

IPv6-szavú Apache

Segédlet az IPv6 szerinti HTTP-kérések Apache alatti beállításához, és néhány előzetes próbaeredmény.

Az IPv6 változat (IPv6) az Internet Protokoll legfrissebb változata, amelyet az IETF tervezett az IPv4 kiváltására. E cikkben arról írunk, hogyan készíthetjük fel az Apache webkiszolgálót az IPv6 alatti futásra.

Az Apache webkiszolgáló

A Netcraft webkiszolgáló-felmérése szerint 1996 áprilisától kezdve az Internet legnépszerűbb webkiszolgálója az Apache. Jelenleg a webkiszolgálók több mint 56 százaléka Apache-ot futtat. Ezek a számok egyáltalán nem meglepőek, ismerve az Apache különféle felületek közti hordozhatóságát, megbízhatóságát, erőteljességét, beállíthatóságát, és nem utolsósorban tudván, hogy ingyenes és jó leírással rendelkezik.

Az Apache 1.3 nagy teljesítményű webkiszolgálóként már bizonyított. Csakhogy a webkiszolgálókkal szemben támasztott követelmények növekednek; a jelenkor igényei között olyan új szolgáltatások szerepelnek, mint nagyobb megbízhatóság, biztonság, teljesítmény és méretezhetőség. Válaszul az Apache folytatta előrenyomulását, és 2.0-s változatában megpróbálja kielégíteni az új elvárásokat: új és fejlettebb képességekkel felvértezett, még erőteljesebb és gyorsabb webkiszolgálót ígér.

Az Apache 2.0 számos teljesítménynövelő fejlesztést mutat be. Nekünk azonban ebben a cikkben most csak egyetlen ilyen képesség lesz fontos, mégpedig az IPv6-támogatás. A 2.0-s változattól kezdve, amennyiben az Apache-ot IPv6-támogatással rendelkező rendszeren futtatjuk, az Apache alapértelmezés szerint hallgatni fog az IPv6-foglatatokon (socket). Ezen felül a `Listen`, `NameVirtualHost` és `VirtualHost` beállítások szintén támogatják az IPv6 formátumú címeket.

A következő részekben bemutatjuk, hogyan illesszük az IPv6-támogatást a Linux-rendszerembe, majd leírjuk, miként telepítsük és futtassuk a legfrissebb Apache-változatot IPv6-támogatással. Végül lefuttatunk néhány teljesítménymérő tesztet, hogy összehasonlíthassuk ugyanannak a kiszolgálónak a teljesítményét IPv4 és IPv6 alatt.

Rendszermagszintű IPv6-támogatás

Ebben a részben röviden ismertetjük, hogyan kell engedélyezni a Linux-rendszerbe IPv6 szolgáltatását, amely előfeltétele az IPv6 HTTP-kérések fogadásának. Elképzelhető, hogy rendszerünk már eleve rendelkezik IPv6-támogatással (erről bővebben lásd még a 24–25. oldalt). A beillesztéshez először is töltsük le a legfrissebb üzembiztos Linux-rendszermagot, majd csomagoljuk ki. A kipróbáláshoz mi a 2.4.8-as változatot töltöttük le a kernel.org-ról. (A jelenlegi üzembiztos rendszermag a 2.4.20-as – a ford.)

Beállítottuk a rendszermagot, hogy támogassa az IPv6-ot. Két lehetőséget kell engedélyeznünk: az egyik a *Code Maturity Level* részben a *Prompt for development and/or incomplete code/drivers* pont. A másik a *Networking Options* részben az *IPv6 Protocol (EXPERIMENTAL)* lehetőség, amelyet engedélyeznünk kell. Ízlés szerint kiválaszthatjuk, hogy az IPv6-támogatást a rendszer-

magba vagy külön modulba szeretnénk-e helyezni. Ezt követően fordítsuk le, majd a szokásos módon telepítsük a rendszermagot és a modulokat, és a gépet IPv6-támogatással indítsuk újra.

