

Appliance a hordozható Linux



Micsoda hülye cím, semmi új nincs benne – mondhatnátok, hiszen a Linux live-változatok száma szinte kimerítetlen. Vannak olyan disztrók, amikben külön program segíti a felhasználót, hogy saját live rendszert készítsen. Magam a már beállított telepítéseimről szoktam ilyen iso-t készíteni, hogy bármikor, különösebb erőfeszítések nélkül visszatérhessek a kiinduló pontra. Már többször használt vettem az ilyen mentésnek.

Nem, most nem erről van szó, hanem a Linuxon belüli hordozhatóságról. Amilyen megnyerő a Linux sokrétűsége és változatossága, annyira meg is keserítheti a felhasználók életét a sokféle csomagkezelő és a rendszercsomagok kavalkádja. Amikor disztribúciót választasz észre sem veszed és kiszolgáltatottá válsz az adott fejlesztői gárda aktivitásának, érdeklődési körének. Tőlük függ, hogy mikor, milyen programokat telepíthetsz, vagy használhatsz. Amíg a szokásos feladatokról van szó, nincs gáz. De ahogy valami különlegeset, vagy nem elterjedten használt programra van szükséged, vagy csak az adott disztribúcióban nem használják, meg vagy lőve. Bár feltehetően tudjátok mire gondolok leírom a saját tapasztalataimat és miért örültem meg az Appliance-nak..

Régebben, amikor hivatalban dolgoztam Windows-hoz voltam kötve, MS Office-t kellett használnom, noha voltak más, szerintem bizonyos területeken fejlettebb programok is. Nem mondom, hogy a MS Word pocsék lenne, sőt. Ám voltak olyan területek, amiken más szövegszerkesztők sokkal jobban teljesítettek. **Word Perfect**: bele lehetett nézni a szerkesztett állomány forrásába és a fölösleges formázási utasításokat törölni. Ugyanazt az eredményt sokkal kisebb fájl méret mellett produkálta. **WordPro**: a formázási segédeszközök mindig kéznél voltak és gyorsan lehetett használni. A szövegszerkesztők csak mostanság kezdik átvenni a koncepcióját. **OpenOffice** (LibreOffice): nemcsak, hogy bármilyen nyelvhez ingyen lehetett telepíteni nyelvi készletet, hanem az adatbázis programokból közvetlen elérhetőségével messze lekörözte a Word-öt. Sokkal könnyebben lehetett sablonokat készíteni és alkalmazni, körleveleket készíteni.



És akkor még csak a szövegszerkesztőkről beszélek, miközben ott vannak (voltak) az **alternatív fájlkezelők, képszerkesztők, webböngészők**, stb. Szóval kellett programok, de érthető okokból, mint egytollú felhasználó nem telepíthettem hivatali gépre. Lehetett ugyan különféle fondorlatokkal nem hivatalos programokat futtatni, de az eredmény mindig kétséges volt. Ekkor megjelentek az ú.n. **hordozható alkalmazások**. Használhattam kedvenc programjaimat a „gyári” helyett. OpenOffice, Total Commander, Firefox, Xnview, VLC és ki emlékszik még miket. A telepített Windows-os rendszeremben a mai napig is így használom a programok jó részét, pedig én vagyok a rendszerem gazdája.

Hogy jön ez ide, a Linux csomagkáoszához? Ha már van, szeretem használni a számítógépet és nem csak böngészésre, levelezésre, vagy írásra, hanem csomó másra is. Vannak kedvenc programjaim, amik nem, vagy éppen nem a megfelelő verzióban találhatóak az általam alapvetően használt disztribúciókban.

