Quiz 2

CSE-644 INTERNET SECURITY

DR. SYED SHAZLI

2/2/2023

Anthony Redamonti
SYRACUSE UNIVERSITY

1(40): What is a firewall? What are the types of the firewall according to filtering directions? What type is the default ufw according to its filtering direction (without setting up anything)? Can ufw be bidirectional? How? If you'd like to use ufw to block traffic, what is the first thing to do in a VM (assume it is already enabled)? What is the reason?

Answer: A firewall is a computer that acts as a guard between the network and the internet. Its goal is to monitor the traffic in either direction (going out or coming in). The firewall will inspect each packet to determine whether to accept the packet and forward it or drop it. If a packet is not accepted (dropped), it is because the packet violates the firewall policy. The packet can also be modified before being forwarded.

The types of firewalls according to the filter direction: Packet Filter (Stateless) Firewall, Stateful Firewall, and Application Firewall. The ingress is the direction coming from the internet into the network. The egress traffic consists of packets originating inside the network traveling to the internet. The Application Firewall can be used to block packets in both directions (ingress and egress).

Another type of firewall is the Network Address Translation (NAT) firewall. It operates inside routers to allow only traffic from the internet that was requested from a machine inside the private network.

Iptables and Uncomplicated Firewall (UFW): There are several tables that make up the Linux Firewall. These are the filter, NAT, and mangle. The filter is for filtering packets based on their header information, and NAT and mangle are used to modify packets. It also uses a stateful firewall mechanism called “connection tracking” which sets up a policy based on the connection tracking status.

By default, UFW blocks ingress traffic (originating from the internet attempting to access the network).

Yes, the UFW can be bidirectional. The UFW resides inside the front-end of the Iptables. It can prevent the client machine from reaching another machine over the internet. For instance, to block a user from telnetting to any external machine, use “sudo ufw deny out from Client_IP to any port 23”.

The UFW commands can be used to set up bidirectional rules that apply firewall policies.
To use UFW to block traffic in a VM, write a command: “sudo deny in from 192.168.0.7 to 192.168.0.5 port 80”. (Webservers run on port 80).

2(20): How many types of firewalls do we have according to its functions? Where does a firewall run: Kernel or user space?

Answer: The firewall types are below.

- **Packet Filter (Stateless) Firewall**: It is the simplest type of firewall that drops a packet based on a set of rules (packet type, IP address, etc.). It analyzes the header of the IP/TCP/UDP layer to determine whether to filter the packet.
- **Stateful Firewall**: Uses the state of the connection to determine whether to forward a packet. If the packet does not belong to an existing connection, it will be dropped.

Stateful firewalls are more powerful than the stateless type because they allow the configuration of a more accurate firewall policy. They also analyze the header of the IP/TCP/UDP layer to determine whether to filter the packet.

- **Application Firewall**: Analyzes the application layer to determine whether to forward the packet. Example: a webproxy that analyzes the website being requested in the packet.

A firewall runs in the kernel space. It allows the programmer to use “hooks” that function as access points for the programmer to insert their firewall policies for the system.

3(40): What module are you using to write the firewall in the lab, is it in the user space or kernel space? How does it work? (not in code detail, in name detail, such as what do you call those connection functions?) Where does this module run, and how does it achieve its purpose, etc.)

Answer: The module is called the Netfilter. It allows the programmer to “hook” or insert filter code into the operating system. The hooks will define policies that the OS will apply to incoming and outgoing packet traffic (Firewall Policies).

One important function is “setUpFilter” which is used to set up a Firewall policy in the kernel and register the policy on the hook. Another important function is the removeFilter method, which removes (unhooks) a policy from the kernel. To inject the module into the kernel, use “module_init(setUpFilter),” and to remove the module from the kernel, use “module_exit(removeFilter).” The programmer must define a function inside of the nf_register_hook() function. The function will define the behavior of the filter (filter implementation) and should analyze the packet header.