Tűzfalak

Sorozatunk előző részében röviden áttekintettük, miként tehetjük biztonságosabbá a Világhálón való cirkálásunkat. Akinek ez nem lenne elég, az segítségül hívhatja a Linux tűzfalszolgáltatásait.

Újságunk hasábjain már többször részletesen kivesézték ezt a témakört, ám úgy gondoltuk, hasznos lehet, ha összegyűjtjük azokat a dolgokat, amelyek egy otthoni rendszer esetén is jól jöhetnek.

Kezdjük egy kis bevezetéssel a tűzfalak világába! Alapvetően két eltérő típusú tűzfalat különböztetünk meg: az úgynevezett állapotfüggő csomagszűréseseket és a proxyalapúakat. Az első kategóriába tartozóak gyorsak, de csak azzal foglalkoznak, hogy a rajtuk átmenő csomagok honnan hová mennek, illetve hogy a bennük lévő adatot milyen protokoll segítségével továbbították. A proxytűzfalak viszont már sokkal magasabb szinten dolgoznak, már a csomagok tartalmába is belelátnak, ezért kiválóan alkalmazhatók például vírusok kiszűrésére vagy a forgalom részletesebb ellenőrzésére. A proxyk ugyanakkor általában ügyféloldali támogatást is megkövetelnek, ami a csomagszűrőkről nem mondható el. A másik lényeges különbség, hogy a csomagszűrés általában az operációs rendszer szintjén történik (így van ez a Linux esetében is), míg a proxytűzfal egy külön alkalmazás képében fut. Hogy mikor melyik tűzfalat érdemes használni, az a pillanatnyi feladattól függ, de ahol összetettebb védelemre van szükség, ott mindkét módszert együttesen alkalmazzák. Nagy örömünkre maga a Linux-rendszermag is tartalmaz egy jól használható csomagszűrő tűzfalat.

Egy otthoni rendszer esetében mire tudjuk használni a csomagszűrő tűzfalat? Segítségével egyrészt letilthatjuk, hogy kívülről bármelyik kapunkat elérjék, így például keresztbe tehetünk a különböző trójai programoknak; másrészt kiszűrhetjük az úgynevezett Denial of Service- (a szolgáltatásmegtagadásos, röviden: DoS) támadásokat.

Egy DoS-támadás esetében a cél nem egy bizonyos jogosultság megszerzése, „csupán” a rendszer megakadályozása abban, hogy elláthassa feladatát. Ezt el lehet érni például azzal, hogy jó sok munkát adnak neki (értsd: túlterhelik), vagy valamilyen úton-módon lefagyasztják, súlyosabb esetben tönkreteszik. Ha tehát rendszerünkkel valami ilyesmi történne, meg kell állapítanunk, hogy DoS-támadás áldozatává váltunk.

Az a tapasztalat, hogy nincs atombiztos rendszer. Valószínűleg nincs olyan bikaerős gép vagy olyan hiperszuper tűzfal, ami egy jól megszervezett nemzetközi összefogáson alapuló DoS-támadásnak ellen tudna állni. Vélhetően ritkán fenyeget minket, átlagfelhasználókat ilyesmi. Nekünk esetleg néhány kisebb lélegzetű támadással kell számolnunk, amelyekkel szemben a Linux eléggé talpraesettnek bizonyult, de a csomagszűrő tűzfal segítségével fokozhatjuk védelmünket.

Az első csomagszűrő tűzfal a Linux-rendszermag 1.1-es változatában bukkant fel, ami az igazat megvallva egy, a BSD-s világból jól „összelopkodott” rendszer volt. Azóta ezt már többször, az alapjaitól kezdve újraírták, a ma forgalomban lévő 2.4-es rendszermagok erre a célra az úgynevezett NetFiltert használják, amely a 2.2-es rendszermag IP Chains módszerét hivatott felváltani.

A csomagszűrő tűzfalat az IP Tables, 2.2-es rendszermag esetében az `ipchains` parancs segítségével állítgathatjuk. Meg kell jegyeznünk, hogy a 2.4-es rendszermag esetében is van lehetőség az `ipchains` használatára (ha a magot úgy fordítottuk), de akkor csak azéra, egyszerre a kettőt nem használhatjuk!

Az *1.* és *2. listán* látható parancsfájlt lefuttatva minden kimenő forgalom engedélyezett lesz (azaz a gépünktől kifelé bármi elérhető), viszont az összes befelé jövő kérést tiltja (tehát a gépünk kívülről elérhetetlen lesz). Egy otthoni rendszer számára talán ez a legjobb beállítás.

