Biztosítsuk be hálózatunkat a Kazaa ellen!

Ha a tűzfalak megkerüléséről van szó,
a Kazaa géptől-gépig rendszer igen ravasz tud lenni. Csakhogy nem eléggé.

Manapság a legnépszerűbb fájlmegosztó alkalmazás a Kazaa. Az ilyen típusú alkalmazásokat géptől-gépig (peer-to-peer, P2P) programnak nevezik, amelyek lehetővé teszik, hogy a felhasználók egymás gépén keressenek, és állományokat töltsenek le. A Kazaa programot jelenleg arra használják a legtöbbször, hogy hangállományokat terjesszenek a szerzői jogok megsértésével.
A Kazaa FastTrack néven ismert üzleti hálózati protokollját számos hasonló termék készítőjének is átadta (licencelte), például a iMesh és a Grokster programoknak. Elérhető továbbá a Kazaa „lebutított" változata is, KazaaLite néven. Rengeteg egyéb P2P-alkalmazás is létezik, de a FastTrack család messze a legnépszerűbb közülük, és a legnehezebben állítható meg olyan csomagszűrő tűzfalakkal, mint a Linux `iptables`.
Számos hálózati gazda szívesen letiltaná a tűzfalán a P2P-forgalmat a nagy sávszélesség-használat, az ellenőrzés nélküli fájlcseréből eredő biztonsági rések és a jogtulajdonosok esetleges jogi ellenlépései miatt. Csakhogy ez nem olyan egyszerű, mint amilyennek hangzik. Ha az interneten rákeresünk az `iptables` alapú FastTrack forgalomblokkolás témára, olyasféle válaszokat fogunk találni, mint „tiltsuk le a 1214 kaput", „készítsünk rendszabályt, és büntessük meg a gazembereket", vagy egyszerűen „ezt nem lehet megcsinálni". Bár a 1214-es kapu letiltása valóban működött a FastTrack korai változataival, a mai változatoknak már nem árt. Ennél kifinomultabb megoldásra van szükségünk. Néhány proxyalapú tűzfal képes ugyan megállítani a FastTrack forgalmát, de az `iptables` alapú tűzfalak esetében megoldást kell találnunk néhány nehézségre. Ebben a cikkben egy új, `P2Pwall` névre hallgató nyílt forrású projektet szeretnénk bemutatni. A projekt célja egy olyan program kifejlesztése, amely képes meggátolni a hálózatunkon próbálkozó P2P-ügyfeleknek a külső gépekhez való csatlakozását. A program `ftwall` összetevője meggátolja a FastTrack-forgalmat. További összetevők is készülnek, amelyekkel más P2P-protokollok ellen lehet védekezni. A fejlesztők között mindenkit szívesen látunk! A programot a következő FastTrack-ügyfelekkel próbáltuk ki: Kazaa 2.1.1, Kazaa 2.5, KazaaLite 2.0.2, iMesh 4.1 (build 132) és Grokster 1.7.

A tűzfalak nehezen boldogulnak a FastTrackkel

A korszerű Linux-terjesztések általában tartalmazzák a Netfilter és az `iptables`-eszközöket. Ezek az eszközök együtt lehetővé teszik, hogy Linux-rendszerünket egyszerű, de hatékony tűzfalként használjuk; csakhogy a FastTrack hálózati protokoll néhány érdekes kihívást rejt magában, amelyek a következők:

- nem használ állandó kapuszámot;
- a kommunikációja nem korlátozódik kis számú gépre: kétszáz gép címét tárolja és induláskor valamennyihez megpróbál csatlakozni; a lista időnként változik és minden gépen más és más;
- a gépkereső eljárás központi adattártól független;
- a protokoll kulcsfontosságú részei erős titkosítást alkalmaznak.

A tűzfalak általában két filozófia egyikét alkalmazzák. Az első igen szigorú: minden kaput letilt, kivéve néhány szükségeset. A második engedékenyebb és aszimmetrikus: szinte bármilyen kimenő kapcsolatot engedélyez, ugyanakkor csaknem minden bejövőt meggátol. Bármelyik megközelítésről legyen szó, a kapukat fürgén váltó FastTrack körbeszimatol, és kihasználja a legálisan nyitva lévő kapukat. Akár a 80-as kaput is használhatja. A szigorú szabályrendszerrel és egy 80-as kapun futtatott proxy együttes alkalmazásával ugyan megállíthatjuk a FastTracket, de ez a megközelítés már túlságosan kötött az olyan hálózatok számára, amelyek tartani szeretnék magukat az engedékeny elképzeléshez, mégis szívesen blokkolnák a P2P-forgalmat.