Ha a Linuxban valamely program a disztró saját tárolóiban nem található, akkor a felhasználó három dolgot tehet(ett eddig): keres olyan disztrót, amiben elérhető, amit keres, telepíti forrásból a programot, vagy használja a Windows-os hordozható változatot, ha van (na ez azért égő). Szívesem csücske a Scribus. Szoktam használni, de általában csak a stabil 1.4.x változat érhető el. Egyszer sikerült az 1.5svn telepítenem PCLinuxOS-re, ami viszont működésében annyival jobbnak bizonyult, hogy még a gyermekbetegségeit is szívesen elviselem. Sajnos az újabb változatnak telepítése azóta sem sikerült, így egyrészt használtam a saját Live-ként mentett rendszeremben, később telepítettem az OpenSuse-t, amire volt kész csomag. Pechemre kijött az új fejlesztői verzió, ami sok javítást tartalmazott. Forrásból nem tudtam telepíteni és OpenSuse-re sem találtam csomagot. Maradt volna a hordozható Windows-os változat wine-nal, de az nem nagyon akart működni régebben, így nem is próbálkoztam vele. Ekkor, a keresgélés közben akadt meg a szemem az [1.5.2 AppImage-en](#). Mivel nem tudtam, hogy eszik-e, vagy isszák, hát [rákerestem a neten](#), mi a franc ez. Azt mondja, hogy mindenhol fut? Hiszem, ha látom! És tényleg... Előbb OpenSuse alatt, majd PCLinuxOS alatt is kipróbáltam. Megy. Azóta is ezt használom.

Szóval AppImage

Amit jelenlegi csomagkezelési mód mellett Linux-ban nem, vagy csak nagyon nehezen lehet megcsinálni, hogy a felhasználó a legfrissebb programváltozatot letöltse és használja esetleg egy másik verzió mellett; tesztelni fejlesztés alatt lévő szoftvert úgy, hogy ne kelljen telepíteni és ezzel a működő rendszer integritását veszélyeztetni; fejlesztőként kiadni programot úgy, hogy ne kelljen a különféle disztribúciók lelki világával foglalkozni.

A probléma abban gyökerezik, hogy amíg Windows és OS X platformként működik, amin lehet programokat futtatni, addig a Linux rendszer, amibe programokat lehet integrálni.

A [honlap](#) szerint Linus Torvalds a 2014-es [DebConf 14 vita részében](#) (05:40-nél) a következőket mondta a csomagkezelés, programtelepítés kérdéséről (nem szó szerinti és nem teljes fordítás):

"..miközben Windows és OS X alá bináris telepítő fájlok készülnek, addig a Linux-ra nem. Miért? Mert Linux asztali alkalmazásokhoz binárist készíteni nagy sz...ás.Készülnek binárisok Fedora 19 és 20-hoz, esetleg még RHEL 5 alá 10 éve, de Debian stabil alá nem, mert a Debian lib fájljai olyan öregek, hogy nem tudja futtatni azt, ami nem a múlt században készült. Ugyanakkor a fejlesztő verzióra, bármi is legyen a neve, készülnek binárisok. Ez is nagyon nehéz, mivel a Debian is követi a szabályt, hogy megosztott könyvtárakat kell használni..."

Továbbiakat [itt](#) olvashatsz.

Noha dolgoznak sokkal [kifinomultabb megközelítéseken](#) is, itt van ez, ami elég egyszerű, általános és megbízhatóan működik.

Célok

Az AppImage formátumot az alábbi célok figyelembe vételével hozták létre.

1. **Legyen egyszerű.** Könnyen érthető, elkészíthető és kezelhető legyen;

