

Miért Linux?

Reuven visszatér az alapokhoz, és áttekintést ad a Linux üzleti világban való fejlődéséről.

Bontsunk pezsgőt! A Linux Journal ebben a hónapban jelenik meg századszorra. Ennek öröme egy kicsit félretereszem szokásos nyílt forrású webes-adatbázisos témáimat és csatlakozom az ünneplőkhöz.

Számos okuk van a Linux felhasználóinak (és általában a nyílt forrású programok híveinek) az öröme. A csúcstechnológiára épülő gazdaság ugyan visszaesett, a nyílt forrású programok fejlesztése azonban továbbra is nagy lépésekkel halad előre. Amikor a Linux Journal első száma megjelent, még kevés ember hallott a finn egyetemista által írott rendszermagról, amely komoly szerepet játszott a Linux és a szabad programok sikerében.

Manapság sokan tudnak a Linuxról, még ha nem is értik meg, hogy mi az és mire jó. Sok ügyfelem tudja, hogy a nyílt forrású megoldásokat részesítem előnyben, és mindig kíváncsiak arra, hogy miért van ez így, és ami még fontosabb, miért érdekük nekik is, hogy ilyen megoldást válasszanak. Vállalva annak a kockázatát, hogy a már hívó emberek között tartsak hittérítést, ebben a hónapban áttekintem azokat az okokat, amiért a Linux kiváló választás a kiszolgálóoldali alkalmazások futtatásánál. Remélem, néhány itt leírt ötlet a jövőben segíteni fog a szabad programokon alapuló megoldások népszerűsítésében munkatársaid és ügyfeleid körében.

Ár és megbízhatóság

A megszállott szakembereket a módszerek és az eszközök érdeklik, amelyek új tudáshoz és új távlatokhoz vezetnek. A való világban viszont az embereket az érdekli, hogy munkájukat a lehető leggyorsabban és legolcsóbban végezzék el. A program csak eszköz a cél eléréséhez, nem maga a cél. Emiatt úgy vettem észre, hogy a legjobban úgy adható el az embereknek a nyílt forrású program, ha azt mondjuk: kevesebbe kerül és többet tud. Egyik tényező sem elég önmagában – könnyen találhatunk drága, jó minőségű programokat, valamint telepítésre sem méltó, ócska és olcsó programokat. A fogyasztók, és így az ügyfeleim is, mindig szívesen szereznek többet kevesebbért, és emiatt a szabad programok tetszenek nekik.

Amikor az ügyfélnek vázolom a megoldást, elkezdem elmagyarázni, hogy olyasmit ajánlok, amit ők lehetetlennek hittek: a szabad és olcsó programok pontosan azt teszik, amit elvárnak tőlük, ráadásul le sem fagynak. Amikor elmondom egy Windows-felhasználónak, hogy még sohasem láttam Linuxot lefagyni hat év és több tucat rendszer után, meglepődnek és hitetlenkednek. Ha elmondom azt is, hogy ez a program szabadon letölthető az Internetről, még nehezebben hiszik el. Ügyfeleim gyakran érdeklődnek, hogy ki támogatja a programot, mi lesz, ha valami elromlik. Megkönnyebbülve veszik tudomásul, hogy nemcsak én ajánlom fel a szükséges támogatást, hanem más is képes erre, ha az én munkámmal nem lennének elégedettek. Ez természetesen szöges ellentétben áll számos tanácsadó cég programtelepítésekkel kapcsolatos viselkedésével és megszorításaival. A nyílt forrású megközelítés

ezért felhasználóbarátabb, mint a hagyományos programfejlesztési modellek: csökkenti a költségeket és erősíti a versenyt.

Természetesen nem minden szabad program jó minőségű, és nem minden tanácsadó tudja, hogy mit kell tenni. A közösségi fejlesztés kitűnő eredményeket hozhat, de ez nem azt jelenti, hogy minden Interneten megjelenő program biztonságos és megbízható. Valójában teljesen világos, hogy sok programot, beleértve egy-két népszerűt is, egyáltalán nem próbáltak ki a kiadás előtt. Az ilyen programok az egész nyílt forrású közösséget lejáratták, és több kárt okoznak, mint amennyi hasznot hajtanak. Évente többször előfordul, hogy felhív az ügyfél, hogy ki kellene javítani egy programot, amelyet letöltöttek, és első ránézésre működött is, de végül kiderült, hogy nem biztonságos, nem megbízható, és tele van hibákkal.

A hibák javítása

Még ha úgy is találsz, hogy a kiszolgáló hibás és nem biztonságos nyílt forrású alkalmazásoktól függ, nincs minden veszve. A szabad programok természetéből következően adott a lehetőség, hogy igényeid szerint módosítsd vagy javítsd, ha valami nem tetszik. A megosztott forrású programok felhasználói szerződéseinek kitalálói nem értik a lényegét, amikor megengedik a forráskód megtekintését, csakhogy nem engedik meg a módosítását vagy a javítását. Ha veszek egy házat vagy egy autót, akkor tökéletesíthetem és meg is javíthatom, ha akarom; miért lenne ez másképp a programokkal?