A P2Pwall projekt ftwall programja

A P2Pwall projekt ezeket a nehézségeket próbálja meg orvosolni azáltal, hogy számos, a P2P-forgalom szűrésével foglalkozó eszközt és dokumentációt nyújt. A FastTrack szűrő `ftwall` lenne az első ilyen eszköz, amelyet a GPL engedély oltalma alatt tölthetünk le a ➲ http://p2pwall.sourceforge.net címről. Az `ftwall` a `QUEUE` cél használatával kapcsolódik a `iptables`-hez. Megvizsgálja a tűzfalon keresztülküldött csomagokat, és a FastTrack protokoll jellemzőinek ismerete alapján eldönti, hogy továbbíthatók-e vagy el kell-e vetni őket. Egyúttal megpróbál a hálózatunkról kifele (így egyúttal befele) igyekvő minden FastTrack-forgalmat letiltani.
Az `ftwall` feladata a kifelé menő FastTrack-kapcsolatok tiltása, mivel feltételezi, hogy a bejövő kapcsolatokat az `iptables` szabályok már eleve meggátolják. Számos tűzfal használ általános blokkolást a befele érkező kapcsolatokra, ahol csak bizonyos számú kiszolgálókapcsolatot engedélyez. Csakhogy ha a belső FastTrack-ügyfél kezdeményez kapcsolatot kifele, a külső géphez, a kívülálló a már meglévő kapcsolaton keresztül visszahívhatja a belső gépet. Így aztán, ha a tűzfalunkra rábízhatjuk a bejövő kapcsolatok blokkolását, az `ftwall` pedig átnézi a kifelé menőket, a probléma megoldható; csak arra kell ügyelnünk, hogy mindkét elem a helyén legyen.
Az `ftwall` telepítése és beállítása annyiból áll, hogy letöltjük a forrást, lefordítjuk és megírunk néhány `iptables` szabályt. Esetleg nehézséget okozhat, hogy a rendszer egyik választható szolgáltatásának a használatához az `ip_string` modulnak benne kell lennie a rendszermagban. A modul jelenleg kísérleti szakaszban van, így sok Linux-terjesztés nem is tartalmazza. Ha használni szeretnénk, valószínűleg magunknak kell bíbelődnünk a felrakásával. További útmutatást a P2Pwall honlapján találunk.

Az iptables QUEUE (sor) célja

Amikor egy `iptables` szabály célként `QUEUE`-t ad meg, a szabállyal egyezést mutató minden csomag valamilyen alkalmazás (például az `ftwall`) által gyűjtött sorba kerül.
A program azután eldobhatja a csomagot vagy további ellenőrzésre és továbbításra adhatja vissza a Netfilternek. A szerke-

zetet meghívó jellemző szabály a következő alakú lesz:

```
iptables -A FORWARD -p tcp -i eth0
➥-dport 123 -syn -j QUEUE
```

Ha a fenti szabály érvényben van, az `eth0`-hoz csatlakozó hálózatról érkező és a távoli gépen a 123-as kaput célzó valamennyi SYN csomag először a programhoz kerül. A program első ízben beolvassa a csomagot, majd a `libipq` könyvtár és a `ip_queue` modul segítségével visszaküldi az ítéletét.
A `QUEUE` szabványos eleme az `iptables`-nek, amelyet a legtöbb terjesztésben megtalálhatunk. Ha ellenőrizni szeretnénk, hogy a rendszerünkön létezik-e ilyesmi, gépeljük be az `insmod ip_queue` parancsot, majd ellenőrizzük, hogy nem kaptunk-e hibaüzenetet. További részleteket olvashatunk a ➲ http://www.netfilter.org/documentation/FAQ/netfilter-faq-4.html címen elérhető Netfilter dokumentumban.

