LAB #1
Networking Tools

temungussie10@gmail.com

Cont ..

- Viewing and changing the configuration of your computer's network interface, such as addresses and other protocol parameters.
- Testing your computer's network connectivity, such as ability to communicate with other computers and statistics of the communication.
- View and analyze traffic sent/received by your computer, as well as other computers on a network.

To do all these things we can use either **graphical user interface GUI** or **command line interface (CLI)**, but we will show examples and expect you to use the command line interface on most occasions.

Cont …

- This is because once you know the command line interface, it is very easy to perform the same operations in the GUI (however, vice versa is not true: if you learn the GUI, it may be hard to understand the options of the command line interface). Also note that some network equipment is managed by a command line interface: e.g. you may log on to a router or switch and set the configuration via the command line interface only.

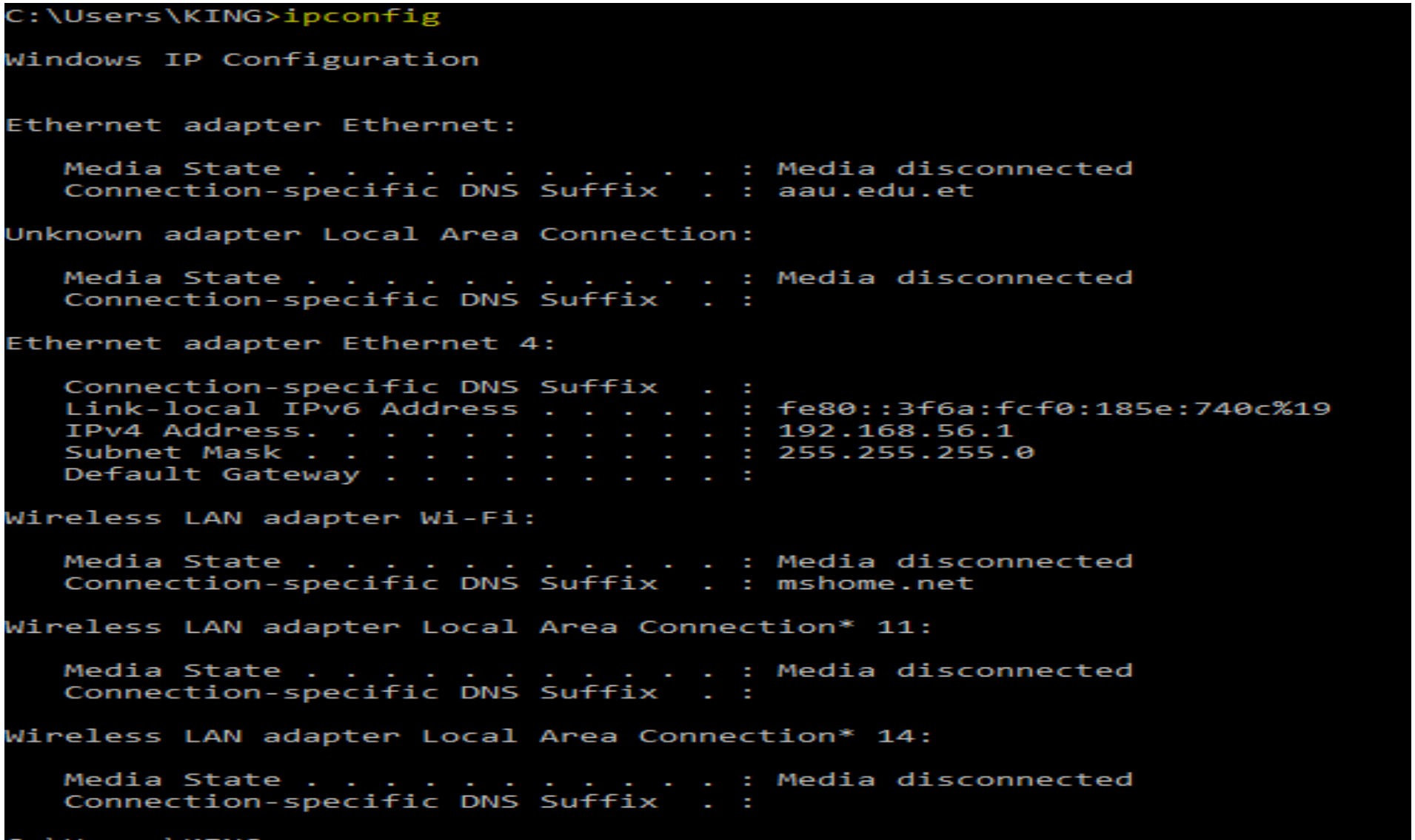
Viewing Network Interface Information

Your computer connects to the LAN via one of its Network Interface Cards (NIC). Almost all operating systems allow the user to view information about the current NIC connection, including:

- MAC (or hardware) address
- IP address and subnet mask
- Addresses of other important nodes (servers) on the network
- Traffic sent/received by the NIC

Ipconfig or ipconfig /all

- **Ipconfig or ipconfig /all** is the main command that shows summary information for your different network interfaces.



Testing Network Connectivity

A basic task for diagnosing the connectivity of a network is to test whether one computer can communicate with another. This is normally performed using the **Internet Control Message Protocol (ICMP).** A user application or utility that implements ICMP for testing connectivity is **ping**

Ping measures the time it takes from sending the message, to when the response is received. That is, the delay to the destination and back, i.e. the **round trip time (RTT).**

>ping DESTINATION ADRESS

```
C:\Users\KING>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=46ms TTL=112
Reply from 8.8.8.8: bytes=32 time=46ms TTL=112
Reply from 8.8.8.8: bytes=32 time=46ms TTL=112
Reply from 8.8.8.8: bytes=32 time=47ms TTL=112

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 46ms, Maximum = 47ms, Average = 46ms

C:\Users\KING>

C:\Users\KING>ping www.aau.edu.et

Pinging www.aau.edu.et [10.90.10.76] with 32 bytes of data:
Reply from 10.90.10.76: bytes=32 time<1ms TTL=62
Reply from 10.90.10.76: bytes=32 time<1ms TTL=62
Reply from 10.90.10.76: bytes=32 time=8ms TTL=62
Reply from 10.90.10.76: bytes=32 time=1ms TTL=62

Ping statistics for 10.90.10.76:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 8ms, Average = 2ms

C:\Users\KING>
```

Testing a Route

- Another useful network connectivity test is to determine the path (or route) that a message takes to reach its destination . An application that implements this in Windows is **tracert**. Like ping, an ICMP message is sent to the destination and returned, but with traceroute the set of routers along the way also send a response to the source.

>tracert DESTINATION

```
C:\Users\KING>tracert 8.8.8.8

Tracing route to dns.google [8.8.8.8]
over a maximum of 30 hops:

  1     13 ms      1 ms      1 ms   10.4.15.1
  2     <1 ms     <1 ms     <1 ms   10.4.3.1
  3      1 ms      1 ms      1 ms   196.189.55.65
  4     34 ms     39 ms     23 ms   10.149.205.2
  5      *         *         *      Request timed out.
  6      *         *         *      Request timed out.
  7     12 ms     13 ms     19 ms   41.189.225.201
  8     74 ms     72 ms     87 ms   41.189.225.170
  9     46 ms     47 ms     46 ms   108.170.246.113
 10     47 ms     47 ms     47 ms   142.251.66.203
 11     72 ms     72 ms     72 ms   dns.google [8.8.8.8]

Trace complete.


C:\Users\KING>tracert aau.edu.et

Tracing route to aau.edu.et [10.90.10.76]
over a maximum of 30 hops:

  1     16 ms     45 ms      1 ms   10.4.15.1
  2      7 ms      2 ms      3 ms   10.1.0.61
  3     <1 ms     <1 ms     <1 ms   10.90.10.76

Trace complete.

C:\Users\KING>
```

Converting Between Domain Names and IP addresses

- We know that the Domain Name Service (DNS) is used for mapping domain names (user-friendly addresses) into IP addresses (computer-readable addresses) or the opposite. In this lab we will use **nslookup**, where we give the domain name as a parameter and returns the corresponding IP address and vice versa.

