



Full Circle

AZ UBUNTU LINUX KÖZÖSSÉG FÜGGETLEN MAGAZINJA

2015 január – 93. szám



Könyvajánló

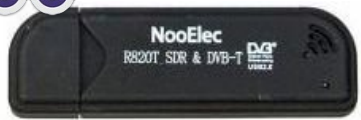


Ubuntu kioszk Könyvtáraknak, kávézóknak, stb.

A Full Circle Magazin nem azonosítható a Canonical Ltd-vel.



Hogyanok



RTL-SDR rádió

13

	A	B	C
1	13	14	15
2	14	15	16
3	15	16	17
4	16	17	18

LibreOffice

15



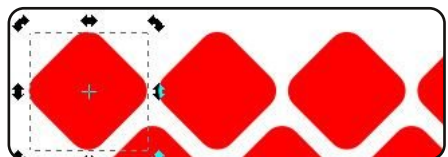
Ubuntu kiosk

17

A jövő hónapban

Blender

XX



Inkscape

21



Grafika



FullCircle

AZ UBUNTU LINUX KÖZÖSSÉG FÜGGETLEN MAGAZINJA

Rovatok

```
#An alias to make the ls
command more detailed
alias ls = "ls -la --
color=always --classify"
```

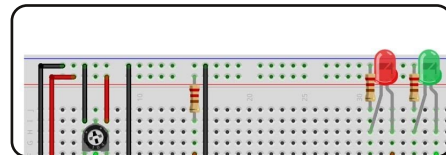
Parancsolj és uralkodj

11



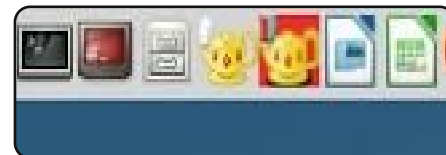
Linux labor

34



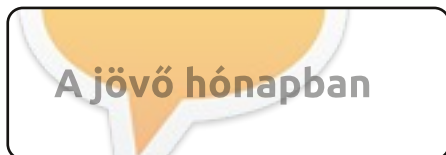
Arduino

26



Fókuszban

37



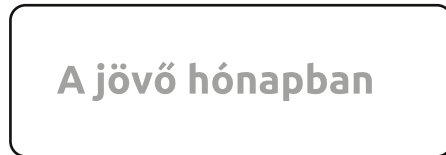
Levelek

XX



Tuxidermy

43



Biztonság

XX



Hölgyek és az Ubuntu

XX

Vélemények



Hírek

04



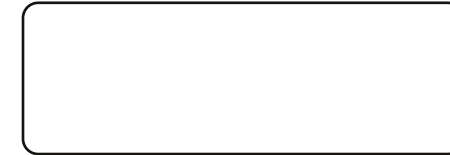
Linux labor

29



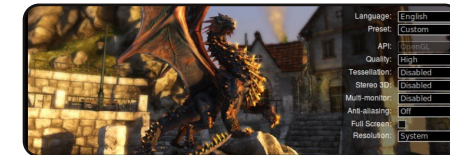
Különvélemény

41



KáVé

44



Játékok Ubuntu

46



Minden szöveg- és képanyag, amelyet a magazin tartalmaz, a Creative Commons Nevezd meg! - Így add tovább! 3.0 Unported Licenc alatt kerül kiadásra. Ez annyit jelent, hogy átdolgozható, másolható, terjeszthető és továbbadható a cikkeket a következő feltételekkel: jelezned kell eme szándékodat a szerzőnek (legalább egy név, e-mail cím vagy url eléréssel), valamint fel kell tüntetni a magazin nevét („Full Circle magazin”) és az url-t, ami a www.fullcirclemagazine.org (úgy terjeszd a cikkeket, hogy ne sugalmazzák azt, hogy te készítetted őket, vagy a te munkád van benne). Ha módosítasz, vagy valamit átdolgozol benne, akkor a munkád eredményét ugyanilyen, hasonló vagy ezzel kompatibilis licenz alatt leszel köteles terjeszteni.

A Full Circle magazin teljesen független a Canonicaltól, az Ubuntu projektek támogatójától. A magazinban megjelenő vélemények és állásfoglalások a Canonical jóváhagyása nélkül jelennek meg.



ÜDVÖZÖLLEK A 2015-ÖS FULL CIRCLE ELSŐ SZÁMÁBAN.

Még mindig egy emberrel kevesebb. Nem hallottam róla, így azt feltételezem, hogy Greg még mindig beteg és félek, ebben a hónapban nem lesz Python. Addig is Python helyett írtam egy gyors cikket az RTL-SDR rádióról. A YouTube-on találkoztam a témával, és gondoltam kipróbálom. Elképesztő, hogy mit lehet kihozni egy olcsó USB eszközből. Elmer itt van a szokásos LibreOffice-résszel és zárja a szentháromságot egy érdekes „Hogyan csináljunk Ubuntu kioszkot?” leírással. Kioszk alatt egy, csak olvasható terminált értünk (jobb nem jutott eszembe), nyilvános információk felhasználásra, vagy egy internetkávézóba vagy hasonló felhasználásra.

A Mate desktop eredetileg a Linux Mintből származik, de már az Ubuntu is átkerült. Ubuntu Mate 14.10 az egyik a két e havi szemléből, amit bemutatunk. A másik A Hivatalos Ubuntu Könyv (The Official Ubuntu Book).

Charles a laborjában a FreeBSD-s benyomásairól ír. Úgy látszik, hogy megküzdött vele, de már a végére ért. És mindeközben a kezeles sorozat ebben a hónapban sajnos a végéhez ér.

Van friss hír az Ubuntu Phone-ről. A Canonical közzé tett egy bejelentést a telefon interfészeiről, a telefon működési paramétereiről és még számos dologról. Az első telefonok február közepén fognak elérhetővé válni a vásárlók számára. Ha minden jól megy, akkor sikerül a kezem rátenni az egyik példányra, hogy kicsomagoljam, majd átnézzem ezen bestia tudását. És tartsd a szemed az FCM honlapján a friss hírekért, melyeket canonicales forrásból frissítünk, mielőtt azokat megkapjuk.

Minden jót kívánok nektek 2015-re!

Ronnie

ronnie@fullcirclemagazine.org



A magazin az alábbiak felhasználásával készült



Full Circle Podcast

Megjelenik havonta, mindig a friss ubuntu hírekkel, véleményekkel, áttekintőkkel, interjúkkal és hallgatói visszajelzésekkel. A Side-Pod egy újdonság, egy extra (rendszeresen) rövid podcast, ami mellékága a fő podcastnek. Leginkább általános technikai és nem-Ubuntu cuccokkal foglalkozik, melyek nem illenek a fő podcastbe.

Műsorvezetők:

- Les Pounder
- Tony Hughes
- Jon Chamberlain
- Oliver Clark



<http://fullcirclemagazine.org>



AUDIO MP3



AUDIO OGG

Letöltés

ELSŐ RÉSZLETEK AZ UBUNTU PHONE-RÓL



Az iPhone megjelenése óta a felhasználói élmények az appok és ikonrácsok körül forognak. A kulcsfontosságú mobilszolgáltatások az iOS-re és az Androidra, az appokon keresztül érkeznek töredezett módon. Ezeket az olyan platform tulajdonosok irányítják, mint a Google, az Apple és a Microsoft, ami az OEM-ekre és a szolgáltatókra másodlagos szerepet oszt.

A felhasználók megérdemelnék egy bőségebb, gyorsabb és kevésbé töredezett tapasztalat azon dolgok köré építve, melyeket a leggyakrabban használnak a készülé-

keiken.

Az Ubuntu telefonnal eltávolodunk az app rácsból az integrált tartalmak és szolgáltatások felé. És tesszük ezt úgy, hogy olyan felhasználói tapasztalatot szolgáltatunk, ami arra összpontosít, hogy a kulcsfontosságú digitális élet szolgáltatásait közvetlenül a képernyőre hozza, amit mi szívből scope-oknak hívunk.

A scope-ok segítségével egységes élményt nyújtunk különböző szolgáltatás-kategóriákban, előtérbe helyezve a felhasználókat, anélkül, hogy tengernyi app és appikon mögé rejtenénk őket. Egy egyszerű UI toolkit segítségével készítjük őket, és ez lényegesen kevesebb fejlesztői és karbantartási költséget eredményez, mint az appok készítése. Két fajta scope létezik.

Az aggregációs scope-ok határozzák meg a készülék alaphangulatát, és ami az Ubuntu mobilt értékesé teszi a felhasználó számára. Ezek engedik az OEM-ek és szolgáltatók számára, hogy olyan felhasználói élményt készítsenek, melyek egyedülállóak a készülékei-

ken, mint például a NearBy scope-ok, ami egyesíti a helyi szolgáltatásokat, arra összpontosítva, amit éppen csinálsz. Még részletesebben kitárgyaljuk a többi aggregációs scope-okat a Phone Glimpse egy következő levelező részében.

Az egyedi scope-ok alkalmazás-szerű élményeket közvetítenek a képernyőre, teljesen testre szabva. Elérhető az alap szoftverközponton keresztül, egy aggregációs scope közvetítésével, vagy úgy is mint az egyik kedvenc képernyő. Ez egy gyorsabb módszer a fejlesztők számára, hogy bőségebb és könnyebben elérhető élményt nyújtsanak a készüléken.

Forrás: Canonical
Beküldte: Canonical

HOGYAN TELEPÍTÜNK LINUXOT EGY CHROMEBOOKRA, ÉS TÁRJUK FEL AZ ÖSSZES LEHETŐSÉGÉT

Van egy pár Chromebook szuper hardverrel szerelve, mint a gyönyörű Chromebook Pixel – de

ezek nem érik el teljes potenciáljukat a Chrome operációs rendszerrel. Megmutatom hogyan telepítsd az Ubuntut és hozd ki a legjobbat a Chromebookodból.

A Chrome operációs rendszer nem rossz, és egy csomó munkát elvégezhetsz a nagyszerű Chrome appokkal. Néha azonban szükséged van egy teljes asztali gépre, hogy elvégezd a dolgokat. Lépj be az Ubuntu-ba: csak egy pár perces munkával kapsz egy teljesen működőképes Linux asztali gépet egy masszív hardverrel rendelkező Chromebookon, így létrehozva egy szuper laptopot.

Egy Crouton nevű eszköz használatával fogjuk telepíteni az Ubuntut (ezer köszönet barátainak a How-To Geek-től), ami a chroot parancsot használja az Ubuntu futtatására a Chrome operációs rendszer felett, ami amúgy is Linux alapú.

Forrás: <http://liferhacker.com/how-to-install-linux-on-a-chromebook-and-unlock-its-ful-509039343>
Beküldte: Arnfried Walbrecht

2015: A NYÍLT FORRÁSKÓD GYŐZELMESKEDETT, DE MÉG NINCS VÉGE

Az új év kezdetén hagyományosan visszanézni az elmúlt 12 hónapra. De amennyire ez a cikk foglalkozik vele, könnyű összefoglalni mi történt eddig: a nyílt forráskód győzött. Kezdjük a csúcsról:

Szuper-komputerek. A Linux annyira domináns a szuper-komputerek 500-as listáján, hogy az már majdnem zavaró. A 2014 novemberi ábrák azt mutatják, hogy az 500 rendszerből, 485-ön a Linux valamelyik formája van. Windows csak egy rendszeren fut. A dolgok még szembetűnőbbek, ha megnézed a szóban forgó magok számát. Itt a Linux 22 851 693-on található, míg Windows csak 30 720-on, ez azt jelenti, hogy a Linux nemcsak dominál, de a nagyobb rendszereken különösen erős is.

Forrás: <http://www.computerworld.com/blogs/open-enterprise/open-source-has-won-3592314/>
Beküldte: Arnfried Walbrecht

A SAMSUNG A JÖVŐBENI TÉ-

VÉIT A LINUX ALAPÚ TIZEN OPERÁCIÓS RENDSZERREL MŰKÖDTETI

Azok, akik még nem tudják, mi is az a Tizen – ne aggódjanak, ez nem az ők hibájuk. A Samsung kísérletének, hogy készítsen egy Linux alapú platformot, az Intel Meego-jával (ami elég szerencsétlenül halált halt) elég rázós útja volt.

A Samsung első Tizen okostelefonja határozatlan időre el van napolva, és egy későbbi kísérlet, a Tizen alapú Z1 indiai indulását eltolták 2014-ről, 2015 januárjára.

Álljunk csak meg! A Samsung új mobil operációs rendszert készít az Intellel? Mi történt a megingathatatlan (majdnem kínos) Android-Samsung kapcsolattal? Egyszerű – a Tizen a Samsung kísérlete, hogy a bevételt a saját irányában térítse, mert a jelenlegi felállásban a profit nagyobb része a Google-é. A Tizen azzal büszkélkedik, hogy a világ legnagyobb neveinek közös erőfeszítésének köszönheti létezését – a Huawei, a Fujitsu, a NEC, a Panasonic, a KT, a Sprint, az SK Telecom, az Orange, az NTT Docomo és a Vodafone igazgatói ülnek a Tizen Association igazgatói tanácsában.

Forrás: <http://thetechportal.in/2015/01/01/samsung-power-future-tvs-linux-powered-tizen-os/>
Beküldte: Arnfried Walbrecht

A NOKIA LUMIA 1020 UBUNTU OPERÁCIÓS RENDSZERÉNEK JELLEMZŐI KISZIVÁROGTAK

Egy új pletyka terjed a mobilkészülékek világában a Team Ubuntu jóvoltából (a GizmoChinán keresztül), mely szerint a Nokia Lumia 1020 okostelefonon a Linux teljes verziójú Ubuntu operációs rendszere fut. Ezt a fajta pletykát nehéz százszázalékosan megerősíteni, de egy képsorozat a forrás weboldalán mutatja, ahogy a mobil készülék hatásosan futtatja az operációs rendszert. A képek felfedik, hogy az okostelefon, az Ubuntu asztali operációs rendszer teljes verzióját futtatja. Bárki, aki használta az Ubuntu Linux operációs rendszert előzőleg, felismerheti az elrendezést a telefonon, mert baloldalra rendezett parancsikonokat és egy alap háttérképet használ a szokásos csempékkel, amiket általában használnak a munkához.

Forrás: [A link már nem él](#) (a szerk.)
Beküldte: Arnfried Walbrecht

AZ ENTERPRISE LINUX DISZTRÓK FONTOS FRISSÍTÉSEINEK NAGY ÉVE

2014 ajánlott egy ritka, egybefolyt kiadási naptárat, ami különösen erős a Linux számára. A három nagy vállalati Linux gyártó, mérnököknek számító frissítéseket adott ki 2014-ben. Április 17-én az Ubuntu 14.04-et, Trusty Tahr kódneven adták ki, mint hosszútávon támogatott (LTS) Ubuntu verzió. (Az Ubuntu két évente jelentet meg új LTS kiadásokat; mindegyik LTS legfeljebb öt évig támogatott.) A Red Hat Enterprise Linux 7 (RHEL) június 10-én jött ki, jelezve a Red Hat első nagyobb RHEL frissítését 2010 óta. A Red Hat tíz évig támogatja a vállalati kiadásait. Október 27-én a SUSE Linux Enterprise 12 jelent meg, jelezve a SUSE zászlóshajós vállalati Linux platformnak az első nagyobb frissítését 2009 óta. Nemcsak a nagy vállalati Linux kiadások voltak az egyedüliek, melyek frissültek. A Red Hat Fedora közösségének Linux disztribúciója megjelentette a Fedora 21-et, december 9-én. Az Ubuntu 14.10, mely a Utopic Uni-

corn kódnév alatt október 23-án debütált, nagy adatképességekkel rendelkezik.

Forrás:

<http://www.eweek.com/servers/slideshows/big-year-for-enterprise-linux-distros-includes-major-updates.html>

Beküldte: Arnfried Walbrecht

ISMERD MEG A MAKULU LINUX XFCE 7.0-ÁT, A LEGSZEBB LINUX DISZTRÓT, AMIT VALAHA LÁTTAM

A Makulu Linux legújabb kiadásának két fontos tulajdonsága van: először is inkább az Ubuntu 14.04-en alapul, mint a Debianon, és másodsorban Xfce asztalt használ. Úgy tűnik a Makulu körbejárja az asztalokat, így talán az Xfce használata nem meglepő.

A kiadási közlemény betekintést nyújt pár érdekes dologról a kiadás háttéréről és fejlesztéséről, valamint a fontosabb jellemzőiről. Ahogy a Makulu Linuxnál megszokhattuk, főleg az esztétikára fókuszál, és rengeteg gyönyörű háttérkép, témát, ikont és betűkészletet tartalmaz. A másik dolog amire

még nagyon fókuszál az a sebesség. Így esett, hogy az Xfce-t választották az asztal számára, és a Firefoxot a Chrome helyett, a Synapse Launchert a Slingscoid helyett, és még sok más csomagot megtartottak, míg egyeseket kizártak.

Forrás:

<http://www.zdnet.com/article/hands-on-with-makulu-linux-xfce-7-0-the-most-beautiful-linux-distro-i-have-ever-seen/>

Beküldte: Arnfried Walbrecht

A LINUX FOUNDATION ÁLTAL BIRTOKOLT ALLSEEN ALLIANCE KITERJESZTI A DOLGOK INTERNETÉVEL KAPCSOLATOS ERŐFESZÍTÉSEIT

Az AllSeen Alliance tegnap jelentett be új kezdeményezéseket és motivációkat, hogy segítsen fejleszteni a szabványokat a növekvő hálózatban, ami a Dolgok Internet (Internet of Things), röviden IoT.

Az AllSeen Alliance multi-résztvényesek közös erőfeszítése alap-

ján jött létre, ami a Linux Foundation Collaboration projektjeként működik. A Linux Foundation 2013 decemberében említette először az AllSeen Alliance erőfeszítést, az AllJoyn kód hozzájárulással a Qualcomm-tól, mely alapjául szolgál. Az AllJoyn egy framework, ami biztonságos és akadálytalan kapcsolatot biztosít, valamint belépést a IoT eszközökhöz.

Most az AllSeen Alliance kiterjeszti a frameworkot az AllJoyn Gateway Agent-tel, ami kiterjeszti az IoT hatásait egy felhasználó helyi környezetén túl, egészen a felhőig.

Forrás:

<http://www.datamation.com/cloud-computing/linux-foundation-allseen-alliance-expands-internet-of-things-efforts.html>

Beküldte: Arnfried Walbrecht

AZ INTEL COMPUTE STICKJE LINUXOT FUTTAT

Az Intelnek van egy új, király zsebkomputere, amit Compute Stick-nek neveztek el. Még nem jelent meg, de lesz Linuxos verzió, valamint olyan is ami Windowst futtat.

A Compute Stick valójában végtelenül egyszerű, a Shenzhen Apec Electronics által készíttethez képest, ami októberben került a boltokba. Az Intel kezdetben két modellt tervez kiadni. Mindkettőt egy-egy négymagos Atom processzor fogja meghajtani. Azt amelyik Windows 8.1-et futtat Binggel, 2 GB RAM-mal és 32 GB belső tárhellyel érkezik, 149 dollárért fogják árúsítani.

Elérhető lesz még egy olcsóbb verzió is, melyre Ubuntu lesz előtelepítve. Ez csak 89 dollárba kerül, de gyengébb hardverrel. A Linux Compute Stick csak 1 GB RAM-ot és 8 GB tárhelyet tartalmaz. Ez még mindig rengeteg erőforrás alapvető informatikai feladatok elvégzéséhez.

Forrás:

<http://www.itworld.com/article/2866536/intel-compute-stick-will-run-linux.html>

Beküldte: Arnfried Walbrecht

3D PRINTER FEJLESZTŐI KIT LINUXOT FUTTAT AZ ÚJ MARVELL ARMV7 SOC-N

A Marvell bejelentette az első Linux alapú 3D printerekhez szánt hardver/szoftver fejlesztői kitet, az új 533 MHz „88PA6120” ARMv7 SoC köré építve.

A Marvell 3D Printer SoC Solutionje, más néven Marvell 88PA6120 3D Printer Development Kit, egy komplett referencia kitet szolgáltat 3D nyomtatók kulcsra-kész fejlesztéséhez, állítja a Marvell. A hardverplatformot az új Marvell 533 MHz órajelű 88PA6120 SoC-ja köré építették. A cég nem osztotta meg a processzor részleteit, de azt mondják, hogy ARMv7 kompatibilis processzor.

Forrás:
<http://linuxgizmos.com/3d-printer-dev-kit-runs-linux-on-new-marvell-armv7-soc/>

Beküldte: Arnfried Walbrecht

REJTETT XOR.DDoS TRÓ- JAI FERTŐZI A LINUX REND- SZEREKET, ROOTKIT-ET TELEPÍT

Egy újonnan felfedezett trójai fertőzi a Linux rendszereket, és valószínűleg gépek arzenálját építi ki, melyeket szolgáltatásmegtagadással járó támadáshoz (DDoS) használnak, áll egy blogban, melyet az Avast posztolt.

Az új fenyegetés, az XOR.DDoS megváltoztatja a telepítését az áldozat Linux környezetétől függően, és később egy rootkitet futtat a felfedezés elkerülése érdekében. Habár egy hasonló trójait már fedeztek fel Windows környezetben, Peter Kálnai, aki malware analízáló az Avastnál, egy, az SCMagazine-nak szerdán adott interjúban azt mondta, hogy ez a trójai egy viszonylag kiaknázatlan területre merészkedett, amikor Linux rendszereket vett célba.

Forrás:
<http://www.scmagazine.com/malware-targets-linux-and-arm-architecture/article/391497/>

Beküldte: Arnfried Walbrecht

LIBRECALC, A 3D-BEN NYOMTATOTT LINUX SZÁMO- LÓGÉP

Főleg arra használom az okoste-

lefonomon levő számológépet, hogy kitaláljam mennyi számlát kell fizetnem, vagy kitaláljam a fizetési csekk összegét. De évekkel ezelőtt volt egy időszak, amikor a matematikaórán vagy vizsgákon használtam számológépet. Létfontosságú volt a megléte egy időben, a középiskolában és azoknak a barátaimnak akik nyomták a matematikát a főiskolán, biztos lehetsz benne, nekik volt egy a hátizsákban. Mi lehet egy számológépnek 3D-s nyomtatóval nyomtatott házzal, és ingyenes szoftverrel az okostelefonok korában? Ez egy jó kérdés, mert a LibreCalc egy új, nyílt forráskódú, programozható számológép, ami letölthető dizájnnal most elérhető.

Forrás:
<http://3dprint.com/36474/3d-printed-calculator-librecalc/>
Beküldte: Arnfried Walbrecht

A GOOGLE CLASSROOM MO- BIL APPAL GÖRDÜL ELŐ, ME- LYET A TANULÓKNAK ÉS TANÁROKNAK SZÁN

A Google egy új Classroom márkájú mobil appot indít el, melynek célja bátorítani a digitális

együttműködést a tanárok és diákok között.

A Google Apps for Education több mint 40 millió tanulót, tanárt és adminisztrátort tart számon a felhasználói adatbázisában, az internet óriásának a tavaly őszi becslése szerint. A Classroom maga durván hat hónappal ezelőtt debütált, annak érdekében, hogy a Google Apps for Education egyszerűbben elérhetővé váljon az oktatók számára. Az oktatók számára elérhető jellemzők segítséget tartalmaznak a tanulóknak, hogy nyomon kövessék a felmerülő feladatokat, ezt követi az instruktorok lehetősége, hogy a feladatokat „készre” jelölhessék, ahogy azok elkészültek.

Indulása óta több mint 30 millió feladat ment keresztül a Classroomon, jegyezte meg Jorge Lugo, egy szoftvermérnök a Google for Education Classroom csapatában.

A Classroom mobil verzióját, ahogy Lugo részletezte, úgy tervezték, hogy még inkább megkönnyítse a kommunikációt tanárok és diákok között, nevezetesen egy tanári feladatok hozzáadásán, és az osztályok archiválásának képességén keresztül.

