Network Security

Professor Gaurav S. Kasbekar
Department of Electrical Engineering
Indian Institute of Technology, Bombay

Week - 09

Lecture - 53

Firewalls and Intrusion Detection Systems: Part 3

Hello, recall that in the previous lecture, we discussed various protocols that are part of the TCP/IP stack, such as ICMP, DNS, and so on. We also discussed traceroute. So, now we'll use this knowledge to understand firewalls better. We'll discuss different types of firewalls in this lecture and the next lecture. So, the different types of firewalls are traditional packet filters, stateful packet filters, and application gateways.

Traditional packet filters examine each packet in isolation and take decisions of whether to pass or block the packet without any state information. Stateful packet filters keep track of which TCP connections are active between a host inside the organization's network and outside the network. And based on this state knowledge of which TCP connections are active, they take decisions on whether to block or pass packets. And then there are application gateways, which are specific to different applications, such as Telnet and web. So, there are different application gateways for different applications.

We'll discuss these three types of firewalls in detail. We start with traditional packet filters. So, a traditional packet filter examines each packet in isolation and decides whether the packet should be allowed to pass or it should be dropped. Traditional packet filters do not store any state about which packets were examined in the past and so on. So, this decision of whether the packet should be allowed to pass or it should be dropped is based on rules configured by the network administrator.

Filtering decisions are typically based on various header fields in the packets. These header fields include the IP source and/or destination addresses, which are in the IP header. Then the protocol type in the IP header, whether the protocol type is TCP, UDP, ICMP, OSPF, and so on. So, this protocol type indicates which protocol is being carried in the payload of the IP packet. Examples are TCP, UDP, ICMP, OSPF, and so on.

Then, filtering decisions can also be based on the TCP or UDP source and destination port numbers. Recall that these are in the transport layer header of the packet. So, filtering decisions can also be based on the port numbers. The port numbers indicate what kind of application process has generated that packet. So, that is useful for deciding whether to filter the packet.

Then, filtering decisions are also based on TCP flag bits, such as SYN, ACK, and so on. If the SYN bit is 1, then that indicates that this is part of a TCP connection establishment phase. We'll see later that by looking at whether the SYN flag is 1 or not, we can decide to let a TCP connection start or not let it start. Hence, the TCP flag bits, such as SYN, ACK, and so on, are also useful for taking filtering decisions by traditional packet filters. And the ICMP message type—for example, is it a ping packet or a redirect packet, and so on?

So, this can also help us to make filtering decisions. And often, different rules are used for packets entering and leaving the network. For example, a system administrator may decide to allow users inside the organization's network to browse webpages of servers that are hosted outside the network. But in the other direction, it may not be allowed. The system administrator may not allow users outside the organization's network to browse web pages hosted by servers inside the network.

So, in general, different rules are used for packets entering and leaving the network. And also, different rules are used for different router interfaces. So, to understand this, note that a firewall sits at the boundary between an organization's network and the internet. So, this is the organization's network, and this is the internet. So, a router may have a number of different interfaces, which are shown here.

So, different rules are applied, in general, for different router interfaces. Now, let's look at various examples of how a traditional packet filter might operate. So, we consider different policies that the system administrator wants to enforce and the corresponding firewall settings which enforce these policies. So, suppose the policy is that there should be no outside web access—that is, users inside the organization's network should not be allowed to access web pages outside the organization's network. So, we use the fact that HTTP is the protocol which supports the web application so that the HTTP server runs on port number 80.

We can use this fact to enforce this rule. The firewall setting is that it drops all outgoing packets to any IP address with port number 80. If any user tries to establish a connection with an HTTP server, then it will have to send packets to some IP address, that is the IP

address of the web server, and port number 80. And these packets will be dropped by the firewall. So, users inside the organization's network won't be able to set up connections and browse web pages.

Policy	Firewall Setting
No outside web access	Drop all outgoing packets to any IP address, port 80
No incoming TCP connections, except those for organization's public Web server	Drop all incoming TCP SYN packets to any IP except 130.207.244.203, port 80
Prevent audio-video traffic from eating up the available bandwidth	Drop all UDP packets, except DNS packets
Prevent your network from being tracerouted	Drop all outgoing ICMP TTL expired traffic

So, HTTP connections will be dropped by this firewall setting. As another example, suppose the policy is that there should be no incoming TCP connections except those for the organization's public web server. So, the organization, for example, may have some website, and we want to allow external users to visit that website. But apart from such connections, there should be no other incoming TCP connections. So, how do we enforce this by a suitable firewall setting?

