2404:6800:4002:80f::200e): icmp_seq=2 ttl=57 time=61.9 ms  
64 bytes from dell1s04-in-x0e.1e100.net (2404:6800:4002:80f::200e): icmp_seq=3 ttl=57 time=65.5 ms  
64 bytes from dell1s04-in-x0e.1e100.net (2404:6800:4002:80f::200e): icmp_seq=4 ttl=57 time=43.9 ms  
64 bytes from dell1s04-in-x0e.1e100.net (2404:6800:4002:80f::200e): icmp_seq=5 ttl=57 time=62.0 ms  
^C  
--- google.com ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 4007ms  
rtt min/avg/max/mdev = 43.369/55.320/65.491/9.643 ms
```

6. netstat

It displays network statistics and active network connections.

```
> netstat -a  
Active Internet connections (servers and established)  
Proto Recv-Q Send-Q Local Address           Foreign Address         State  
tcp        0      0 0.0.0.0:ssh              0.0.0.0:*                LISTEN  
tcp        0      0 localhost:domain        0.0.0.0:*                LISTEN  
tcp        0      0 localhost:ipp            0.0.0.0:*                LISTEN  
tcp        0      0 batcomputer:34122       172.64.148.154:https    ESTABLISHED  
tcp6       0      0 [::]:ssh                [::]:*                  LISTEN  
tcp6       0      0 localhost:ipp            [::]:*                  LISTEN  
tcp6       76      0 2409:40e0:1019:69:51624 2a04:4e42:42::347:https CLOSE_WAIT  
tcp6       76      0 batcomputer:59530       2a04:4e42:42::347:https CLOSE_WAIT  
tcp6       76      0 2409:40e0:1019:69:51580 2a04:4e42:42::347:https CLOSE_WAIT  
tcp6       0      0 batcomputer:33878       ec2-35-153-189-18:https ESTABLISHED
```

7. route

It displays or modifies the IP routing table.

```
> route  
Kernel IP routing table  
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface  
default          _gateway        0.0.0.0          UG    600    0      0 wlo1  
link-local       0.0.0.0         255.255.0.0      U      1000   0      0 wlo1  
192.168.117.0    0.0.0.0         255.255.255.0    U      600    0      0 wlo1
```

8. traceroute or tracert

It traces the route taken by packets from the source to the

destination by sending ICMP packets with increasing TTL values.

```
> traceroute google.com
traceroute to google.com (142.250.194.78), 30 hops max, 60 byte packets
 1 _gateway (192.168.117.102) 23.844 ms 24.192 ms 24.564 ms
 2 255.0.0.0 (255.0.0.0) 44.473 ms 44.452 ms 46.953 ms
 3 255.0.0.2 (255.0.0.2) 46.493 ms 47.701 ms 47.683 ms
 4 255.0.0.3 (255.0.0.3) 46.880 ms 46.864 ms 46.848 ms
 5 172.27.248.115 (172.27.248.115) 46.973 ms 46.957 ms 172.27.248.114 (172.27.248.114) 46.936 ms
 6 192.168.137.188 (192.168.137.188) 46.920 ms 20.991 ms 20.101 ms
 7 * * *
 8 * * *
 9 * * *
10 * * *
11 * * *
12 72.14.195.22 (72.14.195.22) 47.761 ms 44.329 ms 72.14.195.34 (72.14.195.34) 43.901 ms
13 * * *
14 172.253.67.96 (172.253.67.96) 42.160 ms 142.251.49.115 (142.251.49.115) 43.122 ms 172.253.73.194 (172.253.73.194) 42.622 ms
15 142.250.63.52 (142.250.63.52) 45.802 ms 142.251.49.115 (142.251.49.115) 45.749 ms 142.251.49.121 (142.251.49.121) 43.370 ms
16 192.178.83.221 (192.178.83.221) 46.787 ms del12s03-in-f14.1e100.net (142.250.194.78) 52.457 ms 52.417 ms
```

9. tcpdump

It is a command-line packet analyzer that captures and displays network traffic.

```
> sudo tcpdump -i wlo1
```

```
tcpdump: verbose output suppressed, use -v[v]... for full protocol decoding
listening on wlo1, link-type EN10MB (Ethernet), snapshot length 262144 bytes
11:47:37.426186 IP6 2606:4700:8d90:8c0a:8bd9:505:c497:de06.https > batcomputer.3936090], length 24
11:47:37.426424 IP6 batcomputer.40158 > 2606:4700:8d90:8c0a:8bd9:505:c497:de06.https
11:47:37.453755 IP6 2606:4700:8d90:8c0a:8bd9:505:c497:de06.https > batcomputer.53363
11:47:37.453974 IP batcomputer.53363 > _gateway.domain: 31387+ PTR? e.d.
11:47:37.458562 IP gateway.domain > batcomputer.53363: 31387 NXDomain (64 bytes)
```

10. Wireshark

It is a graphical packet analyzer that allows real-time monitoring and analysis of network traffic.