Az ftwall működése

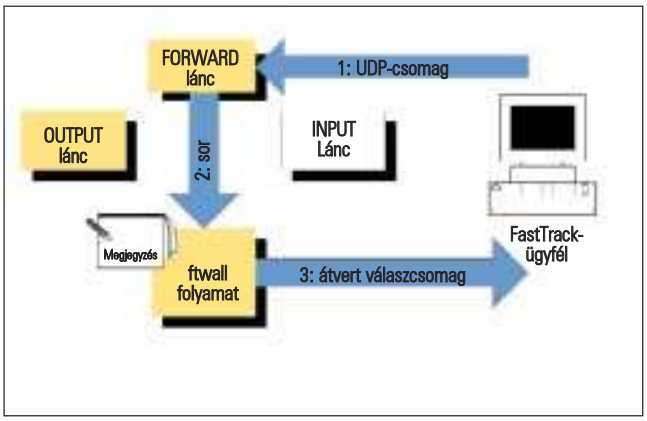
Ha ismertetni szeretnénk az `ftwall` szerkezetét, lépésről lépésre végig kell haladnunk a FastTrack-kapcsolat logikájának bizonyos részének az ismertetésén. A FastTrack három különböző megközelítést alkalmaz a gépekkel való kapcsolatteremtésre nézve: UDP-csomagok áradatát (flood), párhuzamos TCP/IP kapcsolatokat és a kicsit hagyományosabb TCP/IP kapcsolati mintát. Ha úgy érzi, hogy akadályozzák, a program egy másik módra vált át. Az `ftwall` az ügyfeleket igyekszik a lehető legtovább az első módban tartani, hiszen ezt a legkönnyebb azonosítani, valamint lehetővé teszi a célgépek listájának az elkészítését.
Az ügyfél induláskor nagyszámú UDP-csomagot küld keresztül a tűzfalon, amelyeket hosszuk és tartalmuk alapján beazonosíthatunk. A Netfilter ezeket az `ftwall`-on keresztüli feldolgozásba sorolja át (*1. ábra*). Ezután az `ftwall` belső feljegyzést készít a csomagok forrás- és célcímeiről, majd hamis választ küld az ügyfélnek. Ezáltal megakadályozza, hogy az ügyfél arra a következtetésre jusson, hogy az UDP-csomagokat tűzfal blokkolja, és így egy kicsit tovább fut az első módban.
Feltételezve, hogy a saját hálózati csatolófelületünk az `eth0`, az átsorolást a következő `iptables` szabállyal adhatjuk meg:

```
iptables -A FORWARD -p udp -i eth0 -j QUEUE
```

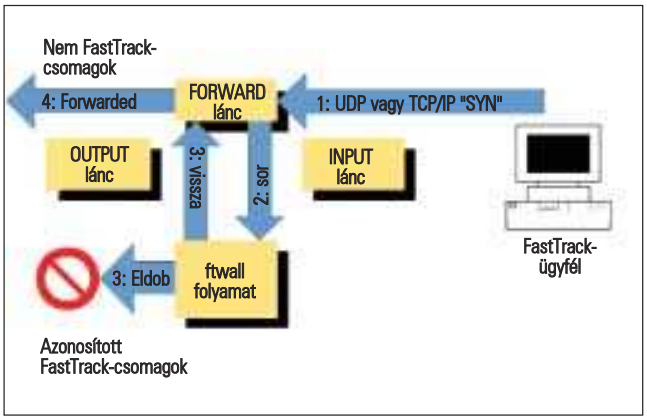
Miután a FastTrack megkapja a hamis választ, UDP-n keresztül megpróbál további adatokhoz jutni, majd ugyanehhez a címhez megkísérli létrehozni a TCP/IP-kapcsolatot. Ezek az UDP- és TCP-csomagok ugyancsak az `ftwall`-hoz kerülnek, amely most már tudja, hogy a célcímek a FastTrackre hivatkoznak, így eldobhatja őket. Az egyéb UDP-, nem FastTrack-csomagok és TCP/IP SYN-csomagok további ellenőrzésre visszakerülnek a Netfilterhez, s ezután céljuk felé továbbítódnak.
A következő szabály a SYN-eket átsorolja az `ftwall`-ba:

```
iptables -A FORWARD -p tcp -i eth0 --syn -j
QUEUE
```

Az ügyfél ezt az UDP és SYN sorozatot ismétli egy ideig – általában (de nem mindig) addig, míg a listájában található valamennyi ismert címet legalább egyszer ki nem próbálta. Ez egyben azt jelenti, hogy a címeket most már az `ftwall` is ismeri, és mint szűrendőket megjegyzi.
Idővel az ügyfél taktikát változtat és erős titkosítással felvértezett, párhuzamos TCP/IP-kapcsolatra vált át. Az `ftwall`

