FreeBSD – a szomszéd vár (8. rész)
Vállalati tűzfal készítése

A vállalati tűzfalak sokrétű feladatokat kell ellássanak: a gépek védelmén túl általában a belső hálózat címfordítását és a forgalom rendszabályozását is el kell végezniük. Ezen feladatokat nyugodt szívvel rábízhatjuk egy FreeBSD alaprendszerre, hiszen tartalmaz minden szükséges eszközt.

NAT – hálózati címfordítás

A hálózati címfordítással megbízott gép a belső hálózat gépeitől kapott csomagokat úgy módosítja, hogy azok feladójaként a saját címét tünteti fel: így a válaszként érkező csomagok is hozzá kerülnek vissza, amelyeket ismét módosítva – immár a belső hálózaton – az eredeti kérést indító géphez továbbít. A belső hálózaton lévő gépek ezáltal meg vannak védve minden kívülről induló támadás ellen, s mégis képesek teljes körű szolgáltatások elérésére (bár van néhány megkötés is). Linux rendszereknél az ***iptables***/***ipchains*** programok végezték a ***NAT*** beállításait, ***FreeBSD*** operációs rendszer esetén viszont erre egy külön program készült: az ***ipnat***. Felesleges keresgélnünk a ports adatbázisban, az ***ipnat*** az alaprendszer része, sőt a rendszermag is tartalmazza, ha az alaprendszerhez tartozó ***GENERIC*** kernelt használjuk. Az ***ipnat*** bekapcsolásához mindössze a

```
ipnat_enable="yes"
```

sort kell elhelyezni a ***/etc/rc.conf*** állományban, s a következő induláskor már használhatjuk is a címfordítást. Alapesetben a ***/etc/ipnat.rules*** állományból olvassa be a program a rá vonatkozó szabályokat, amely helyett más állományt is megadhatunk a ***/etc/rc.conf*** állományba írt

```
ipnat_rules="/root/ipnat.rules"
```

sorral. Ha újraindítás nélkül szeretnénk megúszni az első próbálkozásokat, akkor az

```
# ipnat -C -f /etc/ipnat.rules
```

parancsot kell kiadnunk, amely törli az előzőleg beolvasott szabályokat, majd a megadott állományból újakat olvas be. Ha az aktív ***NAT*** kapcsolatokat is törölni szeretnék a belső táblázatból, akkor a `-F` opciót is megadhatjuk:

```
# ipnat -C -F -f /etc/ipnat.rules
```

Érdemes „konzolközelben” végezni ezen tevékenységeket, mivel egy hibásan megállapított szabállyal könnyedén elvághatjuk magunkat a hálózaton történő további adminisztrációtól...

Alapvetően négy parancs áll rendelkezésünkre ahhoz, hogy a ***NAT*** szolgáltatásunkat megfelelően beállíthassuk, ebből kettőt használunk rendszeresen. ***iptables*** programhoz szokott kézzel először nehézkes lesz a megfelelő szintaxis megadása, pár perc után azonban nevetségesen egyszerűvé áll össze a kép.
Leggyakoribb tevékenység a belső hálózat címfordítása, amelyet rábízunk egy ***NAT*** szolgáltatásra. Ezt a `map` paranccsal tudjuk megtenni, mint a következő példában:

```
map xl0 192.168.1.0/24 -> 195.199.153.218/32
```

Ekkor az `xl0` eszközön át (amely ebből adódóan a külső hálózat felé kapcsolódik), a `192.168.1.0/255.255.255.0` belső privát hálózatot a `195.199.153.218` cím felé irányítjuk címfordítással. Ezzel minden szükséges adatot megadtunk az `ipnat` programnak, s a kliens gépek megfelelő beállításai után már eléri a belső hálózat az internet adta lehetőségeket. Néhány probléma felmerül a megoldással kapcsolatban, amelyek ismert hiányosságai a ***NAT*** szolgáltatásnak. Ha sok gépet engedünk ki egyetlen IP cím alatt, akkor szükségszerűen működésképtelenek azok a programok, amelyek ***PtP*** módon szeretnének működni (játékok, videó/audiókonferenciák, stb). Ennek oka az, hogy a külső hálózatról a tűzfal IP címe felé induló független (nem választ tartalmazó) csomagok nem tudnak bejutni a belső hálózatba, lévén „ismeretlenek” a tűzfal számára, illetve az általuk használandó hálózati kapu sincs nyitva. Ezen csorba részleges javítását a csomagok átirányítása nyújtja, amelyre az `rdr` parancs szolgál. A vállalati intranet külső elérését (például távmunka okán) a tűzfal lehetővé teheti a megadott címekről (lásd előző rész). Persze ehhez egy átirányítást kell készíteni, amely ezt lehetővé is teszi:

```
rdr xl0 195.199.153.218/32 port 80 -> 192.168.1.254
port 80 tcp
```

Ennek megfelelően az `xl0` eszközön érkező csomagot, amely a `195.199.153.218` IP címre érkezik (az ipfw is átengedte), s a TCP 80-as (http) kaput határozta meg célként, a `192.168.1.254` címmel rendelkező gép 80-as kapujára to-

továbbítja. Az átirányítás másik használati lehetősége a terhelés elosztása. Ekkor a tűzfalunknak az a dolga, hogy felváltva dobálja a kéréseket más-más belső hálózaton lévő gép számára:

```
rdr xl0 195.199.153.218/32 port 80 ->
192.168.1.254,192.168.1.253,192.168.1.252 port 80 tcp
```

Ennek megfelelően a három megadott kiszolgáló számítógép felváltva kapja a tűzfal IP címére érkező kéréseket.