Forrás:

<http://www.zdnet.com/article/google-classroom-mobile-ios-android-education/>

Beküldte: Rachel King

GIT SEBEZHETŐSÉGET ZÁRTAK BE AZ UBUNTUKBAN

A Canonical publikált pár részletet egy Git sebezhetőségről, melyet az Ubuntu 14.10, az Ubuntu 14.04 LTS és az Ubuntu 12.04 LTS operációs rendszerekben találtak, és javítottak ki.

Mindegyik új Git verzió rendszerint biztonsági javításokkal érkezik, ez történt a legutolsó ággal is. A fejlesztők elmagyarázták, hogy a Git képes lenne programokat futtatni, ha speciálisan kidolgozott változásokat kapna a szoftverközpontból.

Ez ugyanaz az esemény, amelynek sikerült Linus Torvaldsot felidegesíteni, a HFS+ fájlrendszer gyatra kondíciója miatt.

Forrás:

<http://linux.softpedia.com/blog/Git-Vulnerability-Closed-in-Ubuntu-OSes-470022.shtml>

Beküldte: Silviu Stahie

NVIDIA GeForce GTX 970/980: Windows kontra Ubuntu Linux

Korábban, a héten már posztoltam pár benchmarkot, melyek azt mutatják, hogy az Nvidia Linux drivere folytatja erőteljes menetelését a Windows driverrel szemben, mely magába foglalja a Nouveau driver eredményeit is. Egyébként a Nouveau driver használata közben az újabb GeForce GTX 970/980 kimaradt a tesztelésből, mert a nyílt forráskódú driver nem teljesen támogatja ezeket az újabb Maxwell GPU-kat. Ebben a cikkben visszatérünk, hogy összehasonlítsuk ennek a két csúcskategóriás Nvidia GPU-nak a Windows kontra Linux teljesítményét UHD 4K felbontáson is.

Ahogy a 2014-es korábbi Windows kontra Linux benchmarkokban is, a Windows 7 Pro x64-et az összes elérhető rendszerfrissítéssel, és Linux oldalon az Ubuntu 14.10-et használtuk. A legfrissebb Nvidia Linux driver a 346.22 driver verzió, míg a legfrissebb Windows verzió a teszt idején a 347.09 driver kiadás volt.

Mindkét operációs rendszer naprakész volt, és az alap beállítá-

sokkal volt tesztelve. Ugyanazt az Intel Core i7 4790K Haswell rendszert használtam mind a Linux, mind a Windows teszteléséhez. Egy Seiki 39 colos, 3840 x 2160 felbontású megjelenítőt használtam a teszteléshez, rengeteg OpenGL benchmarkot futtattam 4k felbontással, hogy teljesen leterheljem ezeket az új NVIDIA grafikus kártyákat.

Forrás:

http://www.phoronix.com/scan.php?page=article&item=nvidia_maxwell900_winlin&num=1

Beküldte: Michael Larabel

NEM-LINUX FOSS: KÖNNYŰ ANIMÁCIÓKÉSZÍTÉS

Amennyiben valaha is akartál animációs filmet készíteni, az erre alkalmas szoftverek megtanulási folyamata gyakran igen nehéz. Hála az égnek, megjelent a Pencil nevű program, és bár alapszintű, de egy elég egyszerű módot kínál, hogy nyílt forráskódú eszközökkel animációt készíthess a (Windows, Mac vagy Linux) számítógépeden. Sajnos a Pencil elhanyagolták.

És mégis, ez a legklasszabb része a nyílt forráskódú programnak.

A hihetetlen Pencil programra építkezve, egy új projekt született. A Pencil2D-t aktívan fejlesztik, és a többplatformos alkalmazás segítségével rajzolhatsz és exportálhatsz filmkockáknként animációs képsorokat.

Forrás:

<http://www.linuxjournal.com/content/non-linux-foss-animation-made-easy>

Beküldte: Shawn Powers

A SCOTUS INFORMÁCIÓKAT KÉR AZ IGAZSÁGÜGYI MINISZTERIUMTÓL A GOOGLE ÉS ORACLE JAVA VITÁBAN

Attánok összecsapása a Java miatt véget érhet, hallottuk a Legfelsőbb Bíróságtól, attól függően, hogy a legfőbb államügyész-helyettes mit mond erről. A SCOTUS megkérdezte, vajon az ügy méltó-e a figyelmére. „Ez 2015 legizgalmasabb eseménye lesz”, mondta Al Hilwa technikus. Ez egy olyan ítélet, ami méltányos használatot alkothat a szoftverrel kapcsolatban.

Az Egyesült Államok Legfelsőbb

Bírósága hétfőn meghívta az Obama adminisztrációt, hogy mérlegelje, vajon meghallgatja-e az érveléseket a folyamatban lévő vitában a Google és az Oracle között, a Java szerzői jogokkal kapcsolatban.

Ez a mozgás válaszként szolgál a Google által októberben benyújtott kérvényre a 2014 májusi, az elsőfokú szövetségi bíróság által meghozott, az Oraclenak kedvező ítéletét követően.

A Google érvelése szerint a kód nem vonható szerzői jogvédelem alá a Szerzői Jogi Törvény 102(b) szekciója miatt, ami megvonja a szerzőjogi védelmet bármilyen ötlet, eljárás, folyamat, vagy üzelmeltetési módtól. Az is vitatott, hogy a másolt elemek, vajon kulcsfontosságú elemek voltak-e a Java és az Android közötti átjárhatóság biztosításában.

Számos nagy technológiai cég, beleértve a HP-t, Red Hatet és a Yahoót, nyújtott be írásos véleményt (amicus curiae) a Google pozíciójának erősítése érdekében.

Forrás:
<http://www.ecommerce-times.com/story/81573.html>
Beküldte: Katherine Noyes

APRÓ MINI-PC LINUXOT FUTTAT NÉGYMAGOS AMD SoC-N

A CompuLab leleplezte a legújabb mini PC-jét: a Fitletet. Ez az apró, ventilátor nélküli x86 architektúrájú rendszer csak 0,22-öt fogyaszt, választhatsz a négy- vagy kétmagos 64 bites AMD SoC-k között, 1,6 GHz-es órajellel. Mindez AMD Radeon R3 és R2 Grafikus GPU-val társul. A Fitlet illeszkedik a CompuLab régóta létező „Fit-PC” nevű ventilátor nélküli mini PC-inek vonalába, mely felöleli mind az x86, mind az ARM alapú készülékeket, egyaránt támogatja a Linux és a Windows operációs rendszereket, és tartalmaz egy sorozat, különösen kedvező árú előkonfigurált MintBox mini PC-t, a Linux Mint projekttel való együttműködésen alapulva. A nem Linux Mintes gépek ismerősök a LinuxGizmos olvasóinak. Ilyenek a Fit-PC3, az Intense PC és a strapabíró uSVR, mely dual-core AMD G-T40E és G-T46N APU-kon alapul. Valamint ilyen az Intense PC2, dual-core Intel Core i7 alapokon és a Fit-PC4 AMD, quad-core GX-420CA SoC alapokon Radeon HD 8400E grafikával.

Forrás:
<http://linuxgizmos.com/tiny-fan-less-mini-pc-runs-linux-on-quad-core-amd-soc/>
Beküldte: Arnfried Walbrecht

AZ INDAI KORMÁNY TERVEI SZERINT A HAZAI FEJLESZTÉSŰ BOSS LINUX LESZ INDIA HIVATALOS OPERÁCIÓS RENDSZERE

A BOSS (Bharat Operating System Solution) egy ingyenes operációs rendszer, amit a National Resource Centre for Free/Open Source Software (NRCFOSS) fejlesztett az indiai központi kormány irányítása alatt. Bár már létezik egy ideje, a lelkesedés hiánya az egész országban az állami hivatalnokok és a napi felhasználók között megakadályozta az OS nagyobb mértékű elterjedését. Úgy tűnik, a tavaly hatalomra jutott NDA kormány, ezt megváltoztatni szándékozik.

A Make in India kezdeményezés égisze alatt, melyet Narendra Modi Miniszterelnök álmodott meg, a DeitY (Department of Electronics and Information Technology) utasított számos állami hivatalt, hogy hajtsa végre az átállást a zárt for-

ráskódú Windows rendszerről a BOSS-re. A folyamat elősegítése érdekében, a kormány konzultált egy Csennai-ban székelő IT céggel is, hogy támogatást biztosítson az operációs rendszernek.

A BOSS elsődlegesen Debianon alapul, és jelenleg a C-DAC (Centre for Development of Advanced Computing) tartja karban. A tervek szerint a Microsoft Windows teljes értékű alternatívájává válik, de a BOSS akkor mutatja meg erejét amikor véglegesen el fog helyezkedni. Jelenleg 19 nyelven terjesztik. A legfrissebb stabil verzió, Anokha kódnéven, relatíve elég öreg (2013 decemberében adták ki), de ez az új lökés a kormánytól, technikai támogatással és karbantartással párosulva, biztos új életet lehel abba, amit „India házi készítésű Operációs Rendszerének” hívnak.

Forrás:
<http://computer.financialexpress.com/news/governments-boss-move-now-a-homegrown-system-to-run-computers/7764/>
Beküldte: Anirban Chatterjee

MINTBOX MINI: KOMPAKT LINUX MINT PC ÉRKEZIK HAMAROSAN 295 DOLLÁR-ÉRT

Hasonlóan több más fitlet modellhez, a MintBox Miniben van egy AMD's Mullins architektúrás AMD A4-6400T mikroprocesszor. Tartalmaz egy Radeon R3 grafikus processzort, két USB 3.0 portot, két USB 2.0 portot, dual HDMI portokat (két megjelenítőhöz), microSD kártyaolvasót és egy fejhallgató csatlakozót. A számítógépben van még 80.211/b/g/n WiFi és Gigabit Ethernet. Ezt a modellt 4 GB RAM-mal és 64 GB-os SSD-vel szállítják. A MintBox Mini 2015 második negyedében lesz elérhető 295 dollárért az USA-ban, vagy 295 euróért Európában (az európai ár tartalmazza az ÁFA-t). Az eladásokból származó pénz egy részéből a Linux Mint szoftver fejlesztését fogják támogatni. A MintBox Minire 5 év garancia van.

Forrás:
<http://liliputing.com/2015/01/mint-box-mini-compact-linux-mint-pc-coming-soon-295.html>

Beküldte: Arnfried Walbrecht

AZ UBUNTU LINUX KÉSZEN ÁLL A BERENDEZÉSEIDET ÉS ROBOTJAIT ÜZEMELTETNI

Az Ubuntu Linux elég sok platformon elterjedt 10 éves történelme során, ha nem is mindig sikeresen. Ma azonban a nyílt forráskódú szoftver a legnagyobb kihívás előtt áll: a dolgok internetje. A Canonical kiadta a lecsupaszított finom Ubuntu Core verzióját a csatlakozott készülékekhez, mint az otthoni készülékek, robotok és minden máshoz, amivel a hagyományos PC operációs rendszerek nem tudnak foglalkozni. Úgy tervezték, hogy egyszerű hardveren is fusson (egy 600 MHz-es processzor is megteszi), és könnyen frissíthető, mindemellett megadja a gadget-készítőknek a szabadságot, hogy a szoftvert ahhoz igazítsák amihez tervezik. Ígéret szerint extra megbízható – csak akkor alkalmazza a frissítéseket, ha a kódokat ellenőrizték, így nem veszíted el az okos termosztátod irányítását egy hibás frissítés miatt.

Forrás:
<http://www.engadget.com/2015/01/20/snappy-ubuntu-core-for-iot/>
Beküldte: Arnfried Walbrecht

A CERN ÉS AZ NI EGYÜTTMŰKÖDÉSE, HOGY MEGHATÁROZZÁK A JÖVŐBENI LABVIEW TÁMOGATÁST LINUX 64 BITRE

Az NI (Nasdaq: NATI), azon megoldások szolgáltatója, melyek lehetővé teszik a mérnököknek és tudósoknak a világ legnagyobb mérnöki kihívások megoldását, bejelentette együttműködését a CERN-nel, egy kormányközi kutató szervezettel, hogy megépítsék a világ legnagyobb és legfejlettebb kutató létesítményét. A cél az, hogy az összes CERN ellenőrzési rendszer szabványosítását 64 bites Linux operációs rendszerre tolják át, költséghatékonyan elosztott ellenőrzési rendszereket tervezni, és kiterjeszteni a lehetőségeket a kis és közepes vállalkozások számára a szakértelem segítségével az NI-ban és a nyílt forráskódú technológiában.

Az NI az 1990-es évek elejétől együtt dolgozik a Nukleáris Kutatások Európai Szervezetével (European Organization for Nuclear Research), ismertebb nevén CERN, olyan alkalmazásokon, melyek segítenek elmagyarázni, mi is az univer-

zum, és hogyan jött létre. A figyelemre méltó együttműködés magában foglalja a Nagy Hadronütköztető (LHC) részecskegyorsító rendszerét – ahol LabVIEW rendszerű tervező-szoftverrel volt vezérelve mintegy 120 darab NI PXI léptetőmotor vezérlőrendszer – és a MedAustron ionsugár rákterápiás központot amelyre a CERN, a 2013-as NIWeek rendezvényen három díjat kapott.

Ezek a közös fejlesztések értékes gyakorlati tapasztalatot adtak a mérnököknek a gyorsan fejlődő beágyazott rendszerek piacán, és hosszú távon karbantartható rendszereket hoztak létre a küldetés szempontjából kritikus applikációkban.

Forrás:
<http://www.businesswire.com/news/home/20150126006414/en/CERN-NI-Collaborate-Define-Future-LabVIEW-Support>
Beküldte: Arnfried Walbrecht



Korábban, a 87. számban, bemutattam a Node.JS-t és a „Grunt” feladatfuttatót. Mostanság felfedeztem egy alternatívát, melynek neve Gulp. Ez egy kicsit jobban tetszik nekem (de erről majd később). Körülbelül akkoriban egy olvasó is rákérdezett, hogy nem szentelnék-e egy részt a Gulpnak? Úgyhogy ez következik.

Mi a különbség?

A különbség a Gulp és a Grunt között legfőképpen abban áll, ahogyan a feladatokat futtatják. A Grunt a konfiguráción alapuló rendszerre fókuszál, ahol meghatározzuk minden egyes feladat konfigurációját (fájlok, tömörítés, plugin-használat, stb.). A Gulp pedig a munka folyamatára fókuszál, ami azt jelenti, hogy egy plugint csak egy feladatra használunk, nincs átfedés. Így inkább néz ki program-szerűnek, mint konfigurációs alapúnak. Ez azt jelenti, hogy a Gulp-feladatok úgy néznek ki, mint egy JavaScript függvény, amely függvényhívásokat használ egy-egy konfigurációs lépéshez. Talán egyszerűbb megérteni a különbséget,

ha látjuk működés közben. Ha inkább a különbségek bemutatását szeretnéd, akkor nézd meg a További olvasmányok részt.

PÉLDAKÓDOK

A példák megtalálhatóak az alábbi GitHub helyen:
<https://github.com/lswest/FCM93-Gulp-Example>

GULPFILE.JS

Ez az a fájl, amely létrehozza és rendszerezi a Gulp-feladatokat, amelyeknek le kell futniuk. A fájl első hét sora (a `var name = require('name');`) egyszerűen inicializálja a különböző Node.JS csomagokat, amikre szükségünk lesz. Ezek már telepítve kell, hogy legyenek npm-mel. Amennyiben megvan a `csomagnév.js` fájlunk, akkor az npm install parancs futtatása elegendő.

Az ok, amiért a kiegészítőket változóba helyezzük, hogy könnyebb legyen azok meghívása. És az ok, amiért kötelezőek, mert így teszi lehetővé az npm-csomagok használatát, mint például a Stylust-

t a Gulpból.

A példafájlban aztán létrehozok két feladatot. Az „app” (progi) feladat a 9. sorban kezdődik és tart a 21. sorig. Az „uglify” (elcsúfítás) feladat pedig a 23-28. sorokban található. A feladatok nevei tetszőlegesen választhatóak.

AZ „APP” FELADAT

Ez a feladat azt a célt szolgálja, hogy az előfeldolgozó CSS fájlt, CSS fájlra fordítsa, melyet aztán a stylusnál használunk. Ugyanezen logika mentén működik a SASS és Less is (amennyiben a megfelelő csomagok telepítve vannak).

A feladat először a `gulp.src` fájl futtatásával kezdődik, amely megmondja, hogy hol találja a stylus-fájl(oka)t. Elvileg lehet összefűzni `.styl` fájlokat, vagy a `*.styl` hívásával az összes stylus-állományt megnyitni. De jobb módszer, ha csökkentjük a használandó css fájlok számát – amit az oldalunknak kell megnyitnia –, és egy stylus-fájlt használunk. Alkalmazzuk az `@import` függvényt, más stylus-fájlok

eléréséhez, és végül az egész egy CSS fájlt fog eredményezni.

Aztán pedig van egy csőfüggvény (pipe function), amelyet a `.src`-re hívunk meg, amelyben egy stylus-függvényt hívunk meg, egy pár beállítással. A beállításnál a „use” mondja meg, hogy melyik stylus plugint kell meghívnia (az én példában kizárólag a Jeet-et hívom a Jeet Keretrendszerhez). A „compress” beállítás – ahogyan kitalálható –, vezérli a fájlok tömörítésének beállításait (a CSS fájl minimalizálására). A `source` map fájl létrehozását vezérli, így a vizsgáló eszköz nem a tömörített css fájlra hivatkozik hanem sorszá-mokat produkál, ami az eredeti stílusfájltra vonatkozik. Ez hasznos lehet fejlesztésnél, de ki is hagyható.

Utoljára, de nem utolsó sorban a `.pipe(gulp.dest('css/'))`; beállítja a célmappát, ahol létrehozza a fájlokat.

Remélhetőleg mostanra a „munka folyamata” szóösszetétel jobban értelmezhető. Minden egyes függvényhívásnak egy célja van, a többivel összehangolva úgy,

hogy egyszerre egy lépés történik, míg végül a végső eredményt elérjük.

AZ „UGLIFY” FELADAT

A lényege ennek a feladatnak, hogy az összes JavaScript fájl összekezdve az eredmény a lehető legkisebb legyen. Ekkor már a fájl kevésbé olvasható, de csökkenti a fájl méretét, és a fájlok számát, amit egy weboldalnak használnia kell. Innen a kifejezés „uglify” (elcsúfítás).

Ez a feladat megint csak egy gulp.src függvényhívással kezdődik (amely kiválaszt minden js fájl a js mappában).

Ezek után meghív három cső-függvényt (pipe function). Az első összefűzi az összes js fájl, aztán lefuttatja az elcsúfító függvényt (amely minimalizálja az eredményül kapott összefűzött javascriptet). Az utolsó lépés, mint mindig, a „dest” függvény hívása, amellyel meghatározzuk a célmappát az elkészült fájloknak.

AZ UTOLSÓ SOR

Amikor megírjuk a feladatokat, amire szükségünk van, akkor az utolsó sor létrehoz egy utolsó feladatot, amelyet „default”-nak (alapértelmezett) nevezek. Ez futtatja a másik két függvényt is, ha meghívjuk. Így ahelyett, hogy meghívnanánk a függvényeket egyesével, a Gulp végrehajtja a teljes folyamatot egy szuszra. Ezt a feladatot hívják „default”-nak, mert ezt a függvényt hívja meg a Gulp automatikusan.

Az eredmény két új fájl lesz: egy minimalizált CSS fájl, és egy minimalizált javascript fájl. Hivatkozunk ezekre a fájlokra a HTML „head” részében és már készen is vagyunk.

Remélem ez a cikk segített, hogy a Gulp vonzó legyen azoknak, akik lelombozódtak a Grunt miatt korábban. Számos cikk található arról, hogyan használhatjuk a Gulpot más általános feladatra (mint például SASS-ra). A szintakszisát véleményem szerint könnyebb olvasni mint a Gruntét, amennyiben van valamennyi Javascript tapasztalatunk – ez kezdőkre is igaz. Vannak, akik számára a feladatt futtatók túlbonyolítják azt, amit tenniük kellene. De a könnyű kezelhetőség néhány esetben (főleg a kezdőknél) ellen-

súlyozza a sok helyettesítést. Azok akik az npm-et szeretnék használni, és szeretnék elkerülni a bonyolítást, egyszerűen telepítsék csak a csomagokat (mint például stylus). A dolgok automatizálására használhatunk make fájlt, vagy szkriptet parancsok futtatására. Ezt a feladatot rábízom a felhasználóra. Amennyiben használsz webes keretrendszert, azokban általában a függvényhívások végrehajtásának valamilyen módja létezik.

Ha bárkinek kérdése, javaslata, vagy kérése van, bátran küldjete emailt a következő címre: lswest34+fcm@gmail.com

TOVÁBBI OLVASNIVALÓ

<http://markdalglish.github.io/presentation-build-wars-gulp-vs-grunt/> – Gulp és Grunt különbségek bemutatása

Gulp weboldal: <http://gulpjs.com/>



Lucas a számítógépe folyamatos tönkretételétől a javításig mindent megtanult. Küldj neki emailt az lswest34@gmail.com címre.



EXTRA! EXTRA! OLVASS EL MINDENT!

Nagy becsben tartott hírtudósítóink posztolják a rendszeres hírfrissítéseket a Full Circle weboldalára.

Kattints a NEWS linkre a lap tetején található menüben és máris látni fogod a hírek szalagcímkeit.

Vagy tekints bármelyik lap jobb oldalára, ahol az öt legfrissebb beküldött hírt találod.

Nyugodtan vitassátok meg a hírblokkokat. Lehet, hogy egy olyanról van szó, ami a magazinba is bekerülhet.

Leljétek örömtöket benne!



Hogyanok

Írta: Ronnie Tucker
Fordította: Jancsek Árpád

Szoftveralapú rádió RTL-SDR eszközzel

Még mindig nem hallottam felőle, így feltételezem, hogy Greg ebben a hónapban is gyengélkedik. Nyugodtan küldjétek neki jókívánásokat erre a címre:

greg.gregwa@gmail.com

Mostanában bukkantam rá a Youtube-on egy videóra, amiben látható, hogy milyen könnyű valós időben nyomon követni a repülőket egy egyszerű DVB USB stick használatával. Elhatároztam, hogy veszek egyet és kipróbálom valóban olyan könnyű-e. És tényleg az!

Ezek az eszközök 20\$-ba kerül-



nek az Ebayen és az Amazonon is. Az, amit én vettem, egy RealTek RTL8723BE chipsetre épül. Eredetileg azért készítették, hogy TV-t lehessen nézni a laptopon, de számos ember rájött, hogy valójában hatalmas spektrumú rádiófrekvenciát fed le és sok mindenre lehet használni a HAM-rádiók hallgatásától kezdve a repülőgépek követéséig, ráadásul (jó antennával) időjárási adatokat is le lehet tölteni vele a NOAA műholdakról. Ezek más néven az RTL-SDR eszközök, amiben az RTL a RealTek chipsetre vonatkozik, az SDR pedig szoftveralapú rádiót jelent (Software Defined Radio).

SZÜKSÉGES FELTÉTELEK

Az első dolog, amire szükség lesz, hogy hozzáadd a GQRX PAA-t a forráslistához. Ezt megteheted a csomagkezelő használatával, vagy terminálban is:

```
sudo add-apt-repository  
ppa:gqr/snapshots
```

A következő lépésben Roman Moravcik PPA-ját kell hozzáadni, itt

található az az alkalmazás, amire szükségünk van:

```
sudo add-apt-repository  
ppa:roman-moravcik/gnuradio
```

A probléma most az, hogy ezen tárolók egyike sem rendelkezik Ubuntu 14.10 Utopic támogatással, de ha Trusty vagy korábbi kiadásod van, akkor minden rendben lesz.

