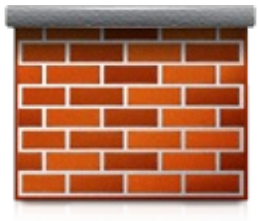
Table Of Contents

nixCraft: Linux Tips, Hacks, Tutorials, And Ideas In Blog Format
http://www.cyberciti.biz/

Configure Linux As Bastion Host

Posted by Vivek Gite <vivek@nixcraft.com>

What is bastion host? How do I configure bastion host under Linux? How do I create a firewall for a bastion host under any Linux distribution?



[1]

A bastion host is high risk host on your network. It can be a dedicated Linux running netfilter or OpenBSD box running PF or a Cisco PIX device. This device is designed to protect your network from external threats.

[2]

```
The Internet
  \\
    \\
Bastion Host
      //
     //
Your Network
```

Usually bastion host placed outside your corporate firewall or in the DMZ itself.

```
The Internet
    \\
+---------------+
| Bastion Host  |  <--- Outside firewall
+---------------+
           //
+---------------+
|    DMZ        | <---- Inside firewall
+---------------+
   \\
     ||
+---------------+
| LAN1 LAN2     |
+---------------+
```

In most cases it has access from the Internet or untrusted parties / computers. In some case a bastion host can be a:

1. Web server
2. DNS Server
3. FTP Server
4. Proxy Server
5. Honey pots
6. Email Server etc

WARNING! These examples needs a dedicated Linux box.You MUST know how to install programs on your computer, how to navigate file system, list open ports, configure iptables, write a firewall script and other advanced admin tasks.

Bastion Host and Screened Subnet

Bastion host adds an extra layer of security to the screened host architecture. It isolate your internal network form the Internet. The end result is that your bastion host is the primary target of Internet attacks. If someone beaks into the

bastion host, your internal hosts are safe as the bastion host is isolated by the perimeter network. The bastion host firewall configuration has more security. Usually following is done on bastion hosts:

1. Firewall works in close all ports and opened required port mode only.
2. Intrusion detection system (IDS/IPS) such as as snort.
3. Security settings to avoid Denial of Service (DoS), spoofing, and flood attacks.
4. Undergo regular auditing.
5. Runs upto date software.
6. May run special kernel security [3] patches.
7. All user accounts are locked down except admin account.
8. Encryption used for logging (ssh) or disk storage.
9. Remove all end user software and other network servers such as Apache, MySQL and so on.
10. TCP/IP stack tuned [4] for network traffic including network buffers.
11. /etc/sysctl.conf customized to improve server security

Usually, the bastion host does act as proxy server. It allows and denies connection as created by your security policy.

How Do I Build Linux As a Bastion Host?

A Linux based bastion host can be build using the following steps:

1. Grab Debian / CentOS CD or your favorite Linux distribution.
2. Install minimum operating system. Avoid installing desktop software or other apps such as MySQL, Apache and other software.
3. Reboot the server.
4. Patch [5] server.
5. Install grsecurity kernel [3] patch and reboot the system.
6. Install additional software such as snort IDS and configure it.
7. Install Advanced Intrusion Detection Environment [6] (AIDE) Software.
8. Make sure all security patches are installed.
9. Disable all network [7] services except ssh.
10. Disable all other daemons [8].
11. Network tuning vis sysctl.conf
12. Configure firewall (see sample script below).
13. Remove centralized authentication such as LDAP.
14. Remove as many utilities and system configuration tools as is practical for your setup. No need to have gcc compilers and other unwanted tools. Use rpm/yum [9] and dpkg command [10] to list all packages.
15. Logging of all security related events and turn on auditing [11].
16. Write protect all log files and only allow them in append only mode using chattr command [12] (e.g. chattr +a /var/log/messages or chattr +i /etc/shadow).
17. Encrypt all database passwords including file systems if possible.
18. Create system recovery DVD or tape [13].

Above all are generic and recommended steps to configure bastion host.

