A loadable kernel module which implements a mini firewall

What is Firewall?

A Firewall is a network security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules .

A Firewall typically establishes a barrier between a trusted internal and an untrusted external network such as internet.

In this project we have implemented a mini packet-filter firewall as a loadable kernal module.

Packet Filter

Packet filters act by inspecting packets transferred between computers. When a packet does not match the packet filter's set of filtering rules, the packet filter either drops (silently discards) the packet, or rejects the packet (discards it and generate an Internet Control Message Protocol notification for the sender) else it is allowed to pass. Packets may be filtered by source and destination network addresses, protocol, source and destination port numbers.

Netfilter

Netfilter is a framework provided by the Linux kernel that allows various networking-related operations to be implemented in the form of customized handlers. Netfilter offers various functions and operations for packet filtering, network address translation, and port translation, which provide the functionality required for directing packets through a network, as well as for providing ability to prohibit packets from reaching sensitive locations within a computer network.

How does it work?

The Linux net filter is a framework in the kernel that allows modules to observe and modify packets as they pass through the protocol stack. Kernel services or modules register custom hooks/filters by identifying both protocol family (e.g., PF INET) and by the point in packet processing (e.g., NF INET LOCAL IN) at which the filter is to be invoked. Only kernel components or installable modules can directly register hooks, never application code.

The main module, which is our firewall kernel module, is the netfilter hook function. Inside init_module(), we fill in the nf_hook _ops structs and then formally register the hook with nf_register_net_hook(). In cleanup(), we unregister the hook with nf_unregister_net_hook(). There are five points in the network stack where netfilter stacks are there: NF_INET_PRE_ROUTING,

NF_INET_LOCAL _IN, NF_INET_FORWARD, NF_INET_POST_ROUTING and NF_INET_LOCAL _OUT. We register our hook function at the pre-routing point which is the first hookpoint where all incoming packets reach.

Our Mini Firewall module thus inspects every packet at the entry point and based on the rules stated, makes the decision of whether to accept the packet or to drop it.

It blocks incoming traffic by dropping the specified packets using the return value NF DROP and accepting other packets by returning NF ACCEPT.

The Mini Firewall is going to filter packets based on 3 factors: IP address,TCP/ICMP protocol and/or port.

Commands That Had To Be Used

- make : builds and maintains programs from source code
- sudo insmod mf_km.ko : loads a module into the kernel, where it remains until you remove it (with rmmod) or shut down your system
- ./mf -h : to view our module information
- dmesg : to see the kernel buffer
- sudo rmmod main : removes the module
- make clean : deletes all the already compiled object files
- ping and ssh : connects to a server or remote computer

THANK YOU