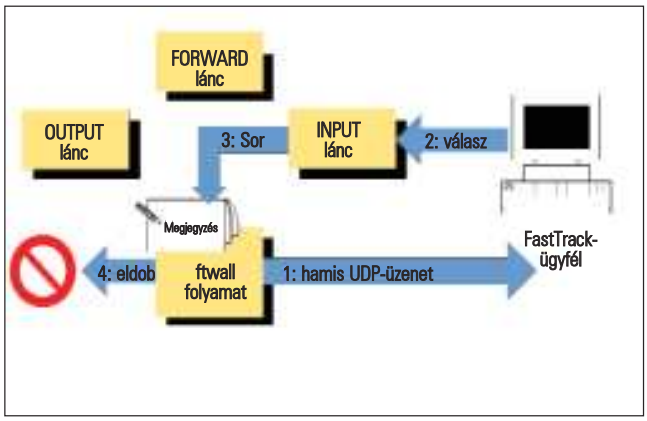


1. ábra Az UDP-csomagok nyitóképe



2. ábra Egyéb UDP- és TCP/IP SYN-csomagok

továbbra is gátol minden kapcsolatot az előző szakaszban feljegyzett címekre. Minden más cím esetében az egyetlen szóba jöhető bizonyíték, amely beazonosíthat egy FastTrack-kapcsolatot, az az, ha rövid idő alatt nagyszámú SYN-csomagot találunk valahol. Ha az `ftwall` kizárólag az UDP-csomagokra alapozná a blokkolást, vereséget szenvedne, különösen, ha az ügyfél az első szakaszban nem próbálta végig az összes ismert címet. A gond megoldása az időzár.
Ebben az új módban az ügyfél keverve próbál TCP-kapcsolatokat megnyitni az `ftwall` által ismert IP-címekre és a még ki nem derítettekre (amennyiben létezik ilyen). Az `ftwall` megjegyzi azt az időt, amikor a legutóbbi ismert címet megpróbálták elérni, és egy előre megadható ideig minden TCP/IP-kapcsolatot blokkol arról a forrás IP-címről. Minden SYN-csomag, ami ismert címre igyekezne, újra nullára állítja a számlálót. Amennyiben a kapcsolatot kellően sűrűn próbálják meg létrehozni, az `ftwall` folyamatosan blokkolni fog.
Ennek a módszernek a mellékhatásaként a lator gép minden TCP/IP-kapcsolata blokkolva lesz: mindaddig amíg a FastTrack próbál kapcsolódni, ideértve web- és az FTP-helyek elérését is. Mondhatjuk azonban, hogy ez elfogadható, hiszen a munkaállomás használója megszegi a szervezet szabályzatát. Amint az ügyfélalkalmazást lezárják, a számláló többé nem frissül, és a lejártát követően a TCP/IP-kapcsolatok ismét engedélyezettek lesznek. Az alapértelmezett beállítások szerint ez két percet jelent.
Miután a FastTrack egy ideig ebben a módban futott, arra a következtetésre fog jutni, hogy a párhuzamos stílusú kapcsolat problémákat okoz, ezért átvált a harmadik módba.



3. ábra UDP FastTrack-állapotpróba

Lelassítja a kapcsolatkereső próbálkozások sűrűségét, és a hagyományosabb, egyszerre csak egy címmel próbálkozó és a próbák között néhány másodperces szünetet tartó megoldást választja. Ez az új megközelítés minden eddigi elképzelésünket romba dönti, és az ügyfél időnként kijut. Ez akár egy órát is igénybe vehet, de azok az ügyfelek, amelyek nem árulják el idejekorán az összes címüket, ebben a szakaszban jó eséllyel képesek lesznek kapcsolatot kiépíteni. És ha már egyetlen egy kapcsolat kiépült, egy teljesen új címkészlet töltődik le. Ettől kezdve ott tartunk, mintha semmiféle blokkolást sem csináltunk volna.

A harmadik mód legyőzéséhez az `ftwall`-nak további adatokra van szüksége, amelyekből megállapíthatja, hogy a FastTrack még használatban van-e. Az egyik lehetséges módszer alkalmazásához még egy kis átejtésre van szükség. Az `ftwall` időről időre egy UDP-csomagot küld az ügyfélnek, mégpedig a pontos másolatát annak, amit az ügyfél maga használt a társához irányuló kapcsolat megnyitásához (*3. ábra*). Amennyiben a gépen fut a FastTrack program, egy könnyen felismerhető csomaggal válaszol, így az időzítő ismét alaphelyzetbe áll vissza. A viszonylag kisszámú és -méretű kutatócsomagok csak kevéssé terhelik a hálózatot.

Minthogy ezt a csomagot nem akarjuk nyilvános címre továbbítani, hanem magának a tűzfalnak szánjuk, létre kell hoznunk egy `iptables` szabályt az `INPUT` láncban, hogy az végül az `ftwall`-hoz kerüljön. A felhasználandó szabály:

```
iptables -A INPUT -p udp -i eth0 -j QUEUE
```

Így az ügyfelet folyamatosan hálózaton kívül tartjuk, de a megoldás nem igazán hatékony. Semmi mást nem kell tennünk, csak helyes időzárértéket alkalmaznunk, úgy, hogy az UDP-csomagokat akkor küldjük, amikor az idő körülbelül félig letelt, és az ügyfelet folyamatosan blokkolva tarthatjuk.