Az Apache letöltése és telepítése

Most, hogy rendszerünk már támogatja az IPv6 protokollt, készen állunk az Apache telepítésére és IPv6-támogatású futtatására. Elsőként töltsük le a legfrissebb Apache-terjesztést a <http://www.apache.org/dist/httpd/> helyről a `/tmp` könyvtárba. E bemutató céljára az Apache 2.0.16-ot fogjuk használni; természetesen ugyanezek a lépések az újabb változatok esetében is érvényesek. Bontsuk ki a forrást:

```
# cd /tmp
# tar xzvf httpd-2_0_16-beta.tar.gz
```

ami létrehozza a forráskódot tartalmazó `httpd-2_0_16` nevű könyvtárat. Az Apache-t hozzá kell igazítanunk a saját felületünkhöz és különleges igényeinkhez. Ezt a `configure` parancsfájl segítségével tehetjük meg, amelyet a telepítés gyökérkönyvtárban találunk meg. Belépünk a `httpd-2_0_16` könyvtárba, majd a parancssorba begépeljük a `./configure` utasítást. Ha kíváncsiak vagyunk, milyen kapcsolókat adhatunk át a beállító parancsfájlnak, írjuk be a `./configure -help` utasítást. Ahhoz, hogy az alapértelmezett kapcsolók megváltoztathatók legyenek, a beállító parancsfájl különféle változókat és parancssori kapcsolókat egyaránt elfogad.

Az egyik ilyen kapcsoló az Apache telepítési helyének előtagja. Alapértelmezés szerint az Apache a `/usr/local/apache` könyvtárba kerül. Ha az Apache-t inkább a `/usr/local/apache-2_0_16` könyvtárba tennénk, akkor a következő parancsot írjuk be:

```
# ./configure --prefix=/usr/local/apache-2_0_16
```

Ugyanez vonatkozik a többi kapcsolóra is. A `configure` futás közben néhány másodperc alatt végigvizsgálja, hogy milyen képességek érhetők el a rendszerünkön, majd elkészíti a kiszolgáló felépítéséhez szükséges `Makefile`-okat. A beállítási parancsfájl lefuttatását követően a `make && make install` parancs kiadásával elkészíthetjük az Apache csomagot alkotó különböző részeket. A parancs lefordítja az Apache-ot, majd telepíti a `/usr/local/apache-2_0_16` könyvtárba. A következő lépés az Apache HTTP-kiszolgáló testreszabása a `/usr/local/apache-2_0_16/conf` könyvtár alatt található `httpd.conf` fájl szerkesztésével. Amennyiben más előtagot adtunk meg, természetesen az `előtag/conf/` könyvtárban kell keresnünk:

```
# vi előtag/conf/httpd.conf
```

A mindenképpen átszerkesztendő beállítási kapcsolók közé tartozik a `ServerAdmin`, ahová a saját levélcímünket kell beírni (így figyelmeztetést fogunk kapni arról, ha a kiszolgálónak gondolni kellene); a `ServerName`, azaz a kiszolgáló

A teljesítménymérés eredményei

	Apache 1.3.19-cel IPv4	Apache 1.3.19-cel IPv6	Apache 2.0.16-tal IPv4	Apache 2.0.16-tal IPv6
Kiszolgálóprogram	Apache/1.3.19	Apache/1.3.19	Apache/2.0	Apache/2.0
Kiszolgáló neve	node01.lmc.ericsson.se	node01.ip6.lmc.ericsson.se	node02.lmc.ericsson.se	node02.ip6.lmc.ericsson.se
Kérések száma másodpercenként	109.28	105.78	104.94	101.37
Készenléti idő	9,15 ms	9,45 ms	9,53 ms	9,87 ms
Átviteli sebesség	1275,38 KB/s fogadott	1234,50 KB/s fogadott	627,35 KB/s fogadott	605,98 KB/s fogadott

IP-je vagy neve; és a `Port`, vagyis az a kapu, amire a kiszolgálónak hallgatnia kell.