Ha Utopic van a gépeden, akkor a sources.list fájl megfelelő sorait kell szerkesztened úgy, hogy az

„utopic main”

végű sorokat átírod erre:

„trusty main”

Most frissítenünk kell a csomag-

listát:

```
sudo apt-get update
```

És végül telepítsük:

```
sudo apt-get install gqr  
sudo apt-get install dump1090
```

Ez eltarthat egy kis ideig, mert a hozzávetőleg 125 MB-os GQRX számos függősége is települni fog. Ilyen például GNURadio is. Ne aggodj, mert ha majd megunod a repülő követését, akkor a GQRX-et használhatod számtalan rádiófrekvencia kereséséhez.

A GQRX valójában nem az, amit akarunk, hanem a függőségei miatt van rá szükségünk. A dump1090 alkalmazás az, ami kell nekünk.

ronnie : dump1090 – Konsole												
File	Edit	View	Bookmarks	Settings	Help							
Hex	Mode	Sqwk	Flight	Alt	Spd	Hdg	Lat	Long	Sig	Msgs	Ti/	
405f78	S								5	2	34	
40610e	S	5441	BEE3EY	23975					10	264	0	
400ad2	S			grnd					9	6	44	
aaf5b1	S	2161	DAL163	36000	461	297			9	52	0	
484166	S	7333	KLM29L	125	132	226	55.897	-4.386	12	147	18	

Hogyanok – RTL-SDR rádió

Csatlakoztasd a kűtyűd, nyiss egy terminált és írd be:

```
dump1090 -interactive
```

Eltarthat 1-2 percig is, amíg keresi és megtalálja a repülőgépeket, de hamarosan a terminál-ablakban megjelennek a repülő ID-i, squawk-kódjai és a vétel minőségétől függően a magassági és a sebességi adatok, valamint a GPS koordináták is.

Tetszik? Térképen is látni akard mindezt? Szerencséd van, mert a dump1090 egy beépített szerveret is tartalmaz.

Ha fut a dump1090, akkor a beállításához nyomj kétszer Ctrl+C-t.

Terminálba gépeld be a következőt:

```
dump1090 --interactive --net  
--net-beast --net-ro-port  
31001
```

És itt a varázslat: nyiss egy böngészőt és navigálj a következő helyre:

<http://localhost:8080>

Szemmel láthatóan az egyetlen repülő a térképen az, ami sugározza a GPS koordinátáit. Elképesztő belegondolni, hogy ezek a gépek

mennyi információt szórnak titkosítatlanul.

Ebben a cikkben csak felületesen érintettem az RTL-SDR eszközöket.

Próbálg meg beállítani egy GQRX-et és keres a frekvenciák között. Kissé hihetetlen a sok furcsaság, amit találni fogsz. Sokat közülük kezdetben nem értesz, de kis kutatás után találni fogsz néhány közönséges FM rádióállomást és pár más adást is, amit például egy garázkapu-nyitó vagy egy autóriasztó sugároz.

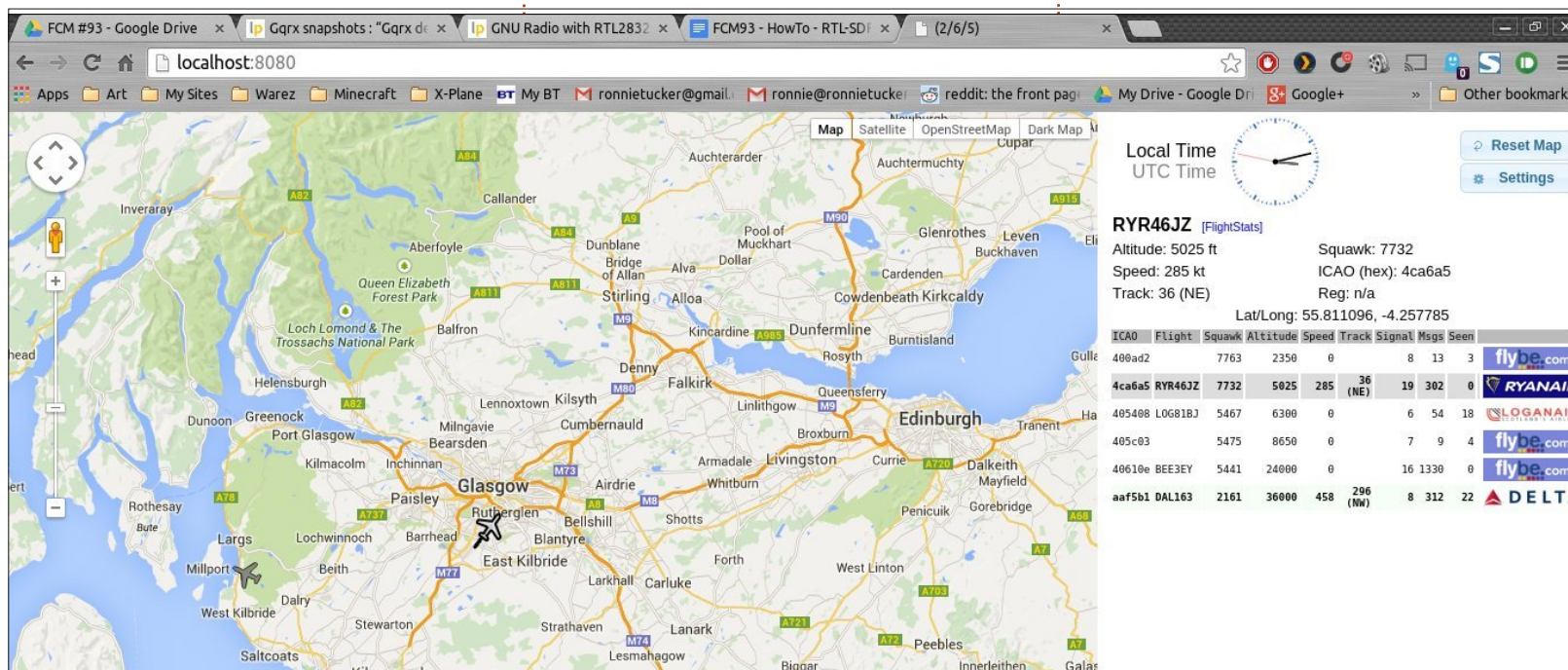


Az Ubuntu Podcast lefedi a legfrissebb híreket és kiadásokat, amik általában érdekelhetik az Ubuntu Linux felhasználókat és a szabadszoftver rajongókat. A műsor felkelti a legújabb felhasználók és a legöregebb fejlesztők érdeklődését is. A beszélgetésekben szó van az Ubuntu fejlesztéséről, de nem túlzottan technikai. Szerencsések vagyunk, hogy gyakran vannak vendégeink, így első kézből értesülünk a legújabb fejlesztésekről, ráadásul olyan módon ahogyan mindenki megérti! Beszélünk továbbá az Ubuntu közösségről is, és a benne zajló dolgokról is.

A műsort a nagy-britanniai Ubuntu közösség tagjai szerkesztik. Mivel az Ubuntu viselkedési kódexnek megfelelően készítik, bárki meghallgathatja.

A műsor minden második hét keddjén előben hallgatható (brit idő szerint), másnap pedig letölthető.

podcast.ubuntu-uk.org





Az utolsó cikkemben bemutatam neked, hogyan használd az Érvénytelen adatok megjelölése eszközt a Calc Detektív eszközeiben. Ezúttal a Detektív menüben lévő maradék eszközöket fogom kifejteni, amelyek abban segítenek nekünk, hogy nyomon kövessük a képletekben használt cellákat. Ezeken az eszközökön keresztül megtanulhatjuk, mely cellák befolyásolják egy képlet eredményét, mely képleteket befolyásol egy cella, és mely cellákat tartalmaz az a képlet, amely hibát eredményez. A munkalap, amelyet használni fogunk kicsi, csak segít neked annak megértésében, hogyan működnek, de ezek az eszközök bonyolult képletekkel ellátott nagy munkalapokon a leghasznosabbak.

A MUNKALAP BEÁLLÍTÁSA

Hozz létre egy új, üres munkalapot. Az A1 cellába írd be a 13-as érté-

2	14	15	16	17	62
3	15	16	17	18	66
4	16	17	18	19	70
5	17	18	19	20	74
6	18	19	20	21	78
7	93	99	105	111	816
8	Complete Sample Sheet				

ket. Válaszd ki az A1:A6 cellákat. Válaszd a Szerkesztés > Kitöltés > Sorozatokat, hogy megjelenítsd a Kitöltés sorozattal párbeszédablakot. Az Iránynál válaszd a Le, a Sorozat típusánál pedig az Automatikus kitöltés opciót. Kattints az OK gombra, és a cellákat kitölti sorszámokkal. Válaszd ki az A1:D6 cellákat. Válaszd a Szerkesztés > Kitöltés > Sorozatokat. Ezúttal válaszd a Jobbra opciót az Iránynál, de ismét az Automatikus kitöltést válaszd a Sorozat típusánál. Kattints az OK gombra, és a jobbra lévő cellákat kitölti sorszámokkal.

Hozz létre összegeket a sorokhoz és oszlopokhoz úgy, hogy kiválasztod a sor vagy oszlop végén lévő üres cellát. Kattints a Képlet eszköztáron lévő Összeg gombra. A sorokhoz a baloldali, az oszlopokhoz pedig a felső cellákat választja ki automatikusan. Nyomd meg az Enter billentyűt az eredmények elfogadásához. Ezt ismételd meg minden egyes sorhoz és

oszlophoz.

Az E7 cella össze fogja adni mind a sorok, mind az oszlopok összegeit. A képlet a következő:

=SZUM (A7 : D7 ; E1 : E6)

ELŐZMÉNYEK FELDERÍTÉSE

Az Előzmények felderítése funkció megmutatja a jelenlegi cellában lévő képlet által használt cellákat. A nyomokat a munkafüzetben kék nyilakkal és a cellatartományok körüli kék kerettel jeleníti meg. Ez minden nyomkövető funkcióra igaz.

Válaszd ki az E7 cellát. Válaszd az Eszközök > Detektív > Előzmények felderítése menüpontot. Kék nyilak és keretek mutatják a képlet által használt cellákat. Ha helyesen írtad be a képletet, akkor ez a sorokhoz és oszlopokhoz tartozó végösszegeket tartalmazza. Hogy kitöröld a munka-

lapról a nyomkövetési jeleket, válaszd az Eszközök > Detektív > Előzmények eltávolítása menüpontot.

FÜGGŐSÉGEK FELDERÍTÉSE

A függőségek felderítése azokhoz a cellákhoz húz nyilakat, amelyek olyan képleteket tartalmaznak, melyek a jelenleg aktív cellára támaszkodnak. Ha kiválasztod a C4-et, majd az Eszközök > Detektív > Függőségek felderítése menüpontot, akkor a C7-re és az E4-re mutató nyilakat kapsz. Mindkét cella olyan képleteket tartalmaz, amelyek a C4-re hivatkoznak a tartományukban. Ugyanúgy, mint az Előzmények felderítésénél, a két képlet tartományait kék doboz veszi körül. A nyomkövetési jelek törléséhez válaszd az Eszközök > Detektív > Függőségek eltávolítása menüpontot.

GYAKORI SZÁMOLÁSI HIBÁK

1	13	14	15	16	17	62
2	14	15	16	17	18	66
3	15	16	17	18	19	70
4	16	17	18	19	20	74
5	17	18	19	20	21	78
6	18	19	20	21		
7	93	99	105	111		816
8	Trace Precedents					

	A	B	C	D	E
1	13	14	15	16	58
2	14	15	16	17	62
3	15	16	17	18	66
4	16	17	18	19	70
5	17	18	19	20	74
6	18	19	20	21	78
7	93	99	105	111	816
8	Trace Dependents				

Mielőtt a hibafelderítésről beszélünk, vessünk egy gyors pillantást néhány gyakori hibára, amit a Calcban láthatsz.

#ZÉRÓOSZTÓ! (DIV/0!) – A képlet megpróbál egy számot nullával osztani. Ahogy régen az elemi matematikában megtanultuk, nem tudsz számot semmivel elosztani. Néhány függvény, mint pl. a VARP és STDEV akkor adja ezt a hibát, ha nem használasz minimálisan szükséges argumentumot.

#HIV! (REF!) – A cella- vagy tartományhivatkozás érvénytelen. Ez főként akkor fordul elő, ha elírod a hivatkozást, vagy olyan cellára vagy tartományra hivatkozol, amely már nem létezik. Például olyan munkalapra mutató hivatkozásod van, amelyet kitöröltél.

#ÉRTÉK! (VALUE!) – A bevitt adat típusa más, mint amit elvártak. Ezt gyakran az okozza, ha egy matematikai függvényben vagy képletben szöveges formátumú cellára hivatkozunk.

Hiba(Err):510 – Egy változó hiányzik a képletből. Ez akkor fordulhat elő, ha két operátort együtt írsz be, mint pl. =1+* 3.

HIBA FELDERÍTÉSE

A Hiba felderítése nyomkövetési nyilakat húz az összes olyan előzménycellába, amely hibát okozott a jelenleg kiválasztott cella képletében. Ennek bemutatásához szükségünk lesz arra, hogy hibát generáljunk. Valószínűleg a #ZÉRÓOSZTÓ!-t a leg-egyszerűbb generálni. Írd be az =C3-D2 képletet az E8 cellába. Az eredmény nulla. Az E9 cellába írd be az =E7/E8 képletet. Ez #ZÉRÓOSZTÓ! hibát fog generálni.

A hiba felderítéséhez válaszd ki az

0	21	78
5	111	816
Trace Error		0
		#DIV/0!

E9 cellát. Válaszd az Eszközök > Detektív > Hiba felderítése menüpontot. Egy nyomkövetési nyilat húz azokon a cellákon keresztül, amelyeket a képlet tartalmaz. A mi esetünkben ez csak két cella. A mínuszt pluszra módosíthatod, és a nyíl törlődni fog, a hiba pedig eltűnik, de most hagyjuk úgy, ahogy van. Van még egy dolog nálam, amit meg szeretnék neked mutatni. Az egyetlen módszer arra, hogy kitörölj egy nyomkövetési jelet a hiba kijavítása nélkül, az Eszközök > Detektív > Minden felderítés eltávolítása menüpont.

A FELDERÍTÉSEK KOMBINÁLÁSA

Egy hiba nyomon követése nem mindig olyan egyszerű, mintha csak egy nyomkövető eszközt használunk. Lehet, hogy szükséged van arra, hogy ezeket kombináld a tényleges probléma megtalálásához. Vegyük a hibánkat példának. Kiválaszthatjuk az E9 cellát és futtathatjuk a hiba felderítését, és láthatjuk, hogy a képlet a fölötté lévő két cellát használja. Viszont ha rájössz, hogy mindkét cella képletet tartalmaz, felderítheted az E8 cellára vonatkozó előzményeket, hogy lásd, megtalálod-e a problémát. Rájössz arra, hogy a képletben használt két cella azonos értékű. Eszedbe

jut; ez azt jelenti, hogy a C6-ot akartad használni nem a C3-at. Módosítsd a C3-at C6-ra az E8-ban lévő képletben, és a hiba, valamint a hibafelderítő nyíl eltűnik. Szintén fontos megjegyezni, hogy az előzmény-felderítések a képlet módosításával együtt mozognak.

A Calcban lévő Detektív felderítő eszközeivel nyomon követheted a képletben szereplő cellákat és meghatározhatod, hogy miért nem a várt eredményeket kapod. Az Előzmények felderítése megmutatja neked az aktuális cellában lévő képlet által használt cellákat. Használd a Függőségek felderítését, hogy megmutassa azokat a cellákat, amelyek olyan képleteket tartalmaznak, amelyek a jelenleg kiválasztott cellát használják. Ha azt próbálsz meghatározni, miért generál hibát egy képlet az aktuális cellában, akkor a Hiba felderítése eszközt használod. Végül kombinálhatod a különböző felderítő eszközöket, hogy átfogóbb képet kapj arról, hogy mi történik, vagy mi nem történik a munkalapodon belül.

	C	D	E
4	15	16	58
5	16	17	62
6	17	18	66
7	18	19	70
8	19	20	74
9	20	21	78
9	105	111	816
Combining Traces			0
			#DIV/0!



Ha egy nyilvános helyen dolgozunk, mint például egy könyvtárban vagy egy kórházban, akkor néha hasznos lehet a látogatók rendelkezésre bocsátani egy számítógépet. Egy ilyen számítógépet, még akkor is, ha billentyűzettel és egérrel van ellátva, úgy kell beállítani, hogy ne használhassák az összes rendszerfunkciót, csak egy korlátozott részt. Például, az emberek csak az Internet eléréséhez használhassák, vagy csak egy belső weboldalhoz férjenek hozzá, amely a cégről vagy a tevékenységekről szolgál információval. Úgy kell beállítani a rendszert, hogy megakadályozza a parancssoros hozzáférést és más, olyan programok elindítását, amelyeket nem akarunk a látogatók számára biztosítani.

Az ilyen számítógép a kioszk. (http://en.wikipedia.org/wiki/Interactive_kiosk).

Végül is beállíthatunk egy PC-t egy nagy TV-re kötve, billentyűzet és egér nélkül, hogy információs diákat vagy videókat mutasson. Az utóbbi célhoz, megnézheted a Xibot, ami egy digitális tartalomkeze-

lő rendszer (<http://xibo.org.uk>).

VÁGJUK BELE!

Kezdjük az Ubuntu Server 14.04 LTS telepítésével. Miért Ubuntu Server? Nos, mert egy kioszkon nincs szükségünk túl sok szoftverre. Itt jegyezzük is meg a `--no-install-recommends apt-get` paramétert.

Figyeljünk arra is, hogy a kioszk-környezet konfigurációja után azt feltételezzük, hogy a számítógépet SSH-n keresztül érjük el, és nem a terminálból.

Tehát, telepítsük az Ubuntu-t, bekapcsolva az automatikus frissítéseket és telepítve az OpenSSH kiszolgálót.

Az első újraindítás után jelentkezünk be mint az (administrator) a felhasználó, amelyet a telepítés során hoztunk létre, majd frissítsük a rendszert:

```
sudo apt update
```

```
sudo apt upgrade
```

```
#!/bin/bash

home_tar="/opt/kiosk/kiosk_home.tar"

rewritehome() {
    echo $home_tar
    if [ -f $home_tar ]
    then
        rm -rf $HOME/*
        tar xvf $home_tar -C /
    fi
}

xset -dpms
xset s off
openbox-session &

rewritehome
xscreensaver -nosplash &

while true; do
    rewritehome
    /usr/bin/firefox
done
```

És telepítsük fel ezeket a csomagokat:

```
sudo apt install --no-install-recommends xorg openbox
firefox xscreensaver plymouth-theme-sabily cups
```

Telepítsük a CUPS kiszolgálót, ha azt tervezzük, hogy engedélyezzük a nyomtatást a felhasználóknak.

Hozzunk létre egy felhasználót,

kiosk néven.

```
sudo useradd -m kiosk
```

```
sudo mkdir /opt/kiosk
```

ÁLLÍTSUK BE A KÖRNYEZETET

Ebben a konfigurációban nem fogunk ablakkezelőt indítani, de meghívjuk az X11-et és csak a Firefoxot indítjuk el.

A fent látható parancsfájlt (előző oldal, jobbra fent) fogja meghívni az upstart a rendszer indulásakor. Ez lényegében lecseréli a kiosk felhasználó saját mappáját a Firefox indítása előtt. A Firefox elindítása egy ciklusba van téve, hogy mindig újrainduljon, ha bezárják.

```
sudo chmod +x
/opt/kiosk/kiosk.sh
```

Most létre kell hoznunk egy upstart parancsfájlt:

```
sudo vi /etc/init/kiosk.conf
```

```
start on (filesystem and
stopped udevtrigger)
```

```
stop on runlevel [06]
```

```
console output
emits starting-x
```

```
respawn
```

```
exec sudo -u kiosk startx
/etc/X11/Xsession
/opt/kiosk/kiosk.sh
```

Mivel az X-nek root jogosultság szükséges az induláshoz, ki kell adnunk ezt a parancsot, és ki kell választanunk az „Anybody” lehetőséget.

```
sudo dpkg-reconfigure x11-
common
```

Most már majdnem elkészül-

tünk. Újra kell indítanunk a gépet.

```
sudo reboot
```

És a Firefoxnak automatikusan el kell indulnia.

Tegyük meg a szükséges módosításokat most, mint például a CA tanúsítványok, biztonsági eszközök vagy egyéb bővítmények hozzáadását, stb.

Végül fel kell telepítenünk az srkiosk bővítményt:

<https://addons.mozilla.org/en-US/firefox/addon/srkiosk/>

Figyelmesen olvassuk el a dokumentumokat (<http://blogoleless.blogspot.it/2012/07/firefox-srkiosk-add-on-full-screen.html>), hogy jobban megértsük a kézzel készült fájlt és könnyebben tudjuk a saját igényeinkre szabni, például lecserélni a kezdőlapot vagy megjeleníteni a címsort.

Zárjuk be a Firefoxot. Mostantól a Firefox lezárt, kiosk-módban fog indulni.

A konfiguráció folytatásához, be kell jelentkeznünk távolról, SSH-val.

UTOLSÓ LÉPÉSEK

A képernyővédő használata hasznos is lehet – nem csak egy szép effekt a kijelzőn egy bizonyos idő eltelte után. Valamint arra is jó, hogy „visszaállítsa” a munkaménetet valamennyi tétlenség után. Például jó dolog visszatérni a kezdőoldalra, ahelyett hogy a kiosk az előző felhasználó utolsó meglátogatott oldalán maradjon. Tehát csak akkor kell beállítanunk a képernyővédőt, ha időnként vissza akarjuk állítani a kioszkot.

Helyezzük az xscreensaver konfigurációs fájlt a kiosk felhasználó saját mappájába. A timeout opció az, amit meg kell változtatnunk hogy megfeleljen az igényeinknek (ebben a példában a kiosk negyedóránként töltődik újra).

```
sudo vi
/home/kiosk/.xscreensaver
```

```
timeout:          0:15:00
lock:             False
verbose:          False
splash:           False
captureStderr:    True
dpmsEnabled:      False
dpmsQuickOff:     False
mode:             one
selected:         0
programs: resetkiosk.sh -root
\n\
```

Ezután szükségünk van a parancsfájltra, amit az xscreensaver meghív a tétlenségi idő lejártá után. Mint mondtam, az xscreensaver, a tényleges képernyővédő helyett, ezt a parancsfájlt fogja futtatni.

```
sudo vi
/usr/lib/xscreensaver/re
setkiosk.sh
```

```
#!/bin/bash
```

```
PID=`ps -u $UID -f |grep
firefox |grep -v grep | awk
'{ print $3 }'`
kill $PID
```

KÉSZÍTÜNK EGY PILLANATKÉPET A SAJÁT MAPPÁRÓL

Végül létre kell hoznunk egy „pillanatképet” a kiosk felhasználó saját mappájáról. Ez a pillanatkép felül fogja írni a saját mappát minden újraindításkor, munkaménetet visszaállításkor vagy a Firefox bezárásakor.

Tehát ne felejtsük el, hogy amikor módosításokat végzünk a kiosk felhasználó saját mappájában, akkor készítenünk kell egy tar fájlt a mappáról, és a megfelelő helyre kell tenni: vessünk egy pillantást a kiosk.sh parancsfájltra.

```
rm -rf /home/kiosk/.cache/

tar cpvf /opt/kiosk/kiosk_ho-
me.tar /home/kiosk/
```

A GRUB KONFIGURÁCIÓJA ÉS A TTY-OK LETILTÁSA

Egy másik hasznos végrehajtandó feladat a rendszer megerősítésével kapcsolatos. Például megakadályozhatjuk a csintalan felhasználókat abban, hogy egyfelhasználós módban indítsák el a rendszert, vagy hogy átváltsanak a virtuális konzolok között (ezek azok a terminálok, amelyeket a funkciógombokkal érhetünk el, mint a CTRL+ALT+F1).

