#include <linux/module.h>

#include <linux/kernel.h>

#include <linux/netfilter.h>

#include <linux/netfilter\_ipv4.h>

#include <linux/ip.h>

#include <linux/tcp.h>

static struct nf\_hook\_ops netfilter\_ops;

unsigned int packet\_filter(void \*priv, struct sk\_buff \*skb, const struct nf\_hook\_state \*state) {

struct iphdr \*ip\_header = ip\_hdr(skb);

struct tcphdr \*tcp\_header;

if (!ip\_header) return NF\_ACCEPT;

if (ip\_header->protocol == IPPROTO\_TCP) {

tcp\_header = tcp\_hdr(skb);

if (ntohs(tcp\_header->dest) == 4444) {

printk(KERN\_INFO "Intercepted C2 connection on port 4444\n");

return NF\_DROP;

}

}

return NF\_ACCEPT;

}

static int \_\_init netfilter\_init(void) {

netfilter\_ops.hook = packet\_filter;

netfilter\_ops.pf = PF\_INET;

netfilter\_ops.hooknum = NF\_INET\_PRE\_ROUTING;

netfilter\_ops.priority = NF\_IP\_PRI\_FIRST;

nf\_register\_hook(&netfilter\_ops);

return 0;

}

static void \_\_exit netfilter\_exit(void) {

nf\_unregister\_hook(&netfilter\_ops);

}

module\_init(netfilter\_init);

module\_exit(netfilter\_exit);

MODULE\_LICENSE("GPL");