A `httpd.conf` sok magyarázatot is tartalmaz és elég könnyű olvasni, értelmezni és testreszabni. Ha mégis további részletekre vagyunk kíváncsiak a beállításokkal kapcsolatban, elolvashatjuk az Apache-kézikönyvet saját gépünk Apache telepítési könyvtárának `docs/manual/` alkönyvtárában vagy a hálózat <http://httpd.apache.org/docs> címén.

Kiszolgáló indítása

Sikeresen lefordítottuk, telepítettük és testreszabtuk Apache rendszerünket. Mint korábban említettük, az IPv6-támogatás ma már része az Apache forráskódjának, így a beindításához nincs szükség további különleges lépésekre. A kiszolgáló indítását az Apache-vezérlést egyszerű parancssoros felülettel lehetővé tevő *Apache Control Script* segítségével végezhetjük el. Ezáltal elindíthatjuk, leállíthatjuk, újraindíthatjuk a kiszolgálót, szemmel tarthatjuk az állapotát, illetve ellenőrizhetjük a beállítások írásmódjának helyességét. Ennek megfelelően a kiszolgáló indításához az `apachectl start` parancsot kell kiadnunk, és a dokumentumokat máris lekérhetjük a `http://kiszolgálónév` vagy a `http://localhost/` címen.

Dokumentumszolgáltatás IPv6 alatt

A 128 bites címek eljövételével egyidejűleg a hosszú IP-címek gépelésének kellemetlensége is megérkezik. Az IPv6-címek helyes írásmódja nyolc, kettősponttal elválasztott, négy hexadecimális karaktert tartalmazó mezőből áll, ami összesen 128 bitet jelent. Emlékezzünk rá, hogy az URL-ekben a kapuszámokat általában kettős ponttal lehet a címhez fűzni, tehát a következőképpen: 149.76.14.14:80. Minthogy az IPv6-címek eleve kettőspontot használnak magában a címben is, az IPv6 URL-ekben a címeket szögletes zárójelbe kell tenni, és csak ezt követően lehet kettős ponttal hozzáfűzni a kapu számát, valahogy így:

```
[3ffe:200:8:1000:250:bbff:fe00:25]:80.
```

Nem minden webböngésző képes értelmezni az IPv6-címeket. A Netscape Navigator 6.x és a Mozilla azonban két kipróbált böngésző, amelyek támogatják az IPv6-címzést. Úgy gondoljuk, hogy bár IPv6-támogatottságuk még nem teljesen kiforrott (hiszen néha lefagynak, és a böngészőfolyamatot ki kell lőni vagy újra kell indítani), mindkettő sikeresen képes elérni a másodneveket (aliased) gépneveket. A másodnevek olyan gépnevek, amelyeket valamilyen egyedi IP-címhez rendeltünk a `/etc/hosts` fájlban. Például másodnevet rendelhetünk egy IPv6-címhez, ha a `/etc/hosts` fájlba a következőket írjuk:

```
:::1 node02-v6-localhost
3ffe:200:8:1000:250:bbff:fe00:25 node02-v6
```

Amikor elindítjuk a Netscape-et vagy a Mozillát, címként begépeljük a megadott álnevet, majd ezekkel a másodnevekkel érjük el az IPv6 alatti weblapokat, a fenti példánál maradván például a `http://node2-v6/` vagy a `http://node2-v6-localhost/` lapokat.