Sample Linux Iptables Bastion Host Rules

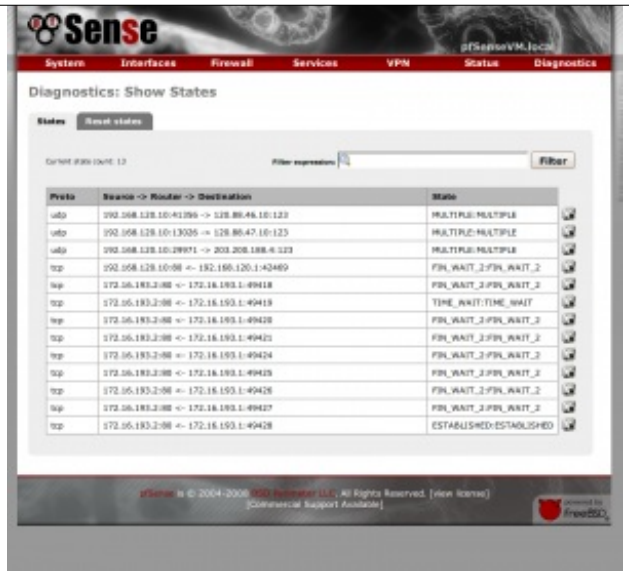
You need at least two network interface one is connected to the Internet via public IP and another private to your Lan.

```
#!/bin/sh
# The bastion host firewall for bhost.lan.nixcraft.net.in
# The bastion host is also:
# (a) Mail server to relay mail to postfix.lan.nixcraft.net.in
# (b) DNS server send zone trasfer to ns1.lan.nixcraft.net.in  and ns2.lan.nixcraft.net.in
# (c) Allow incoming ssh / http / https to bhost.lan.nixcraft.net.in from LAN SUBNET sotha
#     we can manage bhost.lan.nixcraft.net.in via ssh, and read snort stats via ACID web i
#------------------------------------------------------------------------------------------
### Set vars ###
IPT=/sbin/iptables
SYSCTL=/sbin/sysctl
### Set interfaces ###
EXT_IF="eth0"    # The Internet
LAN_IF="eth1"  # Lan
LOOP_BACK="lo"

### Block RFC 1918 private address space range ###
### Block reserved Class D and E IP  ###
```

```
### Block the unallocated address range et all ###
SPOOFDIP="127.0.0.0/8 192.168.0.0/16 172.16.0.0/12 10.0.0.0/8 169.254.0.0/16 0.0.0.0/8 240.

### Set Lan Subnet ###
LAN_SUBNET="192.169.1.0/24"

### Set DNS Server IPs  ###
NS1_SERVER_IP=192.168.1.130
NS2_SERVER_IP=192.168.1.131

### Set Postfix Server IP ###
SMTP_SERVER_IP=192.168.1.132

### Set port numbers ###
SSH_PORT=22
SMTP_PORT=25
HTTP_PORT=80
HTTPS_PORT=443
DNS_PORT=53

### Clean out old fw ###
$IPT -F
$IPT -X
$IPT -t nat -F
$IPT -t nat -X
$IPT -t mangle -F
$IPT -t mangle -X
$IPT -P INPUT ACCEPT
$IPT -P OUTPUT ACCEPT
$IPT -P FORWARD ACCEPT

### Turn on SYN flooding protection ###
$SYSCTL -w net/ipv4/tcp_syncookies=1

### Block out eveything ###
$IPT -P INPUT DROP
$IPT -P OUTPUT DROP
$IPT -P FORWARD DROP

### Allow full access to loopback ###
$IPT -A INPUT -i ${LOOP_BACK} -j ACCEPT
$IPT -A OUTPUT -o ${LOOP_BACK} -j ACCEPT

### Block the RFC 1918 private address space ranges ###
for rfc in $SPOOFDIP
do
 $IPT -A INPUT -i ${EXT_IF} -s ${rfc} -j LOG --log-prefix " SPOOF DROP "
 $IPT -A INPUT -i ${EXT_IF} -s ${rfc} -j DROP
done

### Drop bad stuff ###
$IPT -A INPUT -p tcp --tcp-flags SYN,RST SYN,RST -j DROP
$IPT -A INPUT -p tcp --tcp-flags SYN,FIN,PSH SYN,FIN,PSH -j DROP
$IPT -A INPUT -p tcp --tcp-flags SYN,FIN,RST SYN,FIN,RST -j DROP
$IPT -A INPUT -p tcp --tcp-flags SYN,FIN,RST,PSH SYN,FIN,RST,PSH -j DROP
# FIN-Only
$IPT -A INPUT -p tcp --tcp-flags FIN FIN -j DROP
$IPT -A INPUT -p tcp --tcp-flags ALL ALL -j DROP
$IPT -A INPUT -p tcp --tcp-flags ALL SYN -j DROP
$IPT -A INPUT -p tcp --tcp-flags ALL FIN,URG,PSH -j DROP
$IPT -A INPUT -p tcp --tcp-flags ALL SYN,RST,ACK,FIN,URG -j DROP

# FIN
$IPT  -A INPUT -p tcp --tcp-flags FIN,ACK FIN -j DROP

# NULL packets
$IPT -A INPUT -p tcp --tcp-flags ALL NONE -j DROP

# XMAS
$IPT -A INPUT -p tcp --tcp-flags SYN,FIN SYN,FIN -j DROP
```

```
# Fragments
$IPT -A INPUT -f -j DROP

# sync
$IPT -A INPUT -p tcp ! --syn -m state --state NEW -j DROP

### Allows the bastion host to query remote DNS servers  ###
$IPT -A INPUT -i ${EXT_IF} -p udp --dport ${DNS_PORT} -m state --state  NEW,ESTABLISHED -j
$IPT -A INPUT -i ${EXT_IF} -p tcp --dport ${DNS_PORT} -m state --state  NEW,ESTABLISHED -j
$IPT -A OUTPUT -o ${EXT_IF} -p udp --sport ${DNS_PORT} -m state --state NEW,ESTABLISHED -j
$IPT -A OUTPUT -o ${EXT_IF} -p tcp --sport ${DNS_PORT} -m state --state NEW,ESTABLISHED -j

