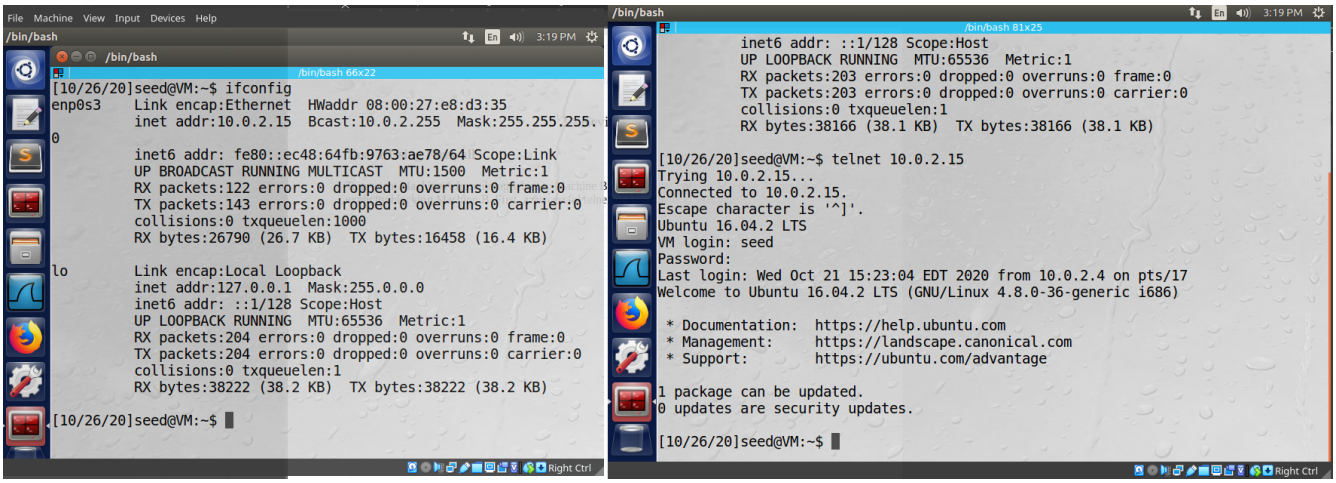
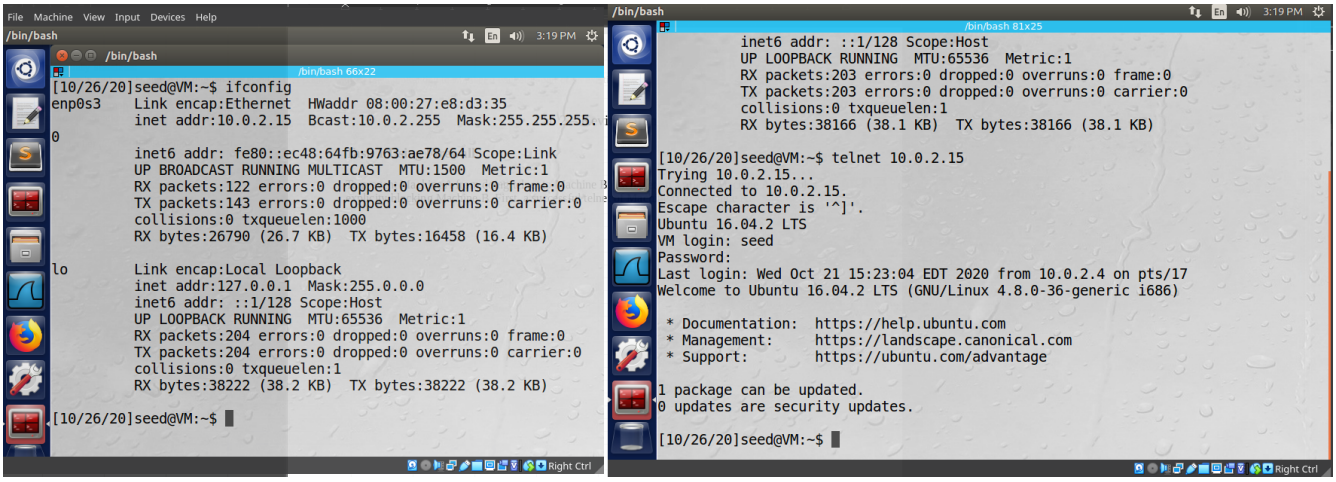
Lab 2 – Kevin Martin

Task 1 Using Firewall

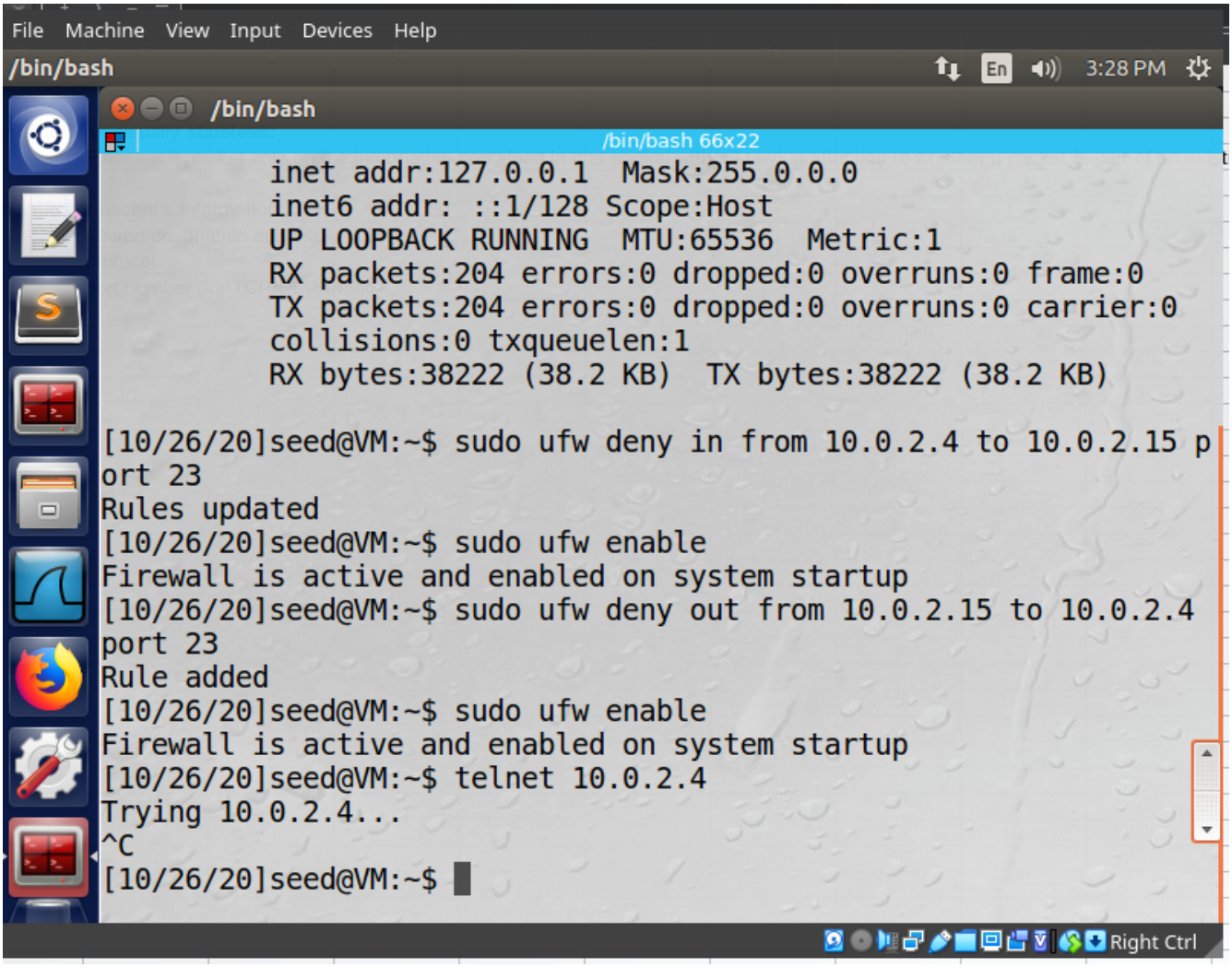
To prevent Machine A from doing telnet to Machine B, I will set up a rule to Machine A's iptables explicity blocking Machine B. First, a successful telnet of Machine B to Machine A:



