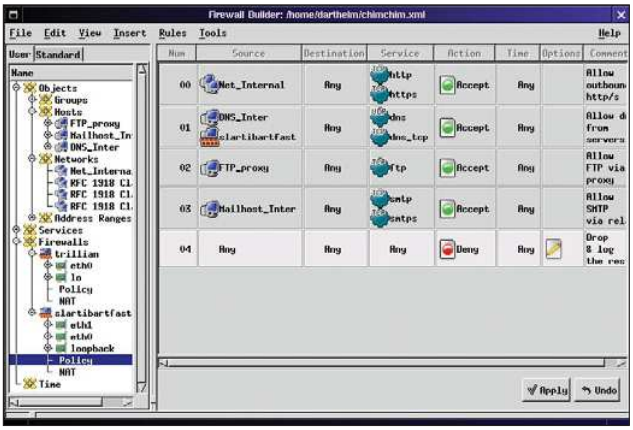
A Firewall Builder használata (1. rész)

Gondoltál már arra, hogy egyetlen könnyen használható, grafikus alkalmazással az összes tűzfalad és kiszolgálód házirendjét kézben tarthatnád?

A 2.4-es Linux Netfilter tűzfalkódja, valamint felhasználói felülete, az IP Tables alaposan kiérdemelte népszerűségét és az őt érő dicséreteket. Ezek segítségével válhattak a Linux alapú tűzfalak a kereskedelmi, állapotalapú csomagszűrő tűzfalakkal egyenértékű megoldásokká, akár a szolgáltatások vagy az intelligencia, akár a biztonság szempontjából vizsgáljuk őket.
A Netfilter kapcsán egyetlen hiányosságot lehetett megemlíteni: a felhasználóbarát jelleg elmaradását. Ha egy tűzfalat jó grafikus felülettel sikerül ellátni, az nem csak a kisebb műszaki érzékkel bíró felhasználók számára jelent segítséget. Idővel még a leginkább kockafejűek is rájönnek, hogy gyorsabban és kevesebb hibával készíthetnek tűzfalházirendeket, ha szemléletes megjelenítés és jelzések segítik őket munkájukban. Az IP Tables-szabályok írásmódjában egyértelműen a biztonsági szabályok szempontjai érvényesülnek, nem az olvashatóságé.
A Firewall Builder (*1. kép*) igazán kiváló grafikus tűzfalfelület. Segítségével állomás-, hálózat- és szolgáltatásobjektumokat hozhatunk létre, amelyeket tetszőleges számú tűzfalszabályban (újra)felhasználhatunk. A szabályokat szemléletes és áttekinthető módon jeleníti meg, és mivel alapvetően operációs rendszertől független, a Firewall Builder segítségével nemcsak Netfilter/IP Tableshez, de a FreeBSD IP Filteréhez, az OpenBSD `pf`-éhez és a Cisco PIX-tűzfalakhoz is készíthetők szabályok.
Ez alkalommal, illetve a következő részben azt mondom el, hogyan szerezheted be és telepítheted a Firewall Buildert, majd szót ejtek arról is, hogy a segítségével miként hozhatsz létre szemléletes és egyszerű módon IP Tables-szabályokat. Elsőként a Firewall Builder telepítését tárgyalom, ezt követően feltöltjük az objektumokat tároló adatbázisát. A szabályok létrehozásáról a következő részben fogok írni.



1. kép A Firewall Builder működés közben

Hová kerüljön a Firewall Builder?

