Network Traffic Filtering

To start with, the filtering process is the process of monitoring the outgoing and incoming data and selecting which parts of the data to allow and which part to deny. This filtering can be based on three criteria; MAC addresses, IP addresses, and port numbers.

1. **MAC address filtering:**
   The MAC address, or sometimes referred to as *hardware address*, is the addressing used in the data-link layer in Ethernet-based networks. Local networks that are not based on Ethernet use a different type of addressing.
   Since the Ethernet frame has a source and a destination MAC addresses, the filtering can be done on anyone of them or both. There are modern Ethernet switches that can have access-control lists to filter frames based on their source and/or destination MAC address. These switches can stop all frames flowing from a certain host or stop all frames going to a certain host. And it can also stop all frames going between two known hosts.
   There are other types of security that some modern switches provide. Some switches have the capability to configure a switch port to receive frames from a defined MAC address such that when the host's cable is removed from the switch, no other host can use the same port. This way, if an intruder disconnects a host from the switch and connects his own host trying to impersonate the original host, he fails.
2. **IP address filtering:**
   Each IP packet has a source and destination IP address. The source IP address is meant to define the sender to the destination, while the destination IP address is used to route the data to the destination.
   Firewalls and some advanced routers can filter these packets based on their source and/or destination IP addresses. This filtering is done to allow and disallow traffic between selected hosts or complete networks. For example, if you like to forbid the traffic between the sales network and the accounting network because it is not functionally necessary, you can create an access-control list on the router connecting these two networks to disallow this traffic.
   IP-address-based filters can be specific, such as disallowing traffic to a server except traffic coming from a known administrator computer.
3. **Port number filtering (service-based filtering):**
   Each transport-layer segment, whether it was a TCP segment or a UDP segment, contains a source and a destination port number.

Usually, for the traffic generated from a user terminal, the source port is chosen randomly and the destination port is defined based on the application-layer protocol used. On the reply segment coming from a server for example, the numbering is inverted. The source port represents a known number depending on the application-layer protocol type, and the destination port number is the random number that was earlier chosen by the sender of the first segment.

Some advanced routers and gateways have the capability of filtering traffic based on port numbers. This filtering provides the ability of blocking certain services such as email traffic or HTTP traffic. For example, blocking or disallowing traffic destined to port number 80 will stop the web-browsing requests from being sent, thus, preventing the users from browsing the web while they can still exchange emails through SMTP and POP3, and can still download files through FTP.