DummyNet – hálózati esélyegyenlőség

Otthoni körülmények között ritkán támad arra igényünk, hogy az „egygépes" kis hálózatunkban a sávszélességet korlátozzuk. Vállalati közegben a felhasználók száma indokol bizonyos korlátozásokat, hiszen többnyire ugyan azzal az ***ADSL*** vonallal van internet kapcsolata az egész cégnek, mint amit – jó esetben – otthon is használunk. Ha egy felhasználó elkezd letölteni egy nagyobb állományt – például egy videót – azonnal használhatatlan lesz a többi résztvevő számára hálózat. Ez jó esetben nem okoz problémát, de ha internetes technológiát igénylő programokkal dolgozni kellene, akkor súlyos bevételkiesést is okozhat a hálózat felesleges terhelése.
Ha a sávszélességet növelni nem lehet, illetve az igényeket sem tudjuk szóbeli figyelmeztetéssel csökkenteni, akkor fizikailag kell beavatkozni, hogy a hálózat terhelését mérsékelni tudjuk. ***FreeBSD*** esetén az alaprendszer tartalmazza a ***DummyNet*** nevezetű eszközt, amely eredetileg hálózati tesztek elvégzésére szolgált, de napjainkra önálló életre kelt. A ***DummyNet*** képes a csomagok egy részét elveszejteni, késleltetni, illetve a hálózat sávszélességét csökkenteni; így kiváló lehetőséget adott a hálózatot használó programok tűrőképességének vizsgálatára. Ugyanakkor az `ipfw` programmal együttműködve lehetőségünk nyílik egyes felhasználók tűrőképességének vizsgálatára is, hiszen az általuk használt gépről indított csomagok egy részét el tudjuk veszejteni, késleltetni vagy a sávszélességüket csökkenteni...
A program használatához a rendszermagot újra kell fordítani a

```
options DUMMYNET
```

megadásával, amely sort például a */usr/src/sys/i386/conf/MYKERNEL* állományba kell írnunk. Ehhez sajnos újra kell indítani a gépet, de ebben a szomorú tényben bőven kárpótol majd minket – gonosz rendszergazdákat – a felhasználóink szomorúsága, amint működni kezd a sávszélesség személyre szabott csökkentése...
Mint említettem, a ***DummyNet*** az `ipfw` programba épül be, mégpedig a `pipe` kulcsszón keresztül, s így a tűzfalunkat beállító programocska a következő sorokat tartalmazhatja:

```
#!/bin/sh

/sbin/ipfw -q flush

# A helyi forgalom engedélyezése
/sbin/ipfw add 1 allow ip from 10.1.1.0/24 to any
/sbin/ipfw add 2 allow ip from me to 10.1.1.0/24
# A kiszolgáló forgalmának engedélyezése
/sbin/ipfw add 60100 allow ip from me to any
/sbin/ipfw add 60200 allow ip from any to me

# A főnök gépének sávszélessége
/sbin/ipfw add 257 pipe 257 ip from any to
➥10.1.1.1/32
/sbin/ipfw pipe 257 config bw 20kByte/s
# Az „nemszeretem" kolléga sávszélessége
/sbin/ipfw add 258 pipe 258 ip from any to
➥10.1.1.2/32
/sbin/ipfw pipe 258 config bw 100Byte/s
# A kedves titkárnéni sávszélessége
/sbin/ipfw add 259 pipe 259 ip from any to
➥10.1.1.3/32
/sbin/ipfw pipe 259 config bw 200kByte/s
```

Látványosan olvasható a ***DummyNet*** működése: megadjuk a megfelelő ***ipfw*** szabályt, a hozzá tartozó cső számát, majd felkonfiguráljuk az adott számú csövet. A `bw` kulcsszó jelenti a sávszélességet, a mértékegység magáért beszél. Ha késleltetést szeretnénk beállítani, azt a `delay` parancs segítségével tehetjük meg:

```
# /sbin/ipfw pipe 258 config delay 30000
```

Ennek megfelelően a közutálatnak örvendő kolléga fél percet kénytelen várni minden csomagjára. Folyamatos letöltésnél nem jelent sokat, de az interaktivitást igénylő programokkal sokat fog szenvedni. Ha nagyon gonosz módon állunk a kollégához, akkor még egy 90%-os csomagvesztéssel is megterhelhetjük a hálózati kapcsolatát:

```
# /sbin/ipfw pipe 258 config plr 0.9
```

Sajnos IP címhez tudunk csak rendelni ilyen ***DummyNet*** szabályokat, ha felhasználókat szeretnénk korlátozni s nem gépeket, akkor ennél sokkal nehezebb dolgunk lesz.

Melyik programot válasszam?

A ***FreeBSD*** alaprendszer és ***ports*** adatbázis ennél a pár programnál sokkal több eszközt tartalmaz, amelyek mind-mind ugyan arra a feladatra más-más módszerrel adnak megoldást, ezek felsorolására sem vállalkozok, a pontos működésüket sem tudom a hely hiányában bemutatni. Csak ajánlani tudom minden kedves ***FreeBSD*** felhasználónak, hogy próbálja ki a lehetőségeket és válassza a hozzá legközelebb álló programcsomagot. Nem érdemes az összes lehetőséget pontosan ismerni, elég egyet mélyebben, ahogy magam is csak az ipfw és az `ipnat` programokat használom gyakrabban (holott használhatnám az `ipf` és a `natd` programokat is :).

Auth Gábor (auth.gabor@enaplo.hu)

KAPCSOLÓDÓ CÍMEK

A FreeBSD projekt honlapja: ➲ http://www.freebsd.org
A magyar FreeBSD honlap: ➲ http://www.freebsd.hu
A magyar BSD honlap: ➲ http://www.bsd.hu
A kézikönyv magyar fordítása
➲ http://www.enaplo.hu/FreeBSD/handbook/