Elsőként tekintsük át, hova kell telepíteni, illetve honnan kell futtatni a Firewall Buildert. Én úgy gondolom, hogy a Firewall Buildert nem célszerű magán a tűzfalgépen, illetve bármely egyéb nyilvánosan elérhető, megerősített gépen, azaz úgynevezett bástyagépen futtatni. Nyilván senki nem gondolt rá, hogy X Window fusson egy ilyen gépen.
A Firewall Buildert egy mindennapos használatra szolgáló munkaállomáson érdemes használni. Az így létrehozott tűzfalparancsfájlokat utolsó lépésként `scp`-n vagy más biztonságos eljárással át kell másolni az őket alkalmazó gépre. A Firewall Buildert eleve ilyen használatra tervezték.
Másrészről viszont, ha a Firewall Buildert egy adott állomás – például egy Linux 2.4-es alapú webkiszolgáló – helyi védelmét szolgáló Netfilter-parancsfájlok létrehozására akarjuk használni, akkor talán nem túl nagy vétség, ha magát a Firewall Buildert is a parancsfájlokat alkalmazó gépen futtatjuk. Az X11 telepítésére ekkor is ügyelni kell, illetve az adott állomásnak megfelelően beállított tűzfal mögött kell lennie.
Fontos, hogy a Firewall Buildert nem szükséges az összes beállítani kívánt állomáson külön futtatni. Nem muszáj tehát egyetlen olyan gépen sem futtatni, amelyen egyébként nem használnál X Window rendszert. Egyetlen Firewall Buildert futtató gépen számos más gép számára hozhatsz létre tetszés szerinti szabályokat. Ennek pontos módjáról rövidesen szót ejtünk.

A Firewall Builder beszerzése és telepítése

A Firewall Builder Projectnek természetesen saját honlapja van, ahonnan a legújabb kiadás, illetve a leírás letölthető (➲ http://www.fwbuilder.org). Ha a weblapon és az itt olvasottak között bármilyen eltérés tapasztalható, akkor az előbbi a mérvadó. A Firewall Buildernek a honlapon található telepítési leírása érthető és pontos. Természetesen semmilyen változás nem zárható ki cikkem írása és a megjelenés időpontja között.

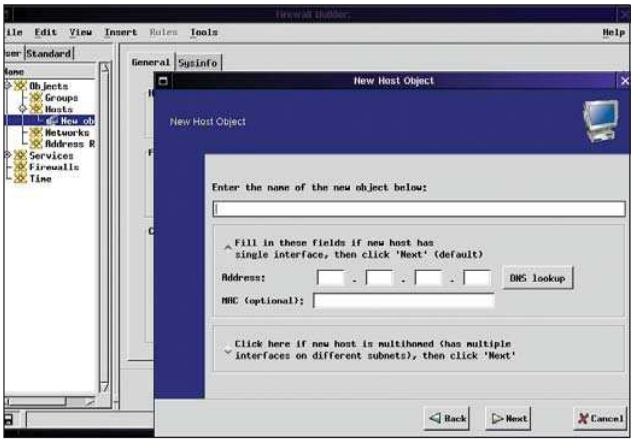
Debian

A könnyebb esettel kezdem. Debian 3.0 alatt a Firewall Builder közvetlenül a Debian telepítési forrásból tehető fel, a Debian ugyanis `fwbuilder` név alatt saját, hivatalos támogatású debcsomaggal rendelkezik. Ez a csomag egyebek mellett a következő Debian-csomagoktól függ: ***libfwbuilder0***, ***fwbuilder-iptables***, ***libgtk1.2***, ***libgtkmm1.2***, ***libxslt1***, ***libxml2*** és ***libsnmp4.2***. A teljes függőségi listát – ha nem gond – most elhagynám.
Ha az `fwbuilder` telepítése `apt-get`-tel történik, akkor az `apt-get` minden szükséges csomag azonosításáról és telepítéséről gondoskodik. A Debian `fwbuilder-doc` csomagjának telepítését is javaslom, és bár a felrakása nem kötelező (és ilyen módon nem is történik meg önműködően, hiszen általa az `apt-get` nem tud semmilyen függőséget feloldani), mindenre kiterjedő és hasznos leírást találunk benne.