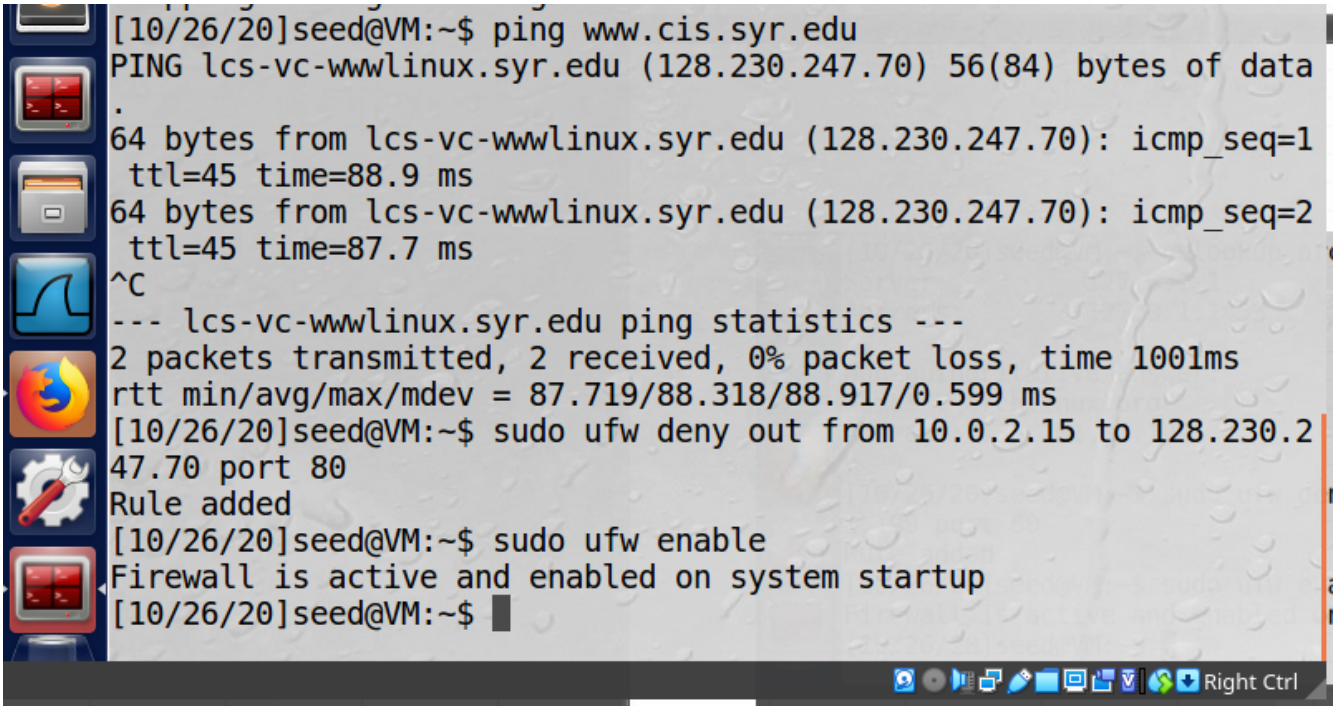
Note how Machine A is 10.2.15 (left VM), and machine B (10.0.2.4, right VM) was able to successfully telnet in. Now, the rule will be added to Machine A, and Machine B will try to establish a connection:

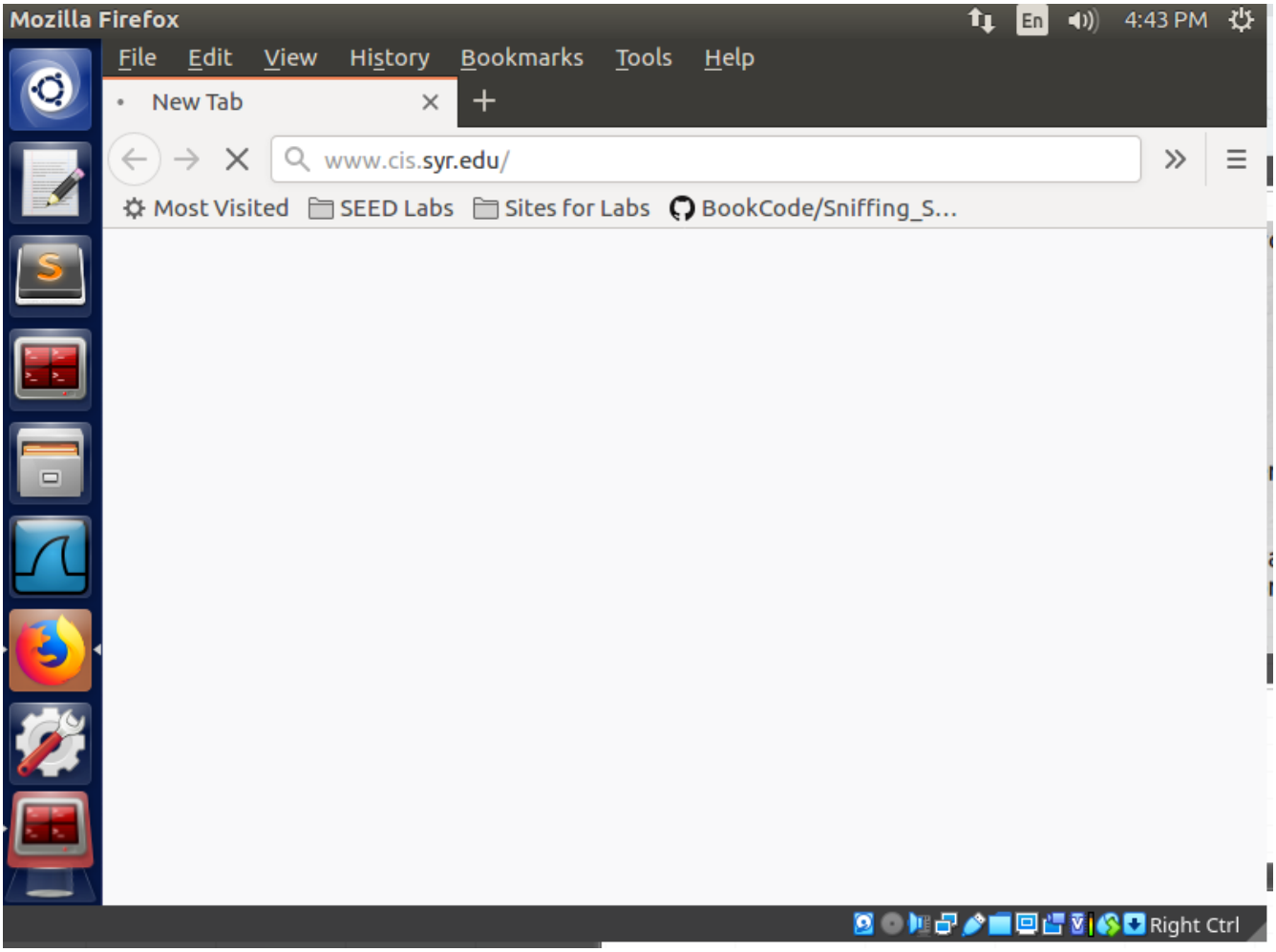


The rule from Machine A successfully prevented the same connection from Machine B. Now Machine A will prevent itself from being able to telnet into Machine B:



Finally, I will prevent A from visiting an external website (Syracuse server). First, a successful ping to the page to get the IP address, then a new rule added, and finally the successful block:

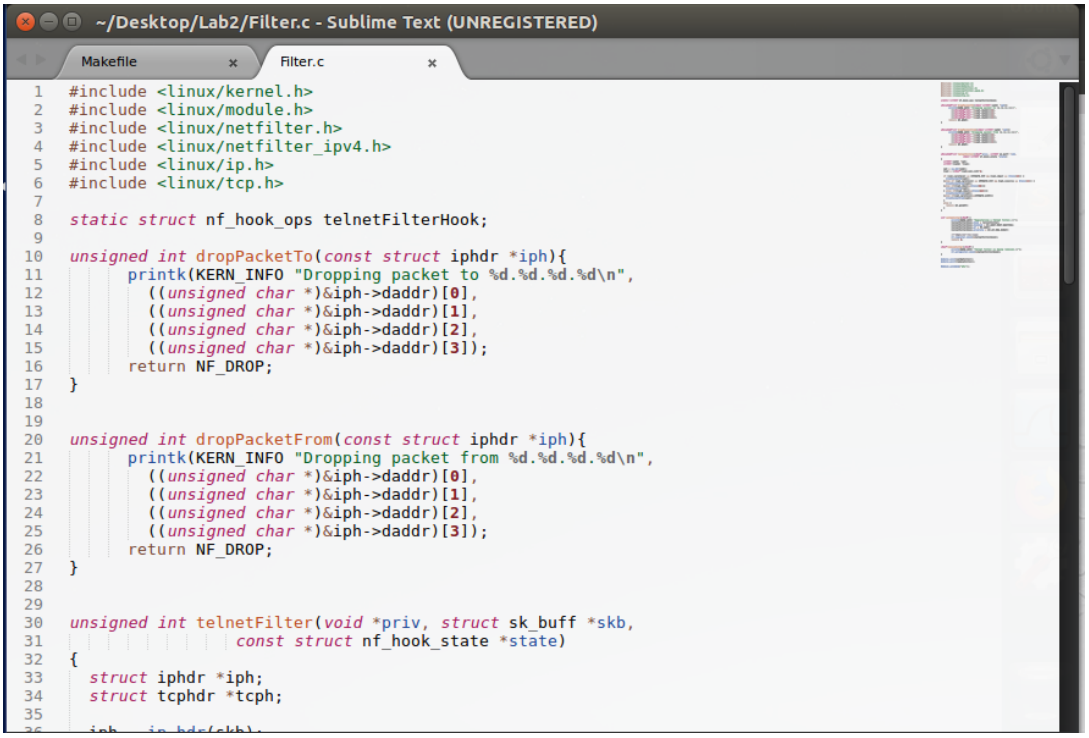

[10/26/20]seed@VM:~$ ping www.cis.syr.edu
PING lcs-vc-wwwlinux.syr.edu (128.230.247.70) 56(84) bytes of data
.
64 bytes from lcs-vc-wwwlinux.syr.edu (128.230.247.70): icmp_seq=1
ttl=45 time=88.9 ms
64 bytes from lcs-vc-wwwlinux.syr.edu (128.230.247.70): icmp_seq=2
ttl=45 time=87.7 ms
^C
--- lcs-vc-wwwlinux.syr.edu ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 87.719/88.318/88.917/0.599 ms
[10/26/20]seed@VM:~$ sudo ufw deny out from 10.0.2.15 to 128.230.2
47.70 port 80
Rule added
[10/26/20]seed@VM:~$ sudo ufw enable
Firewall is active and enabled on system startup
[10/26/20]seed@VM:~$
Right Ctrl



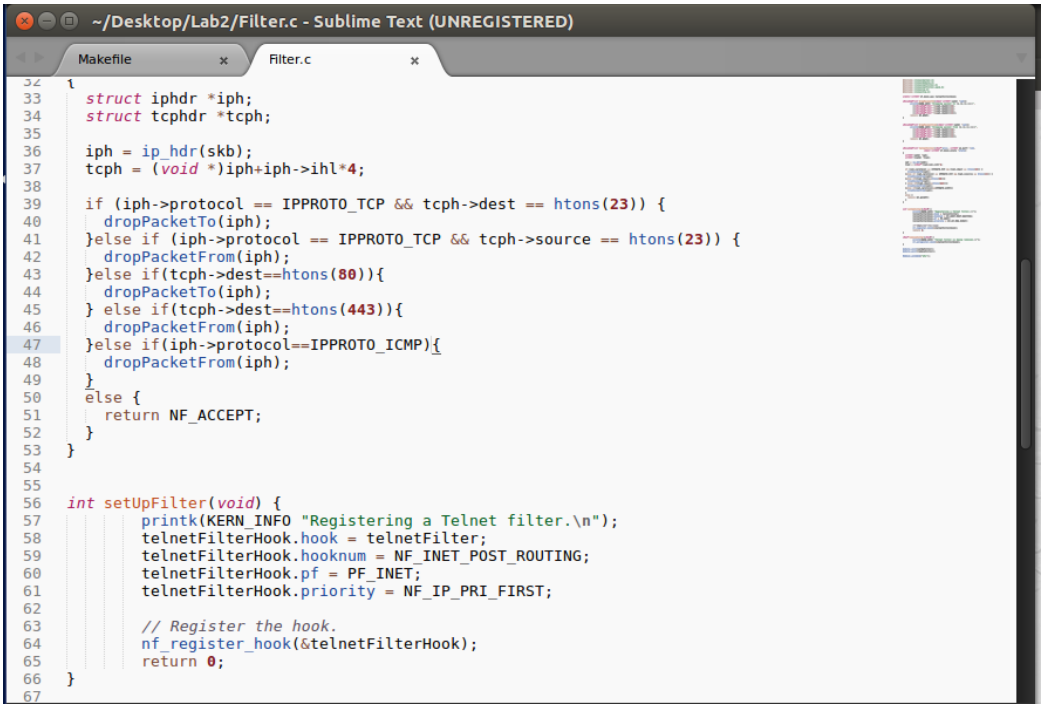
The site will never load now, it just sits here. Note that after this, I ran “sudo ufw reset” to remove the previously set firewalls. I will most likely forget they are there otherwise.