2. **Kompatibilitás fenntartása bináris szinten.** Az Applmage bináris szoftver-terjesztési formátum. Az így létrehozott programcsomagok lehetőleg minél több rendszerrel legyenek kompatibilisek. Minimálisra kell csökkenteni az újrafordítás szükségességét.
3. **Legyen disztribúció-független.** Egy Applmage fusson az összes alap operációs rendszeren (disztribúción). Például, egyszerre kell tudni Ubuntu 9.10, openSUSE 11.2 és Fedora 13 (illetve későbbi verziói) alá készíteni, anélkül hogy az egyes célrendszerekre külön kellene készíteni és karbantartani csomagokat.
4. **Ne legyen szükség telepítésre.** Applmages az alkalmazást olyan formában tartalmazza, hogy közvetlenül az archívumból lehessen futtatni, megelőző telepítés igénye nélkül. Hasonlóképpen mint a Live CD-kénél. A Live CD-k megjelenése előtt az operációs rendszert használatba vétel előtt telepíteni kellett.
5. **Az applikáció tömörítve maradjon.** Mivel az alkalmazás végig becsomagolva marad, sosem bontja ki a merevlemezre. A számítógép az elérésekor menetből (on-the-fly) csomagolja ki, amivel sebességnövekedés is elérhető a helymegtakarítás mellett. Csakúgy, a teljes eltávolítása is gyors.
6. **Az alkalmazások bárhová helyezhetők legyenek.** Az Applmage-ek „áthelyezhetőek”, azaz a felhasználó bárhol futtathatja (közte CD-ROM-ról, DVD-ről, külső merevlemezről, USB-kulcsról).
7. **Az alkalmazás csak-olvasható legyen.** Ha az Applmage-ek kialakításuk következtében csak olvashatóak, a felhasználó biztos lehet abban, hogy az alkalmazás nem módosítja önmagát a működés során.
8. **Ne igényeljen újrafordítást.** Az Applmage-eket lehetőleg már létező binárisokból kell készíteni, újrafordítás igénye nélkül. Ez nagy mértékben felgyorsítja az Applmage készítését, mivel fordító programra nincs szükség. Ezzel lehetővé válik harmadik félnek, hogy zárt forráskódú alkalmazás csomagoljon Applmage-ként.
9. **Az adott operációs rendszert hagyja érintetlenül.** Az Applmage-eket szimpla rendszeren futásra tervezik, amiket rendszergazdának nem kell semmilyen módon előkészíteni. Az Applmage ne függjön kernel-folttól, modultól, vagy bármilyen más alkalmazástól, amivel a cél disztribúció alapbeállításban nem rendelkezik
10. **Ne igényeljen rendszergazda jogokat.** Mivel az Applmage-eket felhasználó általi futtatásra tervezik, nem kérhetnek rendszergazda (root) hozzáférést telepítéshez, vagy futtatáshoz. Ugyanakkor rendszergazda képes legyen azt telepíteni (pl. többfelhasználós környezetben), ha arra lenne szükség.

Az Applmage formátumának alapelve az **egy alkalmazás = egy fájl**. Az Applmage egy alkalmazást és a futásához szükséges összes fájlt tartalmazza. Más szóval, az Applmage-nak csak olyan függőségei vannak, amiket a célzott rendszer(ek) alapból biztosítanak. Noha elméletileg lehetne belőle rpm, vagy deb csomagot készíteni, szinte soha nem csinálnak. Éppen ellenkezőleg, kimondottan nem javasolt és ez az Applmage-ek szokásos használatának módja esetén igaz.

Röviden, az Applmage az a programoknak, mint a Live CD az operációs rendszereknek.

AppImage formátum – áttekintés

Az AppImage egy zisofs tömörítésű ISO 9660-as fájl, egy minimális AppDir-t tartalmazóan (egy könyvtár benne az alkalmazás és a futásához szükséges, a célzott operációs alap rendszerben nem található fájlokkal) és egy kis, fejlécben beágyazott futtatható programmal. Tehát az AppImage egyszerre egy ISO 9660-as fájl (csatolható és megtekinthető) és egy ELF végrehajtható (mai futtatható).

Amikor végrehajtasz egy AppImage-t, a beágyazott kis program csatolja az ISO-fájlt és végrehajtja a tartalmazott applikációt.

Egy AppImage legalább az következőképp néz ki:

ELF	ISO9660 zisofs csomagolt adat benne
beágyazott	AppRun
ISO9660	.DirIcon
fejléc	pár AppFájl

- Az AppRun az AppImage futtatásakor végrehajtott bináris.
- A .DirIcon egy svg ikon, amit az AppImage használ.
- Pár AppFájl tetszőleges, az alkalmazás futásához szükséges fájl.

