FUN WITH IPTABLES - IPTABLES SECURITY BASICS

I like iptables, it's free, it runs on linux, it's easier to use than ipchains, and you can do some pretty funky things with it. But like all things linux, it only plays nice if you ask it properly. A badly configured iptables firewall will be no better than a badly configured any other firewall.

For those of you starting out, there are some user friendly options, the most well known probably being the iptables module of the Webmin project. Webmin is a web based administration tool for linux with modules for just about anything, including iptables. The module actually gives you a rules gui much like the checkpoint firewall's point and click screen. And yes, there are various other options, just google for a list.

Personally, I like the command line approach, so thats what I'm going to be focusing on for some general security settings able to used on almost any iptables firewall (and as always, any suggestions here are taken used at your own risk) ;

1. Ensure that forwarding is enabled
   echo 1 > /proc/sys/net/ipv4/ip_forward

2. Help prevent IP Spoofing
   echo 1 > /proc/sys/net/ipv4/conf/default/rp_filter

3. Ensure the required kernel modules are loaded
   insmod ip_tables
   insmod iptable_filter
   insmod ip_conntrack
   insmod ip_state
   insmod iptable_nat
   insmod ip_conntrack_ftp
   insmod ip_nat_ftp
   insmod ipt_LOG
   insmod ipt_mangle
   --there are more modules these are just the most common--

4. Always flush the old ruleset
   iptables -F
   iptables -t nat -F
   iptables -t mangle -F

5. Always delete any custom chains
   iptables -X
   iptables -t nat -X
   iptables -t mangle -X

6. Setup the default policies for your firewall
   iptables -t nat -P POSTROUTING DROP
   iptables -P FORWARD DROP
   iptables -t nat -P OUTPUT DROP
   iptables -P OUTPUT DROP
   iptables -P INPUT DROP
   iptables -t nat -P PREROUTING DROP
   --make sure any other chains you use are also set to a default of DROP--

7. Allow for local interface to work unimpeded
   iptables -A INPUT -i lo -j ACCEPT
   iptables -A OUTPUT -o lo -j ACCEPT

8. Setup some source address sanity checks for your external interface
   iptables -t nat -A PREROUTING -s 192.168.0.0/24 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 192.168.1.0/24 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 192.168.2.0/24 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 192.168.3.0/24 -i$EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 192.168.254.0/24 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 199.196.196.0/24 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 0.0.0.0/8 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 255.255.255.255 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 127.0.0.0/8 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 224.0.0.0/4 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 240.0.0.0/5 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 169.254.0.0/16 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 192.0.0.0/24 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 192.0.34.0/24 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 10.0.0.0/8 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 172.16.0.0/12 -i $EXT_INT -j DROP
   iptables -t nat -A PREROUTING -s 192.168.0.0/16 -i $EXT_INT -j DROP
   --If you are running services on your firewall change the above table to INPUT--
   --If you are using private ip addresses on your network then check these rules so nothing breaks--

9. Setup some packet flag sanity checks
   iptables -A FORWARD -p tcp --tcp-flags ACK,FIN FIN -j DROP
   iptables -A FORWARD -p tcp --tcp-flags ACK,PSH PSH -j DROP
   iptables -A FORWARD -p tcp --tcp-flags ACK,URG URG -j DROP
   iptables -A FORWARD -p tcp --tcp-flags FIN,RST FIN,RST -j DROP
   iptables -A FORWARD -p tcp --tcp-flags SYN,FIN SYN,FIN -j DROP
   iptables -A FORWARD -p tcp --tcp-flags SYN,RST SYN,RST -j DROP
   iptables -A FORWARD -p tcp --tcp-flags ALL ALL -j DROP
   iptables -A FORWARD -p tcp --tcp-flags ALL NONE -j DROP
   iptables -A FORWARD -p tcp --tcp-flags ALL FIN,PSH,URG -j DROP
   iptables -A FORWARD -p tcp --tcp-flags ALL SYN,FIN,PSH,URG -j DROP
   iptables -A FORWARD -p tcp --tcp-flags ALL SYN,RST,ACK,FIN,URG -j DROP
   --If you are running services on your firewall change the above table to INPUT--

10. Protect against SYN floods
    iptables -t nat -A PREROUTING -p tcp --syn -m limit --limit 10/second --limit-burst 20 -j ACCEPT
    --the figures used here could well differ from situation to situation, please test--

11. Setup the stateful packet inspection rules for each chain
    iptables -A INPUT -m state --state established,related -j ACCEPT
    iptables -A FORWARD -m state --state established,related -j ACCEPT
    iptables -A OUTPUT -m state --state established,related -j ACCEPT
    iptables -t nat -A PREROUTING -m state --state established,related -j ACCEPT
    iptables -t nat -A POSTROUTING -m state --state established,related -j ACCEPT
    iptables -t nat -A OUTPUT -m state --state established,related -j ACCEPT
    --make sure any other chains you use are also setup for packet inspection--

And thats it for this first installment, hopefully some of these tips will help make your iptables firewall a little

more secure. And being more secure is never a bad thing