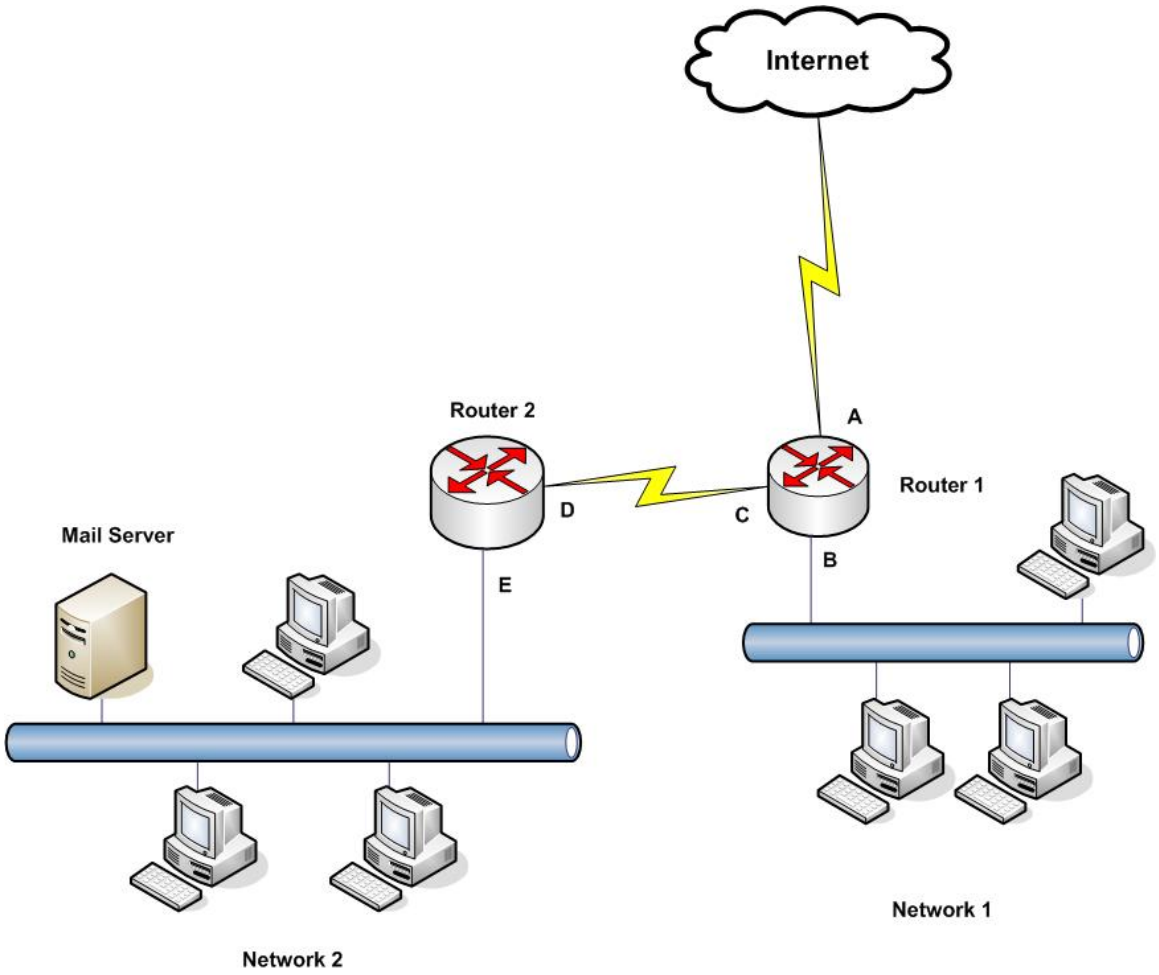
Each one of the three types of filtering mentioned earlier has its advantages and disadvantages. The biggest disadvantage for all types of filtering is the latency caused by the filtering process that delays the arrival of the data from source to destination.

As the access-control list goes longer and longer, the latency caused by it increases.

The complete security system can not be achieved by filtering only. Combining two or more of the filtering techniques mentioned earlier certainly increases the security and let the administrator control the type and route of the traffic inside an internetwork.

Combining the IP address filtering and the port number filtering gives the administrator the ability to deny or allow a certain type of traffic between two known hosts instead of denying or allowing all types of services. For example, instead of denying all traffic between a computer and a server, an administrator can deny the Telnet traffic originated from the computer and going to the server.

To complete our simplified introduction to the filtering subject, another issue must be discussed; the placement of the access-control lists. Placing the access-control list in the right place is very important to achieve the level of security that we intend to. The best way to illustrate this is by the means of an example. Consider the following figure.



Let's consider a filter that denies Telnet traffic (port 23) to the mail server in Network 2 in the figure above. When applying this filter to the incoming traffic at point A, all intruders from the Internet will not be allowed to have Telnet traffic to the mail server. In this case, the computers in Networks1 and 2 are allowed to have Telnet traffic to the mail server.

If the filter is applied to the incoming traffic at point B, the computers on Network 1 will not be allowed to send Telnet traffic to the mail server. On the other hand, intruders from the Internet and computers in Network 2 will be allowed to have Telnet traffic sent to the mail server.

Applying the filter to the outgoing traffic on point C will deny the Telnet traffic from Network 1 and the Internet sent to the mail server. Computers on Network 2 can send Telnet traffic to the mail server.

When the filter is applied at the incoming traffic on point D or outgoing traffic on point E the result is similar to applying the filter to the outgoing traffic on point C.

When applying the filter in any place we wish, we must keep in mind that the closer the filter is to the source, the better the performance of the

network is. This is caused by the fact that discarding the denied data at a point close to the source will reduce the overall traffic traveling through the network, thus increasing the throughput.

From the above example we conclude that we need to be careful in writing the right filter and placing it on the right interface and in the right direction.