Az igaz, hogy a megosztott forráskód lehetővé teszi a kód tanulmányozását, így a megbízhatósági és a biztonsági hibák gyorsabban előkerülnek és kijavíthatók. A forráskód olvasásának lehetősége azonban közel sem olyan fontos, mint a továbbfejlesztés lehetősége. Ráadásul a továbbfejlesztett változat visszajuttatása a közösségi változatba azt eredményezi, hogy mindenki hasznosítja az egyén javítását, és további fejlesztésre nyílik lehetőség. Ezért a közösségi munkában való részvétel minden nyílt forrású program felhasználójának érdeke; ez nem egyszerűen csak szép dolog.

Mivel általában kiforrott eszközöket használok – például Linuxot, Apache-ot, Perl-t és Python-t –, viszonylag ritka, hogy hibát találjak a letöltött programokban. Évente többször is előfordul azonban, hogy az általam használt programokban valami hiányosságot fedezek fel. Mivel hozzáférék a forráskódhoz, a lehető leggyorsabban meg tudom változtatni, és ez azt is jelenti, hogy ezzel a gonddal másoknak már nem kell megbirkózniuk.

Érdekes, hogy ez az érv még ma is használható, tekintve, hogy a nyomtatómeghajtókkal kapcsolatos hasonló gondok indították arra *Richard Stallman*-t, hogy megalapítsa a Free Software Foundation-t, amely a GNU-projektet döntő szerepet játszott a GNU/Linux és a szabad program sikerében. Az is csodálatos, milyen hamar hozzászórtunk, hogy a forráskód



elérhető, és a számítógépes rendszer minden pontját tetszés szerint megvizsgálhatjuk vagy megváltoztathatjuk. Hasonlóképpen a Linux-terjesztések is egyre inkább teljes rendszerré válnak. Nemrég kezdtem el dolgozni egy projekten, amelynek Solarison kell futnia, és rögtön rájöttem, hogy mennyivel gazdagabb és jobban felépített egy átlagos Linux-terjesztés, mint például a Solaris alapkiépítésben. Fél napig is eltart, amíg letöltöm és telepítem az összes szükséges programot, mint a gcc, a Perl, a Python stb. Az elmúlt években hozzászoktam, hogy minden környezetemben levő gépen ott van a gcc, ezért úgy éreztem magamat, mint akit visszadobtak a Unix sötét középkorába.

Nincsenek titkok

A mérnökök nem tudnak titkot tartani, erre *Scott Adams* is rámutatott a Dilbert-képregényekben. Valóban, az egyik dolog, amit nagyon vonzónak találok a nyílt forrású programokban, hogy nincsenek titkok. Az ügyfelek azért bérelnek fel engem, mert időt akarnak megtakarítani maguknak, vagy mert nem értenek egy bizonyos területhez, de nem azért, mert rákényszerülnek. Ebből az okból kifolyólag munkámat az ügyvédi vagy könyvelői munkához hasonlítom tekintem, hiszen tanácsot és megoldásokat ajánlok fel nyilvánosan elérhető tudásanyag alapján.

A „nincs titok” eszmeisége jól működik az ügyfeleimnél, még azoknál is, akiket egyáltalán nem érdekel a programok működése. Tudják, hogy kérdezhetnek, és én a legjobb tudásom szerint válaszolok, nem kell marketingrizzsával, kötelező frissítésekkel és kétértelmű beszéddel fárasztanom őket. Műszaki érdeklődésű ügyfeleim élvezik, hogy ha akarnak, belenyúlhatnak a kódba és elolvashatják a leírást; csak azért nem tudnak annyit, amennyit én tudok, mert nincs hozzá idejük és elég tapasztalatuk.

A nem haszonelvű ügyfelek sok tekintetben tökéletes célpontjai a nyílt forrású programoknak, és sokszor fellelkesednek, ha ilyen eszközöket használhatnak. Különösen az oktatási intézményeknek tetszik a tudás megosztásának és a közösségi munkának az eszméje. Hathatós érv, ha elmondjuk nekik, hogy pénzt takaríthatnak meg, és a hasonló gondolkozású emberek közösségébe tartozhatnak. Ráadásul a nem haszonelvű szervezetek nem érzik fontosnak, hogy a változtatásokat a cégen belül tartsák, így könnyebben részt vehetnek a nyílt forrású közösség munkájában.