Az első parancsfájl az `ipchains`-t, a második az `iptables`-t használja. Ha nem tudjuk pontosan, hogy Linuxunknak melyik lenne a megfelelő, akkor futtassuk le az egyiket, és ha egy rakás hibaüzenetet kapunk, valószínűleg a második lesz a használható.

Ezek után térjünk át második témánkra, az internet-megosztásra, amelyet szintén az `ipchains`, illetve `iptables` alkalmazásokkal állíthatunk be. A most bemutatott módszer akkor tehet jó szolgálatot, amikor a szolgáltatóktól csak egy IP-címet kapunk, de mi a Világhálót több gép számára is elérhetővé szeretnénk tenni. (Ez az úgynevezett álcázás azaz masquerading). A lényeg az, hogy mivel csak egyetlen IP-címmel rendelkezünk, a többi gépnek úgynevezett belső címeket osztunk ki. Ezek olyan IP-címek, amelyek a `10.x.x.x`-es, illetve a `192.168.x.x`-es tartományba esnek, és kifejezetten erre a célra vannak fenntartva. A Linuxszal felszerelt gép egyfajta átjáróként fog szolgálni a többi gép és a külvilág között, tehát egyszerre kapcsolódik az Internethez és a belső hálózatunkhoz. Ez azt jelenti, hogy egyaránt rendelkezik egy külső (valódi) és egy belső IP-címmel.

A belső hálózatban lévő gépek az összes kifelé küldendő csomagot a linuxos gépnek továbbítják. Ekkor egy kicsit ellentmondást nem tűrő lépés következik, ugyanis a csomag fejléce átírásra kerül: a forrás IP-címét a Linux kicseréli a saját külső IP-címére. Erre azért van szükség, mert egy belső IP-címet tartalmazó csomag nem kerülhet ki a Világhálóra, az ilyeneket egyetlen útválasztó vagy átjáró sem továbbíthatja. Amikor pedig megérkezik a válaszcsomag (például egy belső gépről lekért weboldal), a Linux a cél IP-címét cseréli ki a megfelelő gép belső IP-címére, és beengedi a belső hálózatra.

A bújtatás segítségével megoldható, hogy a hálózatunkban lévő összes gép úgy láthasson kifelé, mintha közvetlenül kapcsolódna az Internethez, viszont kívülről csak Linux-átjárónk látszik, mivel egyedül az rendelkezik külső IP-címmel. Ez nagyon jó dolog, mivel belső gépeink nem támadhatóak, másrészről azonban bosszantó lehet, ha például két gépről egyidejűleg szeretnénk NetMeeting-elni, ICQ-zni, illetve olyan alkalmazásokat futtatni, amelyek egyedi IP-cím meglétét követelik meg.

Nézzük, mire is van szükségünk a linuxos gépen! Először is kell egy hálózati kártya, amivel a belső hálózatra csatlakozunk. Ha ADSL-en vagy kábeltelevízión keresztüli internetelérésünk

1. lista Egy célszerű otthoni beállítás ipchainse

```
#!/bin/sh
IPCHAINS=/sbin/ipchains
# ha modemmel vagy ADSL-en kereszt l
# csatlakozunk, akkor az eth0-t cserØlj k ki
# ppp0-ra!
WAN_IFACE="eth0"
LOCAL_PORTS=
➥'cat /proc/sys/net/ipv4/ip_local_port_range
➥|cut -f1':\
➥'cat /proc/sys/net/ipv4/ip_local_port_range
➥|cut -f2'
ANYWHERE= 0/0
$IPCHAINS -F
$IPCHAINS -P forward DENY
$IPCHAINS -P output ACCEPT
$IPCHAINS -P input DENY
$IPCHAINS -A input -i lo -j ACCEPT
WAN_IP='ifconfig $WAN_IFACE |grep inet
➥|cut -d : -f 2 |cut -d -f 1'

 [ -z "$WAN_IP" ] && echo "$WAN_IFACE not
 ➥configured, aborting." && exit 1

$IPCHAINS -A input -p tcp -s $ANYWHERE -d
➥$WAN_IP $LOCAL_PORTS ! -y -j ACCEPT

$IPCHAINS -A input -p udp -s $ANYWHERE -d
➥$WAN_IP $LOCAL_PORTS -j ACCEPT

$IPCHAINS -A input  -p icmp  icmp-type
➥echo-reply -s $ANYWHERE -i $WAN_IFACE
➥-j ACCEPT
$IPCHAINS -A input  -p icmp  icmp-type
➥destination-unreachable -s $ANYWHERE
➥-i $WAN_IFACE -j ACCEPT
$IPCHAINS -A input  -p icmp  icmp-type
➥time-exceeded -s $ANYWHERE -i $WAN_IFACE
➥-j ACCEPT

$IPCHAINS -A input -l -j DENY
```