Kezdjük a GRUB-bal. Először is, olyan csendesnek (kevésbé bőbeszédűnek) kell lennie, amennyire csak lehetséges, már csak esztétikai szempontból is.

```
sudo vi /etc/default/grub
```

```
...
GRUB_TIMEOUT=0
GRUB_CMDLINE_LINUX_DEFAULT="q
uiet splash vga=current
loglevel=0"
...
```

Ezután beállíthatunk egy jelszót, hogy ne engedjük a felhasználóknak az egyfelhasználós módban indítást. Használjuk a grub-mkpasswd-pbkdf2 parancsot egy titkosított jelszó létrehozásához, majd szerkesszünk néhány fájlt.

Tedd a következő sorokat a /etc/grub.d/00_header fájl végére

```
cat << EOF

set superusers="root"

password_pbkdf2 root

<the_sting_obtained_from_the_
grub-mkpasswd-pbkdf2_command>
EOF
```

Ezután felhasználónevet és jelszót fog kérni a GRUB minden egyes menübejegyzéshez, még a Linux rendes indításához is, ez az amit nem szeretnénk. Ezért adjuk hozzá az --unrestricted paramétert minden egyes menübejegyzésnél a konfigurációs fájlokban. Például a /etc/grub.d/10_linux fájlban

```
...
echo "menuentry '$(echo
$title' | grub_quote)'
--unrestricted ${CLASS}
\${menuentry_id_option
'gnulinux-${version}-${type}-
\${boot_device_id}' {" | sed "s
/^/\${submenu_indentation}/"
else

echo "menuentry '$(echo
"$os" | grub_quote)'
```

```
--unrestricted ${CLASS}
\${menuentry_id_option
'gnulinux-simple-
\${boot_device_id}' {" | sed
"s/^/\${submenu
_indentation}/"
...
```

Így a felhasználónevet és a jelszót csak akkor fogja kérni, ha a felhasználó szerkeszteni szeretné a GRUB bejegyzéseket az „e” billentyűvel.

Most ki kell adnunk a sudo update-grub parancsot, hogy a változások érvénybe lépjenek.

Végül letilthatjuk a virtuális terminálokat a tty fájlok törlésével az /etc/init mappából:

```
sudo rm /etc/init/tty*
```

TOVÁBBI MUNKÁK ÉS KONK-LÚZIÓ

Ahhoz, hogy megerősítsük a gépet, célszerű BIOS-jelszót beállítani, és megakadályozni a CD/DVD-ről vagy USB-portról történő indítást. És még az USB-ről: a portok használatát érdemes úgy megakadályozni, hogy a felhasználók ne férjenek hozzá a PC-hez.

Egyéb teendők? Állítsunk be egy saját témát indítóképernyőnek. Keressünk módszereket a lehetséges gyorsbillentyűk letiltására.

A Firefox futtatása helyett, lehet hogy egy videolejátszót vagy diavetítést szeretnénk futtatni: megpróbálhatjuk úgy, hogy egy másik parancsot írunk bele a while ciklusba.

Ez a cikk egy itt található bejegyzésen alapul:

<https://thepcspy.com/read/building-a-kiosk-computer-ubuntu-1404-chrome/>

Lehet, hogy egy kiosk-megoldás jobban is kivitelezhető, de amennyire tudom, nem létezik jobb alternatíva. Ez a megoldás megfelel az igényeimnek: lehet hogy neked is hasznos lesz vagy kiindulópontként használható egy jobb megvalósításhoz.

Get *unlimited* access to a cutting-edge technology and business library with **Apress Access!**

For **\$199**

YOU GET:

- Unlimited access to Apress titles for a full year
- Instant access to each new Apress publication
- Compatibility with any device—desktop, laptop, or mobile
- Use of our new exclusive-to-Apress reader with unparalleled search functions
- Option to download any eBook for just \$4.99 for a limited time



www.apress.com |  @apress

Want more info? Check out www.apress.com/subscription



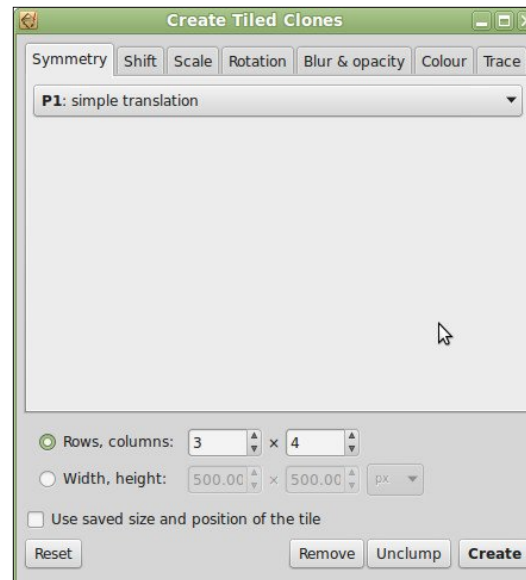


Hogy befejezzük a kalandozásainkat a klónok világában, néhány cikket az Inkscape „Csempézett klónok létrehozása...” párbeszédablak bemutatásával töltök (amire ezután csak Csempézett klónok ablaknak hívok). Ez talán a leg-sokoldalúbb és leginkább zavarba ejtő párbeszédablakok egyike az egész alkalmazásban, így szánjunk rá néhány percet, hogy felkészüljünk mielőtt belemerülünk.

Már láthattuk az egyszerű módszerrel egy klón létrehozására: válasszuk ki a szülőobjektumot és nyomjuk meg az ALT-D billentyűkombinációt (vagy használjuk a Szerkesztés > Klónozás > Klón létrehozása menüpontot). Ha egy második klont is szeretnénk ugyanattól a szülőtől, akkor vagy megismételjük a folyamatot, vagy az első klont másoljuk le (ezúttal a CTRL-D-vel). Egy maréknyi klont szeretnénk? Húzzuk át az elsőt a képernyőn, és szúrjunk be másolatokat a szóköz billentyű időnkénti lenyomásával. De mi van akkor ha száz klont szeretnénk, vagy ezret? És mi van, ha pontosan szeretnénk őket elhelyezni? Vagy minden klont el sze-

retnének forgatni vagy átméretezni egy kicsit? A Csempézett klónok ablak ezt mind tudja, sőt jóval többet is.

Szóval kezdjünk azzal, hogy létrehozunk egy objektumot, hogy a klónjaink szülője legyen. Az egyszerűség kedvéért egy lekerekített sarkú négyzetet fogok használni, de a szülő lehet szinte bármilyen egyedi objektum, vagy csoport. A 3D dobozok viszont nem működnek a párbeszédablakkal – bár először átalakíthatjuk ezeket egyszerű alakzatok csoportjává a hatás eléréséhez, ha nem bánjuk hogy elveszítjük a lehetőséget a szülőobjek-



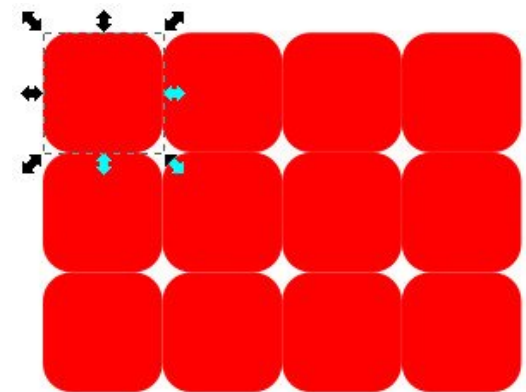
tum 3D dobozkénti szerkesztésére. Egy szülőobjektumot kiválasztva nyissuk meg a Csempézett klónok ablakot a Szerkesztés > Klónozás > Csempézett klónok létrehozása... menüpont segítségével.

Kezdetben a párbeszédablak nem tűnik túl összetettnek, de ha végig lépkedünk a lapokon, hamar megláthatjuk hogy sok mező és vezérlő el van rejtve. Az első lap: a Szimmetria, csupán egy legördülő menüből áll, de az is 17 különböző beállítást tartalmaz. Egyelőre az elsőnél maradunk: „P1: egyszerű eltolás”. Ez lehetővé teszi, hogy egyszerű szögletes klón-tömböket hozzunk létre és ez a legegyszerűbb módja annak, hogy megértsük ezt a párbeszédablakot.

A lapfülek között van néhány globális beállítás a létrehozandó klónok darabszámára vagy a lefedni kívánt területre vonatkozóan. Figyeljük meg, hogy az mondtam „lefedni” és nem „kitölteni”. Gondoljunk egy fürdőszoba csempézésére: elég csempének kell lennie hogy lefedje a falat, még akkor is ha túlcsúszik, és néhányat el kell

vágni a sarkoknál. Hasonlóan, az Inkscape elég csempét fog létrehozni, hogy lefedje a megadott területet (szélességgel és magassággal megadva), ránk bízva hogy levágjuk ha esetleg túlnyúlik. A „csempe elmentett méretének és pozíciójának használata” beállítás legyen kikapcsolva egyelőre – egy későbbi cikkben erre is kitérünk.

A csempék létrehozásához először nyomjuk meg a Visszaállítás gombot. Ez visszaállítja az értékeket ésszerű eredeti értékekre, ez általában egy jó kiindulópont. Most adjunk meg valamilyen értéket a „sorok, oszlopok” mezőkhöz. Én 3x4-es klón-tömbbel fogok kezdeni. Végül a szülő objektumot kijelölve, kattintsunk a Létrehozás gombra.

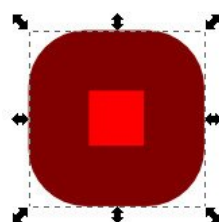


Van néhány dolog amit érdemes megfigyelni a frissen létrehozott klónozott tömbről. Először is, vegyük észre hogy a szülő objektum maradt kijelölve. Ez egyszerűvé teszi, hogy a Törlés gombot megnyomva eltávolítsuk az összes klónt, ha nem vagyunk elégedettek a végeredménnyel. Tartsuk észben, hogy az első klón közvetlenül a szülő objektum felett van. Ha megváltoztatjuk a fókuszt, akkor újra ki kell jelölnünk a szülőt, a bal felső, sarokra kattintva, többnyire a klónt fogjuk kiválasztani. Az egyszerű válasz az hogy csak jelöljük ki bármelyik klónt, és nyomjuk meg a SHIFT-D-t vagy használjuk a Szerkesztés > Klónozás > Eredeti kijelölése menüpontot.

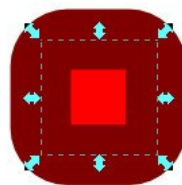
Vessünk egy pillantást a klónok pozíciójára. Miután visszaállítottuk a párbeszédablak paramétereit az alapértékekre, az alapértelmezett viselkedés az hogy a klónok egymást érintve helyezkednek el. Amikor objektumokat jelölünk ki az Inkscape-ben, akkor a szaggatott vonalat, ami jelzi a kijelölést, „határoló téglalap”-nak hívjuk. A doboz méretei határozzák meg a klónok pozícionálásának alapját. Az első oszlop a téglalap szélességének 100%-ával kerül eltolásra. A máso-

dik 200%-kal, és így tovább. A sorok azonos szabályokat követnek a téglalap magasságával. Ez kicsit absztrakt módszernek tűnik a klónok pozícionálásának leírására, de az Eltolás és Méretezés lapfülek a „csempeszélesség/magasság százalékát” használják mértékegységként, így a legegyszerűbb ezeket a kifejezéseket használni.

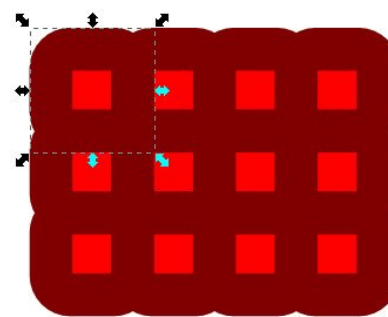
Hogy bonyolítsuk a dolgokat, az Inkscape-nek két különböző határoló téglalapja van: a vizuális határoló, ami tartalmazza az objektum körvonalát, és a geometria, amely csak magán az objektumon alapul, függetlenül a körvonalától. Kiválaszthatjuk hogy melyiket használja az Inkscape kijelöléskor a Szerkesztés > Beállítások párbeszédablakon. A kettő közti különbség világos, ha kijelölünk egy vastag körvonalú objektumot.



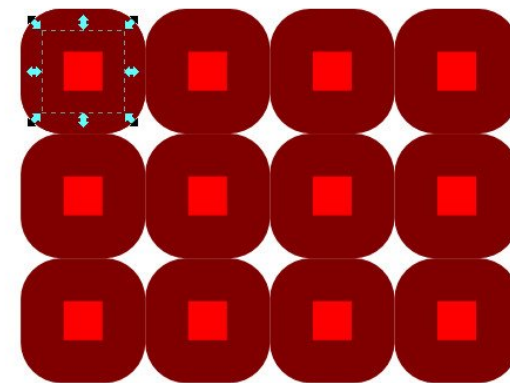
**Visual
Bounding
Box**



**Geometric
Bounding
Box**



**Visual
Bounding
Box**

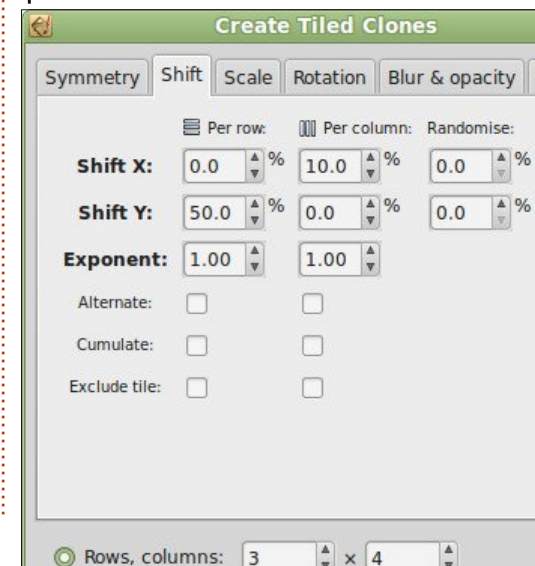


**Geometric
Bounding
Box**

A dokumentáció szerint, az Inkscape mindig a geometriai határoló téglalapot használja a klónok létrehozásakor. Ez nem egyezik a tapasztalataimmal az Inkscape 0.48-ban, Linux Mint 17-en. Úgy találtam, hogy a geometriai és a vizuális határoló téglalap beállítás módosításának van hatása a csempézésen. Még meglepőbb, hogy a működés pont fordítottja annak amit várnánk. Nézzük meg ugyanazt a vastag körvonalú négyzetet fent amikor a két beállítással van csempézve.

Innentől kezdve, többnyire maradok a körvonal nélküli alakzatoknál a csempézett klónok bemutatása során, így ez a különbség nem lesz probléma. De tartsuk ezt észben, hogy ha körvonalas objektumokat

akarunk csempézni és a klónok nem ott jelennek meg ahol várnánk. Az alapértelmezett elrendezés lehet hogy megfelelő ha csak arra akarjuk használni a párbeszédablakot hogy sok klónt hozzunk létre. De az igazi ereje a számtalan pozícionálási és transzformációs le-



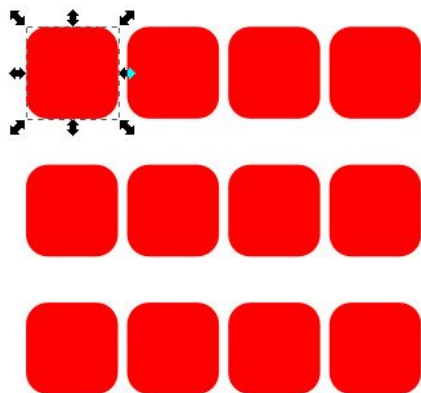
hetőségben rejlik. Kezdjük a tömb fellazításával, az Eltolás lapfül segítségével.

A lapfül megértésének kulcsa az, hogy a vezérlőelemek három oszlopba vannak rendezve. Az első oszlop azt határozza meg hogy, mennyivel lesznek eltolva a sorok; a második azt hogy, mennyivel lesznek az oszlopok eltolva; a harmadik minden klónra érvényes és véletlenszerű eltolást ad hozzá az x és y tengelyek mentén.

Kezdetnek egy egyszerű példa: megtartva a négyszögletes tömböt, miközben egy kis plusz helyet adunk a klónok közé. Az első dolog amit akarunk, az a hely hozzáadása a felső-középső vezérlőelembe. Ez a mező módosítja az x-pozíciót a klónok minden oszlopában. A 10 érték itt a határoló téglalap szélességének 10%-át fogja hozzáadni minden oszlop x-pozíciójához, így ahelyett hogy 100%, 200%, 300%...-nál helyezné el őket, 110%, 220%, 330%...-nál lesznek elhelyezve – minden következő pozíció 110%-kal növelt az alapértelmezett 100% helyett.

Emellett beírtam az 50 értéket a második sor első vezérlőelemébe is. Ez a határoló téglalap magassá-

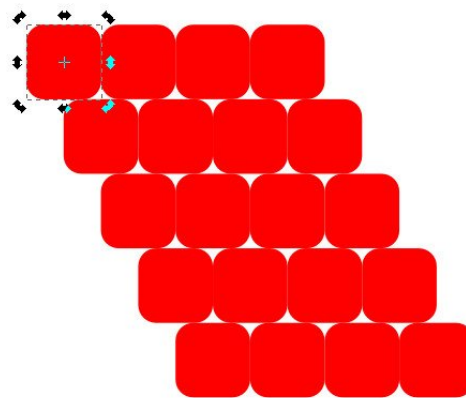
gának 50%-át fogja hozzáadni minden sor y-pozíciójához. Így a sorok 150%, 300%, 450%...-nál lesznek. Végeredményként fellazítottuk a tömböt, több függőleges térközzel mint vízszintessel.



Ha nem akarjuk hogy a klónok ilyen szigorúan legyenek elhelyezve, egyszerűen írunk be egy pozitív számot az egyik (vagy mindkét) Véletlenszerűség mezőbe a harmadik oszlopban. Az érték amit ide írunk egy felső korlátot jelent, így 20-at beírva az x irányú eltolás vezérlőelembe megengedi minden klónnak az eltolást a határoló téglalap szélességének legfeljebb 20%-ával. Ez a többi eltolással együtt kerül hozzáadásra, így ugyanúgy fellazíthatjuk az egész tömböt, és hozzáadhatunk egy kis véletlenszerűséget is. Másképp pedig használhatjuk a Manipuláló eszközt (lásd 22. rész), vagy

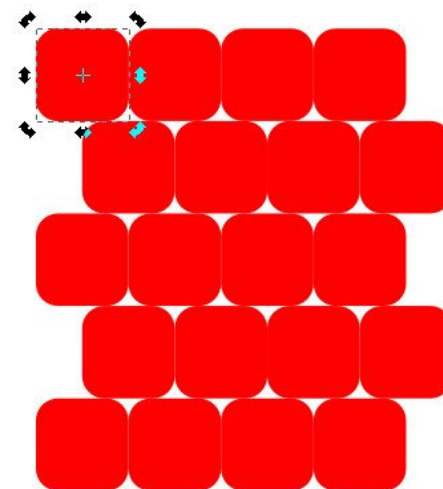
kattintsunk az Egyenletesítés gombra a párbeszédablakon, ami változtat minden klón x és y koordinátáján egy kicsit. Az utóbbi néha hasznos lehet, ha egy kicsit „természetesebb” kinézetet szeretnénk, például egy levél vagy hópehely klónozásánál.

A sima, négyszögletes tömb mellett, az eltolás lapfül érdekesebb eredményeket is el tud érni. Egy egyszerű téglafalat kell rajzolni? Kezdjük egy téglával, de tégy 50% eltolást az első mezőbe. Ez a határoló téglalap szélességének 50%-át fogja hozzáadni minden sor klón x pozíciójához. Ennek hatására minden sor a szülő szélességének a felével mozdul el.

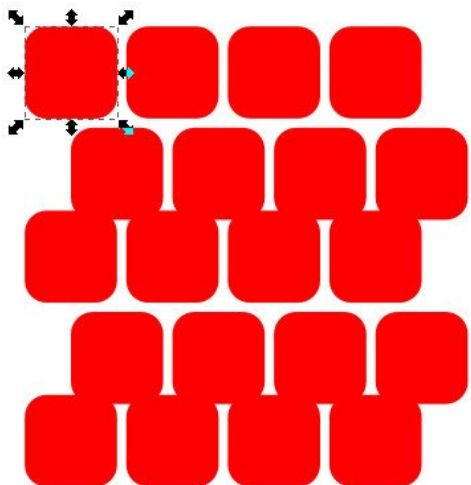


Ezt még falszerűbbé tehetjük a Váltakozás bejelölésével a Soronként oszlopban. Ez azt okozza,

hogy az eltolások egyszer pozitív, másszor negatív értéként kerülnek alkalmazásra. Így a sorok először jobbra tolódnak 50%-kal, majd vissza balra ugyanennyivel, aztán megint jobbra, és így tovább.



A falunk jól néz ki, de nem segítene egy kis habarcs? Vonzó lenne 10%-ot írni néhány másik mezőbe. Ez ugyan sikeresen elosztaná vízszintesen, de a függőleges elosztási kísérletek sikertelenek lesznek, mert a Váltakozás beállítás megváltoztatja az Y irányú eltolás mezőben megadott térközt. Ahelyett hogy állandó 10% térközt tennénk a sorok közé, valójában 10%-ot fogunk hozzáadni, aztán 10%-ot elvenni, majd megint hozzáadni, és így tovább.



Van néhány módja a probléma megoldásának: mind arra épít, hogy hozzáad a szülő határoló téglalap méretéhez, így nincs szükség további térköz hozzáadásához a klónok létrehozása során. Megadhatunk egy vastag körvonalat is és

biztosíthatjuk, hogy a geometriai határoló téglalap legyen használatban. Hozzunk létre a klónokat csak az 50% értékkel az első mezőben és kapcsoljuk be a Váltakozás beállítását hogy további térköz kapjunk a körvonal vastagságától függően. Aztán csak töröljük a körvonalat a szülőobjektumról, és így minden klón azonnal meg fog változni.

Egy másik megoldás erre a problémára, ha adunk némi elmosást a szülőobjektumra. Ez hatással van a vizuális határoló téglalapra is, így ha be van állítva az Inkscape beállításában hogy a geometriai határoló téglalapot használja (emlékezzünk, hogy ez rosszul működik ebben a párbeszédablakban), a klónok további térköz kapnak maguk

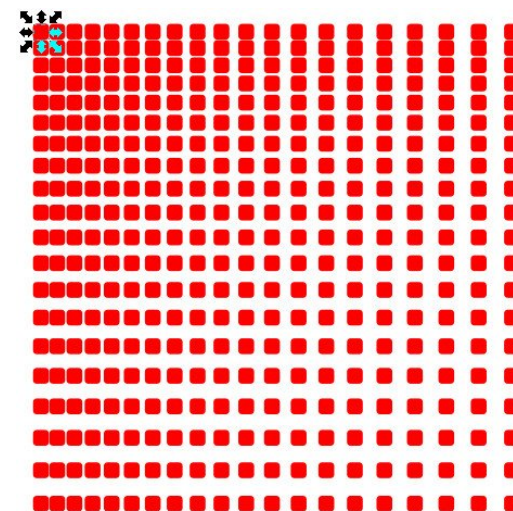
	Per row:	Per column:	Randomise:
Shift X:	50.0 %	10.0 %	0.0 %
Shift Y:	10.0 %	0.0 %	0.0 %
Exponent:	1.00	1.00	
Alternate:	☒	☐	
Cumulate:	☐	☐	
Exclude tile:	☐	☐	

körül. Ezután töröljük az elmosást a szülőről.

Az utolsó megközelítés az hogy a szülőt egy nagyobb objektummal egy csoportba tesszük. A második objektum csak a csoport határoló téglalapjának megadásához kell. Hozzuk létre a klónjainkat, aztán lépünk bele a szülő csoportjába és töröljük az ideiglenes objektumokat. Ez a módszer csoportok klónját eredményezi, nem sima objektumokét, de nem szükséges hozzá az Inkscape beállításainak módosítása.