Jó minőségű eszközök

Amikor cikkeket kezdtem írni a Linux Journalba, a legtöbb kiszolgálóoldali alkalmazás kézzel írott CGI-program volt. Rengeteg webhely használ még ma is ilyen programokat. De ahogy a web egyre bonyolultabbá vált, az emberek olyan eszközöket kezdtek igényelni, amelyekkel könnyen és gyorsan lehet jó minőségű és méretezhető webalkalmazásokat készíteni. Nem túl meglepő módon seregnyi zárt forrású kereskedelmi program jelent meg, hogy kielégítse ezeket az igényeket. Az már viszont felháborító, hogy mennyi pénzt kérnek el ezekért a programokért – ráadásul kötelező szerződést kötni velük, hogy az ő tanácsadóikat bérelj fel a testreszabáshoz. Szerencsére a nyílt forrású világ válaszolt a kihívásra. Számos nyílt forrású eszköztár áll rendelkezésre bonyolult kiszolgálóoldali alkalmazások készítéséhez. A Zope, mint az elmúlt hónapokban láttuk, egy rendkívüli (bár bonyolult) alkalmazás-kiszolgáló, amelynek segítségével adatbázisokhoz és más adatforrásokhoz kapcsolódni képes webalkalmazásokat készíthetünk. A következő hónapban megnézzük az OpenACS-t,

amellyel az online közösségi rendszerek létrehozása és módosítása válik egyszerűbbé. Továbbá az Apache Software Foundation által támogatott `mod_perl`, Mason és számos Javához és XML-hez kapcsolódó eszköz egyre inkább azt a kedvezőtlen folyamatot gerjeszti, hogy a megfelelő eszköz kiválasztása a telepítéssel és a használatával összemérhető nehézségű feladattá válik.

Bármily csodálatosak is legyenek ezek az eszközök, ne feledjük, hogy nem mindenkit lehet a mi oldalunkra állítani. A legdurvább élményem ezzel kapcsolatban az, hogy tavaly egy leendő ügyfél nem engem választott egy egyszerű tartalomkezelő rendszer megvalósítására, amely a termékeit jelenítette volna meg a weben. Megtudtam, hogy az ajánlatom 800 000 dollárral volt kevesebb, mint a legközelebb álló versenytársamé. Mivel azonban én nyílt forráskódra építettem, a versenytárs neve viszont ismert volt tartalomszolgáltatás világában, vesztettem (az ügyfél azóta sorozatos elbocsátásokra kényszerült, negyedéves eredményei egyre romlottak, és weboldalát még mindig kézzel rakják össze, így legalább némi erkölcsi győzelmet elkönnyelhetek).

Azt is meg kell jegyeznünk, hogy a nyílt forrású világ nem minden résztvevője tartja be a közösségnek tett ígéreteit. A nyílt forrás számos támogatója meglepődött és csalódott, amikor a Lutris tavaly „bezárta” az addig nyílt forrású Enhydra Enterprise Java alkalmazáskiszolgáló kódját. Szerencsére vannak helyettesítő termékek, nemcsak a GPL-es JBoss alkalmazáskiszolgáló tett szert óriási népszerűsége az elmúlt év során, hanem nemrég a Sun is bejelentette, hogy a nem üzleti, nyílt forrású J2EE-megvalósítások is hivatalos tanúsítványt kaphatnak a következő hónapokban. Ez csökkentheti néhány üzleti szereplő idegenkedését a nyílt forrású programokkal szemben.

Még ha érnek is kudarcok, ne hagyj magadat becsapni: az IBM, a HP és még a Sun is elismeri ma már, hogy a Linuxot és a nyílt forrású programokat komolyan kell venni, mert sokoldalúak és megbízhatóak. A „világuralom” ideje még nem jött el, de a márkanévet már elismerik, a pénzügyi tényezők és az akadémiai és üzleti világból érkező elismerés segít minket a továbblépésben.

Boldog születésnapot!

Írásomat azzal fejezem be, hogy az újság létrejöttét lehetővé tevő keményen dolgozó munkatársakat elismerés és nagyrabecsülés illeti. A Linux Journalt egy évvel azelőtt kezdtem olvasni, hogy az első cikkem megjelent benne, és azóta is minden hónapban olvasom. A cikkek a világ legalkotóbb programozóinak sokféle érdeklődési területét mutatják be. A tény, hogy a Linux Journal a 100. számához érkezett, minden kétség nélkül azt jelenti, hogy a Linuxnak és a nyílt forrású programoknak helyük van ezen a világon. Remélem a 200. számba is írhatok egy hasonló cikket!

Linux Journal 2002. augusztus, 100. szám



Reuven M. Lerner

(reuven@lerner.co.il) kisebb webes és internetes módszerekkel foglalkozó tanácsadó cég tulajdonosa és vezetője. A cikk megjelenésének időpontjában valószínűleg már végleg elkészült Core Perl című könyvével, melyet idén jelentet meg a Prentice-Hall. Az ATF honlapon érhető el (☞ <http://www.lerner.co.il/atf/>).