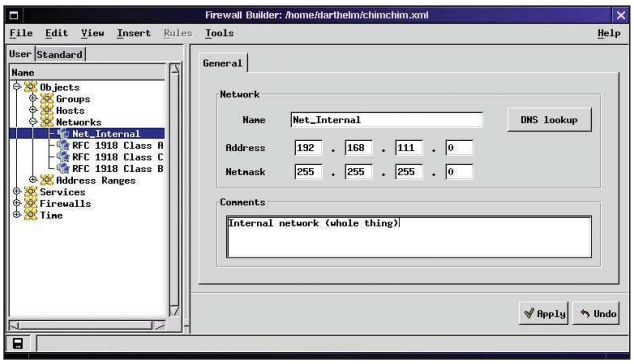
Red Hat

A Red Hat 8.0 (cikkem írásakor ez a legújabb Red Hat-változat) terjesztésnek hivatalosan még nem része a Firewall Builder.
A Firewall Builder fejlesztői csapat ugyanakkor számos Red Hat-terjesztéshez készített RPM fájlokat – ezeket a Firewall Builder

➲ http://sourceforge.net/project/showfiles.php?group_id=5314 letöltések oldalán találhatjuk meg
Az *fwbuilder* és a *libfwbuilder* csomagra lesz szükségünk, illetve az *fwbuilder-ipt*, *fwbuilder-ipf* vagy *fwbuilder-pf* csomag valamelyikére, attól függően, hogy Linux Netfilter/IP Tables, FreeBSD IP Filter vagy OpenBSD `pf` számára akarunk-e szabályokat létrehozni. Az előbbi három csomagból akár mindegyiket is feltelepíthetjük, ha szükséges. Mivel a Firewall Builder végeredményként ASCII formátumú parancsfájlokat állít



2. kép Az Insert Host (állomás beszúrása) párbeszédpanel



3. kép Az Insert Network (hálózat beszúrása) párbeszédpanel

elő, Linux alatt más operációs rendszerek számára is nyugodtan létrehozhatunk szabályokat.
A Firewall Builder-csomagok telepítése előtt a következő normál Red Hat-csomagoknak kell jelen lenniük a rendszerben: bind-utils, gdk-pixbuf, glib, glibc, gtk+, gtkmm, libfwbuilder, libsigc++, libstdc++, libxml2, libxslt, openssl-0.9.6b, ucd-snmp és XFree86-libs.
Ezek mellett szükség lesz a gtkmm (the GIMP Tool Kit Minus Minus) csomagra is, ami a GTK+ C++-kötéseit tartalmazza. A csomag a Ximian Gnome része, azonban a ➲ http://www.freshrpms.net címről is letölthető.

SuSE

A Red Hathez hasonlóan jelenleg még a SuSE sem építette be a Firewall Buildert a hivatalos terjesztésbe. A SuSE 8.1 RPM-ek (nem hivatalos, külső forrásból származók) a Firewall Builder letöltések oldalról (➲ http://sourceforge.net/project/showfiles.php?group_id=5314) érhetők el.
Az *fwbuilder* és a *libfwbuilder* csomagra mindenképpen szükség lesz, illetve az *fwbuilder-ipt*, az *fwbuilder-ipf* és az *fwbuilder-pf* csomagok közül egyet vagy többet úgyszintén le kell tölteni. A telepítéshez a következő normál SuSE-csomagok szükségesek: gcc, gdk_pixbuf, glib, glibc-2.2.4, gtk, gtkmm, libsigc++, libstdc++, libxml2, libxslt, libz, openssl-0.9.6b, ucdsnmp és xshared.