Ugyanakkor az automatikus generáláshoz, feldolgozáshoz és a több metaadat érdekében az AppImage formátum ennél egy kicsit kidolgozottabb konvenciót követ:

ELF	ISO9660 zisofs csomagolt adat benne
beágyazott	AppRun
ISO9660	appneve.desktop
fejléc	usr/share/appneve
	usr/share/icons/ /iconneve.svg
	usr/share/appneve/valamisúgó fájl
	.DirIcon

Az ISO 9660-as fejlécben beágyazott ELF mindig az AppRun fájlt hajtja végre, ami az ISO9660-as fájlban található.

Az ISO9660-as fájlban belüli az AppRun történetesen nem végrehajtható, hanem egy kis kisegítő bináris, ami megkeresi és futtatja a tényleges alkalmazást. Az általános AppRun fájlokat a bash és a C vezette be az AppImageKit részeként. Az AppRun C-verziója az AppRun.c a preferált, mivel gyorsabb és hordozhatóbb.

Az AppRun általában nem tartalmaz fixen bekódolt információt az alkalmazásról, hanem az appneve.desktop fájlból veszi a Desktop fájl-specifikációból.

Egy minimális tartalmú appneve.desktop fájl, ami az AppImage futásához elegendő ezt tartalmazza:

```
[Desktop Entry]
Name=AppNeve
Exec=appneve
Icon=ikonneve
```

stb. stb. Akit érdekel, olvasson utána. Szerintem az általános bemutatáshoz ennyi éppen több, mint sok.

Akkor a lényeg, jelenleg milyen [programokat lehet AppImage formában](#) elérni? Lásd a hivatkozást.

A lap alján azt írja, hogy ezzel a paranccsal lehet frissíteni a listát:

```
wget "https://github.com/probonopd/AppImages/tree/master/recipes" -O -
| grep "master/recipes" | cut -d '"' -f 10 | sed -e 's|.| * **\0**'
[Recipe] (https://github.com/probonopd/AppImages/blob/master/recipes/\0/Recipe) \
[Download] (https://bintray.com/probono/AppImages/) | g'
```

Úgy vélem, ahogy a felhasználók kezdik majd megismerni ezt a csomagfajta egyre inkább és gyorsuló tempóban növekedni fog az elérhető programok száma. Legalábbis remélem.

Jó szórakozást! Remélem valaki csak talál magának olyasmit, amit használni tud.

Ja, és még valami. Van egy AppImageKit nevezetű valami – a szövegben feljebb is esett említés róla –, amit ilyen hordozható alkalmazások csomagolásához találtak ki. Akit érdekel [itt bővebben](#) olvashat róla.

Bizonyára a Linux-zsenik megvetően viszonyulnak majd ehhez az újításhoz, hiszen alkalmas arra, hogy „Linux-zombikat” termeljen ki, akik csak arra képesek, hogy kettőt hármat kattintsanak, ugyanakkor fogalmuk sincs, mi történik a gépben. Szerintem meg, ha megkönnyíti a Linux használatát, akkor javítja az alternatív operációs rendszer versenyfeltételeit a Windows-szal és az OS X-szel szemben. Az elterjedtség mindannyiunk érdeke, mert igazi versenyre szorítja az óriásokat.

Még két megjegyzés a végére.

Belegondolva, előbb-utóbb megjelenhetnek programok mindenféle adathordozón, amiket bármely, szokásos operációs rendszerrel működő számítógépbe bedugva az annak a rendszernek megfelelő változatot indítják el. A kedvenc programjaimat egy-egy, vagy egyetlen USB-kulcsra vihetem magammal és bármely gépen használhatom, otthon vagyok velem. Ez attól lenne jobb a szokásos Live-rendszereknél, hogy nem kellene a boot-olási dolgokkal foglalkozni lévén ez egy sima adathordozó.

A cikk PDF-változatát is Scribus 1.5.2 AppImage-t futtatva készítettem.