Task 2: Implementing a Simple Firewall

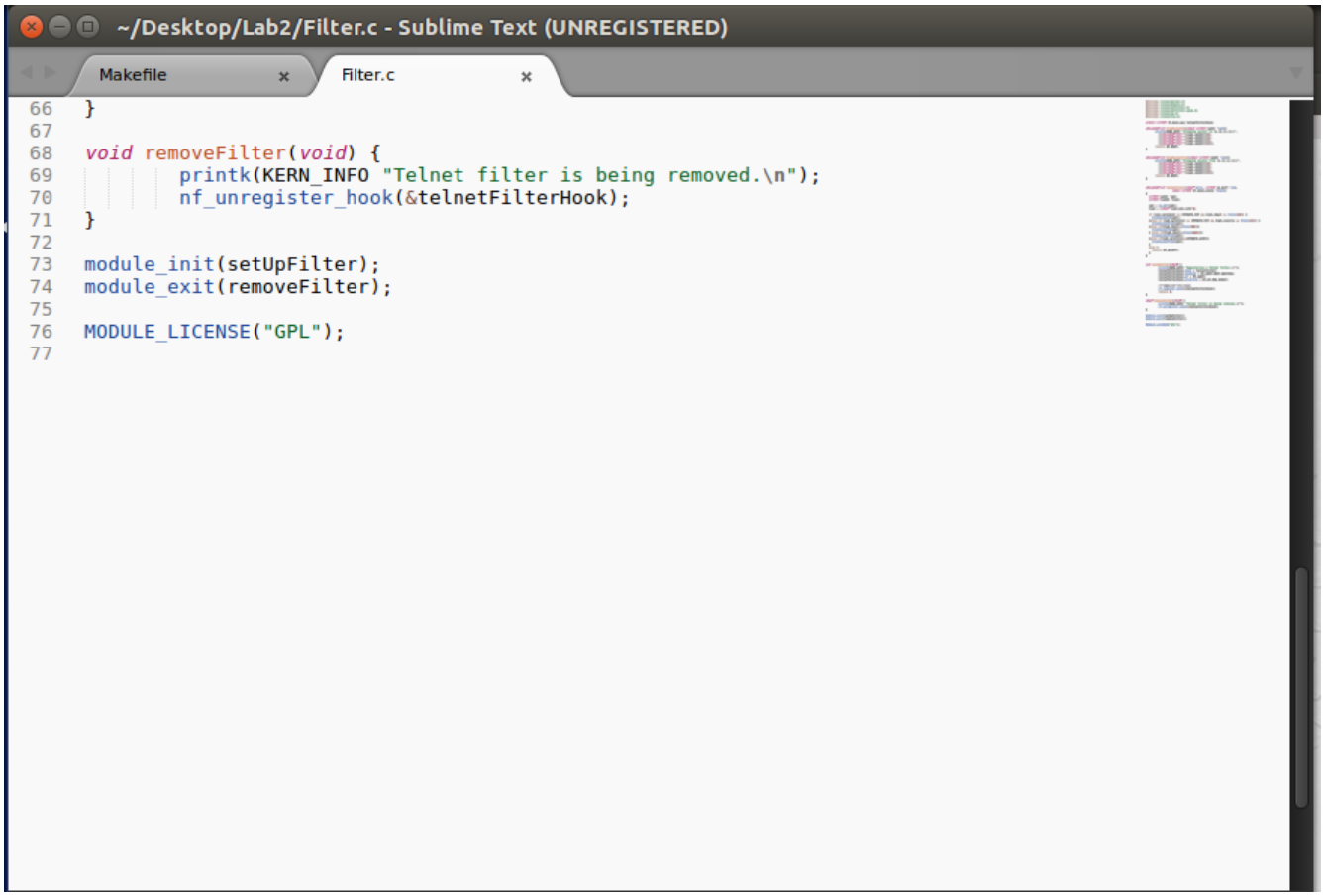
To implement a firewall, I’m using code from the book and modifying to capture more scenarios. It is compiled with “make” and I’m using the standard Makefile provided. The code:




```c
#include <linux/kernel.h>
#include <linux/module.h>
#include <linux/netfilter.h>
#include <linux/netfilter_ipv4.h>
#include <linux/ip.h>
#include <linux/tcp.h>

static struct nf_hook_ops telnetFilterHook;

unsigned int dropPacketTo(const struct iphdr *iph){
        printk(KERN_INFO "Dropping packet to %d.%d.%d.%d\n",
          ((unsigned char *)&iph->daddr)[0],
          ((unsigned char *)&iph->daddr)[1],
          ((unsigned char *)&iph->daddr)[2],
          ((unsigned char *)&iph->daddr)[3]);
        return NF_DROP;
}


unsigned int dropPacketFrom(const struct iphdr *iph){
        printk(KERN_INFO "Dropping packet from %d.%d.%d.%d\n",
          ((unsigned char *)&iph->daddr)[0],
          ((unsigned char *)&iph->daddr)[1],
          ((unsigned char *)&iph->daddr)[2],
          ((unsigned char *)&iph->daddr)[3]);
        return NF_DROP;
}


unsigned int telnetFilter(void *priv, struct sk_buff *skb,
                          const struct nf_hook_state *state)
{
  struct iphdr *iph;
  struct tcphdr *tcph;

  iph = ip_hdr(skb);
```