Objektumok létrehozása

A csomagok telepítése után a Firewall Builder készen áll a használatra. Mindössze egyetlen parancsot kell megjegyezni: a `fwbuilder`-t. A parancs kiadásakor az X Window rendszernek már futnia kell. A program nem csak rendszergazdaként használható, sőt nem is javaslom, hogy különösebb indok nélkül így futtassuk, hiszen ki tudja, mit nézünk el.
Az *fwbuilder* ablak megnyílása után nekiláthatunk az objektumok létrehozásának (*2. kép*). A Firewall Builder felfogásának az az alapja, hogy a szabályokat újrafelhasználható, húzd és ejtsd módszerrel elhelyezhető objektumok segítségével hozzuk létre, tehát az objektumoknak még a szabályok megalkotása előtt rendelkezésre kell állniuk. Még a Firewall Builder önműködő szabálylétrehozó varázslói sem használhatók, ha nem hoztuk létre a szükséges objektumokat.
Az objektumok hálózati állomásokat, hálózatokat (ezeket IP-cím és alhálózati maszk azonosíthatja), címtartományokat, TCP/IP-szolgáltatásokat, tűzfalakat (többlaki tűzfalrendszereket és bástyagépeket), időtartományokat és más objektumok csoportjait képviselhetik. Mindenki tetszése szerinti mennyiségben hozhat létre objektumot – annyit, amennyire szüksége lesz a saját szabályaiban. Értelemszerűen legalább egy tűzfal- és legalább egy hálózat- vagy állomásobjektumra szükség van. Számos általánosan használt TCP/IP-szolgáltatáshoz előre megadott objektumokat találunk.

Hálózati állomásobjektumok

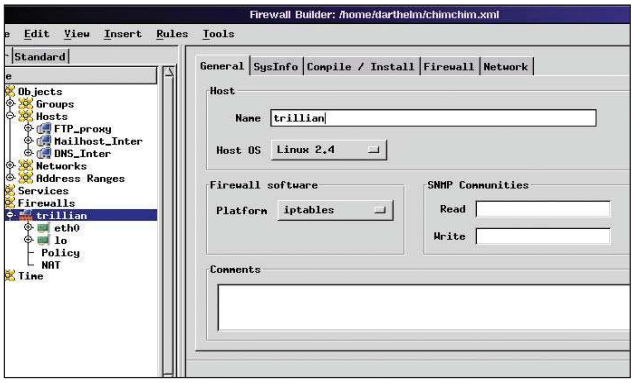
Az objektumok a Firewall Builder ***Insert*** (beszúrás) menüjével hozhatók létre. A 2. képen az állomások létrehozására szolgáló ***Insert host*** (állomás beszúrása) párbeszédpanel látható. A szabályok létrehozásakor az állomások legfontosabb jellemzője az IP-címük. Ha az állomásokat a MAC-, illetve az ethernetcímük alapján azonosító szabályokat akarunk írni, akkor ilyen címeket is megadhatunk. Mint a képről is kitűnik, az IP-címek kézzel és DNS-keresés alapján is megadhatók. Az utóbbi szolgáltatás ugyan hasznos, de ne feledjük, hogy csak olyan állomások esetében használható, amelyek címe a Firewall Buildert futtató gépről feloldható.

Hálózatobjektumok

A *3. képen* az ***Insert network*** (hálózat beszúrása) párbeszédpanel látható. Az állomás beszúrásával ellentétben – ami külön ablakban jelenik meg – a hálózat beszúrását egy egyszerű ***New object*** (új objektum) űrlapon végezhetjük el, a főablak jobb oldali részén. Az űrlap tulajdonképpen egyszerűbb, mint az állomás beszúrására szolgáló párbeszédpanel, mindössze a megfelelő hálózat IP-címét és alhálózati maszkját kell megadnunk, nevet kell adnunk az objektumnak, és szükség szerint megjegyzéssel is elláthatjuk.

Tűzfalobjektumok

Az objektumok közül messze a legbonyolultabb a tűzfalobjektum. Magukon az alapbeállításokon könnyedén túl lehet jutni, mindössze a tűzfal hálózati felületét vagy felületeit kell megadni az IP-cím és az alhálózati maszk által. A tűzfalobjektum hozzáadása és az *fwbuilder* ablakának bal oldali, a felhasználói objektumokat tartalmazó listában való megjelenése



4. kép A tűzfal tulajdonságai

után kattintsunk rá az ikonjára. Öt adatlap (fül) fog megjelenni az ablak jobb oldalán (*4. kép*).