Teljesítménymérési eredmények

Az Apache teljesítményének méréséhez a fenti lépéseket követve feltelepítettük a legfrissebb rendszermagot IPv6-támogatással, majd felraktuk az Apache 2.0.16-as változatát. A gép egy 256 MB memóriával rendelkező 1U Celeron 500 MHz szekrénybe szerelhető egység volt, a rendszer Red Hat 7.0 alatt működött. A teljesítményméréshez az Apache webkiszolgálóval érkező ingyenes ApacheBench programot használtuk. Két tesztet futtattunk le: Apache 2.0.16 IPv4 alatt, majd Apache 2.0.16 IPv6 alatt. Az összehasonlítás kedvéért ugyanezt a próbaprogramot lefuttattuk egy azonos beállításokkal rendelkező gépen is, amelyen Apache 1.3.19 futott, így összehasonlíthattuk a két változat IPv6-támogatásának sebességét. Hogy az Apache 1.3.19 alatt is elindíthassuk az IPv6-ot, a Kame Projekt honlapjáról letöltöttük az IPv6 Apache-foltot, majd az 1.3.19 forrásfájrára alkalmaztuk. Ezt követően lefuttattuk a `configure` parancsfájlt:

```
# ./configure --enable-rule=INET6
```

és engedélyeztük az INET6 lehetőséget. Végül elvégeztük a `make` és `make install` parancsot, ami lefordította és telepítette a Kame IPv6-támogatással kiegészített Apache 1.3.19 változatot. A táblázatban láthatjuk a próbaeredményeket. Minden teljesítménypróba 1000 darab 1-es konkurenciaszinten lefuttatott kérés eredményeként adódott. Egyetlen hibás kérelem vagy írási hiba sem jelentkezett.

Az eredménnyel kapcsolatban néhány megjegyzéssel kell élnünk. Az Apache 1.3.19 másodpercenként több kérelmet volt képes kiszolgálni, mint az Apache 2.0.16. Az Apache általunk kipróbált változatában IPv6 alatt kevesebb kérelmet értünk el másodpercenként, mint az IPv4 használatakor; ennek egyik oka az lehet, hogy az Apache-hoz adott IPv6-kód nincsen módosítva végigellenőrizve és javítva. Másrészt, ha az Apache 1.3.19 átviteli sebességét tekintjük, azt tapasztalhatjuk, hogy jelentősen magasabb a Apache 2.0.16 teljesítményénél. Ez még további vizsgálatokat igényel.

Az Ericsson Open Architecture Research Labnál mostanában építettünk fel egy teljesítménymérő környezetet, amelyben Linux-telepek és alkalmazáskiszolgálók képességeit vizsgálhatjuk (ide értve az Apache, a Tomcat és a Jigsaw webkiszolgálókat). Csakhogy a környezet és a próbák IPv4-hez, és nem IPv6-hoz készültek. Azt tervezzük, hogy a környezetet IPv6 HTTP-kérelmek kipróbálására építjük át, hogy az Apache (és más webkiszolgálók) teljesítményét komoly IPv6-terhelés

mellett is meghatározhatjuk. Jelenleg összesen 100 darab 1U méretű egységgel rendelkezünk (Celeron 500 és Pentium III-as gépek vegyesen, 256 MB és 512 MB memóriával), amelyek hamarosan egyetlen csomópontra menő IPv6-forgalmat hoznak majd létre, ahol összegyűjtjük és megszerkesztjük az eredményeket. Amint befejeztük a munkát, az eredményeket közzétesszük. Ez az összegzés már sokkal többet fog mondani, mint a cikkben szerepeltetett előzetes próbák eredményei.

Összegzés

Az IPv6 kezd valósággá válni. A következő néhány évben azonban alkalmazáskiszolgálóinkon képesnek kell lennünk az IPv4 és az IPv6 protokollok együttes támogatására, még mielőtt a teljes IPv6-átállás megtörténhetne. Az Apache fejlesztői ügyeltek erre, és az IPv6-támogatást már be is építették az Apache kódjába. Előzőleg az IPv6 csak letölthető folt formájában volt elérhető (többek között a Kame Projekt honlapjáról),

amit aztán a forrásfába kellett illeszteni.