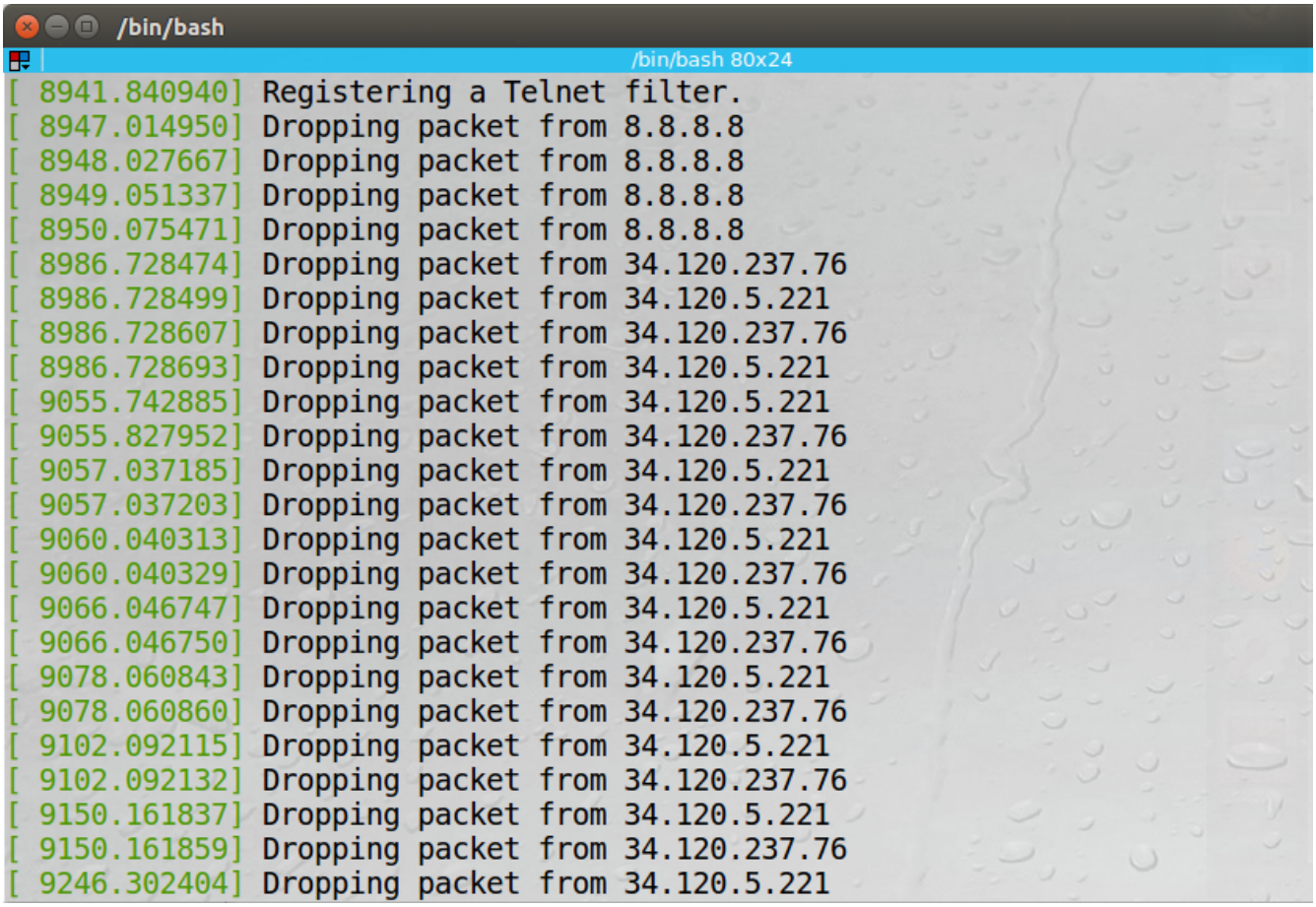

```c
{
  struct iphdr *iph;
  struct tcphdr *tcph;

  iph = ip_hdr(skb);
  tcph = (void *)iph+iph->ihl*4;

  if (iph->protocol == IPPROTO_TCP && tcph->dest == htons(23)) {
    dropPacketTo(iph);
  }else if (iph->protocol == IPPROTO_TCP && tcph->source == htons(23)) {
    dropPacketFrom(iph);
  }else if(tcph->dest==htons(80)){
    dropPacketTo(iph);
  } else if(tcph->dest==htons(443)){
    dropPacketFrom(iph);
  }else if(iph->protocol==IPPROTO_ICMP){
    dropPacketFrom(iph);
  }
  else {
    return NF_ACCEPT;
  }
}


int setUpFilter(void) {
        printk(KERN_INFO "Registering a Telnet filter.\n");
        telnetFilterHook.hook = telnetFilter;
        telnetFilterHook.hooknum = NF_INET_POST_ROUTING;
        telnetFilterHook.pf = PF_INET;
        telnetFilterHook.priority = NF_IP_PRI_FIRST;

        // Register the hook.
        nf_register_hook(&telnetFilterHook);
        return 0;
}
```



To run, the code is invoked with “insmod” and stopped with “rmmod”. I first ping 8.8.8.8 and note no issues. Then run the code, and now the ping requests do not go through:

```
[10/27/20]seed@VM:~/.../Lab2$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=11.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=114 time=9.49 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 9.498/10.744/11.990/1.246 ms
[10/27/20]seed@VM:~/.../Lab2$ sudo insmod Filter.ko
[10/27/20]seed@VM:~/.../Lab2$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3060ms

[10/27/20]seed@VM:~/.../Lab2$
```

Finally, the dmesg output showing the packets being dropped. Note that after pinging 8.8.8.8 I tried to load google.com, and you can see the packets there were also dropped (the webpage timed out in the browser):



Task 3a: Evading Egress Filtering

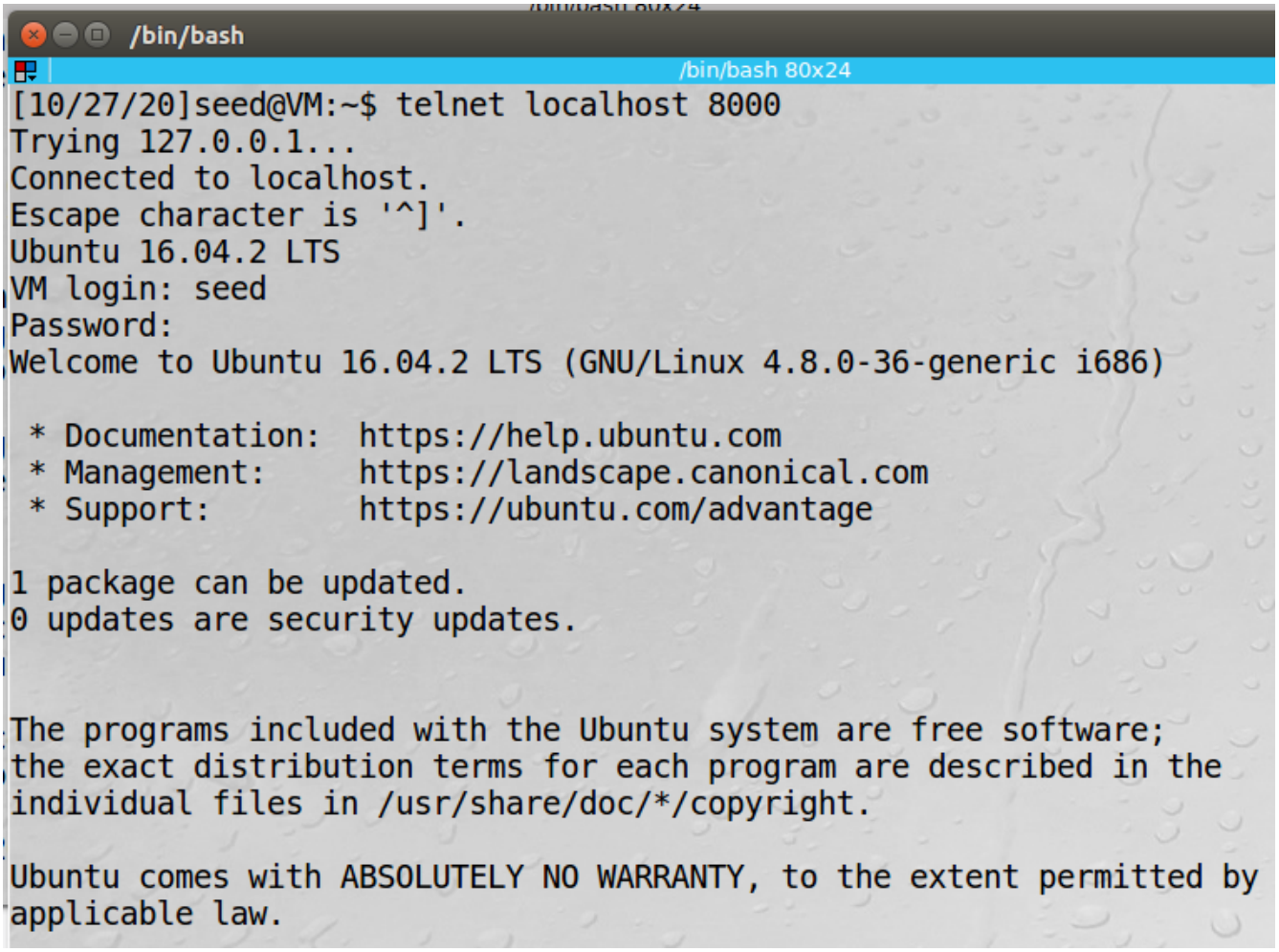
I am going to add the firewall on VM A. Note that I removed the rules at the end of Task 1. First, I reinstate them, then show that they are active (VM A is 10.0.2.15) and that VM A cannot telnet to VM B (10.0.2.4):

```
[10/27/20]seed@VM:~/.../Lab2$ sudo ufw deny out from 10.0.2.15 to any port 23
Rule added
[10/27/20]seed@VM:~/.../Lab2$ sudo ufw deny out from 10.0.2.15 to 31.13.65.36
Rule added
[10/27/20]seed@VM:~/.../Lab2$ sudo ufw enable
Firewall is active and enabled on system startup
[10/27/20]seed@VM:~/.../Lab2$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 23                         DENY OUT    10.0.2.15                  (out)
[ 2] 31.13.65.36                DENY OUT    10.0.2.15                  (out)

[10/27/20]seed@VM:~/.../Lab2$ telnet 10.0.2.4
Trying 10.0.2.4...
^C
[10/27/20]seed@VM:~/.../Lab2$
```

