Bryan Diaz

Final Project Design: Network Packet Filtering Module

List of All Actors

Our Linux-based system will ping several internet-connected servers using the program developed. It will also ping another computer on the same network to both send and request/receive data packets.

Information/Control Flow

The Linux-based system being used will be both sending and receiving data packets to the other actors involved. It will also regulate this data flow based on a blacklist of IP addresses. It may also restrict any and all data flow into, out of, or both from the system.

Supported Use Cases

The netfilter should allow its users to perform multiple software packet filtering tasks. It should give the user the option to block or allow all incoming traffic. Similarly, it should provide the option to block or allow all outgoing traffic. The netfilter should also be more specific than these all-or-nothing functions. It should block or allow all traffic (incoming & outgoing) to a specific IP address. As IP addresses are blocked, it should keep a record of these blocked IPs and provide the option to display that list of IP addresses. Finally, it should also keep a record of the number of packets that have been blocked (both incoming and outgoing) for each blocked IP address.

Expected Fault Cases

Successfully sending to or receiving from a blacklisted IP address. Any traffic into/out of the system when either incoming or outgoing traffic should be blocked. Being unable to receive data from an IP that is not blacklisted. Being unable to send data to an IP that is not blacklisted.

Programming Language

Done in Linux-based C programming.

External Libraries

Will use several files from the /include/linux/ (https://elixir.bootlin.com/linux/v4.16/source/inclu de/linux). Files to be used: /include/Linux/skbuff.h, /include/Linux/ip.h, /include/Linux/netfilter.h, and /include/Linux/udp.h.

Testing Procedure

Multiple pinging tests will occur. Try to receive ping while all incoming traffic blocked. Try to receive ping while all incoming traffic unblocked. Try to send ping while all outgoing traffic blocked. Try to send ping while all outgoing traffic unblocked. Try to send/receive ping while all traffic blocked. Try to send/receive ping while all traffic unblocked. Try to send ping to a blacklisted IP address. Try to receive ping from a blacklisted IP address. Display blacklist while blacklist is empty. Display blacklist while all traffic blocked. Display blacklist while blacklist contains some IP addresses. Display number of packets blocked to/from IP address on blacklist. Display number of packets blocked to/from IP address not on blacklist.

Design Assumptions

This design assumes several things. First, it assumes that the system using the program has internet access. Next, it assumes that the system can reach the desired IP address at any given time. It also assumes that the desired IP address it not blocking it. It assumes that the desired functions described above will be the only final functions that the program will perform. Finally, it assumes that the program will be used only on Linux based systems.

Work Breakdown

Seeing as I was unable to find a partner in time for the deadline (and with the approval of Professor Dubey) I will be working on this project by myself.