A ***General*** (általános) lapon a tűzfalobjektum létrehozásakor megadott állomásnevet láthatjuk. Fontos, hogy a megfelelő ***Host OS*** (futtató operációs rendszer) és Platform (***célrendszer***) beállításokat megadjuk, hiszen a Firewall Builder így tudja majd kiválasztani a megfelelő fordítómotort, amikor a szabályokat az adott tűzfal számára lefordítja.

A SysInfo (***rendszerinformációk***) csak az SNMP-vel kapcsolatos beállításokat tartalmaz (lásd még a széljegyzetet). A ***Compile/Install*** (fordítás/telepítés) lap a tűzfalházirend önműködő telepítésének beállításait tartalmazza. Ha a telepítést kézzel akarjuk végezni, a lap tartalmával nem kell foglalkoznunk. Valamikor a – remélhetően nem túl távoli – jövőben a Firewall Builder képes lesz arra is, hogy önműködően, SSL felett továbbítsa és telepítse a tűzfalparancsfájlokat. Írásom elkészültekor az `fwbd` démon, amit ennek a szolgáltatásnak a használatához majd a céltűzfalon kell futtatni, még nem jelent meg.

A ***Compile/Install*** lap ***Installer*** (telepítő) beállítását az alapértelmezett `fwbd` értéken is hagyhatjuk, ekkor – a szolgáltatás támogatásának hiánya ellenére – sem fog baj történni, a lefordított tűzfalszabályokat a program a kezdőkönyvtárunkba menti. A ***Rules*** (szabályok) menü ***Install*** (Telepítés) eleme természetesen szürkén jelenik meg. Ha viszont úgy döntünk, hogy az ***Installer*** beállításnak ***Install Script*** (parancsfájl telepítése) értéket adunk, akkor a ***Policy Install Script*** (szabálytelepítő parancsfájl) mezőben saját parancsfájlunk elérési útját írhatjuk be, illetve parancssori átadott értékeket is megadhatunk hozzá. A saját parancsfájl futtatására akkor kerül sor, amikor a szabályok lefordítása után a ***Rules/Install*** (szabályok/telepítés) parancsot választjuk.

Ezzel a módszerrel kényelmesen, parancsfájlból indíthatunk például `scp`-t, ami elvégzi a szabályok másolását a céltűzfalra. Ilyen telepítő parancsfájlra példákat is találhatunk a Firewall Builder letöltések oldalán (➲ http://sourceforge.net/project/showfiles.php?group_id=5314); közülük is az `fwb_install` érdemes kiemelt figyelemre.

A telepítőbeállítástól függetlenül a Firewall Builder a lefordított parancsfájlokat egy helyi ASCII-fájlba írja, aminek egy, a tűzfalobjektuméval megegyező nevet ad, kiterjesztésnek pedig a *.fw*-t választja. Például a 4. képen is látható Trillian nevű tűzfalhoz készített parancsfájlokat ***trillian.fw*** név alatt menti.

Folytatva a tűzfalobjektum tulajdonságainak vizsgálatát: a ***Firewall*** (Tűzfal) lap szolgál a ***General lapon*** kiválasztott célrendszerre – ami ez esetben a Netfilter/IP Tables – egyedileg jellemző beállítások megadására (*5. kép*). Az alapértelmezett beállítások valószínűleg a legtöbb felhasználónak megfelelnek, néhány lehetőséget mégis érdemes áttekinteni.