Next, I build an SSH Tunnel from VM A to VM C (10.0.2.5):

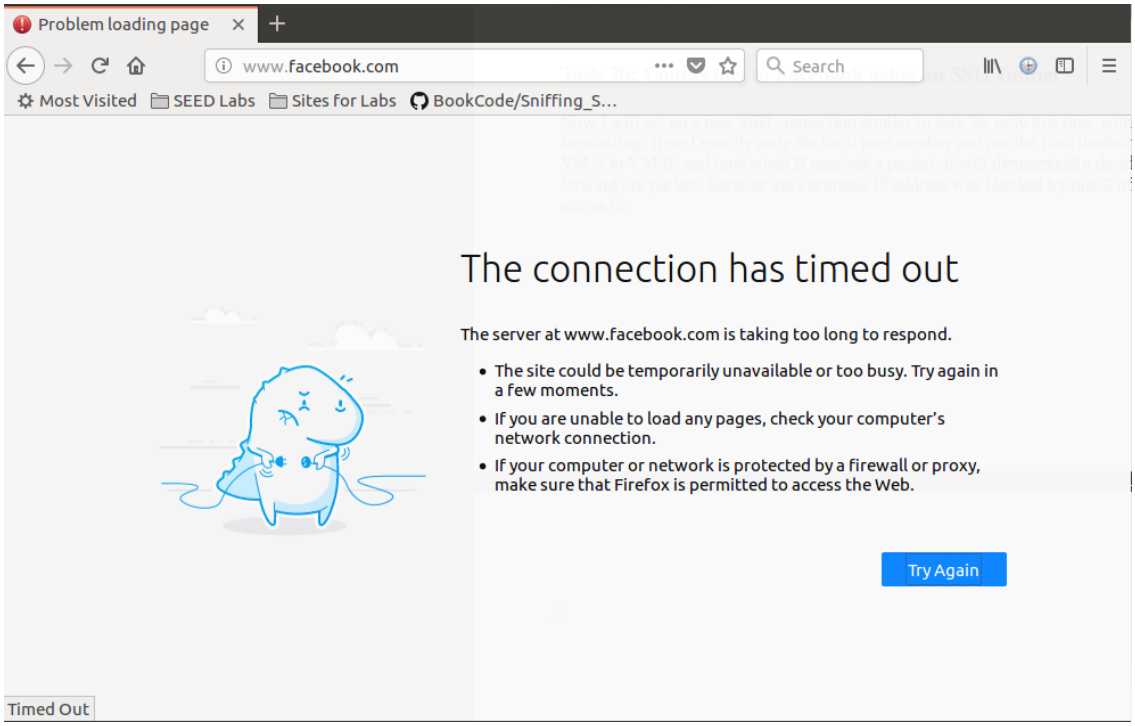
```
[10/27/20]seed@VM:~/.../Lab2$ telnet 10.0.2.4
Trying 10.0.2.4...
^C
[10/27/20]seed@VM:~/.../Lab2$ ssh -L 8000:10.0.2.5:23 seed@10.0.2.4
The authenticity of host '10.0.2.4 (10.0.2.4)' can't be established.
ECDSA key fingerprint is SHA256:p1zAio6c1bI+8HDp5xa+eKRi561aFDaPE1/xq1eYzCI.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.4' (ECDSA) to the list of known hosts.
seed@10.0.2.4's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

1 package can be updated.
0 updates are security updates.


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

[10/27/20]seed@VM:~$
```

Finally, I open a second terminal window on VM A and telnet into machine B (via the localhost, provided by C):