So, we drop all incoming TCP SYN packets to any IP address except 130.207.244.203 and port number 80. This IP address is the IP address of the organization's public web server, and the web server obviously runs on port number 80. So, this firewall setting enforces this policy because, recall that at the time of TCP connection establishment, there are three packets exchanged. The first packet is the TCP SYN packet, then the next packet is the SYNACK packet, and the third packet is the ACK packet. So, the first packet that a user will have to send to set up a TCP connection is the TCP SYN packet.

But if all TCP SYN packets are dropped, except those going to this IP address and this port number, then all incoming TCP connections will be dropped except those for the organization's public web server. Thus, this firewall setting enforces this rule. Then suppose another policy we want to enforce is we want to prevent audio-video traffic from eating up the available bandwidth. This is a common scenario. So, if there is a lot of audio-video traffic, then that can block the network resources and prevent other applications, such as web traffic and FTP and so on, from running.

To enforce this policy, we can use the fact that audio-video traffic typically runs over UDP. So, the corresponding firewall setting which would work is we drop all UDP packets except DNS packets. DNS packets, as we have seen in the previous lecture, run over UDP. So, if the packet is a DNS packet then we should let it pass but apart from this all other UDP packets should be dropped. So, this will prevent audio-video traffic from passing through the firewall, and hence it will enforce this rule.

Now, how can we determine whether it's a DNS packet or not? So, that can be observed from the port number of the packet. So, if it has a DNS port number, then we should pass it, otherwise it should be dropped. Another example of a policy we may want to enforce is preventing our network from being tracerouted. We discussed in the previous lecture how traceroute functions.

So, a source host sends a series of packets with increasing TTL values, and whenever the TTL becomes 0, the router sends a TTL expired packet, and from that, the source host gets to know the IP address of the router. So, if we drop all outgoing ICMP TTL expired traffic, then that will prevent our network from being tracerouted. Consider an attacker who is outside the organization's network, and that attacker sends ICMP TTL. So, that attacker, who is outside the network, runs the traceroute program, and that generates a series of packets with different TTL values, and these are sent inside the organization's network. But when the TTL expires at a router inside the network, that router will send a TTL expired traffic message to the source host, which is the attacker in this case. But if the firewall drops all outgoing ICMP TTL-expired traffic, then that will prevent the attacker from knowing the IP addresses of the hosts inside the network.

So, this firewall setting will prevent the network from being tracerouted. Let's look at some additional examples of how we can enforce appropriate firewall settings. Suppose we want to permit internal hosts to initiate or accept Telnet connections to and from only external hosts from a pre-specified list. So, how can we configure this policy? So, we know the IP addresses of the external hosts from this list.

So, we want to only allow Telnet connections to or from external hosts from this list of IP addresses. So, this can be configured as follows. Telnet runs on port number 23. So, the packet filter forwards only those Telnet packets—that is, those with port number 23—initiated by or to the external hosts from the pre-specified list of IP addresses. So, this will enforce this policy.

So, notice that this filtering policy is based on a combination of IP addresses and port numbers. So, we check the IP address in the packet and compare it with the IP addresses from the pre-specified list. And also we check whether the port number is equal to 23 or not. Hence, this filtering policy is based on a combination of IP addresses and port numbers. So, this shows that a traditional packet filter can apply a filtering policy that is based on a combination of different parameters, such as IP addresses and port numbers.

As another example, suppose an organization wants to allow its internal hosts to initiate TCP connections to external hosts but wants to prevent external hosts from initiating TCP connections to internal hosts. So, how can we enforce this policy? So we can use the following fact which we discussed in the previous lecture. This fact is that the first packet in every TCP connection has the ACK bit set to 0, but all the other packets have the ACK bit set to 1. So, the first packet in a TCP connection is the SYN packet.

It has the SYN flag equal to 1, but the ACK flag in that packet is 0. The second and subsequent packets have the ACK bit equal to 1. So, we can use this fact. The packet filter can drop all incoming TCP packets with the ACK bit set to 0. So, this will prevent external host from initiating TCP connections to internal host because if an external host wants to initiate a TCP connection to an internal host, it will have to send a SYN packet to the internal host.

But this SYN packet will have the ACK bit equal to 0, and the packet filter will drop that packet. So, since the SYN packet is dropped, the TCP connection is not established. But if an internal host wants to set up a TCP connection to an external host, it will send a SYN packet. The packet filter will not drop it because it only drops incoming TCP packets with the ACK bit set to 0. So, it will allow this packet to pass even though the ACK bit in it is 0.