Az Összeadódás jelölőmezők meghatározzák hogy az eltolások hogyan adódnak hozzá a klónok alappozícióihoz. Általában az eltolás egyszer kerül hozzáadásra a szélességhez vagy a magassághoz, és atz az egy értéket használja minnden egyes sor és oszlop klón esetén. A jelölőmező bejelölése azt jelenti, hogy az eltolás újra és újra hozzáadódik minden sorhoz és oszlophoz, így egyre nagyobb értéket eredményez.

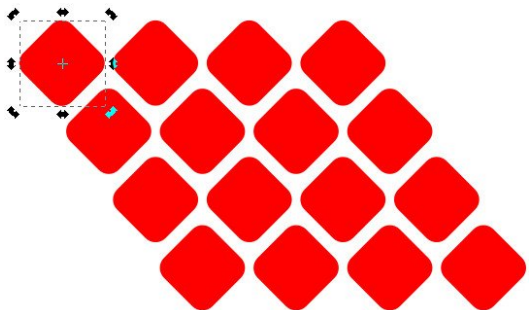
Hasonló hatás érthető el a Kitevő értékének 1-nél nagyobbra állításával. Az ellenkező hatás – a különbség csökkentése az eltolások között – úgy érhető el, hogy 1-nél kisebb Kitevőt állítunk be.



Az utolsó vezérlőelempár ezen a lapfűlön, a „Csempe ne számítson” jelölőelem, amely levonja a határoló téglalap méretét a klónok számított pozíciójából. Azok a beállítások, amik eddig 110%, 220%, 330%-ot jelentettek, most 10%, 20%, 30%-ra változnak, amikor a jelölőmező aktív. Ezzel tudjuk a szülőhöz képest nagyon kis eltolással klónokat létrehozni – egyszerű négyszögletes tömb esetén általában egymást valamilyen módon fedve.

Még egy utolsó dolog: lehetséges negatív eltolást megadni az x és y pozíciókhoz. Ez balra (x esetén) vagy felfelé (y esetén) történő eltolást eredményez, és átállítja a szabványos 100%, 200%, 300% pozíciókat 90%, 180%, 270% értékre,

ha az eltolás 10%. Ez egy másik módja az egymást fedő klónok létrehozásának, de a szülőobjektum alakjától függően lehet hogy pontosan ez kell, hogy minden szépen illeszkedjen.



Próbáljunk játszani néhány érték kombinációjával és a beállítással az Eltolás lapfűlön. Hamar rá fogunk jönni, hogy könnyű létrehozni vad és váratlan klónelhelyezéseket – szerencsére van Visszaállítás gomb! Próbáljuk megérteni, hogy az egyes vezérlőelemek hogyan állítják a klónok elhelyezkedését, és hogyan vonatkozik a három-

oszlopos elrendezés a klón soraira és oszlopaira, mert legközelebb fel fogjuk használni ezt a tudást. hogy felfedezzünk néhány másik lapfület a Csempézett klónok párbeszédablakban.



Mark Inkscape-et használ három webes képregényének elkészítéséhez, a „The Greys”, „Monsters, Inked” és „Elvie” címűekhez, amit a következő oldalon nézhetsz meg: <http://www.peppertop.com/>

	☰ Per row:	☷ Per column:	Randomise:
Shift X:	55.0 ▲▼ %	10.0 ▲▼ %	0.0 ▲▼ %
Shift Y:	-20.0 ▲▼ %	0.0 ▲▼ %	0.0 ▲▼ %
Exponent:	1.00 ▲▼	1.00 ▲▼	



Hogyanok

Írta: Ronnie Tucker
Fordította: Meskó Balázs

Arduino

A múlt hónapban megmutattam hogyan készítettem el az automata tyúkólajtómat. Nos, ahogy az lenni szokott, sikerült egy problémát megoldanom – és aztán kétszer olyan bonyolulttá tennem. A rendszerem az egyszerű szervó és LDR helyett most már egy teljes szervóból, LDR-ből, páratartalom-érzékelőből és LCD kijelzőből áll.

Egy szervó normális esetben 0 és 180 fok között forog, és a kódban megadhatjuk, hogy milyen szögbe forgassuk. Egy teljes szervó (legalábbis amennyire értem) valahol egy motor és egy szervó között helyezkedik el félúton. Úgy lehet kezelni a kódban mint egy szervót, de inkább olyan mint egy motor, abban az értelemben, hogy 360 fokban forgatható, de nem adható meg irányszög. Ez egy kisebb problémát jelentett amikor előrefelé kellett a szervót mozgatni az ajtó felemeléséhez, majd hátrafelé a leengedéséhez. Végül sikerült megoldani a problémát egy kis kontárkodással, a nyitás paranccsal:

```
myServo.write(openDoor);
```

és a map paranccsal, amelybe bele-

dobtam a megfordított számokat:

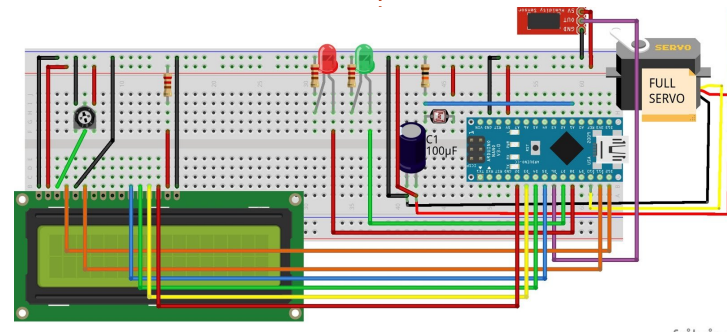
```
reverse=map(openDoor, 0, 1023, 180, 0);
```

```
myServo.write(reverse);
```

Semmi esetre sem elegáns, de működik.

A LCD kijelző már ismerős a lézeres drótakadály projektből. A megvalósítás nagyrészt egy másolás/beillesztés volt, de hozzáadva a páratartalom-érzékelő nehézségeit.

A páratartalom-érzékelő egy DHT11 modell; egyszerűen csak be kell importálni a dht11 függvénykönyvtárat. Ha nincs meg, akkor a fontos fájlok itt megtalálhatóak: <http://playground.arduino.cc/main/DHT11Lib>. A pára- és hőmérő funkciók kicsit trükkösek voltak, hiszen elég bonyolult matek alakítja át a számokat Celsiusban megjeleníthető értékekre. Nem is fogok úgy tenni, mintha érteném, hogy ezek a függvények mit csinálnak. Más kódokból másoltam át őket a DHT11 oldaláról és néhány Google keresésből. A loop()-ban található chk változó megkapja a pin értékét, ha a szenzor be van kapcsolva.



```
int chk =  
DHT11.read(DHT11PIN);
```

Később futtatunk egy switch parancsot hogy ellenőrizzük, hogy az érzékelő be van-e kapcsolva (ismétlem, nem az én kódom). Aztán néhány sor kiírja az átalakított értékeket a kijelzőre:

```
Serial.print("Humidity (%): ");
```

```
Serial.println((float)DHT11.humidity, 2);
```

```
Serial.print("Temperature (°C): ");
```

```
Serial.println((float)DHT11.temperature, 2);
```

Ezeket megtartottam, mert jó megnézni, hogy az érzékelő működik-e az LCD hozzáadása előtt. A végén ugyanazok az LCD kiírási parancsok, mint a lézeres drótaka-

dálnál, csak a páratartalom értékekkel.

```
lcd.setCursor(0, 1);  
lcd.print("Temp. ('C): ");  
lcd.print(DHT11.temperature);
```

Most hogy elkészült a prototípus, már csak be kell raknom az Arduino Nanot (a próbapanellal) egy műanyag csőbe, és hosszabb vezetékekkel kell kivezetnem az LCD kijelzőt és az érzékelőt egy megfelelő helyre.

A végső kódom itt található:
<http://pastebin.com/ES421PHV>.



Ronnie alapítója és szerkesztője a Full Circle magazinnak (még!), rész munkaidős művész, és újdonsült Arduino hacker.



IRÁNYELVEK

Az egyetlen szabály, hogy a cikknek **valahogy kapcsolódnia kell az Ubuntuhoz, vagy valamelyik változatához – Kubuntu, Xubuntu, Lubuntu, stb.**

SZABÁLYOK

- Nincs korlátozva a cikk terjedelme, de a hosszú cikkeket több részre bontva közöljük sorozatban.

- Segítségül olvasd el a **Hivatalos Full Circle Stílus iránymutatást** a <http://url.fullcirclemagazine.org/75d471>

- A cikket bármilyen programmal írhatod, én ajánlom a LibreOffice-t, de a lényeg: **ELLENŐRIZD A HELYESÍRÁST ÉS A NYELVHELYESSÉGET!**

- A cikkedben jelöld meg, hogy hová szeretnél elhelyezni képet, úgy, hogy egy új bekezdésbe írod a kép nevét, vagy ágyazd be a képet, ha ODT (Open-Office) dokumentumot használasz.

- A képek JPG típusúak legyenek, 800 pixel szélességnél ne legyenek nagyobbak és alacsony tömörítést használj.

- Ne használj táblázatot vagy *dőlt, kövér* betűformázást.

Ha a „Fókuszban” rovathoz írsz, kövesd az itt látható irányelveket.

Ha kész vagy elküldeni a cikket, akkor ezt e-mailban tedd az articles@fullcirclemagazine.org címre.

FORDÍTÓKNAK

Ha szeretnéd saját anyanyelvedre lefordítani a magazint, küldj egy e-mailt a ronnie@fullcirclemagazine.org címre és adunk hozzáférést a nyers szövegekhez. Ha kész a PDF, akkor fel töltheted a Full Circle magazin weboldalára.

Hogyan írjunk a Full Circle-be

FÓKUSZBAN

JÁTÉKOK/ALKALMAZÁSOK

Ha játékokról, alkalmazásokról írsz, légy szíves érthetően írd le a következőket:

- a játék nevét
- ki készítette a játékot
- ingyenes, vagy fizetni kell a letöltéséért?
- hol lehet beszerezni (letöltési-, vagy honlapcím)
- natív Linuxos program, vagy kell-e hozzá Wine?
- hogyan osztályoznád egy ötös skálán?
- összegzés a pozitív és negatív véleményről

HARDVER

Ha hardverről írsz, világosan írd le:

- a hardver gyártója és típusa
- milyen kategóriába sorolnád
- a hardver használata közben fellépő hibákat
- könnyű működésre bírni Linux alatt?
- kell-e hozzá Windows driver?
- hogyan osztályoznád egy ötös skálán?
- összegzés a pozitív és negatív véleményről

Nem kell szakértőnek lenned, hogy cikket írj – írd azokról a játékokról, alkalmazásokról és hardverekről, amiket mindennap használasz.



Online
BACKUP

Secure
SYNC

Easy
SHARING

Whether you need to access a document you have stored on a remote server, synchronize data between a Mac, Windows or Linux device, share important business documents with your clients, or just rest easy knowing all of your data is safely, securely, and automatically backed up - SpiderOak's free online backup, online sync and online sharing solution can handle all your needs!

SpiderOak offers a different approach to online backup by combining a suite of services into one consolidated tool - free online backup, synchronization, sharing, remote access, and storage. This difference is further measured in our zero-knowledge privacy policy - the first one ever employed in this setting. Our flexible design allows you to handle data from any operating system (Mac, Windows and Linux) or location (external drives, network volumes, USB keys, etc...) using just one centralized account.

- ➔ Access all your data in one de-duplicated location
- ➔ Configurable multi-platform synchronization
- ➔ Preserve all historical versions & deleted files
- ➔ Share folders instantly in web ShareRooms w / RSS
- ➔ Retrieve files from any internet-connected device
- ➔ Comprehensive 'zero-knowledge' data encryption
- ➔ 2 GBs Free / \$10 per 100 GBs / Unlimited devices

<https://spideroak.com>

Download mobile clients
for **iOS & Android**

JOIN SPIDEROAK NOW
Get 2 Free GBs

Get 25% off any SpiderOak package
with the code: **FullcirclemagFans**



Sorozatunk utóbbi részében bemutattunk néhány apró változtatást a Linux kernelen, módosítást hajtottunk végre a kódon, ami a processzorunkról szolgáltat információkat a /proc virtuális fájlrendszerben. Most az utolsó részben megpróbálunk létrehozni néhány teljesen új kódot, beillesztjük a kernelforrásba, lefordítjuk és mindenestől futtatjuk.

Ehhez ahelyett, hogy képességeket adnánk a fő kernelhez, könnyebbnek tűnik egyszerűen létrehozni egy új modult. Ez kényelmesebb abból a szempontból, hogy nem kell aggódni a kernel elrontása vagy egy nem reagáló rendszer miatt – ne felejtse el, hogy ne egy mindennapi használatban lévő gépen csináld! – és a teszt során az új modul bárhányszor betölthető és eltávolítható a memóriából. Valamint egy külön modul fordítása sokkal kevesebb időbe telik, mint egy teljes kernelé.

Ebben a részben is a /proc fájlrendszeren belül fogunk maradni és megnézzük, hogy ez miként használható kommunikációra a

kerneltől a felhasználó felé vagy fordítva, információk vagy parancsok kernelbe juttatására. Ennek bemutatására írni fogok egy egyszerű modult, ami létrehoz egy új /proc/hostname bejegyzést. A kiíratáskor ez a virtuális fájl megadja az aktuális rendszerünk gépnevét. De mivel a /proc rendszer kétirányú, ebbe a fájlba írni is lehet. Ebben az esetben a kernelnek fogadnia kellene a szöveget és ennek megfelelően át kellene állítania a gépnevet.

Igazából ez nem valami eredeti felhasználása a /proc fájlrendszernek, mert már létezik egy bejegyzés, ami pont ezt csinálja: /proc/sys/kernel/hostname. De a modulépítés példajaként fogjuk használni és írási jogot adunk minden felhasználónak, valamint nem engedélyezzük a gépnév rootra módosítását, mint ahogy a létező bejegyzésnél is ez a normális viselkedés (talán nem véletlenül).

A kernel forráskód-fában a modul forrásfájl vagy -fájlok tetszés szerint bárhol elhelyezhetők vagy ha úgy tetszik, akkor egy teljesen különálló mappába is rakhatóak. Mivel a /proc keretein belül fogunk

dolgozni ezért én az fs/proc mappán belül hoztam létre egy új fájlt, amit hostname.c-nek neveztem el.

EGY ÚJ MODUL FELÉPÍTÉSE

Egy kernelmodul fordításához szükséges alap lépések a következők:

- kódot kell írunk, ami inicializálja a modul betöltésekor létrehozott adatszerkezeteket és /proc bejegyzéseket.

- kódot kell még írunk az állapot megfelelő visszaállítására, amikor a modul eltávolításra kikerül a memóriából. Ez egy kicsit eltér más operációs rendszerektől, mert a Linux kernelmodulok minden további nélkül eltávolíthatóak a rendszer futása közben, ha nincs rájuk többé szükség.

- végül callback függvényeket kell írunk, amik akkor hívódnak meg, amikor feldolgozásra kerül a /proc fájl.

A callback függvények logikája inkább az operációs rendszerek lét-

rehozásánál megszokott. A rendszer részeként, amikor arra várunk, hogy valami történjen – például egy billentyűzeteseményre – választhatunk, hogy milyen stratégiát kövessünk. Az egyik a billentyűzet periodikus lekérdezése, amikor ellenőrizzük, hogy történt-e billentyűleütés és várakozik-e valami feldolgozásra. De ez nem valami hatékony, mert rengeteg feldolgozást igényel még akkor is ha ténylegesen senki sem ül a billentyűzetnél.

Az eseményekre való válaszolás másik módszere a megszakítás-mechanizmus használata. Lényegében ez a hardverre hárítja a várakozást. Ilyenkor az történik, hogy inicializáláskor a meghajtóprogram beállít egy konkrét függvényt a billentyűleütések feldolgozására. A függvény még nem fut, de készenlétben áll a memóriában. Ennek a címe (egy mutató) kerül átadásra a megszakításkezelő rendszernek. Egy billentyűzet megszakítás észlelésekor – amikor valaki megnyomott egy billentyűt – meghívódik ez a függvény.

Ez a callback függvény akkor is használható, amikor a kernelen kívülről érkező más típusú eseményekre várakozunk. Ezek lehetnek fizikai események, mint az egérkattintás, vagy hálózati adapterre érkező csomagok, vagy a mi esetünkben logikai (szoftver) események, mint amikor a felhasználó olvas vagy ír egy fájlba.

A MODUL INICIALIZÁLÁSA ÉS ELTÁVOLÍTÁSA

Az első, amit végre fogunk hajtani az, hogy beillesztjük a modul utolsó két sorát, ezeket:

```
module_init(hostname_proc_init);
```

```
module_exit(hostname_proc_exit);
```

A „module_init” azt a függvényt adja meg, aminek akkor kell meghívódnia, amikor a modul betöltődik a memóriába, a mi esetünkben „hostname_proc_init”, míg a „module_exit” ugyanezt teszi, csak a függvény akkor hívódik meg, amikor a modul eltávolításra kerül. Vegyük észre, hogy itt egy elnevezési szokást követünk ami igen elterjedt a kernel-forrásban: minden függvényünk a modulunk nevével kezdődik, ami

```
static int __init hostname_proc_init(void)
{
    printk(KERN_INFO "hostname loading\n");
    hostname_entry = proc_create("hostname", 0666, NULL, &hostname_proc_fops);

    if (hostname_entry == NULL)
        printk(KERN_INFO "hostname could not create /proc entry\n");
    else {
        hostname_entry->proc_iops = &hostname_proc_iops;
        printk(KERN_INFO "hostname /proc entry created\n");
    }

    return 0;
}
```

„hostname” ezt a „proc” követi, hogy jelezzük, ez a kód a /proc fájlrendszeren belül működik és végül mindegyik függvénynek egy megkülönböztető nevet adunk ami a feladatáról árulkodik.

Ugyanezt a szokást követjük, amikor a modulra való hivatkozáshoz egy kezelőt (handle) hozunk létre, ami jól jöhet akkor, amikor azt ellenőrizzük, hogy sikeres volt-e a telepítés:

```
static struct proc_dir_entry
*hostname_entry = NULL;
```

Most meg kell írunk a modult inicializáló kódot (fent látható). A függvény interfésze szabványos, és igazodni kell hozzá. Az új /proc bejegyzést 0666 hozzáférési móddal fogom létrehozni, amivel minden felhasználó írási és olvasási jogot

kap (tulajdonos, csoport és minden más felhasználó). Továbbá elég sok információt fogok a kernellogba küldeni, így hibakereséskor használható a „dmesg” parancs.

Vedd észre a két táblacím jelenlétét: „hostname_proc_fops” és „hostname_proc_iops”. Ezek további callback függvényekre tartalmaznak hivatkozásokat, például a „hostname_proc_permission”, ami a felhasználó hozzáférési jogosultságának ellenőrzésére használatos,

vagy „hostname_proc_open” ami a szöveget szolgáltatja amikor a /proc bejegyzésünket olvassuk és a „hostname_proc_write” ami beolvassa és feldolgozza a felhasználói adatot a /proc bejegyzés írásakor (lent).

A többi műveletet az alapértelmezett függvény kezeli például „seq_read” stb.

A mi esetünkben a modul memóriából való eltávolításakor (jobb

```
static const struct inode_operations hostname_proc_iops = {
    .permission = hostname_proc_permission,
};

static const struct file_operations hostname_proc_fops = {
    .open      = hostname_proc_open,
    .read      = seq_read,
    .write     = hostname_proc_write,
    .llseek    = seq_lseek,
    .release   = single_release,
};
```

fent látható) nagyon kevés takarítási műveletet kell végrehajtani, csak el kell távolítani a /proc bejegyzést a „proc_remove” függvénnyel. Mint ahogy az előbb is, most is próbálok a hibakeresés megkönnyítésére sok naplóbejegyzést biztosítani. A legtöbbet közülük el kellene távolítani, ha ez a modul kiadható állapotba kerülne.

REAGÁLÁS A /PROC BEJEGYZÉS OLVASÁSÁRA ÉS ÍRÁSÁRA

Mint ahogy a sorozatunk előző részében, a felhasználó /proc bejegyzést olvasó kérésére válaszoló művelet két függvényre van bontva. A „hostname_proc_open” függvény a tényleges callback függvény, ami a „hostname_proc_fops”-ban található. De ennek csak a /proc bejegyzésfájlhoz és inode mutatókhoz van hozzáférése. A hozzáférés megkönnyítésére jó megoldás a „single_open” függvény használata, ami egy szekvenciális fájlhozzáférés mutatót biztosít („m”), ami a „printf” függvénnyel használható a formázott kimenetek fájlba írására. A kód jobbra a második.

Vedd észre az „utsname” mechanizmus használatát, ami a rend-

```
static void __exit hostname_proc_exit(void)
{
    printk(KERN_INFO "hostname unloading\n");

    if (hostname_entry == NULL)
        printk(KERN_INFO "hostname /proc entry does not exist, not removing\n");
    else {
        proc_remove(hostname_entry);
        printk(KERN_INFO "hostname /proc entry removed\n");
    }
}
```

```
static int hostname_proc_open(struct inode *inode, struct file *file)
{
    return single_open(file, hostname_proc_show, NULL);
}
```

```
static int hostname_proc_show(struct seq_file *m, void *v)
{
    seq_printf(m, "system hostname is currently: %s\n", utsname()->nodename);
    seq_printf(m, "write new name to this file to change hostname\n\n");
    return 0;
}
```

szerünk jelenlegi gépnevét adja vissza. Ahogy az uname manuáldalokon is olvasható, a POSIX megfeleltetés szempontjából valójában nem követelmény, hogy a hostname és a nodename ugyanaz legyen. Azonban a Linuxnál úgy látszik, hogy igen.

Ha ez helyén van, akkor a felhasználó képes lesz olvasni a /proc bejegyzésünket:

```
cat /proc/hostname
```

```
system hostname is currently:
alan-vaio
write new name to this file
to change hostname
```

Egy kicsit bonyolultabb lekezelni a bejegyzésünkbe való írást. A felhasználói folyamat által írt adat a „user_data” mutatóval lesz elérhető, a rendelkezésre álló karakterek száma pedig a „len”-ben található. De ez az adat egy felhasználói névtérben lévő adatszerkezetben található, és mielőtt dolgozhathatnánk vele, át kell másolni egy ugyanolyan, de a kernel névtéré-

ben lévő táblába.

Valamint nagyon körültekintően kell leellenőrizni a felhasználói beírt karakterek összetételét. Bármilyen kernelkódban véghezvitt tevékenység, ami helytelen felhasználói beírt karakterekkel kezdődik, kihatással lehet az egész rendszerre, tehát jobb ha itt nagyon óvatosak vagyunk.

A mi esetünkben a felhasználó által beírt karakterek egyszerű átmásolásról van szó egy nem-nyomtatható karakter felbukkanásáig

(például bármi, aminek a kódja kisebb mint a szóköze). Ha a karakterek száma kevesebb mint az elfogadható gépnév-hossz (ellenőrzés! ellenőrzés! és megint csak ellenőrzés!) másold át az új gépnevet a vonatkozó utsname táblába.

Végül a függvény befejezése előtt vissza kellene adni a felhasználói bevitelből ténylegesen beolvasott karakterek számát. Ez a rendszernek lényeges, mert ha az elérhető karaktereknél kevesebb került tényleges feldolgozásra, akkor a rendszer még egyszer meghívja ezt a függvényt – vagy ahányszor még szükséges – a teljes bevitel lekezelésére. Így mi egyszerűen a „len”-el egyező karakterszámot adjuk vissza (bemenet hossza), hogy jelezzük, mindegyiket lekezeljük és nincs szükség további hívásra. A kód jobbra látható.