> **nslookup DOMAIN #** returns IP address
> **nslookup IPADDRESS #** returns domain name

```
C:\Users\KING>nslookup 8.8.8.8
Server:  UnKnown
Address:  10.90.104.31

Name:    dns.google
Address:  8.8.8.8


C:\Users\KING>nslookup hilcoe.net
Server:  UnKnown
Address:  10.90.104.31

Non-authoritative answer:
Name:    hilcoe.net
Address:  109.70.148.48
```

Viewing the Routing Table

- IP uses routing table to determine where to send datagrams. This applies to end hosts (like PCs), as well as routers, however a routing table on a host is typically quite simple, since all packets are often sent to a local (default) router. You can view your routing table using the route command:

>route PRINT

```
C:\Users\KING>route print
===========================================================================
Interface List
  7...........................WireGuard Tunnel
 19...0a 00 27 00 00 13 ......VirtualBox Host-Only Ethernet Adapter
 18...94 53 30 4d cf 69 ......Realtek RTL8723BE 802.11 bgn Wi-Fi Adapter
 14...96 53 30 4d cf 69 ......Microsoft Wi-Fi Direct Virtual Adapter
  4...94 53 30 4d cf 69 ......Microsoft Wi-Fi Direct Virtual Adapter #4
 16...98 e7 f4 db fb 5c ......Realtek PCIe FE Family Controller
  1...........................Software Loopback Interface 1
===========================================================================

IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0      10.4.15.1      10.4.15.126     35
        10.4.15.0    255.255.255.0         On-link      10.4.15.126    291
      10.4.15.126  255.255.255.255         On-link      10.4.15.126    291
      10.4.15.255  255.255.255.255         On-link      10.4.15.126    291
        127.0.0.0        255.0.0.0         On-link        127.0.0.1    331
        127.0.0.1  255.255.255.255         On-link        127.0.0.1    331
  127.255.255.255  255.255.255.255         On-link        127.0.0.1    331
     192.168.56.0    255.255.255.0         On-link     192.168.56.1    281
     192.168.56.1  255.255.255.255         On-link     192.168.56.1    281
   192.168.56.255  255.255.255.255         On-link     192.168.56.1    281
        224.0.0.0        240.0.0.0         On-link        127.0.0.1    331
        224.0.0.0        240.0.0.0         On-link     192.168.56.1    281
        224.0.0.0        240.0.0.0         On-link      10.4.15.126    291
  255.255.255.255  255.255.255.255         On-link        127.0.0.1    331
  255.255.255.255  255.255.255.255         On-link     192.168.56.1    281
  255.255.255.255  255.255.255.255         On-link      10.4.15.126    291
===========================================================================
```

Converting IP Addresses to Hardware Addresses

- Remember that IP addresses are logical addresses, but For a computer to send data to another computer on the same LAN they must use hardware (or MAC) addresses. For example, if computer A wants to send an IP datagram to computer B (on the same network as A) with IP address 192.168.1.3, then computer A must know the **hardware address** of computer B. Hence, the Address Resolution Protocol (ARP) is used to find the corresponding hardware addresses for a given IP address, and the it puts in a data structure called **ARP table**

> arp –a shows the **arp table** of the system

```
C:\Users\KING>arp -a

Interface: 10.4.15.126 --- 0x10
  Internet Address      Physical Address      Type
  10.4.15.1             34-a2-a2-8a-6e-01     dynamic
  10.4.15.58            1c-4d-70-fe-6a-ff     dynamic
  10.4.15.255           ff-ff-ff-ff-ff-ff     static
  224.0.0.2             01-00-5e-00-00-02     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

Interface: 192.168.56.1 --- 0x13
  Internet Address      Physical Address      Type
  192.168.56.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.2             01-00-5e-00-00-02     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static

C:\Users\KING>
```

Network Statistics

- Along with different options, a tool that allows you to view many different network statistics is **netstat**. For example, you can view interface statistics (similar to ipconfig), routing table statistics (same as route print), connection statistics and TCP/IP packet statistics.

