```c
/* ANRC RHKI */
/* Lab19: Networking Lab */
#include <linux/module.h>
#include <linux/kernel.h>
#include <linux/init.h>

#include <linux/netfilter.h>
#include <linux/netfilter_ipv4.h>

#define DRIVER_AUTHOR "ANRC"
#define DRIVER_DESC   "Lab19"

MODULE_LICENSE("GPL");           // Get rid of taint message by declaring code as GPL.

/*  Or with defines, like this: */
MODULE_AUTHOR(DRIVER_AUTHOR);    // Who wrote this module?
MODULE_DESCRIPTION(DRIVER_DESC); // What does this module do?

/* IP Hooks */
/* After promisc drops, checksum checks. */
#define NF_IP_PRE_ROUTING       0
/* If the packet is destined for this box. */
#define NF_IP_LOCAL_IN          1
/* If the packet is destined for another interface. */
#define NF_IP_FORWARD           2
/* Packets coming from a local process. */
#define NF_IP_LOCAL_OUT         3
/* Packets about to hit the wire. */
#define NF_IP_POST_ROUTING      4
#define NF_IP_NUMHOOKS          5

static struct nf_hook_ops netfilter_ops_in; /* NF_IP_PRE_ROUTING */
static struct nf_hook_ops netfilter_ops_out; /* NF_IP_POST_ROUTING */

/* Function prototype in <linux/netfilter> */
unsigned int main_hook(unsigned int hooknum,
                  struct sk_buff **skb,
                  const struct net_device *in,
                  const struct net_device *out,
                  int (*okfn)(struct sk_buff*))
{

        /* Drop ALL Packets */
}

int init(void)
{
        printk(KERN_INFO "init_module() called\n");


        /* build netfilter_ops_in struct */
        /* build netfilter_ops_out struct */

        /* register hooks */

    return 0;
}

void cleanup(void)
{

        /* unregister hooks */

        printk(KERN_ALERT "Unloading netdead ...\n");
}

module_init(init);
```

```
module_exit(cleanup);
```