Amikor ez a helyére került minden felhasználó képes lesz szöveget írni a bejegyzésünkbe, hogy megváltoztassa a gépnevet:

```
$ echo "our-new-hostname" > /proc/hostname
```

```
$ hostname
```

```
our-new-hostname
```

```
static ssize_t hostname_proc_write(struct file *file, const char __user *user_data, size_t len, loff_t *offset)
{
    int buffer_size = 0;
    int i;
    char buffer[MAX_BUFFER_SIZE];
    printk(KERN_INFO "hostname_proc_write, len=%ld\n", len);

    buffer_size = len > MAX_BUFFER_SIZE ? MAX_BUFFER_SIZE : len;
    i = 0;
    buffer[0] = '\0';
    while ((i < buffer_size) && (user_data[i] > ' ')) {
        buffer[i] = user_data[i];
        i++;
    }
    buffer[i] = '\0';
    buffer_size = i+1;

    printk(KERN_INFO "wrote %d bytes\n", buffer_size);
    printk(KERN_INFO "hostname=%s\n", buffer);

    if (buffer_size <= __NEW_UTS_LEN)
        for (i = 0; i < buffer_size; i++)
            utsname()->nodename[i] = buffer[i];

    return len;
}
```

A KÉSZ MODUL FORRÁSKÓDJA

A modulunk még nem fordul le mert kihagytam az include sorokat, amik a függvény prototípusok megadásához szükségesek. Ezeket a fájlnk elejére kell beszúrni. Közölhetünk a fordítóval néhány egyéb adatot is, például mit csinál a modul, ki a szerzője és milyen licenc alatt érhető el:

```
MODULE_AUTHOR("Alan Ward");
MODULE_LICENSE("GPL v2");
MODULE_DESCRIPTION("hostname module for Full Circle Magazine");
```

Ez az információ benne lesz a modulban és lekérdezhető a „module_info” paranccsal:

```
$ modinfo hostname.ko
filename: /home/alan/Esriptori/linux/fs/proc/hostname.ko
description: hostname module for Full Circle Magazine
```

```
license: GPL v2
author: Alan Ward
srcversion:
431F7E34A05708273893D24
depends:
vermagic: 3.13.0-24-generic
SMP mod_unload modversions
```

Amikor mindennel készen vagyunk a végső modul kód a következőképpen áll össze:

<http://pastebin.com/5d6KxCRZ>

AZ ÚJ MODUL FORDÍTÁSA ÉS TELEPÍTÉSE

Amikor az új modulunk fordítása kész, össze kell szerkeszteni a létező kernel adatszerkezetekkel és függvényekkel. Ennek a legegyszerűbb módja ha beintegráljuk az új modult a létező forráskód-könyvtár make struktúrájába. Szerkeszd az azonos mappában lévő Makefile-t és add hozzá:

```
obj-m := hostname.o
```

Most már lefordíthatjuk az összes modult, többek között a miénket is, ha a kernel forráskód gyökérmappájába megyünk és futtatjuk a make-et a következő paranccsal:

```
cd ../../
```

```
make modules
```

Másik lehetőségként praktikus lehet, ha csak a munkakönyvtárban lévő modulokat fordítjuk a következő paranccsal:

```
$ make -C /lib/modules/`uname -r`/build M=$PWD modules
```

Amikor ez kész, telepíthetjük a modult a futó kernelbe:

```
sudo insmod hostname.ko
```

és a leírtak szerint használjuk. A modul szükség szerint eltávolítható:

```
sudo rmmod hostname
```

A helyes működés leellenőrzéséhez tanácsos a rendszer kernel-logjának áttekintése:

```
dmesg | tail
```

```
[ 7501.047170] hostname loading
```

```
[ 7501.047178] hostname /proc entry created
```

```
[ 8095.253713] hostname_proc_write, len=17
```

```
[ 8095.253722] wrote 17 bytes
```

```
[ 8095.253726] hostname=our-new-hostname
```

```
[ 8381.501772] hostname unloading
```

```
[ 8381.501784] hostname /proc entry removed
```

Ezzel végére értünk a kernelfordításról szóló hatrészes sorozatunknak. Talán a kernel módosítása, fordítása és telepítése az egyik legtöbbet követelő tevékenység, amit a kedvenc operációs rendszerünkön végrehajthatunk, de ez tö-

kéletesen kivitelezhető egy kis türelemmel és szisztematikus megközelítéssel. Na jó, még egy kis kitartás is jól jöhet, amikor nem az elvártak szerint mennek a dolgok az elején.

Akárhonnan is nézzük, a lényeg, amit ki szeretnék emelni az hogy ez végrehajtható – nem csak számítógép zsonglőrök és profi hekkerek, hanem „normál” emberek által is, egyszerűen csak ezt az utat kell követni. Ha így teszel, biztosan rengeteg tapasztalattal gazdagodsz, mivel olyan dolgokat ismerhetsz meg a számítógépedről, amire nem is gondoltál. Én így voltam vele amikor erre a sorozatra készültem.

Ez ugyanakkor azt is jelenti, hogy egy nyílt forrású operációs rendszer nyílt forrású kernelével lehetőség nyílik arra, hogy azt többen nézzék át, fejlesztési javaslatokat vagy javításokat kapjon hogyha szükség van rá. Bár lehet, hogy az olvasó személy szerint nem vizsgálja át a kernelt, de mindnyájan profitálunk a lehetőség pusztá létezéséből egy stabilabb és naprakészebb számítógépes platform létezésével.

Köszönet mindenkinek aki részt vett a Linux kernel fejlesztésében

és elérhetővé tette ezt a világ számára; ezt nem lehet elégszer mondani.



Alan számítástechnikát tanít az Escola Andorrana de Batxillerat gimnáziumban. Tartott GNU/Linux előadásokat az Andorrai Egyetemen és most is GNU/Linuxot és rendszeradminisztrációt oktat a Katalán Nyílt Egyetemen (OUC)



Mi a közös a Yahoo, a Yandex.ru (Oroszországban népszerűbb kereső, mint a Google), a Playstation 4, pfSense, Hotmail és a Netcraft oldalakban? Egy ideig mindegyik a FreeBSD-t vagy valamelyik leszármazottját futtatta. Bár a Hotmail (outlook.com) már jó pár éve Windows szerveren üzemel, bizonyára vannak köztetek olyanok, akik még emlékeznek, hogy milyen gyors volt a Hotmail, amikor még FreeBSD-n futott.

Sok évvel ezelőtt én is FreeBSD-t használtam, amíg bele nem futottam egy problémába, az éppen megvásárolt HP Deskjet nyomtató meghajtójával összefüggésben. Azelőtt egyike voltam azoknak, akik a Walnut Creek-nél előfizettek a FreeBSD-re. Egy félresikerült kiadás után (az egyik kiadásban a /stand/ sysinstall hibás volt, a megoldás pedig egy fájl letöltése volt az internetről, mely értelmetlenné tette a CDs verzió megvételét), s mivel a szélessávú internet egyre elérhetőbb lett a környékünkön, így megszüntettem az előfizetésemet. A CD-s verzió értékes volt számomra, de minden változik, s én is inkább a Linux iránt kezdtem érdeklődni (1997-ben Slackware 96-ot

használtam).

Az egyik dolog, ami vonzott a FreeBSD-ben, az a ports rendszer érdekes szoftvergyűjteménye volt. A portok alapvetően make fájlok, melyek hivatkozások a forráskód letöltési útvonalaira (vagy a CD-re/DVD-re). Hasonlóképpen kell telepíteni a szoftvert, mint például a Arch, Gentoo, vagy Slackware Linux disztribúciókban. Másik nagyszerű dolog a FreeBSD-ben, hogy széleskörű hardvertámogatottsággal rendelkezik. A FreeBSD 10.1-RELEASE (a cikk írásának pillanatában a legfrissebb kiadás) lemezképei egyaránt elérhető-

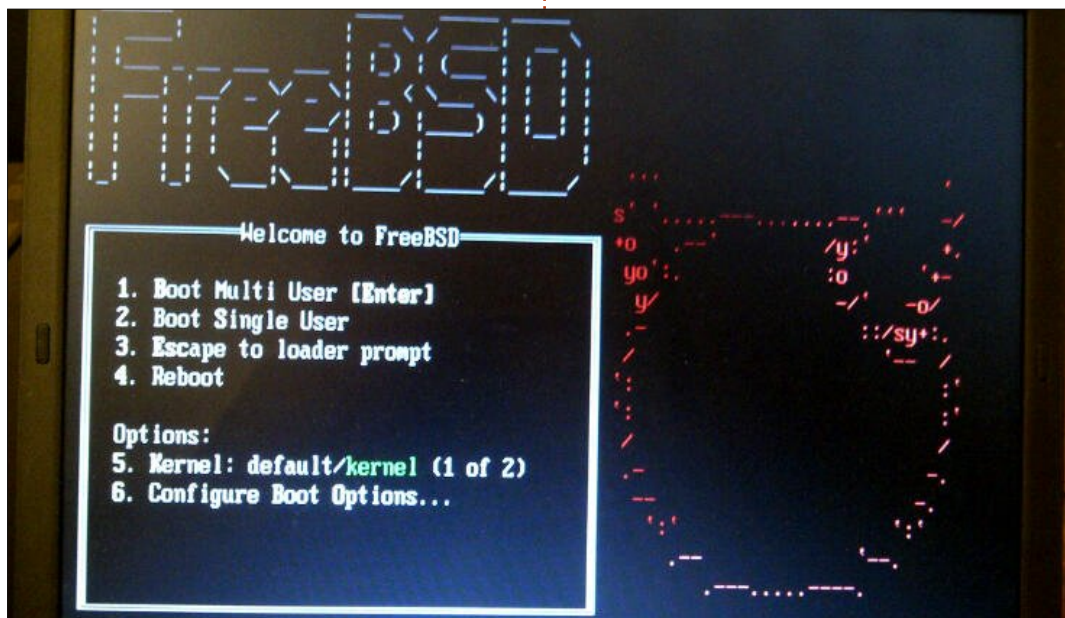
ek amd64, i386, ia64, powerpc, powerpc64, sparc64 és armv6 architektúrára. Emlékeim szerint a FreeBSD villámgyors volt. Ami pedig elhalványult emlékeimben van, hogy egy picit mindig utána kellett olvasni és trükközni kellett a dolgokkal a rendszer futása érdekében, hasonlóképpen, mint a Gentoo, Arch vagy Slackware esetén.

A FreeBSD telepítése nem különösképpen nehéz. Alapjában véve hasonló folyamat, mint egy Linuxnál: billentyűzetkiosztás kiválasztása, gépnév megadása, néhány opcionális komponens kijelölése (extra doku-

mentáció, játékok, a ports fa és a rendszer forráskódja), merevlemez particionálása, alapsomagok telepítésének kivárása, root jelszó, hálózat, időzóna beállítása, néhány szolgáltatás elindítása (sshd, ntpd, powerd, stb.), további felhasználók hozzáadása. Ezután még bármilyen egyéb változtatás alkalmazható, majd az újraindulás után beléphetünk a telepített rendszerbe.

Akinek van egy kis tapasztalata már egy Linux telepítésében, mind ismerősnek fogja találni a FreeBSD-t is. A FreeBSD projekt elfogadható online dokumentációval rendelkezik, a kézikönyv, <https://www.freebsd.org/doc/handbook/>, bárkit végigvezet a telepítés folyamatán.

Nálam mindösszesen körülbelül 15 percbe telt a telepítés egy 2 GB RAM-mal és 5400 rpm-es (lassú) 80 GB-os merevlemezrel rendelkező Pentium 4 1.7 GHz-es laptopon. Egy problémám akadt a telepítés során, az Intel vezetéknélküli kártya meghajtója nem akart feltelepülni az üzembe helyezés kezdetén, ezért vezetékes kapcsolatot kellett használnom. A FreeBSD 10.1-et telepítettem, mely alapesetben a GPT



partíciós sémát használja, s akár 3 TB-nál nagyobb merevlemez is képes kezelni. Lehetőség van a Zetabyte fájlrendszer (ZFS) használatára is. Ellentétben másokkal, én jobban szeretem figyelemmel kísérni az egyes szolgáltatások indulását, mint egy pörgő forgó logót nézni.

Miután befejeződött a telepítés, azon kaptam magam, hogy a parancssort bámulom, ami minden új Linux-felhasználó rémálma. Akik egy kulcsrakész rendszert keresnek, azoknak a PC-BSD egy jó megoldás lehet: <http://www.pcbsd.org/>.

Fontosnak tartom megemlíteni, hogy a FreeBSD kézikönyv tartalmazza a sysinstall-t is. A FreeBSD 10.1 a bsdinstall-t alkalmazza a sysinstall helyett. A /sbin mappa böngészése közben találtam egy bsdconfig nevű csomagot, mely úgy tűnt, hogy ugyanazt végzi el, mint a sysinstall, azaz csomagokat telepít a portok helyett. Azonban mikor kipróbáltam a bsdconfig parancsot, a rendszer a FreeBSD ftp szervereinek lekérdezése közben hosszú ideig nem válaszolt. Nem tudom, hogy csak én voltam-e türelmetlen a lista letöltésére várva, vagy valami más gond volt. Bármilyen is legyen, úgy tűnt, hogy a FreeBSD-nek túl hosszú ideig tartott az FTP szerver lekérdezése (pin-

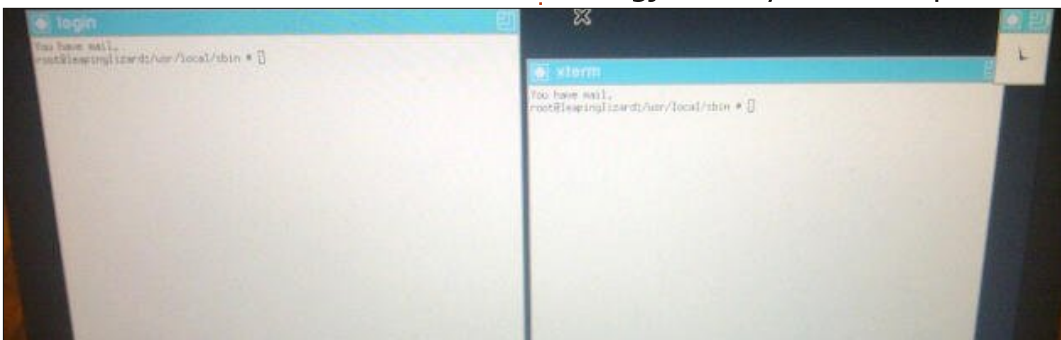
gelni tudtam a kiszolgálót). Egy idő után kaptam egy hibaüzenetet „error no pkg database found.”. A DVD-t használva megpróbáltam telepíteni a csomagokat, de ismét belefutottam ugyanabba a hibaüzenetbe „no pkg database”. Találtam néhány fórumbejegyzést a hibával kapcsolatban: <https://forums.freebsd.org/threads/bsdconfig-no-pkg-8-database-found.49216/> <https://forums.freebsd.org/threads/error-no-pkg-8-database-found.48142/>

Sajnos egyik bejegyzés sem segített megoldani a dolgot, így megpróbáltam a pkg parancssal.

Sajnos egyik bejegyzés sem segített megoldani a dolgot, így megpróbáltam a pkg parancssal.

```
pkg update
pkg upgrade
```

Mindkét parancs működött, és frissítette a rendszer számára elérhető csomagokat az internetről. Ezt követően az alábbiakkal próbálkoztam:



```
pkg install xfce
```

Egy rakás csomag és függőség települt fel, majd kiadtam a következő parancsot:

```
pkg install xorg
```

Még több csomag települt. Ezúttal a startx parancssal próbálkoztam, s íme megjelent a munkaasztal (nem XFCE hanem TWM... de mégiscsak munkaasztal). Sajnos mikor a menü Exit parancsával próbáltam kilépni a TWM felületből, a FreeBSD leállt, a TWM ablak díszítményei eltűntek, de maga az ablak megmaradt. Próbáltam átkapcsolni a virtuális terminálok között, s kiadni egy kill parancsot, de nem volt szerencsém.

A pkg parancs az apt-hoz hasonlóan működik. pkg search <package-name> rákeres a <csomagnév>-re, ugyanúgy mint ahogy az apt-cache search képes megtalálni a csomagokat egy Debian/Ubuntu alapú rend-

szerben.

Jelen állás szerint, a FreeBSD működik és képes megtalálni, telepíteni az új csomagokat. A következő hónapban még több közvetlen tapasztalattal fogok jelentkezni a ports fáról. Tíz év után tértem vissza a FreeBSD-hez, s most becsülöm meg csak igazán, hogy az Ubuntu milyen egyszerűvé teszi a desktop felhasználók életét.

A FreeBSD-t Linuxhoz hasonlítóssá lehet varázsolni és ugyanazt a felületet lehet futtatni rajta, bár mindez olyan, mintha az almát a narancssal hasonlítanánk össze. Biztos vagyok benne, hogy egy picit több utánaolvasás és munka révén minden emlék visszajön. Következő hónapban érkezik a FreeBSD kipróbálásának második része.



Charles Mccolm - az Instant XBMC írója, ugyanakkor egy non-profit számítógép újrahasznosítás projekt menedzsere. Amikor nem számítógépeket büttyöl és nem rosszindulatú programokat (malware-t) távolít el, a GNU/Linux támogatására bíztatja az embereket, Ubuntu lokális órákat tartva. Charles a <http://www.charlesmccolm.com/> weboldalon blogol

[PACKT] CLASSICS

DID YOU MISS ANY?

New to Packt? Check out some of our all time classics to build your essential learning library and make sure you start 2015 ahead of the curve.

<https://www.packtpub.com/books/packt-classics>



Nemrég úgy döntöttem, hogy kipróbálom az Ubuntu 14.10-et, melynek igen változatos eredményei voltak. A Unity asztal a betöltése után kb. tíz másodperccel lefagyott. Ezt a problémát három különböző telepítés is produkálja. Gyanítom, hogy a Nouveau videóillesztőprogram lehet a ludas, de még sosem volt időm az Nvidia illesztőprogram telepítésére.

Kipróbáltam a Gnome 3 asztalt is, és az jól működött, azonban rengeteg olyan, általam nem szeretett alkalmazást telepít előre, mint a Brasero vagy a Rhythmbox. Ráadásul lehetetlen eltávolítani őket, mivel hozzánőttek a Gnome asztalhoz, így eltávolításukkor viszik azt is. Nincs semmi baj a Rhythmboxal, csak én jobban szeretem a Clementine-t. Valakinek talán jó a Brasero, de az általam tesztelt disztrókon mindig azt mondta, hogy nincs írható CD/DVD a meghajtóban – ilyen gond nincs az xfburnnel vagy a K3b-vel.

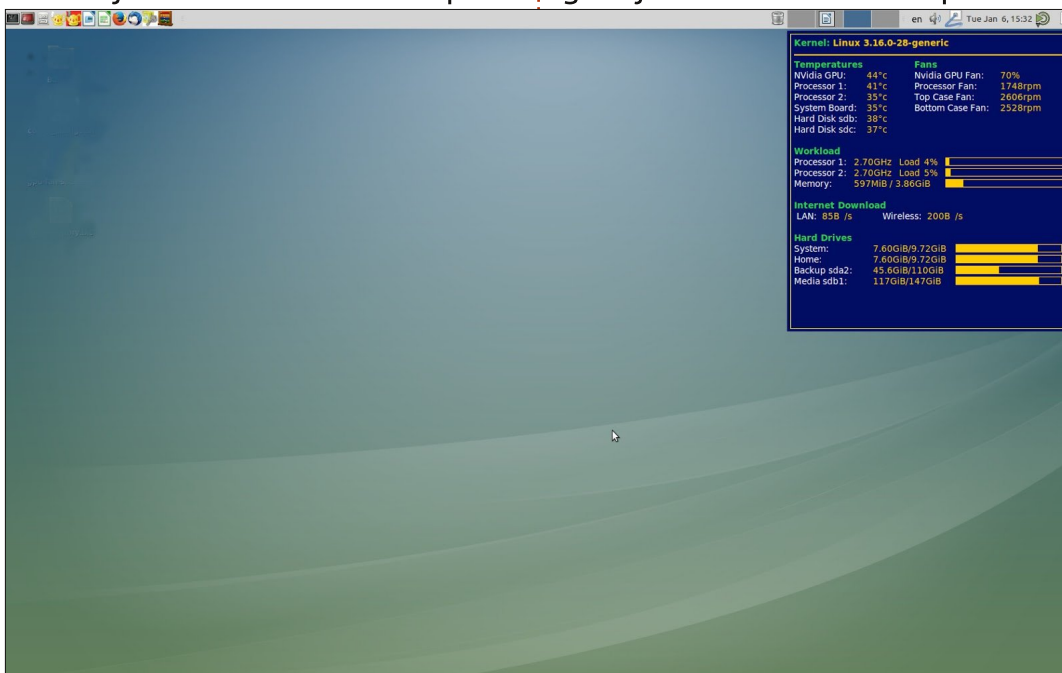
Továbbá nem igazán rajongok a Gnome 3-ért sem. Utálom az asztalon rendezetlenül elszórt ikonokat.

Még Calibrét sem telepíthetek, mert inkompatibilis más illesztőprogramokkal, ezzel pedig elvesztett engem.

Türelmetlenül nyitottam meg múlt héten az FCM 92-es kiadását és szembe jött az Ubuntu Mate változata. A Distrowatch nem tesz róla említést, így ellátogattam a <https://ubuntu-mate.org>-ra további információért. Ez a verzió az USA-ból származik. Csak 983 MB-ot kell letölteni, de ez is időbe telik, ezért javaslom a torrent használatát. Teljesen különbözik az alapér-

telmezett Ubuntutól, így megérdemel egy teljesen saját bemutatást is. Természetesen, az Ubuntu alapjait használja, ami az Ubuntu csapat eredménye, ugyanakkor a végfelhasználói élmény nagyban a Mate csapat érdeme. Ők ragaszkodnak a jól tesztelt alkalmazásokhoz, ezért minden úgy működik, ahogy annak kell.

Telepítés előtt mindig létrehozok egy partíciót az új rendszernek, így nem akad össze a telepítettekkel. Néhány felhasználó jelzett, hogy gondja van az Ubuntu telepítővel



végzett particionálással. Mivel én egyénileg szoktam particionálni, nem kellett aggódnom emiatt. A telepítés gondtalan volt és mindössze 30 percig tartott. Újraindítás után, a valaha látott legjobb GRUB-menü fogadott. A bejelentkezéstől az asztalig látható grafika jól mutatja a Mate csapat művészi tehetségét. Azonnal figyelmeztetett, hogy frissítések érhetőek el, amiket észrevétlenül telepített is. Az első benyomás megismételhetetlen és le voltam nyűgözve. Azonnal szembesültem a szép, üres asztallal, és hogy minden felhasználói akciót el lehet intézni a panelen és a hagyományos menükben. Néhányan mondhatják hogy „régimódi” és nem „hagyományos”. Akárhogy is, ez egy jól ismert rendszer, mely az alkalmazásokat értelmes kategóriákba rendezi.

Első dolgom volt, hogy az ízlésemnek megfelelően állítsam be, ami a Mint LMDE és Cinnamon asztal keverése. Vezérlőközpontban van a További illesztőprogramok szekció, ahol automatikusan kiválasztotta az NVidia kártyámnak megfelelő illesztőprogramot és telepítette is.