When the external host to which the TCP connection is established responds, the ACK bit will be one in it, so the packet filter will not drop it. So, this will allow the internal host to initiate TCP connections to an external host. So, in this way, we can configure this policy in the traditional packet filter. As another example, consider an organization whose hosts have IP addresses of the form 222.22/16. That is, the IP addresses start with the prefix 222.22.

What types of traffic are allowed and blocked by the access control list in this figure? So, the first rule from the top to bottom that matches is applied. So, the firewall first looks at the first rule in this row, and if it applies, then it allows that traffic because the action here

is ‘allow’. Then, if the first rule does not apply, it applies the second rule, and so on and so forth. So, it keeps on doing this comparison until one of the rules applies.

Action	Source address	Dest. address	Protocol	Source port	Dest. port	Flag bit
allow	222.22/16	outside of 222.22/16	TCP	> 1023	80	any
allow	outside of 222.22/16	222.22/16	TCP	80	> 1023	ACK
allow	222.22/16	outside of 222.22/16	UDP	> 1023	53	--
allow	outside of 222.22/16	222.22/16	UDP	53	> 1023	--
deny	all	all	all	all	all	all

So, the default rule is this: the last one, which denies the packet. So, what types of traffic are allowed and blocked by the access control list in this figure? So, let’s understand what this access control list is saying. If the source address is this—that is, if the source address is that of a host inside the organization’s network, and the destination address is outside the organization’s network, and the protocol is TCP, and the source port is greater than 1023 (that is, this is a client kind of port), and the destination port is 80 (that is, it corresponds to HTTP), and the flag bit is any, then the packet should be allowed. And this is the other rows can be interpreted similarly.

This says that if the source address is from outside the network and the destination address is from inside the network, if it’s a TCP packet and the source port is 80 (that is, HTTP) and the destination port is greater than 1023, and the flag bit ACK is 1, then the packet should be allowed, and so on and so forth. So, these third and fourth rows, these are for DNS, and the first two rows are for HTTP. So, let’s understand what types of traffic are allowed and blocked by this access control list. So, the first two rules together allow internal users to access the web because if a host from inside the organization’s network initiates an HTTP session with a web server outside the network, then the source address will be this—the address of the host in the organization’s network. The destination address will be that of the web server, which is an address outside this organization’s network.

The protocol will be TCP, and the source port will be greater than 1023. The destination port will be that of HTTP, that is, 80, and all such packets are allowed. And when the web server responds, the source address will be outside this list of prefixes, and the destination address will be of this form. The protocol will be TCP, and the source port will be 80 (that is, HTTP), and the destination port will be greater than 1023. And the ACK bit will always be one because the session is initiated by the host inside the network.

So, whenever the web server responds, the ACK flag is always 1. So, all such packets are allowed; hence, internal users are allowed to surf the web. But external sources are not allowed to establish a TCP connection with the web server inside the organization because in that case, when we look at the first rule, then the source address is 222.22/16. The web server, in this case, is inside the organization, so the source port will be 80, and that does not match this rule or this rule, so hence, it will be denied. So, external sources are not allowed to establish a TCP connection with the web server inside the organization. Similarly, the third and fourth rules together allow DNS packets to enter and leave the organization's network.

So, let's look at the third and fourth rules. If some host wants to find out the IP address corresponding to a particular external host name, so in that case, some host from inside the network, which will have an IP address of this form, will send a packet to a DNS server, which is outside the network, and hence it has an IP address of this form. It will be a UDP kind of packet. The source port will be greater than 1023, and since it is sent to a DNS server, which runs at port number 53, the destination port will be 53. So, this DNS query will be passed by the firewall, and the response of the DNS server, which will be of this form, that is also passed by the firewall.

So, these third and fourth rules will together allow DNS packets to enter and leave the organization's network. In particular, they will allow users from inside the network to query DNS servers and obtain the IP addresses corresponding to web page names. In summary, this access control list blocks all traffic except web traffic initiated from within the organization and DNS traffic. So, this is an example of how an access control list can be used to enforce certain policies in a traditional packet filter. So, in summary, we discussed different types of firewalls.

We saw that there are three types of firewalls: traditional packet filters, stateful packet filters, and application gateways. We discussed traditional packet filters in detail in this lecture. In the next lecture, we will discuss stateful packet filters and application gateways. Thank you.