www.spkaa.com
Ph: 888-310-4540

SPK and Associates
900 E Hamilton Ave, Ste.100
Campbell, CA 95008

Security Basics with IPTables

IPTables is the firewall configuration software used on Redhat. The default configuration for iptables allows everything. This isn't a good idea for Internet accessible hosts. You can see the current rules on a Linux host as follows. In this case, it's the default rules which allow everything:

```
iptables --list

Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

If a host is deployed on the internet, it's suggested to deny all incoming and allow only ports that you wish opened. Edit the default configuration file: /etc/sysconfig/iptables. Enter these lines into the configuration. Denying all traffic into the host is as simple as:

```
# Configure default policies (-P).   This is what's used if no
# other rule is matched.  By default, don't allow anything into
# the host and drop all packets.   Anything originating at the
# host can be allowed out.
#
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT
```

Now, we need to add rules for desired traffic. Some basics:

```
#
# Permit packets into the host from existing outbound connections
#
iptables -A INPUT -i eth0 -m state --state ESTABLISHED,RELATED -j
ACCEPT
```

Open a hole in the firewall for services that we want available to outside users. These rules allow access to SMTP, HTTP, HTTPS, and SSH respectively:

```
iptables -A INPUT -p tcp -s 0/0 -d 69.36.111.222/32 --
destination-port 25 --syn -j ACCEPT
```

www.spkaa.com
Ph: 888-310-4540

SPK and Associates
900 E Hamilton Ave, Ste.100
Campbell, CA 95008

```
iptables -A INPUT -p tcp -s 0/0 -d 0/0 --destination-port 80 --
syn -j ACCEPT
iptables -A INPUT -p tcp -s 0/0 -d 0/0 --destination-port 443 --
syn -j ACCEPT
iptables -A INPUT -p tcp -s 0/0 -d 0/0 --destination-port 22 --
syn -j ACCEPT
```

Finally, there comes a time when someone on the internet is being unfriendly. Here's how to deny all SSH traffic from IP: 69.36.22.11 to your host's IP: 69.36.99.10.

```
iptables -A INPUT -p tcp -s 69.36.22.11 /32 -d 69.36.99.10 /32 --destination-port 22 --syn
-j DROP
```

The above example by itself is not very useful. SSH shouldn't be open on the internet. Best practice is to deploy a VPN service to securely authenticate into the LAN. If you need to open SSH, it's better to limit to specific IPs. This allows SSH into 69.36.99.10 from any other machine on the 69.36.99.1 subnet (69.36.99.1-69.36.99.254)

```
iptables -A INPUT -p tcp -s 69.36.99.1 /24 -d 69.36.99.10 /32 --destination-port 22 --syn –
j ACCEPT
```

If you wish an open the SSH port, I'd recommend the following minimal steps to secure the port:

- Allow only one or two non-root users access to SSH. This is done in /etc/ssh/sshd_config with the: AllowUsers command. Use an uncommon login name – don't use john, admin, super, etc.
- Monitor the failed ssh login log regularly using "lastb". Ideally, this is done by a cron or background job. Setup maximum failed attempt policies that will trigger some action.
- Once triggered, deny future activity from that IP with a new iptables rule. Automate this with a script that's run with cron every few minutes to stop them quickly.

There are many options available with iptables to handle any level of complexity. IPTables can be used as your sole firewall. However, I prefer to use it in conjunction with firewall s from Cisco or Juniper. The above example is does not have a complete set of firewall rules. If you wish to use IPTables as your only firewall, I recommend using one of the tools available on the internet to configure iptables such as:
http://easyfwgen.morizot.net/gen/

As with all security work, do not take a configuration from any internet website without reviewing it and understanding what each line allows or denies. Whenever you encounter a serious threat, work with CERT: http://www.cert.org/.