Task 3b: Connecting to Facebook using an SSH Tunnel

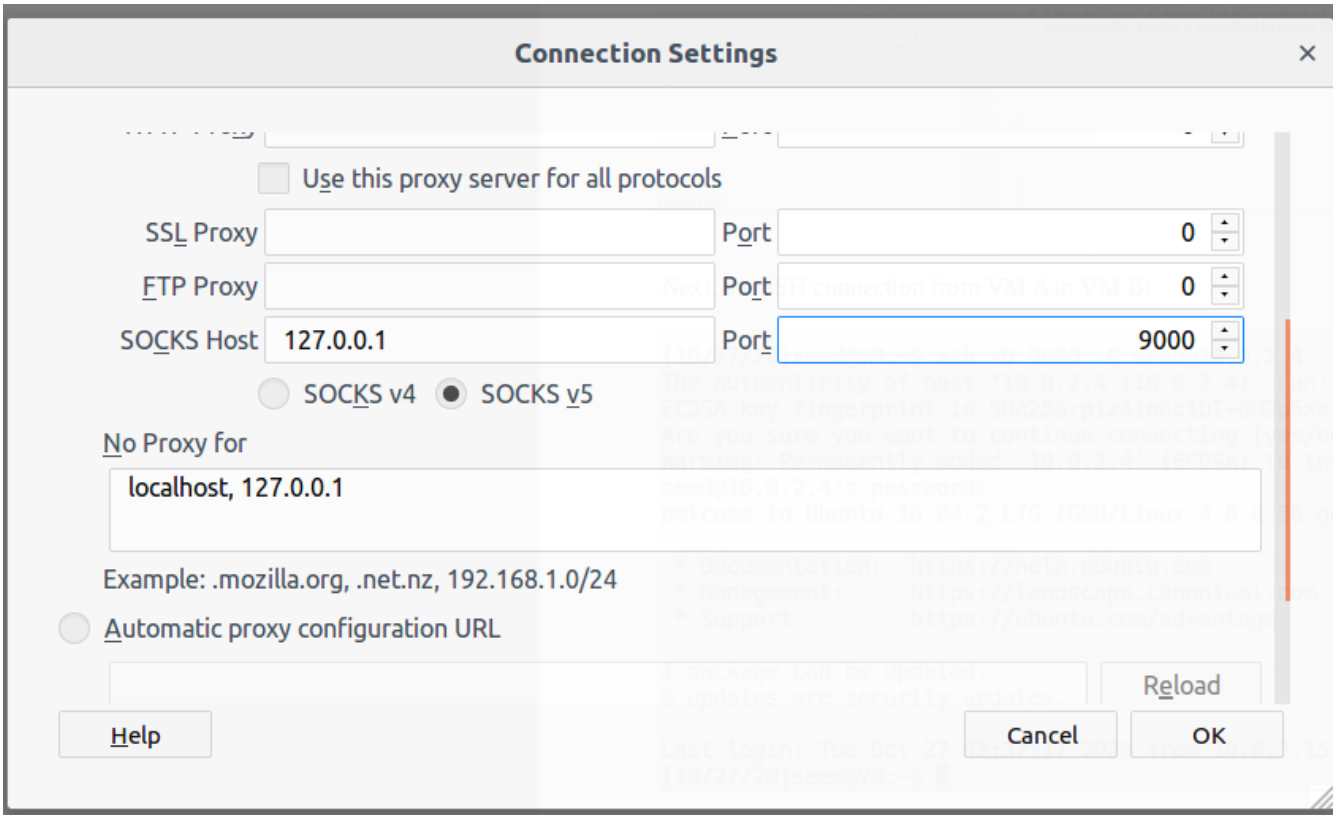
Now I will set up a new SSH connection similar to task 3a, only this time with dynmaic port forwarding. Here I specify only the local port number and not the final destination. I will SSH from VM A to VM B, and now when B receives a packet, it will dynnamically decide where it should forward the packet. Because the Facebook IP address was blocked by rule 2 of my firewall, we cannot access it:



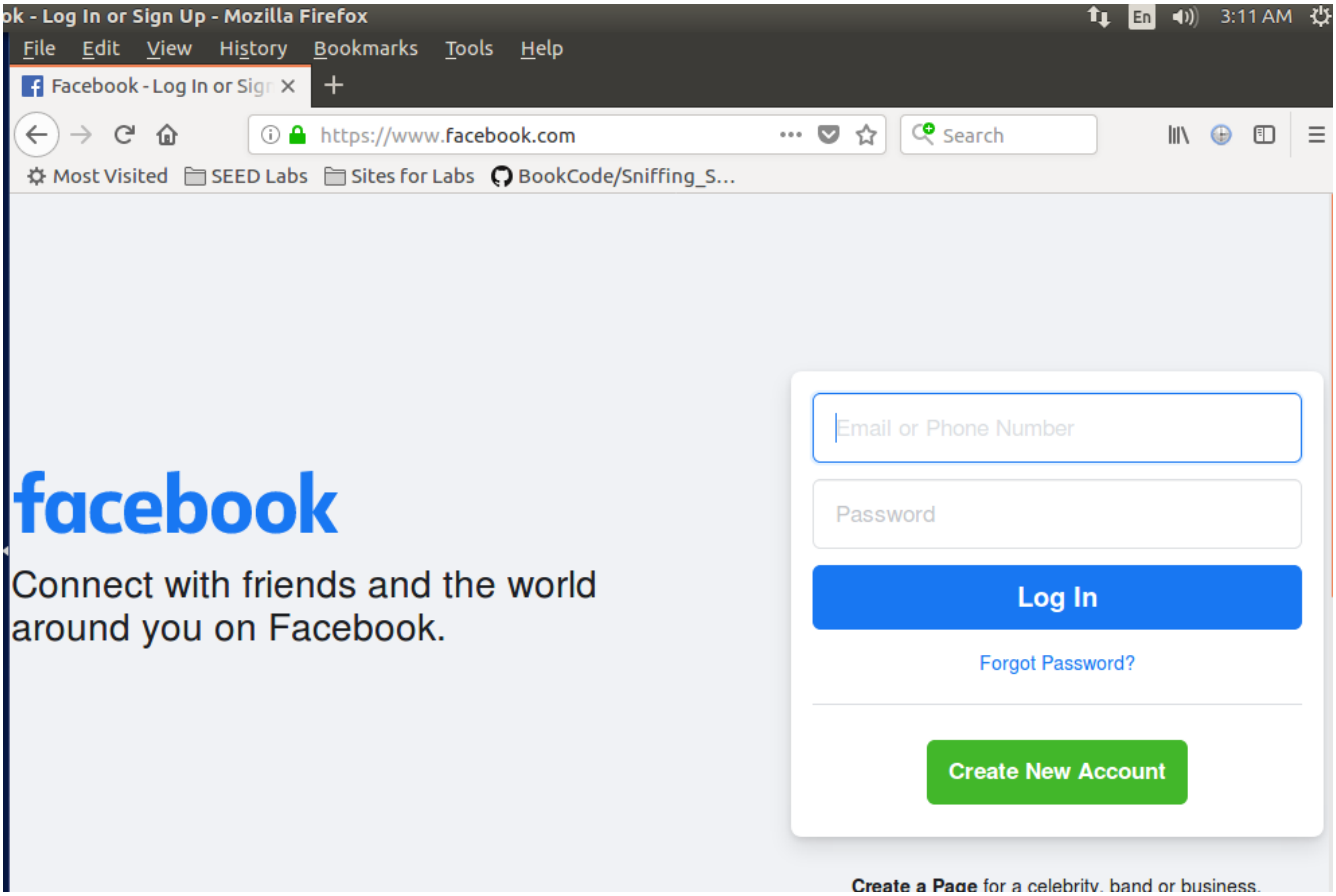
Next, the SSH connection from VM A to VM B:

```
[10/27/20]seed@VM:~$ ssh -D 9000 -C seed@10.0.2.4
The authenticity of host '10.0.2.4 (10.0.2.4)' can't be established.
ECDSA key fingerprint is SHA256:p1zAio6c1bI+8HDp5xa+eKRi561aFDaPE1/xq1eYzCI.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.4' (ECDSA) to the list of known hosts.
seed@10.0.2.4's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

1 package can be updated.
0 updates are security updates.

Last login: Tue Oct 27 02:37:17 2020 from 10.0.2.15
[10/27/20]seed@VM:~$
```