A tűzfalak és az SNMP

A Firewall Builder kiterjedt Simple Network Management Protocol (SNMP) támogatással rendelkezik. Az SNMP könnyen használható eszköz az SNMP-képességekkel rendelkező hálózati eszközök és állomások beállításainak lekérdezésére, illetve beállításaik frissítésére, feltöltésére (a Firewall Builder egyébként csak lekérdezést végez). Én biztonságra törekvő környezetben soha sem szerettem az SNMP-t használni. Az SNMP-átvitelek hitelesítése közösségi karakterláncok, avagy jelszók segítségével történik, amiket a felek mindenféle titkosítás nélkül, nyílt szövegként továbbítanak. Emiatt egy általános, megosztott átviteli közeget használó – például kapcsoló nélküli ethernet vagy kábelmodemes – hálózatban nem túl bonyolult feladat az SNMP-jelszavak lehallgatása, sőt egyes esetekben ez még kapcsolóval rendelkező ethernethálózaton is megoldható. Az SNMP tehát nem megbízható hálózatokban meglehetősen kockázatos eszköz a berendezések beállításainak módosítására, és még részben megbízható hálózatokon sem feltétlenül jó választás. Ne feledjük, a hálózat biztonságát a legtöbb esetben belső személyek fenyegetik.

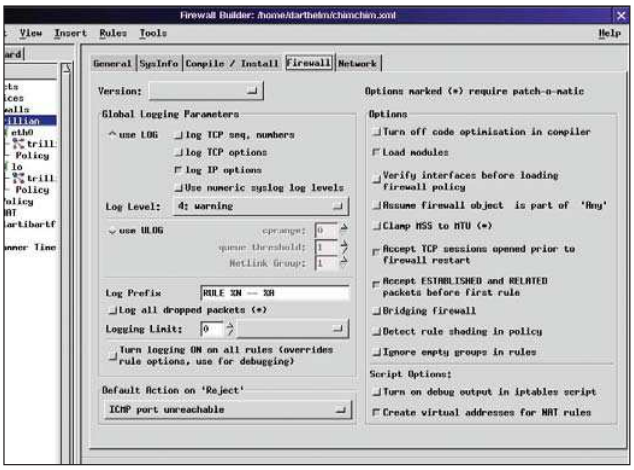
További adalék, hogy a legtöbb Linux-terjesztésben megtalálható UC-Davis SNMP-csomagban korábban jó néhány biztonsági hiányosságot találtak. Bástyagépen, tűzfalon tehát gyakorlatilag semmilyen körülmények között nem javaslom, hogy ezt az SNMP-démont – vagy bármilyen más fajtát – futtassuk. Az, hogy a Firewall Buildernek szüksége van az SNMP-könyvtárakra, nem okoz különösebb gondot, hiszen – mint már írásom elején említettem – a Firewall Buildert nem magán a tűzfalon vagy a bástyagépen kell futtatni.

Természetesen mindenki maga dönti el, hogyan és mekkora mértékben használja ki a Firewall Builder SNMP-szolgáltatásait. A hogyannal azonban – álláspontommal összhangban – én nem fogok foglalkozni.

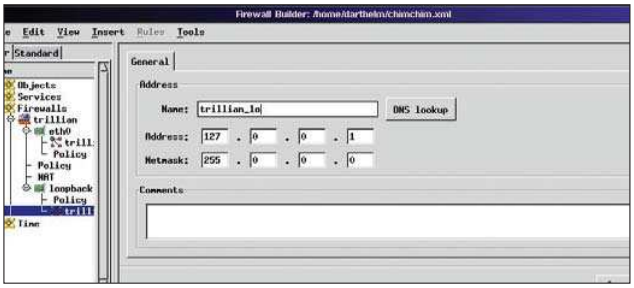
A ***Global Logging Parameters*** (általános naplózási beállítások) lap segítségével szabályozhatjuk, hogy a Firewall Builder hogyan készíti a naplóbejegyzéseket. Az alapértelmezett ***Log Level*** (naplózási szint) a ***6***-os (Info is okay). Jómagam csak az eldobott és visszautasított csomagokat naplózom, vagyis a 4-es (Warning) szintet használom.