> **netstat:** alone shows active connections along with their state

```
C:\Users\KING>netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    10.4.15.126:51775      a95-101-20-209:https   CLOSE_WAIT
  TCP    10.4.15.126:51779      13.107.213.63:https    CLOSE_WAIT
  TCP    10.4.15.126:51781      13.107.213.254:https   CLOSE_WAIT
  TCP    10.4.15.126:51789      205.128.93.254:http    TIME_WAIT
  TCP    10.4.15.126:52819      20.90.152.133:https    ESTABLISHED
  TCP    10.4.15.126:52820      55:https               ESTABLISHED
  TCP    10.4.15.126:52928      13.69.239.77:https     ESTABLISHED
  TCP    10.4.15.126:56394      a95-101-20-209:https   CLOSE_WAIT
  TCP    10.4.15.126:56395      a95-101-20-209:https   CLOSE_WAIT
  TCP    10.4.15.126:56396      a95-101-20-209:https   CLOSE_WAIT
  TCP    10.4.15.126:56397      a95-101-20-209:https   CLOSE_WAIT
  TCP    10.4.15.126:62844      sof02s49-in-f14:https  TIME_WAIT
  TCP    10.4.15.126:62851      mrs09s13-in-f13:https  ESTABLISHED
  TCP    10.4.15.126:62852      a2-17-161-65:https     ESTABLISHED
  TCP    10.4.15.126:62853      162.159.153.247:https  ESTABLISHED
  TCP    10.4.15.126:62856      162.159.153.247:https  ESTABLISHED
  TCP    10.4.15.126:62858      204.79.197.239:https   ESTABLISHED
  TCP    10.4.15.126:62859      ec2-52-207-122-56:https  ESTABLISHED
  TCP    10.4.15.126:62860      a-0003:https           ESTABLISHED
  TCP    10.4.15.126:62861      a2-21-14-187:https     ESTABLISHED
  TCP    10.4.15.126:62862      a2-21-14-187:https     ESTABLISHED
  TCP    10.4.15.126:62864      68.219.88.97:https     ESTABLISHED
  TCP    10.4.15.126:62865      a-0001:https           ESTABLISHED
  TCP    10.4.15.126:62868      server-18-165-242-110:https  ESTABLISHED
```

> netstat -n –t

- "-n" option: This option instructs "netstat" to display numerical IP addresses and port numbers instead of attempting to resolve them to hostnames and service names.
- "-t" option: This option filters the output of "netstat" to display only TCP (Transmission Control Protocol) connections and related information.

Cont …

```
C:\Users\KING> netstat -n -t

Active Connections

  Proto  Local Address          Foreign Address        State           Offload State

  TCP    10.4.15.126:52819      20.90.152.133:443      ESTABLISHED     InHost
  TCP    10.4.15.126:52820      34.117.65.55:443       ESTABLISHED     InHost
  TCP    10.4.15.126:53040      10.90.104.31:53        TIME_WAIT       InHost
  TCP    10.4.15.126:53041      96.17.193.13:443       ESTABLISHED     InHost
  TCP    10.4.15.126:53042      96.17.193.13:443       ESTABLISHED     InHost
  TCP    10.4.15.126:53043      13.89.179.8:443        TIME_WAIT       InHost
  TCP    10.4.15.126:53044      3.233.117.176:443      ESTABLISHED     InHost
  TCP    10.4.15.126:53045      13.89.179.8:443        TIME_WAIT       InHost
  TCP    10.4.15.126:53046      54.225.175.183:443     ESTABLISHED     InHost
  TCP    10.4.15.126:62859      52.207.122.56:443      ESTABLISHED     InHost
  TCP    10.4.15.126:62878      3.228.185.195:443      ESTABLISHED     InHost
  TCP    10.4.15.126:62924      52.70.125.53:443       ESTABLISHED     InHost
  TCP    10.4.15.126:62931      151.101.1.44:443       ESTABLISHED     InHost
  TCP    10.4.15.126:62951      2.16.149.133:80        ESTABLISHED     InHost
  TCP    10.4.15.126:62953      104.82.150.7:80        ESTABLISHED     InHost
  TCP    10.4.15.126:62954      35.208.249.213:443     ESTABLISHED     InHost
  TCP    10.4.15.126:62955      35.213.89.133:443      ESTABLISHED     InHost
  TCP    10.4.15.126:62956      104.82.150.7:80        ESTABLISHED     InHost
  TCP    10.4.15.126:62995      2.17.209.189:443       CLOSE_WAIT      InHost
  TCP    10.4.15.126:62996      2.17.209.189:443       CLOSE_WAIT      InHost
  TCP    10.4.15.126:62997      2.17.209.189:443       CLOSE_WAIT      InHost
  TCP    10.4.15.126:62998      2.17.209.189:443       CLOSE_WAIT      InHost
  TCP    10.4.15.126:63001      96.17.206.197:443      CLOSE_WAIT      InHost
  TCP    10.4.15.126:63002      184.25.204.57:443      CLOSE_WAIT      InHost
```

> netstat -s

- The output is typically divided into sections for different network protocols, such as “Ip" for IP (Internet Protocol), "Tcp" for TCP (Transmission Control Protocol), and "Udp" for UDP (User Datagram Protocol). Each section provides specific statistics related to the corresponding protocol, including packet counts, connection details, segment information, and error counts and so on depending the system you are on.