Mint azt a teszteredmények sugallják, az IPv6 alatti dokumentumszolgáltatás valamivel lassabb lesz (lekérés/másodperc értelemben), mint amilyen az IPv4 alatt volt; ami érthető is, hiszen az IPv6-támogatás még gyerekcipőben jár. A fejlesztés előrehaladtával az Apache is valószínűleg eléri, majd meghaladja az IPv4 alatt elért teljesítményt, hogy továbbra is a világ legkedveltebb webkiszolgálója maradjon.

Köszönetnyilvánítás

Köszönet az Ericsson Researchnek (Open Architecture Research Lab) linuxos és nyílt forrású munkáink támogatásáért, illetve hogy engedélyezték a cikk megjelenését.

Linux Journal 2003. január, 105. szám



Ibrahim Haddad (Ibrahim.Haddad@Ericsson.com)
A montreali Ericsson Research Open Architecture Lab kutatója, ahol elsődleges feladata a hírközlési osztályú kiszolgáló csomópontok kutatása valós idejű, teljesen IP alapú hálózatok területén. Jelenleg DrSc-jelölt a Concordia Egyetemen.



David Gordon (davidgordonca@yahoo.ca)
Most fejezte be együttműködését a Kanadai Ericsson Researchs-szel, ahol a közel telekommunikációs szintű Linux-telepek IPv6-támogatásán dolgozó csapat tagja volt. Jelenleg Sherbrooke-i egyetem informatikus hallgatója.

KAPCSOLÓDÓ CÍMEK

Apache webkiszolgáló ➔ <http://www.apache.org>
 ARIES Projekt ➔ <http://www.risq.ericsson.ca>
 Kame Projekt ➔ <http://www.kame.net>
 Linux-rendszermag ➔ <http://www.kernel.org>
 Mozilla ➔ <http://www.mozilla.org>
 Netscape ➔ <http://www.netscape.com>
 Red Hat ➔ <http://www.redhat.com>

Ismét maratoni hétvége – Február 7–9.

Ahogy azt már két évvel ezelőtt is írtam, a linuxos társadalom számára az egyik legfontosabb feladat az, hogy a rendszer magyartítását, honosítását elvégezzük. Szerecsére nagy örömmel mondhatjuk, hogy a munka dandárját már elvégeztük, sőt nagyon sok rendszer rendelkezik magyar felülettel és leírással. A rendkívül sok emberórát igénybe vevő folyamat szerecsére a mai napig folytatódik és erősödik. Az egyik legtöbbet „szem előtt lévő” program, az OpenOffice.org irodacsomag magyartítása nagymértékben **Somogyi Péter** szervezőmunkájának köszönhető, aki több segítővel tavaly áprilisban maratoni „honosítási hétvégét” szervezett. Ma már mindenki könnyedén használhatja az elkészült rendszert, akár *.deb* csomagból is.

Sajnos a munkát tavaly nem tudtuk befejezni, a programban még pontatlan vagy részleges fordításokkal találkozhatunk, magyar sűgó pedig még most sem tartozik hozzá. Hogy az eltelt idő alatt összegyűlt hibákat

orvosolni tudjuk, Péter ismét egy maratoni hétvégét szervez. A tervek szerint húsz–huszonöt ember gyűlik majd össze egy erre megfelelő helyen, és a tavaly bevált rendszert is üzembe helyezve (amelyen keresztül külső

segítők is bekapcsolódhatnak a munkába) a hétvége alatt az egész fordítást elkészítik. Ez első hallásra kivitelenül feladat, a tavalyi tapasztalatok mégis azt mutatják, hogy a csapatnak ismét nagy esélye van megmutatni: a nyílt rendszerek élnek! Amennyiben kedvet és erőt érzel, hogy részt vegyél a hétvégén, vagy bármilyen módon támogatni tudod a csapatot (például

ellátmánnyal, helyszín biztosításával), írd közvetlenül Péternek, vagy akár nekem! Várunk!

Kapcsolódó cím

OpenOffice.org magyarul ➔ <http://office.fsf.hu>

Szy György (Szy.Gyorgy@linuxvilag.hu)