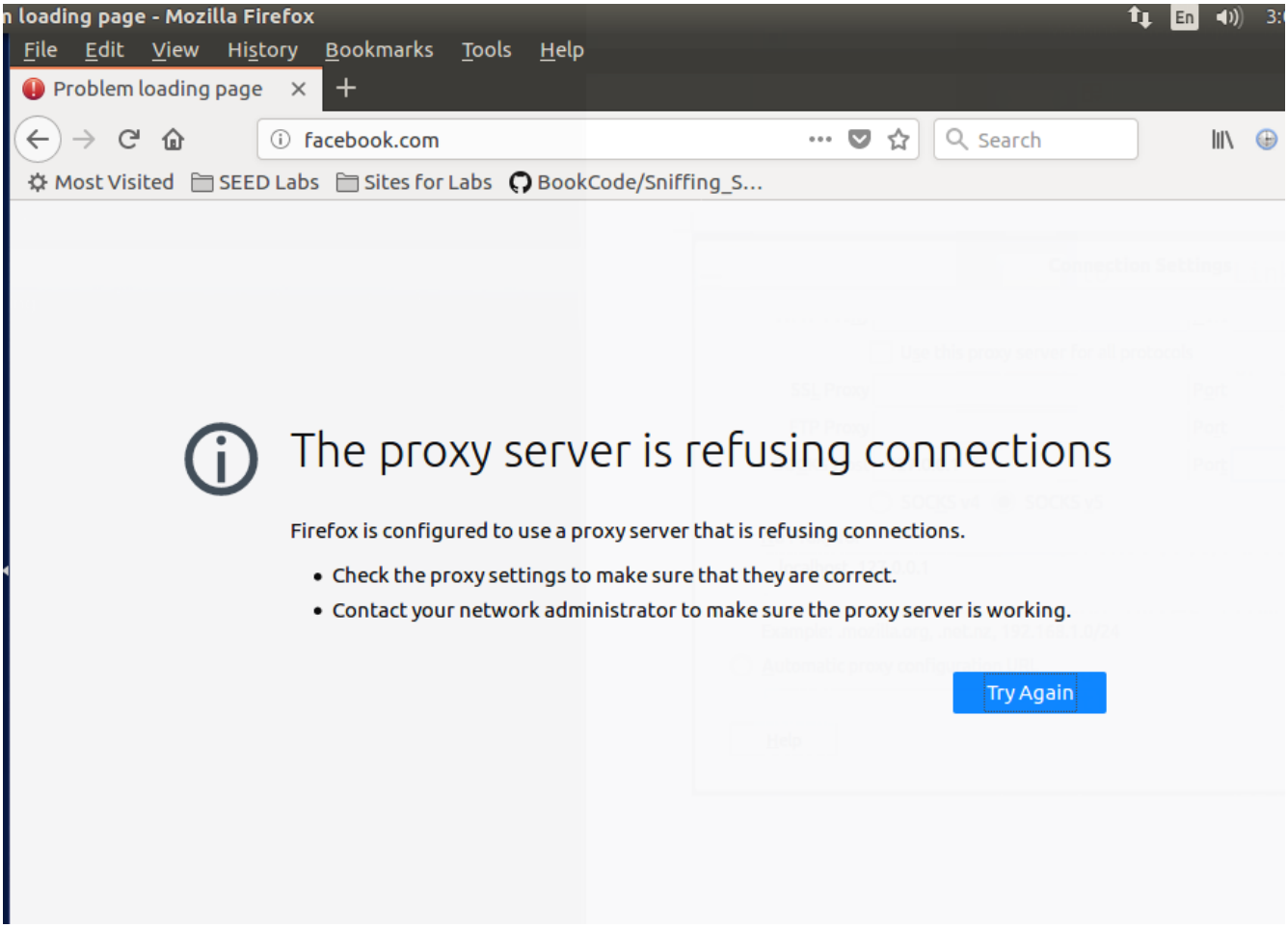
Once the connection is established, we force Firefox to connec to localhost:9000, which allows traffic from VM A to be transferred through the SSH tunnel to VM B. Firefox proxy:



After establishing the tunnel:


ok - Log In or Sign Up - Mozilla Firefox
En
3:11 AM
File Edit View History Bookmarks Tools Help
Facebook - Log In or Sig
https://www.facebook.com
Search
Most Visited
SEED Labs
Sites for Labs
BookCode/Sniffing_S...
facebook
Connect with friends and the world around you on Facebook.
Email or Phone Number
Password
Log In
Forgot Password?
Create New Account
Create a Page for a celebrity, band or business.

And after breaking the tunnel, we lose access again:



To recap, after adding the firewall to VM A, it could not longer telnet or access Facebook (as identified by a specific IP address). Creating an SSH tunnel to a third VM, which did not have a firewall restriction, allowed access to the target second VM. As for Facebook, dynamically forwarding the ports to the second VM (which again, did not have any restrictions) allowed access to the server via an SSH tunnel.

Task 4: Evading Ingress Filtering

Now I will set up a firewall blocking VM B from accessing VM A through port 80 and 22. See firewall from VM A:

```
[10/27/20]seed@VM:~$ sudo ufw deny in from 10.0.2.4 to 10.0.2.15 port 80
Rules updated
[10/27/20]seed@VM:~$ sudo ufw deny in from 10.0.2.4 to 10.0.2.15 port 22
Rules updated
[10/27/20]seed@VM:~$ sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
[10/27/20]seed@VM:~$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 10.0.2.15 80               DENY IN     10.0.2.4
[ 2] 10.0.2.15 22               DENY IN     10.0.2.4

[10/27/20]seed@VM:~$
```

Note the existing SSH connection from the previous section was interrupted. To confirm, a new (failed) ping attempt from VM B to VM A:

```
[10/27/20]seed@VM:~$ ping 10.0.2.15
PING 10.0.2.15 (10.0.2.15) 56(84) bytes of data.
^C
--- 10.0.2.15 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3075ms

[10/27/20]seed@VM:~$
```

To bypass, we will use a reverse SSH tunnel from VM A to VM B, and then try the SSH again. First, the reverse SSH to an unallocated port 19999:

```
[10/27/20]seed@VM:~$ ssh -R 19999:localhost:22 seed@10.0.2.4
seed@10.0.2.4's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

1 package can be updated.
0 updates are security updates.

Last login: Tue Oct 27 03:36:33 2020 from 10.0.2.4
[10/27/20]seed@VM:~$
```

And finally, from VM B, a successful SSH back into VM A using the local host and specified port:

```
[10/27/20]seed@VM:~$ ssh localhost -p 19999
The authenticity of host '[localhost]:19999 ([127.0.0.1]:19999)' can't be establi
shed.
ECDSA key fingerprint is SHA256:p1zAio6c1bI+8HDp5xa+eKRi561aFDaPE1/xq1eYzCI.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[localhost]:19999' (ECDSA) to the list of known hosts
.
seed@localhost's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

1 package can be updated.
0 updates are security updates.

Last login: Tue Oct 27 03:48:01 2020 from 10.0.2.4
[10/27/20]seed@VM:~$
```