Cont …

- Keep in mind that the actual output and the statistics provided can vary between different operating systems, versions, and configurations.

Cont …(**> netstat -s**)

```
C:\Users\KING> netstat -s

IPv4 Statistics

  Packets Received                   = 38868
  Received Header Errors             = 0
  Received Address Errors            = 10
  Datagrams Forwarded                = 0
  Unknown Protocols Received         = 0
  Received Packets Discarded         = 367
  Received Packets Delivered         = 41084
  Output Requests                    = 32859
  Routing Discards                   = 0
  Discarded Output Packets           = 250
  Output Packet No Route             = 25
  Reassembly Required                = 0
  Reassembly Successful              = 0
  Reassembly Failures                = 0
  Datagrams Successfully Fragmented  = 0
  Datagrams Failing Fragmentation    = 0
  Fragments Created                  = 0
```

> netstat -s (ICMPv4) statistics

```
ICMPv4 Statistics

                          Received    Sent
  Messages                17          43
  Errors                  0           0
  Destination Unreachable 17          43
  Time Exceeded           0           0
  Parameter Problems      0           0
  Source Quenches         0           0
  Redirects               0           0
  Echo Replies            0           0
  Echos                   0           0
  Timestamps              0           0
  Timestamp Replies       0           0
  Address Masks           0           0
  Address Mask Replies    0           0
  Router Solicitations    0           0
```

> netstat -s (TCP) statistics

```
TCP Statistics for IPv4

  Active Opens                            = 591
  Passive Opens                           = 7
  Failed Connection Attempts              = 30
  Reset Connections                       = 147
  Current Connections                     = 22
  Segments Received                       = 54429
  Segments Sent                           = 52205
  Segments Retransmitted                  = 335
```

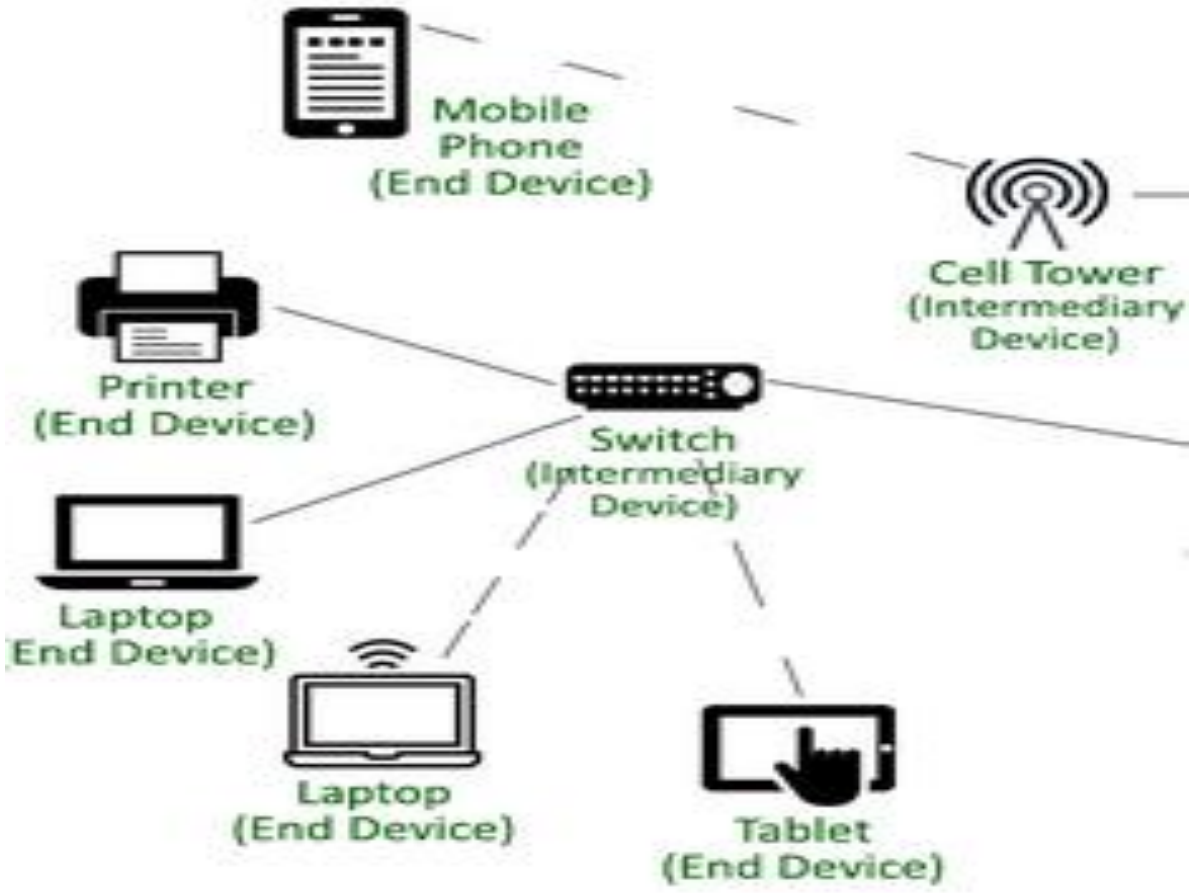
> netstat -s (UDP) statistics

```
UDP Statistics for IPv4

  Datagrams Received    = 14541
  No Ports              = 368
  Receive Errors        = 3
  Datagrams Sent        = 8167
```