A ***Firewall*** (Tűzfal) ablakban érdemes engedélyezni az ***Assume firewall object is part of Any*** (Feltételezze, hogy a tűzfalobjektum része a Bármely csoportnak) beállítást. A beépített ***Any*** (bármely) ***Source/Destination*** (forrás/cél) objektum esetében az alapértelmezett értelmezési mód a „bármely hálózati állomás, kivéve a tűzfalat”. A tűzfalak szabályainak létrehozásakor ez nem szokatlan elgondolás, de időnként meglepő viselkedést válthat ki. Ha például a házirend utolsó szabálya egy source=any, destination=any, service=any, action=drop és logging=on szabály, akkor nyilván azt várjuk, hogy a rendszer elhárít és naplóz minden, a tűzfalhoz való, az előbbi szabályoknak meg nem felelő kapcsolódási kísérletet. A rendszer valóban eldobja őket, de nem emiatt a szabály miatt. Elvetésükre az `INPUT` lánc alapértelmezett házirendje miatt kerül sor, amit a Firewall Builder mindig `DROP` értékre állít. Az említett szabály tehát csak a tűzfalon keresztüli kapcsolódási kísérletek esetében jut szerephez. Mivel magát a tűzfalat nem tekintjük az ***Any*** csoport részének, a fenti szabály csak a `FORWARD` láncban lép életbe, az `INPUT` vagy az `OUTPUT` láncban nem.

Az ***Assume firewall object as part of Any*** beállítás megfordítja ezt, és a fenti szabály végre úgy fog működni, ahogy azt erede-



5. kép Célrendszertől függő tűzfaltulajdonságok



6. kép A hurokfelület címének megadása

tileg szerettük és vártuk volna. Ugyanakkor figyelembe kell venni, hogy más dolgokat viszont bonyolultabbá tesz, például a tűzfal hálózati felületeire vonatkozó címhamisítás-védelmi szabályokat. Tehát: az éremnek két oldala van. Én ezt a beállítást inkább kikapcsolva hagyom. Ezután úgy alakítom a Firewall Builder-parancsfájlokat, hogy legalább az `INPUT` lánc esetében tartalmazzák a `log` és `drop` sorokat, vagy hozzáadok egy külön tűzfalbemeneti `log` és `drop` szabályt a házirendhez. Ha valami nem biztos, próbálgatni kell, majd szükség szerint javítani. Ilyenkor jó szolgálatot tehet a ***Global Logging Parameters*** rész ***Log all dropped packets*** (Naplózzon minden eldobott csomagot) beállítása, bár a működéséhez a Netfiltert a ***Patch-O-Matic Dropped Table*** folttal kell lefordítani – ha a Linux-terjesztésünk alapváltozat szerinti rendszermagját használjuk, akkor ez valószínűleg nincs benne.

A tűzfalobjektum tulajdonságait tartalmazó lapok közül a ***Network*** (***hálózat***) az utolsó. Ez a ***General*** lapon kiválasztott futtató operációs rendszerrel kapcsolatosan tartalmaz beállításokat. Az itt található beállítások a rendszermag viselkedését közvetlenül befolyásolják – ha ez valakit megrémít, nyugodtan átugorhatja ezt a lapot. Egy apróság: ha jelentős terheléssel küzdő, egy teljes hálózat védelmét biztosítani hivatott tűzfalról, és nem csupán egy megerősített, de egyszerű gépről van szó, akkor nem árt bekapcsolni a ***Packet Forwarding*** (csomagtovábbítás) beállítást.

Hurokfelületek

Bármilyen hihetetlen, de még mindig nem végeztünk a tűzfalobjektum beállításával. A 4. képet szemlélve talán feltűnt, hogy az ablaknak az objektumokat hierarchikusan megjelenítő bal oldali részében a Trillian nevű tűzfal két felülettel bír, ez az `eth0` és a `lo`. Az `eth0` hálózati csatoló alág a tűzfalobjektum beszúrásakor önműködően létrejött, ellenben a `lo` felületet – ami a Trillian hurokeszközét jeleníti meg – kézzel kellett létrehozni. Kicsit furcsa, hogy a létrehozásáról nem gondoskodik magától a program. Minden tűzfal, még a többlaki rendszerek vagy a bástyagépek esetében is szükség van olyan szabályokra, amelyek lehetővé teszik a hurokeszközök működését, és ezzel megelőzik a helyi folyamatok megszakítását.