van, akkor egy másik hálózati kártyára is szükségünk lesz. Felmerülhet a kérdés, hogy nem lehetne-e ezt csupán egyetlen kártyával megoldani? Abban az esetben igen, ha meg tudjuk valósítani, hogy a kártya egyidejűleg lássa a modemet és a belső hálót is, ugyanis a rendszermag IP-aliassing (IP-álnevesítő) szolgáltatásával egy kártyához több IP-címet is rendelhetünk. A hálózati kártyákhoz egyébként az `ifconfig` nevű paranccsal rendelhetünk IP-ket. Az első kártya esetében például `ifconfig eth0 10.1.1.1`, a másodikhoz például `ifconfig eth1 10.1.1.2`. Második IP-címet a következő módon adhatunk meg: `ifconfig eth0:1 10.1.1.3`.
Az utóbbi csak abban az esetben működik, ha a rendszermag tartalmazza az IP aliassing szolgáltatást – szerencsére ez a legtöbb terjesztés alapmagjában megtalálható.
A bújtatást a legegyszerűbben a következő módon állíthatjuk be: `ipchains -A forward -j MASQ`, illetve `iptables` esetében: `iptables -t nat -A POSTROUTING -j MASQUERADE`. Innentől kezdve a Linuxon átmenő összes

2. lista Egy ügyes otthoni beállítás iptables-re

```
#!/bin/sh
IPTABLES=/sbin/iptables
# ha modemen vagy ADSL-en kereszt l
# csatlakozunk, akkor az eth0-t cserØlj k
# ki ppp0-ra!
WAN_IFACE="eth0"
ANYWHERE="0/0"
modprobe ip_conntrack_ftp
$IPTABLES -F
$IPTABLES -P FORWARD DROP
$IPTABLES -P OUTPUT ACCEPT
$IPTABLES -P INPUT DROP
$IPTABLES -A INPUT -i lo -j ACCEPT
$IPTABLES -A INPUT  -p icmp  icmp-type
➥echo-reply -s $ANYWHERE -i $WAN_IFACE
➥-j ACCEPT
$IPTABLES -A INPUT  -p icmp  icmp-type
➥destination-unreachable -s $ANYWHERE -i
➥$WAN_IFACE -j ACCEPT
$IPTABLES -A INPUT  -p icmp  icmp-type
➥time-exceeded -s $ANYWHERE -i $WAN_IFACE
➥-j ACCEPT
$IPTABLES -A INPUT -m state  state
➥ESTABLISHED,RELATED -j ACCEPTT
$IPTABLES -A INPUT -m state  state NEW
➥-i ! $WAN_IFACE -j ACCEPTT
$IPTABLES -A INPUT -j LOG -m limit  limit
➥30/minute  log-prefix "Dropping: "
```

csomag fejléce átíródik. Ha a parancs kiadásakor valamiféle hibaüzenetet kaptunk, minden bizonnyal nincs a rendszermagunkban ez a szolgáltatás. A legtöbb terjesztés ilyen maggal szállítja termékeit.
Ha látszólag minden rendben van, de mégsem működik a dolog, akkor valószínűleg az a gond, hogy ki van kapcsolva az úgynevezett IP forwarding. Semmi pánik! Az `echo 1 > /proc/sys/net/ipv4/ip_forward` utasítás segítségével egyszerűen csak be kell kapcsolnunk.
Meg kell jegyeznünk, hogy mind az `ipchains`, mind az `iptables` beállításai elvesznek a rendszer leállításakor, ezért ha nem akarjuk minden alkalommal bepötyögni a parancsokat, írjuk be valamelyik rendszerindító parancsfájlba!
A többi gépen a saját belső IP-címükön kívül két dolgot kell beállítanunk. Az első, hogy az alapértelmezett átjáró a Linuxunk belső IP-címe legyen. Ha az a gép szintén Linux, ezt a következőképpen tehetjük meg: `route add default gw 10.1.1.1` (a `10.1.1.1` helyére értelemszerűen átjárónk belső IP-címe kerül). A második a DNS-kiszolgáló, amit a Linux esetében a */etc/resolv.conf* állományban adhatunk meg a `nameserver` után, például: `nameserver 195.72.32.131`. Ezek után úgy kell látnunk kifelé, mintha közvetlenül kapcsolódnánk a Világhálóra.

Garzó András
(garzoand@interware.hu) körülbelül három éve foglalkozik Linux- és más Unix-rendszerekkel. Legjobban az operációs rendszerek lelkivilága érdekli, de nyitott egyéniség. Kedvenc étele a palacsinta, és van egy Richard nevű macskája. Minden észrevételt, megjegyzést, levelet szívesen fogad.