Network Hardware Tools

End Devices: These devices are typically the endpoints of a network and are responsible for generating, receiving, and processing data.

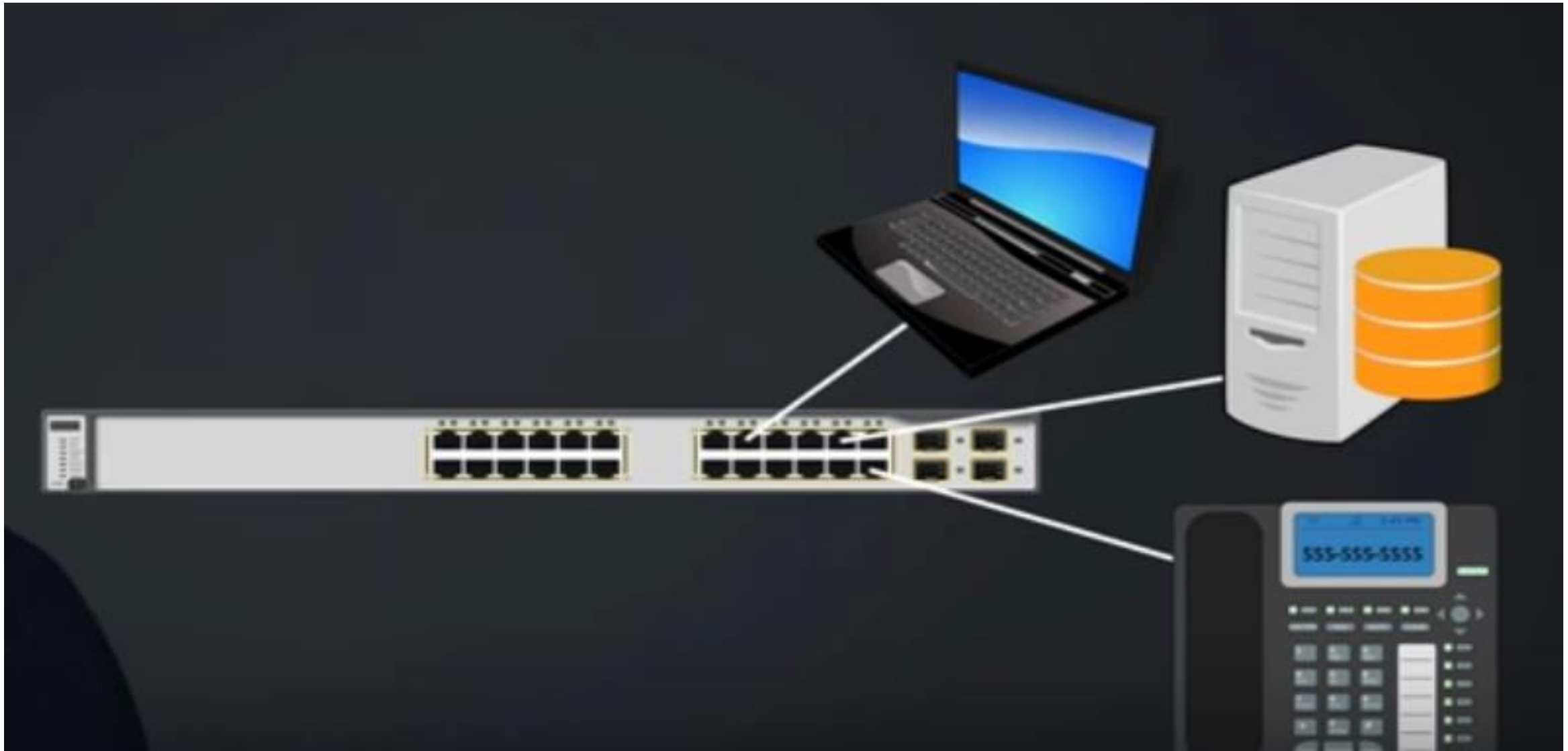


The HUBs In networking, a hub is a networking device that connects multiple devices in a local area network (LAN). It operates at the physical layer (Layer 1) of the OSI (Open Systems Interconnection) model. **However, it