### Allow internal DNS i.e. zone trasfer between the bastion and 2 LAN ns1 & ns2 ###
$IPT -A INPUT -i ${EXT_IF} -p udp -s ${NS1_SERVER_IP} --dport ${DNS_PORT} -m state --state
$IPT -A INPUT -i ${EXT_IF} -p udp -s ${NS2_SERVER_IP} --dport ${DNS_PORT} -m state --state
$IPT -A INPUT -i ${EXT_IF} -p tcp -s ${NS1_SERVER_IP} --dport ${DNS_PORT} -m state --state
$IPT -A INPUT -i ${EXT_IF} -p tcp -s ${NS2_SERVER_IP} --dport ${DNS_PORT} -m state --state

### Allow outgoing DNS and Zone transfers btw the bastion host and two 2 LAN ns1 & ns2 ###
$IPT -A OUTPUT -o ${EXT_IF} -p udp -d ${NS1_SERVER_IP} --sport ${DNS_PORT} -m state --state
$IPT -A OUTPUT -o ${EXT_IF} -p udp -d ${NS2_SERVER_IP} --sport ${DNS_PORT} -m state --state
$IPT -A OUTPUT -o ${EXT_IF} -p tcp -d ${NS1_SERVER_IP} --sport ${DNS_PORT} -m state --state
$IPT -A OUTPUT -o ${EXT_IF} -p tcp -d ${NS2_SERVER_IP} --sport ${DNS_PORT} -m state --state

### Allow LAN workstation to get into the bastion host via SSH but no access from the Inte
$IPT -A INPUT -i ${LAN_IF} -p tcp -s ${LAN_SUBNET} --dport ${SSH_PORT} -m state --state NEW
$IPT -A OUTPUT -o ${LAN_IF} -p tcp -d ${LAN_SUBNET} --sport ${SSH_PORT} -m state --state ES

### Allow LAN workstation to get into the bastion host via HTTP to read SNORT stuff via wel
### Read ACID stats ###
$IPT -A INPUT -i ${LAN_IF} -p tcp -s ${LAN_SUBNET} --dport ${HTTP_PORT} -m state --state NE
$IPT -A OUTPUT -o ${LAN_IF} -p tcp -d ${LAN_SUBNET} --sport ${HTTP_PORT} -m state --state E
$IPT -A INPUT -i ${LAN_IF} -p tcp -s ${LAN_SUBNET} --dport ${HTTPS_PORT} -m state --state N
$IPT -A OUTPUT -o ${LAN_IF} -p tcp -d ${LAN_SUBNET} --sport ${HTTPS_PORT} -m state --state

### External SMTP Rules ###
$IPT -A INPUT -i ${EXT_IF} -p tcp --dport ${SMTP_PORT} -m state --state NEW,ESTABLISHED - j
$IPT -A OUTPUT -o ${EXT_IF} -p tcp --sport ${SMTP_PORT} -m state --state NEW,ESTABLISHED -j

### Internal SMTP Rules ###
$IPT -A INPUT -i ${LAN_IF} -p tcp -s ${SMTP_SERVER_IP} --sport ${SMTP_PORT} -m state --stat
$IPT -A OUTPUT -o ${LAN_IF} -p tcp -d ${SMTP_SERVER_IP} --dport ${SMTP_PORT} -m state --sta

### Add your other rules below ###

### End no editing below ###

### Log ###
$IPT -A INPUT -m state --state INVALID -j LOG --log-prefix " INVAID DROP "
$IPT -A INPUT -m state --state INVALID -j DROP

$IPT -A INPUT -i ${EXT_IF} -j LOG --log-prefix " INPUT DROP "
$IPT -A OUTPUT -o ${EXT_IF} -j LOG --log-prefix " OUTPUT DROP "
```

Above script is basic and can be modified as per your requirements. You can also use firewall distributions such as pFsense [14] or IPcop [15] to automate lost of stuff.



[16]

Fig.01: pFsense in Action (click to enlarge)

References:

1. Thinking about firewalls [17]
2. An overview of network Firewall [18]
3. Linux DMZ tutorial [19] using iptables
4. Snort [20] - A free lightweight network intrusion detection system for UNIX and Windows.
5. Refer your Linux distribution documentations to perform required steps.
6. man page sysctl, and iptables

http://www.vtcif.telstra.com.au/pub/docs/security/ThinkingFirewalls/ThinkingFirewalls.html

[18] network Firewall: **http://www.cas.mcmaster.ca/wiki/index.php/Network_firewall**

[19] Linux DMZ tutorial: **http://www.cyberciti.biz/faq/linux-demilitarized-zone-howto/**

[20] Snort: **http://www.snort.org/**