Kirakósjátékunk utolsó darabja egy biztonsági háló, amire elméletileg soha nincs szükség. A fenti logika felismerhető UDP-csomagokra alapoz, amelyek az `ftwall`-t a szükséges adatokkal látják el, de gondolnunk kell arra az esetre is, amikor ezek a csomagok egyáltalán nem érkeznek meg – például azért, mert a felhasználó az UDP-átvitelt kikapcsolja a munkaállomás tűzfalán. Ebben az esetben semmilyen módon nem tudhatjuk meg a társgépek címeit.

Ilyenkor is maradt még egy lehetőségünk: vizsgáljunk meg minden TCP/IP-adatcsomagot, és próbáljuk meg felfedezni bennük a fájlok átvitelére utaló nyomokat. A FastTrack titkosítási eljárása csak a kapcsolatteremtő kézfogásra és a keresésekre korlátozódik. A megosztott állományok egyszerű szöveges HTTP formátumban utaznak, de nincsenek a 80-as kapuhoz kötve. A HTTP-kérelemfejlécek számos mezőt tartalmaznak, amelyek a FastTrack-felhasználót, a protokollt és a szupercsomópont címét tartalmazzák (ez az a csomópont, amelyik az indexadatokat nyújtja). Amennyiben ezeket a csomagokat ellenőriztetjük, az `ftwall`-lal be fogja azonosítani közülük azokat, amelyek gyanúsan hasonlítanak FastTrack-fájlletöltés elejére. A HTTP-fejlécekben tárolt adatokból a blokkolandó címek listájára felveszi a cél és a szupercsomópont IP-címét, az ügyfelet pedig beszúrja azoknak a listájába, akikre az időzármódszert alkalmazni kell.

A telepítés áttekintése

A `ftwall` telepítésének folyamatát részletesen bemutatja a programhoz csatolt ***INSTALL*** állomány, illetve elolvashatjuk a projekt weblapján, de röviden a következő lépéseket kell megtenni:

- Töltsük le a forrásokat a ➲ http://P2Pwall.sourceforge.net címről, és csomagoljuk ki őket.
- Amennyiben még nincs telepítve, telepítsük a ***libipq*** könyvtárat. Néhány rendszeren (ide tartozik a Red Hat 7.x és 8) ez egyet jelent az `iptables` forrásának letöltésével és befordításával.
- Fordítsuk le és telepítsük az `ftwall`-t a `make` és `make install` parancsokkal.
- Adjunk egy bejegyzést a */etc/rc3.d* behúzófájlba, amely majd elindítja az `ftwall`-t.
- Ellenőrizzük, hogy elérhető-e a `QUEUE` mechanizmus, és tegyük fel, ha nem. A legtöbb mai Linuxban a helyén szokott lenni, de a rendszermag foltozásával és újrafordításával a többihez is hozzá lehet adni.
- Készítsük el az `INPUT` és `FORWARD` láncokba kerülő szabályokat.
- Ha „biztos, ami biztos” alapon telepíteni szeretnénk a fájlletöltések HTTP-fejlécének a vizsgálatát végző lehetőséget is, például abban az esetben, ha a hálózatunkon az UDP nem működne, adjuk a rendszermaghoz és az `iptables`-hez a `string` modult. Ehhez a rendszermagot meg kell foltoznunk és újra kell fordítanunk.
- Indítsuk újra a gépünket.

Összefoglalás

Az `ftwall` az írásunk születésekor használatos valamennyi FastTrack-ügyféllel megbirkózik. Elképzelhető, hogy a FastTrack-protokoll a jövőben megváltozik, ebben az esetben az `ftwall`-t is valószínűleg módosítani kell.

A megközelítés hátránya, hogy kizárólag a FastTrack-rendszerekre összpontosít; igaz, a P2Pwall projekt egyik célja, hogy a jövőben más P2P-protokollokra is kiterjessze a hatáskörét. Amennyiben valaki hajlandóságot érezne magában, és részt kívánna venni egy ilyen irányú fejlesztésben, írjon nekem levelet a chris@lowth.com címre.

Linux Journal 2003. október, 114. szám

Chris Lowth (chris@lowth.com)
az Intercai Mondiale (➲ http://www.intercai.co.uk) Egy angol telekommunikációs, IT- és üzleti tanácsadással foglalkozó cég munkatársa. Feleségével, három fiával és golden labradorjával Londonban él.