A WiFi-illesztők a rendszer telepítésekor automatikus felkerültek. Két panelem volt, de az alsó csak két panelalkalmazást tartalmazott. Ezeket átmozgattam a felső panelre és töröltem azt. Növeltem a panel szélességét 24-ről 32 pixelre és az ikonok nagyra, valószínűleg 32 pixel szélesre nőttek. Kísérletezéssel rájöttem, hogy nekem a 31 képpont szélességű panel és 24 képpont nagyságú ikonokkal az optimális. Egyszerű a további alkalmazások telepítése, melyekkel a legtöbb igényt (nekem az összeset) ki lehet elégíteni. A Menü mellett használom a Dátum, Hangerő, Hálózati, Nyelv, Munkaterület-váltó, Leállítás és Kuka panelalkalmazásokat. Továbbá telepítettem a nyitott ablakokat megjelenítő alkalmazást is. Mindet egyszerű beállítani annak ellenére hogy a segítség nem működik, de nincs is rá szükség. A menüsor három kategóriát tartalmaz: Alkalmazások, Helyek és Rendszer. Úgy gondoltam, hogy ezek túl sok helyet foglalnak el a felső panelen és nincs szükségem a „Helyekre”. Nagyon egyszerű volt ennek az eltávolítása és az alapértelmezett Mate menüben a lecse-

rélése. Jobb klikkel hozzáadni, eltávolítani vagy mozgatni lehet menüpontokat. Hozzáadhatsz egy haladó menüt is, amiben benne vannak a kedvencek is, de ebben egy újabb kattintásnyi távolságba kerülnek a többi alkalmazások. Ráadásul lassú is, mivel a kedvenc alkalmazások nagy ikonjait tölti be. Egyszerűen lehet hozzáadni, kísérletezni és eltávolítani bármilyen panelalkalmazást.

A kedvencem a Témák. Széles témaválaszték érhető el a Vezérlőközpontban, egy sor csodálatos háttérképpel együtt, így biztosan megtalálod a számodra tetsző összeállítást.

Elérkeztünk a szoftverekhez. Mindenkinél más ízlése van. Én úgy szeretem, ha a disztró egyszerű marad és engedi a felhasználóknak a kedvenceinek későbbi hozzáadását. A Mate ezt az egyensúlyt az olyan, csak szükséges alkalmazások szállításával tartja meg, mint többek között: Firefox, Thunderbird, Képnézegető, Pluma szövegszerkesztő és Simple Scan. Csak a Pidgin, Brasero és Rhythmbox, amit el

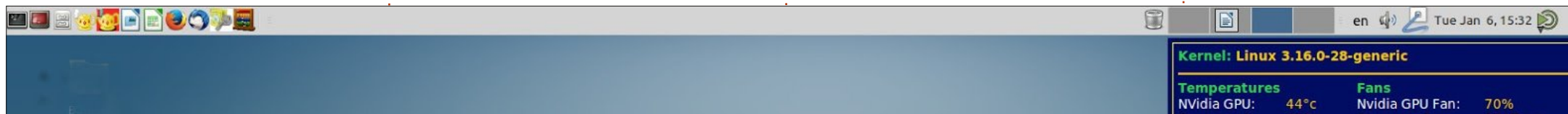
szerettem volna távolítani. Sajnos, mind a három hozzá van növe a GNOME-asztalhoz, így eltávolíthatatlanok. De a menüpontjaik törölhetők. Sok más alkalmazást és segédprogramot telepítettem, melyek mind megtalálhatóak a tárolókban. Kivéve a grub-customizer-t, de ennek az összes függősége rendelkezésre állt, szóval nem volt problémám a telepítésével.

Ez az egyetlen tesztelt disztró, melynél elgondolkodtam, hogy használhatnám a Mint/Cinnamon helyett. Minden egyszerűen működik, pont úgy, ahogy annak kell, vilámgyors, nincs felesleges grafikus effekt. Jól működik a Nouveau illesztőprogrammal, miközben nálam a Cinnamon és KDE az Nvidia drivert igényli. Szeretem a panelt, mert könnyű hozzáadni/mozgatni/eltávolítani indítókat és panelalkalmazásokat. Az egyetlen bajom a Cinnamon asztallal, hogy nehéz ugyanezt megtenni. Másrésről, a Cinnamon menü megnyílik ha csak felé viszem az egeret, miközben a Mate menüre kattintani kell. A Mate menü nem tartalmaz kedvenceket, de ez nem gond mert az

enyéim mind kint vannak a panelen.

Egy furcsaságot látok a Conkyban, mely talán érdekelheti a többi felhasználót is: Conkyval jelenítem meg a rendszer hőmérsékleteit és a ventilátorok fordulatszámát az összes munkaterületemen. Ezt a .conky.rc „own_window_type desktop” beállításával teszem és általában jól működik. A Mate-ben az ablak eltűnik az összes munkaterületről, mikor bármelyik asztali ikonra kerül a fókusz. Megoldódik, ha a sort „own_window_type override”-ra változtatod.

Ha új Linux felhasználó vagy és egy jól használható Linux rendszerre van szükséged, melyben minden azonnal működik, akkor ez neked való!





Kiadó: Prentice Hall (Pearson)

Oldalak száma: 322

Szerzők: Matthew Helmke, Elizabeth K. Joseph, José Antonio Rey and Philip Ballew

ISBN-13: 978-0-13-390539-7

ISBN-10: 0-13-390539-X

A Hivatalos Ubuntu könyv című olvasmány egy, a Canonical által jóváhagyott útmutató az Ubuntu Linux operációs rendszerhez. A nyolcadik, frissített kiadás a 14.04-es hosszú távon támogatott Ubuntu kiadással kapcsolatos lényeges információkat tartalmazza. A könyv

célja egyszerű, a napjainkban futtatott Ubuntu környezetek még kényelmesebbé és hatékonyabbá tétele számodra.

A cél elérése érdekében a szerzők kitérnek a telepítésre, hasznos alkalmazásokra, a médiára, az adminisztrációra, Unity-s tippekre és trükkökre, szoftverek forrásból való telepítésére, illetve az Ubuntu ökoszisztémára épülő közösségi és néhány szerverhez kapcsolódó témára.

Ami tetszett a könyvben, hogy míg az új Ubuntu-felhasználókat célozza, ennek ellenére a már tapasztalattal rendelkező felhasználók számára is tartogat hasznos dolgokat. A teljes könyv tíz fejezet hosszúságú. Az első három fejezet a telepítést és a rendszerfrissítést fedi le. A negyedik fejezet az Ubuntu elérhető érdekesebb alkalmazásokról szól. Az ötödik a Unity testre szabhatóságára terjed ki. A hatodik fejezetben a Linux fájlrendszeréről, a gyorsbillentyűkről, csomagokról és az alapvető fordítási technikákról olvashatunk. A hatodik fejezetben volt egy „hé, ez itt szuper” pillanat, amikor a PPA eltávolításá-

nak lépésein ment végig (vicces, hogy mennyi weblapon írnak arról, hogy hogyan add hozzá, de arról nincs egy szó sem, hogyan távolítsd el) és ami még jobb, hogy hogyan telepíts forrásból úgy, hogy az törlhető legyen.

Egy kurzusnak vagy könyvnek abban látom részben a sikerét, függetlenül attól, hogy tanultam-e valami új dolgot vagy sem, hogy hatékonyabbá tesz. Még akkor is, ha a kurzus vagy könyv célközönsége a kezdőknek, gyakran a már haladó felhasználóknak is értékes, különösen, ha a kurzus vagy könyv ad valamit, amitől még hatékonyabbá válsz.

Korábban már említettem a könyv egyszerű célját, még kényelmesebbé és produktívabbá tenni az Ubuntut, mivel találtam számos tippet, amivel a tudásomat kibővítettem (a hatodik fejezetben és azon is túl). A könyv sikeresen eléri ezt a célját, annak ellenére, hogy nem vagyok túl figyelmes hallgatóság.

Ha félelmeid vannak a parancssoros utasításoktól, akkor a hetedik fejezet ad egy tömör bevezetést a

témába. Sok linuxos alap könyv foglalkozik az alaputasításokkal, mint az ls, mv, cp, more, grep és a ps, de ez az olvasmány kiterjeszti ezeket a lsb_release, iwconfig, head és a chgrp, a helyettesítők és a Byobu, mint GNOME-terminál alternatíva használatára.

Az asztali felhasználók talán kihagynák a nyolcadik fejezetet, ami különböző szervertémákat jár körbe mint a RAID és az LVM, ami talán felkelti az olyan kezdők figyelmét, akik érdeklődnek a meghajtó professzionális beállításai iránt.

Az utolsó néhány fejezet közösségi témával foglalkozik. Ezek közül egyesek túlmutatnak egy kezdőknek szóló könyv keretein. Miért kellene aggódnia egy új felhasználónak a „bazaar revision control system” vagy a MOTU (Masters Of The Universe) miatt? Talán az útjuk kezdetén nem törődnek vele, de mint számítógép-felújító, rendelkezem tapasztalattal ezen a téren, és ne mondjunk gyors ítéletet egy „tapasztalatlan felhasználó” képessége felett. Gyakran meglepődök, amikor az új Linux után érdeklődő felhasználók egyszer hozzáférést



kapnak az információhoz és a technológiákhoz, és ekkor felsülnek. A tudás nem mindig jön azonnal. Emlékszem arra az időre amikor elkezdtem az első MySQL tanulmányomba. Négy fejezetet elolvastam, és itt muszáj volt leraknom a könyvet. Néhány hónap múlva értelmet nyertek az akkor még értelmetlennek tűnő dolgok, és én örültem ennek. Mert van egy remek közösség az Ubuntu körül és megéri tudni arról, hogy hogyan férj hozzá a közösséghez a különböző aspektusokból, legyen az egy IRC csatorna vagy egy hiba-nyomkövető rendszer.

A könyv magába foglalja az olyan Linux disztribúciókat mint: Kubuntu, Xubuntu, Lubuntu, Linux Mint, Edubuntu, stb. Ez egy rövid bepillantás képernyőképek nélkül, amelyek bemutatnák a vizuális különbségeket. A tömör bemutatást nem tekintem rossz dolognak, mivel a könyv hangsúlya az Ubuntu-átlomány megismertetésén van. Megkísérelni hasonlóan bemutatni a KDE, LXDE, XFCE, MATE, Cinnamon és azokat a szoftvereket melyek minden egyes disztribúcióval jönnek, túl sok lenne egy kezdőknek szóló könyv számára. A tömör bevezetés mindegyikhez jobbnak tűnik.

Érdemes megemlíteni, hogy a 44. oldalig nincs kép a könyvben (a 2. fejezet az Ubuntu telepítése). A könyv sok képernyőképet tartalmaz, de ha te egy vizuális típus vagy, ez nem egy olyan könyv lesz ami mindent képekkel mutat be, itt az olvasásra is szükséged lesz a bemutatott téma megértéséhez.

Az első fejezet bemutatja az Ubuntu történetét nem igazán sok képernyőkép használatával, de érdekes olvasmány mely érint néhány nem Ubuntuval kapcsolatos (SLS/ Slackware) történet és előrevetíti a könyv néhány későbbi témakörét.

Ismétlem, számomra egy könyv vagy tanfolyam értéke abban rejlik, hogy az új dolgokat melyeket tanultam, képes legyek használni, és ez hatékonyabbá tegyen. Ennek az érzésnek a fenntartása volt a könyv célja, és sikerült is új trükkökre tanítania engem – melyeket azóta is használok – annak ellenére, hogy egy kicsit több tapasztalattal rendelkezem mint egy kezdő. Ugyanakkor a könyv a kellő mennyiségű információt szolgáltatja az új felhasználóknak az induláshoz, illetve utat mutat a közösségi erőforrásokhoz arra az esetre, ha valahol elakadnának, vagy részt vennének

egy közösségi projektben.



Charles McColm a szerzője az Instant XBMC-nek, egy rövid könyv a XBMC-buntu telepítésről és beállításról (egy *buntu+XBMC disztribúció) és projekt menedzsere egy számítógép újrahasznosító non-profit projektnek. Amikor nem PC-t épít, vagy vírust írt, és nem embereket biztat a GNU/Linux használatára, akkor Charles azon dolgozik, hogy blogját újra felfedezzék <http://www.charlesmccolm.com/>.



A fekete, jellegtelen háttért csak a sárga színű kurzor folyamatos villogása töri meg. Ritmikus villogása arra emlékeztet, hogy rengeteg gombot kell benyomnom ahhoz, hogy ez a szöveg megjelenjen. Eltekintve egy vékony szövegtől alul, ami arról tájékoztat, mennyi sort, betűt és karaktert állítottam elő, nincs egyéb választási lehetőség, nincsenek megjegyzések, javaslatok vagy egyéb figyelemelterelések.

A modern kor követelményei szerint, ebből az applikációból hiányzik rengeteg tulajdonság. Nincs formázó menü, nincsenek szójavaslatok, szövegre vonatkozó menük vagy megosztás gombok. Mégis ez az egyik legproduktívabb szoftver a számítógémemen: segítségével írom ezt a cikket.

A számítógép, amelyen futtatom ezt az applikációt szintén „gyenge” a mai normák szerint, és alul árazott is. A Raspberry Pi számítógémem a nevetségesen alacsony memóriájával, gyenge CPU-jával és kevés tárhelyével, kevésbé erőteljes mint az okoste-

fon a zsebemben, mégis úgy döntöttem ez lesz a kiválasztott gépem. Miért?

Nos, van egy szeretem-utálok kapcsolatam a napjaink számítógépeivel. Az erőteljes CPU-ik és a bonyolult operációs rendszereik a digitális létezésünk csúcspontja. Bőséges választékkal és lehetőségekkel, összecsomagolva jönnek, amik illeszkednek minden szükségleteinkhez. Ez egyébként a végzetük is, tandemben a túlságosan összefonódott applikációkkal és szolgáltatásokkal, amik semmi mást nem akarnak, mint, hogy kicselezzenek vagy csiripeljenek mindenről amit csinálsz... egészen addig a pontig, amíg a termelékenységed nullára nem csökken. Néha ez már „túl sok”.

Ezért ugrottam bele, és mentem vissza az „alapokhoz” pár napra. A Raspberry Pi használata, a mérsékelt képességével segített újragondolnom és újraértékelnem, mire van szükségem ahhoz, hogy termelékeny legyek.

A Raspbian operációs rendszer-

nek nincs túl sok képessége, csak az alapok, hogy segítsen megcsinálni, amit meg kell. Van egy pár applikáció feltelepítve, de amikor extra szoftvert keresel, vedd figyelembe a limitált képességeit. Ezért elkezdtem gondolkodni, milyen applikációkra van szükségem, hogy termelékeny legyek, és azoknak mik a minimális követelményeik.

Szóval, kell egy böngésző, de kell, hogy legyen 5000 bővítménye? Úgy tűnik nem. Szükségem van egy szövegszerkesztőre, de kell, hogy legyen 39 gombja? Nem, nem kell. Az első gondolatom, mely szerint a kreativitásomat komolyan visszafogja az „egyszerű” környezet limitált teljesítménye, rossz volt. Az egyszerűsített környezet és az „egyszerű” applikációk segítettek arra fókuszálnom, mit kell tennem, ahelyett, hogy elterelte volna a figyelmemet maga az eszköz.

A „lassú számítógépnek” van még egy előnye. A böngészőnek kell pár másodperc, hogy életre keljen (egy végtelenség a másik rendszeremhez képest), de ez nagy segítség, amikor „sürgősen ellen-

őriznem kell a Facebookot”, és ez elterelné a figyelmemet. Nem tartathatok nyitva 50 ablakot, mert nincs hozzá RAM, de ez hozzásegít, hogy arra az egy dologra koncentráljak, ami után kutatok.

Tömören: a „lassú számítógép” segít koncentrálnom. Hagy nekem időt, hogy gondolkodjam a dolgokról, mielőtt másik hyperlinket nyomok. Mentés a túl összefonódott dolgoktól. Valamint, visszatart attól, lemenjek 45 percre a nyúl üregébe csak azért, hogy megtaláljam a „megfelelő háttérképet” a dupla képernyős asztalomhoz. Persze biztos lázadozni fogok a rendszer limitált képességei ellen... és ha tényleg így lesz, akkor még mindig ott van a másik erőteljes laptopom tartalékban. De az anyag mennyisége, ami „elkészül”, mielőtt ezt meglépek, lenyűgöző. A lassú számítógép segít koncentrálni! Próbáld ki!



Levelek

Fordította:

Ha szeretnéd, hogy leveled nyilvánosságra kerüljön amely lehet köszönet vagy reklamáció, akkor küldd az alábbi címre: letters@fullcirclemagazine.org.
FONTOS: terjedelmi okokból a levelek szerkesztésre kerülhetnek.

Csatlakozz:



goo.gl/FRTMI



facebook.com/fullcirclemagazine



twitter.com/#!/fullcirclemag



linkedin.com/company/full-circle-magazine



ubuntuforums.org/forumdisplay.php?f=270

A FULL CIRCLE-NEK SZÜKSÉGE VAN RÁD!

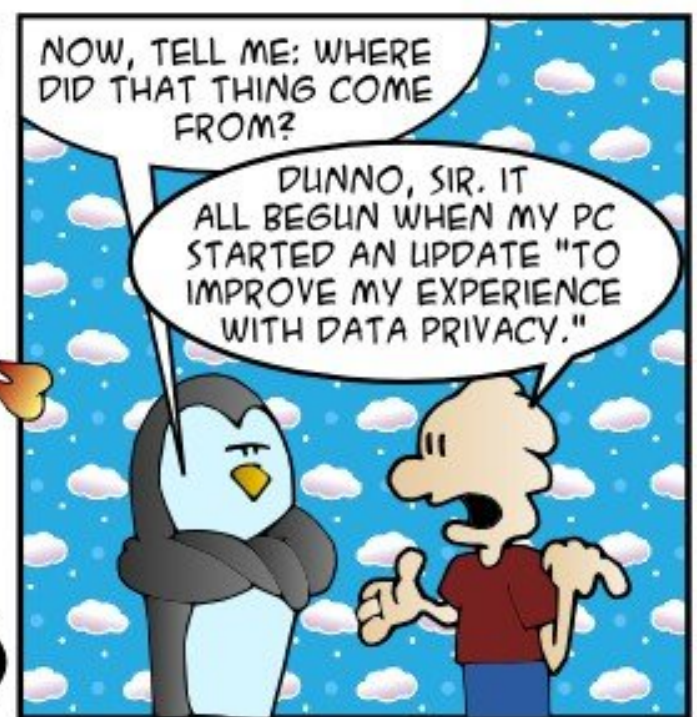
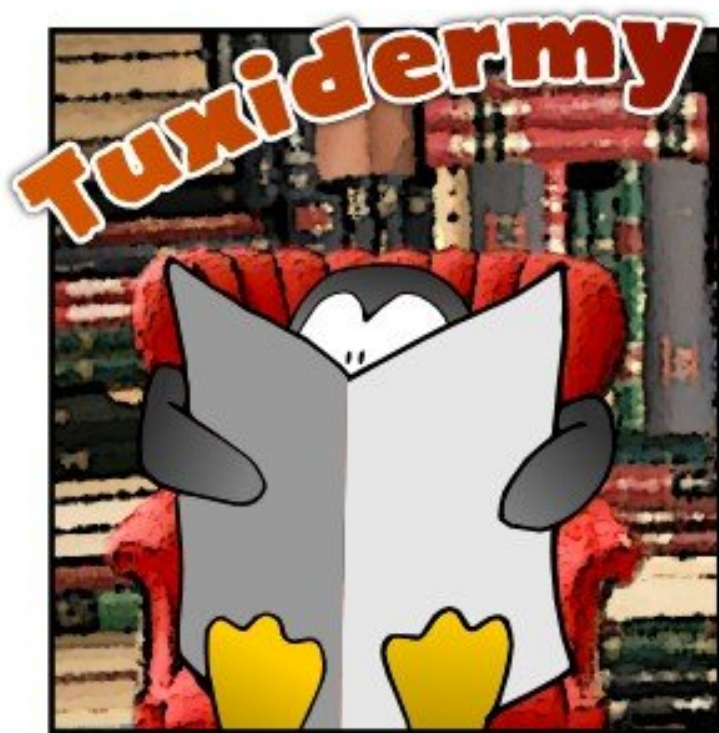


Olvasói tartalom nélkül a **Full Circle** egy üres PDF fájl lenne (amit szerintem nem túl sokan találnának érdekesnek). Mindig várunk cikkeket, termékbemutatókat, teszteket, vagy bármit. Még az olyan egyszerű dolgok, mint egy levél, vagy egy képernyőkép is segít megtölteni a magazint.

Az irányelveinkről a „**Hogyan írjunk a Full Circle-be**” oldalon olvashattok. Ha betartjátok ezeket, garantált a siker.

Az utolsó oldalon találjátok, hogy hova kell küldeni a cikkeket.





*Monster dialogues are too offensive to be shown here



Kávé

Összegyűjtötte: Gord Campbell
Fordította: Veres László

Ha Ubuntuval kapcsolatos kérdéseid vannak, küldd el őket a questions@fullcirclemagazine.org címre és Gord válaszolni fog rá valamelyik későbbi számban. Kérjük, annyi információt küldj a problémáddal kapcsolatban, amennyit csak tudsz!

K Az ECS Liva gépemen Mythbuntu 12-t használok és nincs hang a HDMI-n keresztül.

V Tanácsolom, hogy nézz bele azokba az ismertetőbe melyek az ECS Liva-val kapcsolatosak a Newegg oldalon. Számos hozzászólás javaslatokat is tartalmaz. Már az első a listából ajánlja a legújabb kernel használatát. Rengeteg leírás létezik kernelfrissítéssel kapcsolatosan, például ez itt: <http://ubuntu-handbook.org/index.php/2014/08/install-upgrade-linux-kernel-3-16/>

K Szeretném megtudni, hogyan konvertálhatom a FLAC-et AAC-be?

V Használd a Sound Convertert, melyet lehet, hogy telepítened kell.

K Arra vagyok kíváncsi, hogy miként tudok mentést készíteni a Windows 8.1-ről az otthoni szerveremre, melyen Ubuntu szerver fut.

V Feltételezve, hogy a szerverednek vannak megosztott mappái, így például a Z: meghajtót a Windowsban levő megosztott mappához rendelheted. Telepítsd a Macrium Reflect ingyenes verzióját, és utasítsd arra, hogy a lemezképet a Z: meghajtóra mentse.

K Hogyan találom meg egy USB meghajtó elérési útját?

V A fájlkezelődben, vidd az kurzort a felcsatolt USB neve fölé, így az elérési út meg fog jelenni. Másik lehetőség, ha a terminálba kiadod a df parancsot. Az USB úgy jelenik meg majd mint /media/néhány-szám-és-betű.

K Éppen feltelepítettem a 14.04-et és amikor Youtube videókat nézek a böngészőben, állandóan újraindul a gépem. De amikor letöltöm a videókat és VLC-ben lejátszom, akkor kitűnően működnek.

V Az Opera böngésző használata megoldja a problémát.

A LEGJOBB KÉRDÉSEK AZ ASKUBUNTUN

• Töröljünk minden fájlt egy mappából, kivéve a PDF-fájlokat
<http://goo.gl/EBTQu3>

• Lehetséges a Unity felső paneljét automatikusan elrejtetni?
<http://goo.gl/bZYbgP>

• Mi a különbség a „>” és a „>” között?
<http://goo.gl/KdBrhE>

• Szöveg keresése több .pdf és .doc fájlban
<http://goo.gl/Cn9NSC>

• A Wine-on keresztül szerzett vírusok csak akkor lépnek működésbe amikor a Wine fut?
<http://goo.gl/1y1t2W>

• Hogyan lehet a számítógépet ki-csatolni a billentyűzetről?
<http://goo.gl/u2aiwv>

• Hogyan szerezhetem vissza a hozzáférési jogomat a titkosított home mappámhoz, ha megváltoztattam a jelszavam?

<http://goo.gl/ej1sVA>

• Swap létrehozása az Ubuntu 14.04 LTS telepítése után
<http://googl/GANX5Z>

Gyárilag beépített képernyő letiltása
<http://goo.gl/Q30eBR>

TRÜKKÖK ÉS TECHNIKÁK



Nem olyan nehéz az

B arangolásaim során szoktam keresni érdekes linuxos kérdéseket. Nem szokatlan olyanokat látni, mint „a leírás szerint jártam el, de a nyolcadik, (vagy a 14., vagy éppen a 37.) pontnál hibába ütköztem.” Mondom is magamnak legtöbbször, hogy „kevesebb pontból álló leírásokat kellene keresni.”

