Workshop in Information Security – HW 1

As instructed in the assignment, I wrote a device driver that passes/blocks packets in the network as they get to the machine, according to their type – their source/destination, and prints (to the kernel) an indicative message.

In order to implement the restrictions, I used 3 of the nf_hook_ops structs we were shown in class:

1. nf_local_in – Represents packets that are sent to the FW machine.
2. nf_local_out – Represents packets that are sent from the FW machine.
3. nf_forward – Represents packets that are sent between host1 and host2.

Inside the initialization function I filled all the data of each struct as follows:

- The hook is the hook function that would be explained later.
- The hooknum is the inspection/hook point – for example for nf_local_in it would be NF_INET_LOCAL_IN.
- The pf would be PF_INET for IPV4.
- The priority would be NF_IP_PRI_FIRST – highest.

After filling in all the data I registered each of the structs.

The hook functions:

We already know what would be (in this case) the packet's fate as it "got to the function" according to the hooknum that represents the inspection/hook point.

All we have to do is print and return NF_ACCEPT/NF_DROP accordingly.

In the cleanup function I unregister the hooks.

I got some guidance in how to fill up the data in the structs in the initialization function from http://www.roman10.net/2011/07/22/how-to-filter-network-packets-using-netfilterpart-1-netfilter-hooks/.