is important to note that hubs are less commonly used today compared to switches, which provide more advanced functionality.**

HUB

Layer2/Access Switches

a switch is a networking device that connects devices within a local area network (LAN). It operates at the DDLlayer (Layer 2) of the OSI

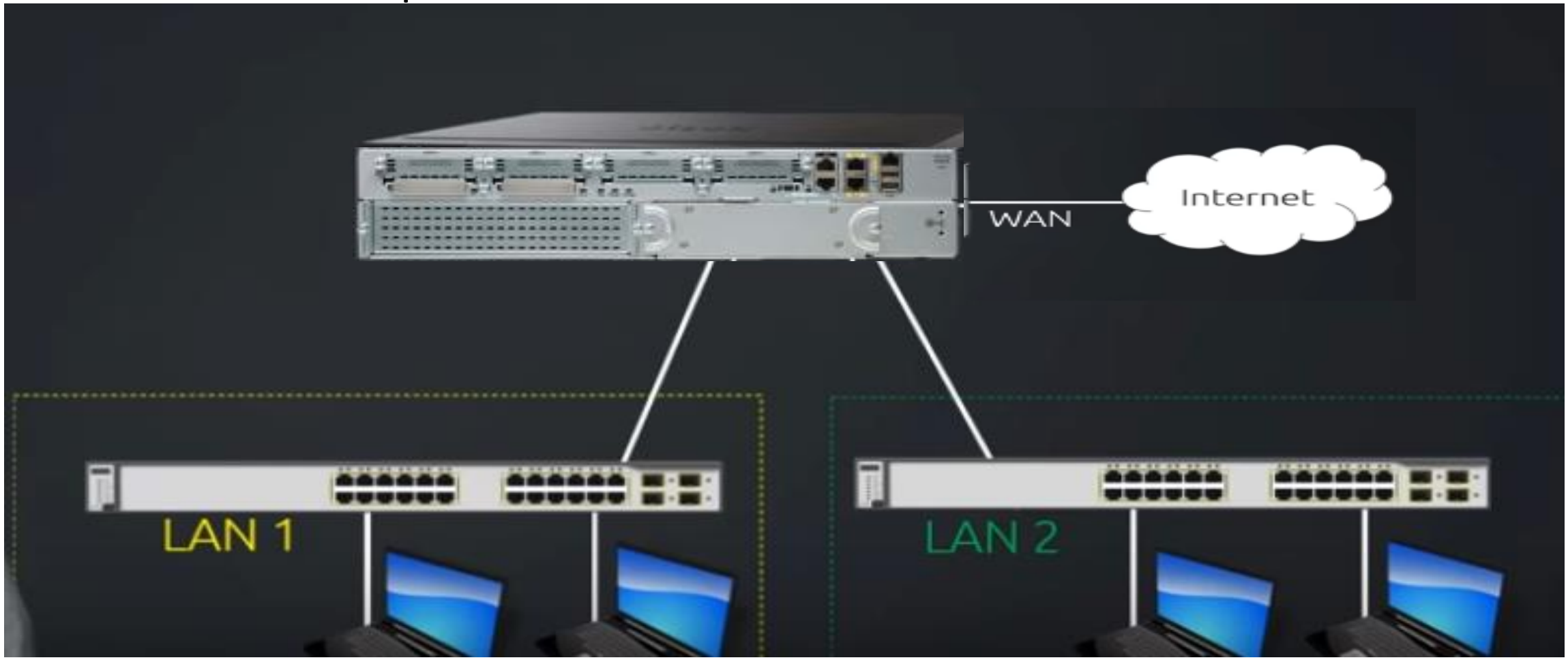


Here are some key aspects and functions of a switch in networking:

- ✓MAC Address Learning:
- ✓Forwarding and Filtering:
- ✓Unicast, Broadcast, and Multicast Traffic:
- ✓PoE (Power over Ethernet) Support:
- ✓Management and Configuration:
- ✓VLAN Support:
- ✓Collision Domain Separation:
- ✓Stacking technology

Router

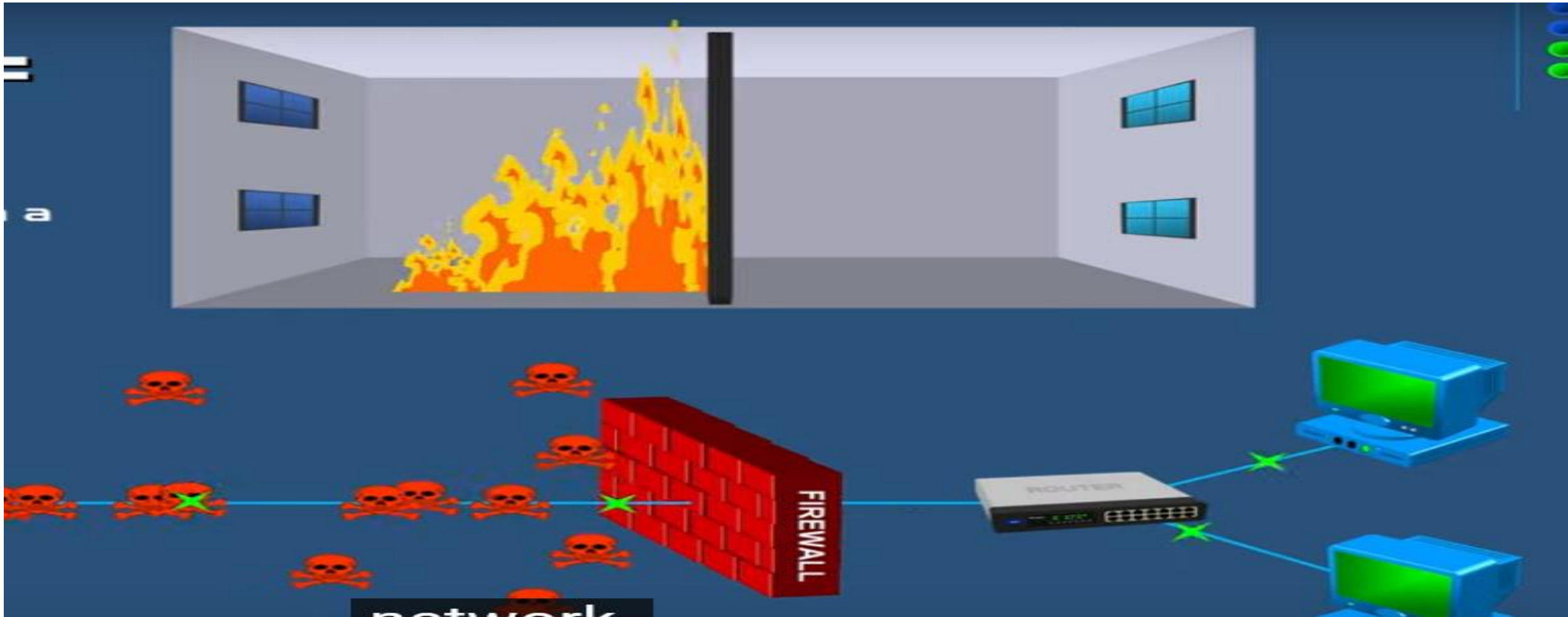
Is a fundamental networking device that plays a crucial role in connecting and directing data packets between different computer networks.



Here are some key aspects and functions of a router in networking:

- ✓Network Interconnection
- ✓IP routing
- ✓Packet forwarding
- ✓Network address translation (NAT)
- ✓Firewall and security
- ✓Quality of service (QoS)
- ✓Virtual private network (VPN)

Firewall



Here are some key aspects and functions of a firewall in networking:

In networking, a firewall is a network security device that acts as a barrier between an internal network (such as a local area network, or LAN) and external networks (such as the internet).

- ✓Traffic Filtering
- ✓Access Control
- ✓Stateful Inspection
- ✓Intrusion Detection and Prevention
- ✓Logging and Reporting