Ha hurokfelületet akarunk létrehozni, válasszuk ki a listából a tűzfalobjektum ikonját, nyissuk meg az ***Insert*** menüt, majd válasszuk az ***Interface*** (felület) parancsot. Az ***Interface*** pont mindaddig szürke marad, amíg egy állomás- vagy tűzfalobjektumot ki nem választunk. A tűzfalobjektum alatt egy új felületikon jelenik meg, az új felület tulajdonságai pedig a jobb oldalon jelennek meg. Írjuk be a felület nevét a ***Name*** (név) mezőbe (példa: lo), majd kapcsoljuk ki a ***This Interface is External*** (külső felület) beállítást, ez ugyanis csökkentené a biztonságot. A beállítást csak külső felületek és DMZ-felületek objektumain kell engedélyezni.

Következő lépésként, miközben az új felület objektuma ki van választva, újra nyissuk meg az ***Insert*** menüt, és válasszuk az `Address` (cím) parancsot. Egy cím alobjektum bukkan fel az új felület alatt, jobb oldalon pedig a tulajdonságai jelennek meg (*6. kép*). Adjunk meg egy nevet, IP-címként 127.0.0.1-et, hálózati maszkként pedig 255.0.0.0-t (az utóbbit a program magától is beírja). Bizonyos helyzetekben a rendszer több hurokfelülettel is rendelkezhet, ilyenkor a megadott cím más is lehet (127.0.0.2 stb.). Az esetek túlnyomó részében csak egy ilyen felület van, és ennek IP-címe 127.0.0.1. Ha nem vagyunk biztosak a dolgunkban, a tűzfalgépen adjuk ki az `ifconfig` -a parancsot.

Ha az összes objektum megadásával végeztünk, vagy legalábbis eleget gyűjtöttünk össze ahhoz, hogy a szabályokat elkészíthessük, a ***File*** (fájl) menü `Save` (mentés) parancsával mentsük az objektumokat. Az alkalmazás egy fájlnevet kér, a fájlt pedig ***.xml*** kiterjesztéssel a kezdőkönyvtárunkba fogja menteni. Egyes parancsfájlok azt várják, hogy az objektumokat ***objects.xml*** névvel mentsük és a ~ könyvtárban tároljuk, de ezt módosítani lehet. Más szavakkal: úgy nevezzük el az objektumokat tartalmazó fájlt, ahogy tetszik, és oda mentjük, ahova akarjuk. A nevét és a helyét azonban ne felejtsük el, hiszen ha az `fwb_installt` akarjuk módosítani, vagy más házirend-telepítő parancsfájlt akarunk készíteni, még szükségünk lesz ezekre az adatokra.

A következő lépések a következő hónapban

Mindenki szüksége szerint, az adott környezetnek megfelelően hozzon létre további állomásokat, hálózatokat és tűzfalobjektumokat. A ***Network Range*** (hálózattartomány) és a ***Time*** (idő) objektumokról nem szóltam, ám mindkettő használatát könnyű megérteni – ha máshogy nem, kísérletezni kell velük egy kicsit, vagy bele kell olvasni a ➲ http://www.fwbuilder.org címen található leírásba. A jövő hónapban szabályokat fogunk létrehozni a már meglévő objektumok felhasználásával. Addig is már megszerzett tudásunk alapján próbáljunk tovább ismerkedni a Firewall Builderrel.

Jó szórakozást!

Linux Journal 2003. május, 109. szám

Mick Bauer (mick@visi.com)
Hálózati biztonsági tanácsadó az Upstream Solutions Inc.-nél Minneapolisban (Minnesota). Mick a szerzője a hamarosan megjelenő új O'Reilly könyvnek, amelynek címe „Building Secure With Linux”.