Ebben a hónapban teljesítettem feleségem nagyon összetett kérését. Öt lépésből állt a válasz, amit a Google adott. (Az első és a harmadik lépés nekem eszembe se jutott volna.)



Néhány feladatnak tényleg sok lépésből áll a megoldása, de sokkal könnyebb ha feldaraboljuk a problémát, és egyenként keresünk rájuk. A „hogyan építhetem meg Rómát” biztos kudarc, de a „hogyan készíthetem elő a terepet a Colosseumnak”, az már jó eséllyel kecsgetet.

Tökéletes példa a kedvenc videószerkesztőm. A „hogyan szerkeszthetek videókat Cinerellaban” keresésre nem fogunk használható segítséget kapni, míg a „hogyan tudok Kern Burns effektust készíteni a Cinerellaban” keresés már csodaszép leírást fog eredményezni.

Tehát az én tanácsom az, hogy ha sok lépésből álló leírást találsz, próbáld meg több részre felosztani a kérdésed, és lehetséges, hogy jobb leírásokat kapsz majd minden egyes részre.

És csak úgy mellékesen, álljon itt egy weboldal, melyet bárcsak én készíthettem volna:

<http://linux.oneandoneis2.org/LNW.htm>

Kedvenc idézetem a „Linux nem Windows” címszó alatt, a „Linuxnak nincsenek vásárlói.”



Gord a számítógépes iparág egyik régi bútordarabja. Egy időre visszavonult a szakmától, aztán nemrég azon kapta magát, hogy egy 15 fős, „The IT Guy” nevű cégnél dolgozik Toronto belvárosában.



Az FSEconomy (FSE) legnehezebb része valószínűleg a weboldal használata. Mi az FSE? Az FSE hozzáad az X-Plane-hez egy gazdasági rendszert, ahol (kezdetben) bérelsz egy repülőt, elvégzel egy munkát, elrepülve A pontból B-be, és begyűjtöd a jutalmat. Végül összegyűjtesz elég pénzt ahhoz, hogy vegyél egy saját gépet, és így több pénzt tudsz csinálni. Nyilvánvalóan, fel kell mérned az összes munka előnyeit és hátrányait. A repülő bérleti díja olyan magas, hogy felemészti az összes fizetséget? Igen – egy pár figyelmetlen munkavállalás után végezheted úgy, hogy több pénzzel tartozol, mint amennyid ténylegesen van.

ELŐFELTÉTELEK

Az FSE telepítése előtt fel kell telepítened a Python plugint. Ezt már pár számmal ezelőtt megtettük. Szóval, ha esetleg még nem tetted meg, itt az ideje. Még ezen kívül is telepítve kell legyen a Python a számítógépedre. De ha az X-Plane pluginok működnek, akkor több mint valószínű, hogy a Python telepítve van már.

TELEPÍTÉS

Mivel már tudod, hogyan kell telepíteni a Python plugineket az X-Plane-hez, itt van egy URL a plugin-hez:

<https://github.com/ksqy/x-economy>

Nyilvánvalóan neked csak a .PY fájlra van szükséged, amelyik, mint azt tudod, a .../plugins/Python-Scripts mappádba megy.

Annak érdekében, hogy meggyőződj, fel van-e telepítve, izzítsd be az X-Plane-t, és ellenőrizd a plugins menüt. Lenni kell egy bejegyzésnek az X-economy-hoz.

REGISZTRÁLÁS

Ez egy kicsit furcsa lesz. Először is regisztrálnod kell az oldalon: <http://www.fseconomy.net/>. Ezzel tudsz belépni az oldalra. Most regisztrálnod kell a fórumba (<http://www.fseconomy.net/forum/index>), és üzenetet küldeni a Game World Account Request szekciónak. <http://www.fseconomy.net/forum/>

X-Plane és FSEconomy

[new-account](#) Egy moderátor készíteni fog neked egy fiókot, majd erről értesítést kapsz. Érkezik majd egy e-mail a felhasználóneveddel és egy jelszóval, amit nyilvánvalóan meg kell változtatnod.

Már majdnem kész vagyunk!

BEJELENTKEZÉS

Vissza az X-Plane-hez. Az X-economy ablakban beírod az FSE felhasználónevedet és a jelszót. Ismét ez egy kicsit zavaros, mert egy csomó dolog az FSE oldaláról nem bukkan fel a plugin ablakában, amíg el nem indítottad, vagy legalább félig vagy a munkában. Röviden, amíg azt mutatja az oldal, hogy úton vagy, és a plugin bejelentkezett, addig minden rendben van.

MUNKÁT TALÁLNI

Végre! Elértünk arra a pontra, ahol munkát tudunk vállalni!

Az FSE oldalán, menj az oldal menüjéhez, és válaszd az FSE Game World > New Browser Page-et. Ez eljuttat téged az FSE munka oldalá-

ra (<http://www.fseconomy.net:81/>).

Nos, találjunk egy példamunkát. Kattints a képernyő tetején levő Airport linkre.

Itt kereshetsz munkát repülőterenként, gépenként, stb. Én általában az Eclipse 550, a kedvenc gépem alapján keresek munkát. „Airports that have this craft” a neve annak a lenyíló menünek ahol kiválasztod a repülőt... Erre rákattintok és kiválasztom az Eclipse 500-at (a következő legjobb dolog).

Ezután ellenőrizd az ablakot a lenyíló menü mellett. Ez mutatja azokat a repülőtereket, ahol ezeket a gépeket lehet bérelni. Még nincs saját gépünk az FSE-ben, így bérelnünk kell.

Végül kattints a GO-ra, hogy lásd az eredményt.

Rendezheted az eredményeket repülőtér, ország, stb. szerint.

Lássuk! Van egy a Benbeculában (EGPL). Kattintsunk rá, és nézzük mit kapunk.

Search Airports

By ICAO: By aircraft registration:

By name or city:

Airports that have this aircraft:

Eclipse 500 ☐

rentable

Mielőtt ránézünk ezekre a munkákra görgessünk lejjebb egy kicsit, nézzük meg mennyibe kerül a bérlet.

Aú! Ez 751 dollár per óra, tanolkolva. Vagyis megtankolva, felszállásra készen. Szóval tovább kell nézegetnünk a munkákat, és egy olyat találni, ami megéri.

Tengeri mérföld (Nautical Miles, NM) szerint szeretem csoportosítani a munkákat, mert nem szeretem ami túl sokáig tart.

Mondjuk a „3 Benbecula Nurse” munkát szeretnénk. Azt, amelyik 1132 dollárt fizet. Emlékezz, ha egy óra kell a munka elvégzésére (márpedig nem kell, mert csak 27 NM), akkor marad körülbelül 400 dollár

\$1,763.00	EGPL	EGPF	137	134	4 Patient Charter	T	[not provided]	17 hours	Aircraft
\$1,109.00	EGPL	EGAC	179	163	2 Gov't Charter	T	[not provided]	1 hours	Aircraft
\$852.00	EGPL	EGPR	27	185	2 Benbecula Passengers	T	[not provided]	3 days	Aircraft
\$1,132.00	EGPL	EGPR	27	185	3 Benbecula Nurse	T	[not provided]	2 days	Aircraft
\$8,962.00	EGPL	EINN	292	191	15 Passengers	T	[not provided]	20 hours	Aircraft
\$3,972.00	EGPL	EISG	197	193	Domestic Air Cargo 745kg	T	[not provided]	2 days	Aircraft

rod. Ha csak 30 perc kell, akkor fizetsz 350 dollár bérleti díjat, és a maradék a tied.

A munka elvállalásához kattints az üres négyzetbe a munka mellett, és a gombra (a munkák alatt) melyen áll az Add Selected Assignments To -> opció, és a lenyíló menüben mellette látod a My Flightot. Az oldal újratölt, és a munka amit elvállaltál most már listázva lesz ahogy kérted.

Görgess le a repülőig, és kattints a „Rent Wet-re” a gép mellett, amit választottál. Ez az Eclipse 500 az én esetemben. Rendben. Van munkád, van géped. Kattints a My Flightra az oldal tetején.

Ez az oldal mutatja a munkád részleteit. Ott van az indulásod EG-

PL-ből EGPR-be, ami 27 NM, és van egy repülőd 33% üzemanyaggal (elég lesz a munkához), és hogy a munka készen áll, hogy elkezd.

Vissza az X-Plane-hez, és győződj meg, hogy bejelentkeztél az X-economy pluginbe, aztán kattints a Start Flying gombra. Lehet, hogy azonnal mutatja a munka részleteit, de lehet, hogy csak miután felszálltál. A plugin néha furcsán viselkedik, mint ez is. Töltsd újra az FSE oldalt, és azt mutatja, hogy úton vagy.

Akárhogy is, végezd el a munkát. Fékezz le, állítsd le a motort. Ha minden jól ment az X-Plane felbukkan egy üzenettel, mely szerint a munka elvégezve. Úgy találtam, hogy néha elég csak befékezni a munka befejezéséhez. Néha le kell

húlnie a gépednek, és elsötétednie (minden lekapcsolva). Néha meg kell nyomnom a V billentyűt, hogy a kézféket behúzzam.

Menj vissza az FSE munkák oldalára, és nézd meg az oldal jobb felső részét. Itt fent láthatod a jelenlegi egyenleged.

Biztonságos repülést!

Logged in as **ronnietucker**
Cash Balance: \$1,347.76
Bank Balance: \$0.00
Hours Flown: 0.00 in last 48
[Log out](#)

Dick Thomas-nak jár a pont, mert megmutatta, hogy használgam a nem túl intuitív FSE oldalt.

Aircraft						
Id	Type	Equipment	Home	Rental Price	Bonus	Action
G-GM2*	Beechcraft 18	IFR/AP/GPS	EGKA	[Hour]	\$0.00	
OH-CNH	Beechcraft Baron 58	IFR/AP/GPS	EFKE	\$205 Dry/\$312 Wet [Hour]	\$58.00	Rent dry Rent wet
G-IVDR*	Douglas A-26	IFR/AP/GPS	EGNF	[Hour]	\$0.00	
C-FSGN	Eclipse 500	IFR/AP/GPS	CYVR	\$420 Dry/\$751 Wet [Hour]	\$71.00	Rent dry Rent wet
TF-QXQ	Howard Aero 500	IFR/AP/GPS	BIRK	\$579 Dry/\$1529 Wet [Hour]	\$370.00	Rent dry Rent wet
AJ-Y-8*	Maule M-7	VFR	EGNC	\$220 Dry/\$330 Wet [Hour]	\$50.00	Rent dry Rent wet
G-TDSE	Stinson L-5B Sentinel	VFR	EGMA	\$29 Dry/\$56 Wet [Hour]	\$31.00	Rent dry Rent wet



Játékok Ubuntu

Írta: Oscar Rivera

Fordította: Molnár Tibor

Szóval, most, hogy 2014-et magunk mögött tudjuk, visszatérve elmondhatjuk milyen nagy-szerű év volt ez a Linux játékok számára. A rettenetesen nagy számú játék, amely bekerült a Linux játék könyvtárába, valamint ezen játékok arányának folyamatos növekedése tették lehetővé számos duál bootos felhasználó számára, hogy kiirtsák a Windowsos partíciójukat, amit csakis a játékok miatt tartottak meg.

Szóval most, hogy ennyi játék elérhető, a rendszerünk felkészült a legfrissebb AAA játékokra, melyekre oly régóta várunk? Nos, jó ha tudjuk, van egy mód mellyel ki lehet deríteni. Mostanra már több mint egy éve írok az FCM-nek cikkeket a játékokról, gyakran kell támaszkodnom a közzétett benchmarkra, hogy tájékozottabb legyek a játékokról, amelyekkel játszok, és/vagy amelyekről beszámolót írok. Többnyire a processzorom, a merevlemezem, és RAM-om egyezik a mai játékok által támasztott igényekkel, és azt hiszem ez így van az olvasóim zömével is. Habár, amikor szóba kerül a videokártya, és a

processzor, ez az a pont, ahol kiderül néhányunknak nincs meg a szükséges minimum, ami kell a régóta türelmesen várt játék futtatásához. Szóval, azon gondolkodtam, van-e megbízható benchmark-szoftver Linuxra, amivel kitalálhatnám, vajon a rendszerem megfelel-e a legfrissebb, legigényesebb videojátékoknak? Azt a célt tűztem ki magam elé, hogy megtaláljam a legegyszerűbb, és legmegbízhatóbb Linuxos benchmark-szoftvert. Habár zárt forráskódú, és nem a legegyszerűbb

rúbb telepíteni Linuxra, a Unigine's Heaven Benchmark univerzális, és mint ilyen talán az egyetlen, legtöbbet használt benchmark applikáció. Az Unigine's Heaven Benchmark elérhető Windowsra, Mac OS X-re és Linuxra. Ahogy korábban jeleztem, a telepítés összetettebb folyamat, mint Windows-on, vagy Macen. Miután találtam pár weboldalt nagyon eltérő utasításokkal, hogyan kell telepíteni és futtatni a programot, sikeresen telepítettem, és futtattam a Heavent,

elkerülve minden nagyobb katasztrofát.

TELEPÍTÉS

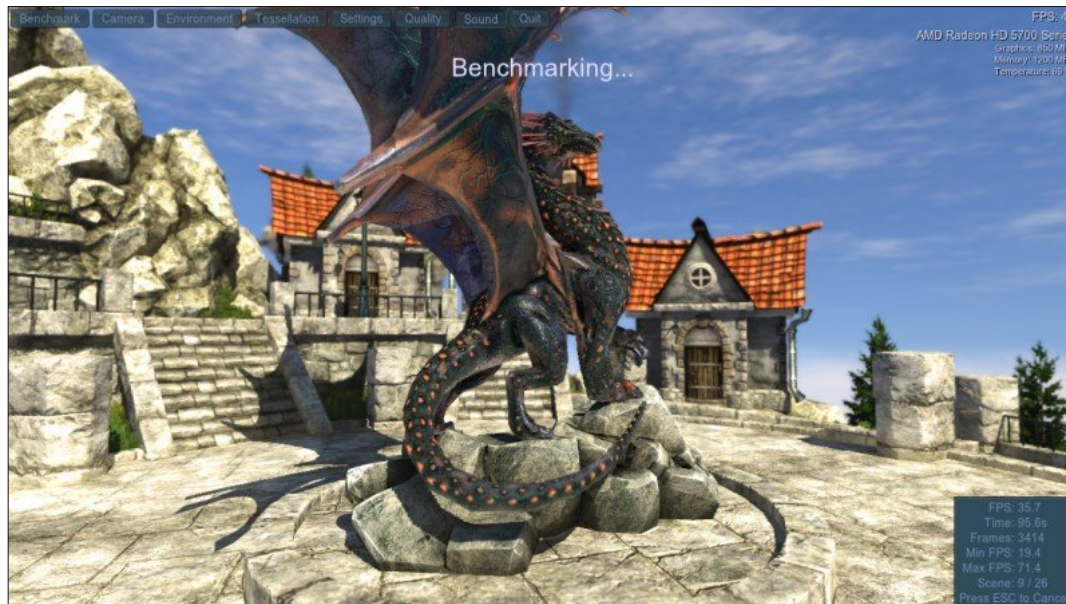
A legbonyolultabb dolog a Unigine's Heaven 4.0 használatával kapcsolatban nekem a telepítés volt. Sajnos a Unigine Heaven nem elérhető egyetlen hivatalos Ubuntu tárolóból sem, ami azt jelentette, hogy meg kellett nyitnom ezt az oldalt <https://unigine.com/products/heaven/>, ahonnan letölthettem a telepítő csomagot. A Unigine's weboldalán három lehetőséget találunk: Basic Edition ami ingyenes, Pro Edition vagy Advanced Edition. Van egy lista is, melyben a különbséget megtalálhatjuk a három kiadás között.

A cikk megírásához és amire én használni akarom, a Basic Edition elégséges. Az Advanced Edition 14.95 dollárba, míg a Pro Edition 495 dollárba kerül. Az átlagembernek, aki csak a rendszerét szeretné tesztelni, csak a Basic Editionre van szüksége. A fájlnak, amit letöltöttem a neve Unigine_Heaven-4.0.run, és körülbelül 280 MB. a



mérete. Miután letöltöttük a telepítőt, meg kell változtatni az engedélyeit, hogy végrehajtható fájlként fusson, és talán változtatnunk kell a szkript egy során, amennyiben 64 bites operációs rendszerünk van. Néhány hasznos információ található itt: <http://ubuntuxtre-me.com/howto/how-to-run-unigine-heaven-3-0-in-ubuntu-12-04-64bit/3/>, de ne felejtsük el átírni az „Unigine_Heaven- 3.0”-át „Unigine_Heaven-4.0”-ra ahogy követjük az instrukciókat, mert amit telepíteni fogunk az valószínűleg a 4.0 lesz, és nem a 3.0 (ami egy régebbi verzió). Találtam egy másik csokor nagyszerű instrukciót itt: <http://ubuntuforums.org/showthread.php?t=2144692>, és a végén egy kicsit mindkettőből használtam, hogy feltelepítsem, és futtasam a rendszeremen a Heavent. Mikor végre feltelepítettem, egyáltalán nem tudtam a szoftvert futtatni. Át kellett váltanom a Downloads mappámra (ahova telepítettem a Heavent), és így tudtam futtatni a szoftvert egy terminálból:

```
cd ~/Downloads/Unigine_Heaven- 4.0
./heaven
```



LEHETŐSÉGEK, ÉS EREDMÉNYEK

Páran közületek elgondolkodhatnak azon, mi célból futtatunk egy ilyen benchmark szoftvert, mint az Unigine's Heaven. Rengeteg előnye van annak, ha a hardveinket rendszeresen leteszteljük. Amennyire megnyugtató azon emberek számára, akik játékaikat olyan konzolokon játsszák, mint a Playstation 3/4 vagy Xbox360/XBox One, hogy a játék, amit pont most vettek, minden probléma nélkül, szaggatás nélkül, a lehető legjobb grafikával fog futni a választott konzoljukon. Nekünk, akik a számítógépünkön játszunk, semmi támpontunk nincs, hogy bármelyik

játék, amit épp most vettünk (vagy a megvételén gondolkodunk), játszható-e egyáltalán. Egy rendes számítógép teszt jó kiindulási pontot biztosít, mely játékok futnak simán maximum beállításokkal, minimum beállításokkal, vagy egyáltalán nem működnek.

A Heaven felhasználói felülete ablakos, és körülbelül az asztalom egyharmadát foglalja el. Az egyik beállítási lehetőség, mielőtt ténylegesen futtatnánk a benchmarkot az az, hogy ablakos módban, vagy teljes képernyőn akarjuk futtatni a benchmarkot. Mivel a játékaim 99,9%-át teljes képernyőn játszom, ezért azt találtam értelmesnek, hogy a Heavent csak teljes képer-

nyőn futtassam, és így is tettem. Van még néhány másik beállítási lehetőség, mint például Nyelv, Minőség, Tesszelláció, Stereo 3D, Multi-monitor, Antialiasing, Teljes képernyő, és Felbontás. Aztán ezen választási lehetőségek alatt van egy „Run” feliratú gomb, ezt kell megnyomni, ha készen állunk a benchmark futtatására. A benchmark maga egész jól néz ki...

Van egy lebegő város, egy sárkánnyal a város terének a közepén, 360 fokban szögben megtekinthetjük a sárkányt, valamint az egész várost, mielőtt felszáll egy léghajó-szerűséggel az égbe. Megnyomjuk a „Run”-t, és csak élvezzük a grafikát, és a zenét, ami társul hozzá, de amikor felkészültünk a rendszerünk tényleges tesztelésére, és néhány infót begyűjteni, akkor használnunk kell az egeret, menjünk a képernyő bal felső sarkába, és kattintsunk a gombra melyen ez áll „Benchmark”. A benchmark lefut, és megadja nekünk az eredményeket, ahogy végzett. A teljes benchmark 26 jelenetet tartalmaz – ami a jobb alsó sarokban nyomon követhető, mialatt az aktuális benchmark fut. A jobb felső sarokban látható néhány információ a rendszerünkről, úgymint: milyen grafikus kártyánk van, a memóriából mennyit

használ, a jelenlegi hőmérséklete (ami ugyebár változik, mert a benchmark fut). Láthatod mennyi képkocka-per-másodpercet (FPS) tud a rendszer kipróbálni magából a teszt futása közben. A képernyő jobb alsó sarkában (és csak azután látható, ha a „Benchmark” gombra kattintottunk), látható néhány további információ a benchmarkról, például mennyi ideig fut (Time), képkockák, minimum képkocka-per-másodperc, maximum képkocka-per-másodperc, és a jelenlegi pontjaink. Ha nem főzi meg a videokártyánkat, megpróbálhatjuk lefuttatni az egész teszt teljes folyamatát, annak érdekében, hogy még pontosabb eredményeket kapjunk. Amikor a benchmark véget ért, a végeredmény megjelenik a képernyőn, és lehetőségünk lesz elmenteni HTML formában. Azonkívül, hogy elmentjük az eredményeket, javasolom, hogy készítsünk egy képernyőmentést is. Azért javaslom a képernyő mentést, mert ha össze szeretnénk hasonlítani az eredményt, és megosztani, néhány fórum megköveteli, hogy küldjük el a képernyőmentést, igazolva, hogy ténylegesen lefuttattuk a tesztet. Az egyik weboldalnak, a techpowerupnak van egy listája az eredményekről, amit arra fogok használni, hogy kitaláljam melyik videokártyát

vehetném figyelembe frissítéskor, különösen, mert az én eredményeim jobbak, mint pár újabb grafikus kártyáé, amit láttam a listán.

RENDSZERKÖVETELMÉNYEK

Igen, ha hiszed, ha nem vannak minimális rendszerkövetelményei a Heaven Benchmarknak; végül is a játékhoz szükséged van egy félig-meddig tisztességes rendszerre. Egy dolog, amit a weboldal nem árul el, hogy zárt forráskódú grafikus kártya-meghajtóra van szükséged a Heaven futtatásához.

Ezt mutatják a Unigine Heaven honlapján:

- GPU:
- ATI Radeon HD 4xxx és magasabb

- Nvidia GeForce 8xxx és magasabb
- Intel HD 3000 és magasabb
- Videó Memória: 512 MB
- Tárhely: 1 GB
- A hardver tesszelációhoz kell egy videokártya DirectX 11/OpenGL 4.0 támogatással, és MS Windows Vista/7/8, vagy Linux.

AZ ÉN RENDSZEREM

Én a Unigine Heaven Benchmark-ot rendelésre készült asztali PC-men futtattam, ami a következőkből áll: egy AMD FX-6100 3,3 GHz CPU (túlhúzva 3,5 GHz-re), egy Asus M5A97-EVO alaplap, egy Sapphire Radeon HD 5770 videokártya, 8 GB Kingston Hyper X RAM, és egy 1 TB-os Seagate Barracuda merevlemez. A szoftver, amit használtam

Ubuntu 14.04.1 LTS volt, Unity asztallal AMD 13.9 zárt forráskódú grafikus meghajtóval.

VÉGKÖVETKEZTETÉS

Elteltekintve a küszködéstől, amivel mi, Linux felhasználók a Unigine Heaven telepítése közben bizonyosan találkozni fogunk, minden más szempontból a benchmark majdnem tökéletes minősítést kapott. A jelenetek, melyeken keresztül tesztel, gyönyörűek, és nagyszerű módja annak, hogy felfedezzük, mire képes ténylegesen a videokártyánk. Ráadásul működés közben mutatja az FPS-t (képkocka-per-másodperc), és egyéb videokártya-információkat (köztük a hőmérsékletét). Az eredmények, melyeket a végén ad, nélkülözhetetlenné teszi minden Linux játékos számára.



Oscar a CSUN-n szerzett diplomát, jelenleg zenei igazgató/tanár, béta tesztter, Wikipedia szerkesztő és Ubuntu Fórumok résztvevője. Követheted itt: www.gplus.to/7bluehand; twitteren: @7bluehand vagy küldhetsz neki emailt: www.7bluehand@gmail.com

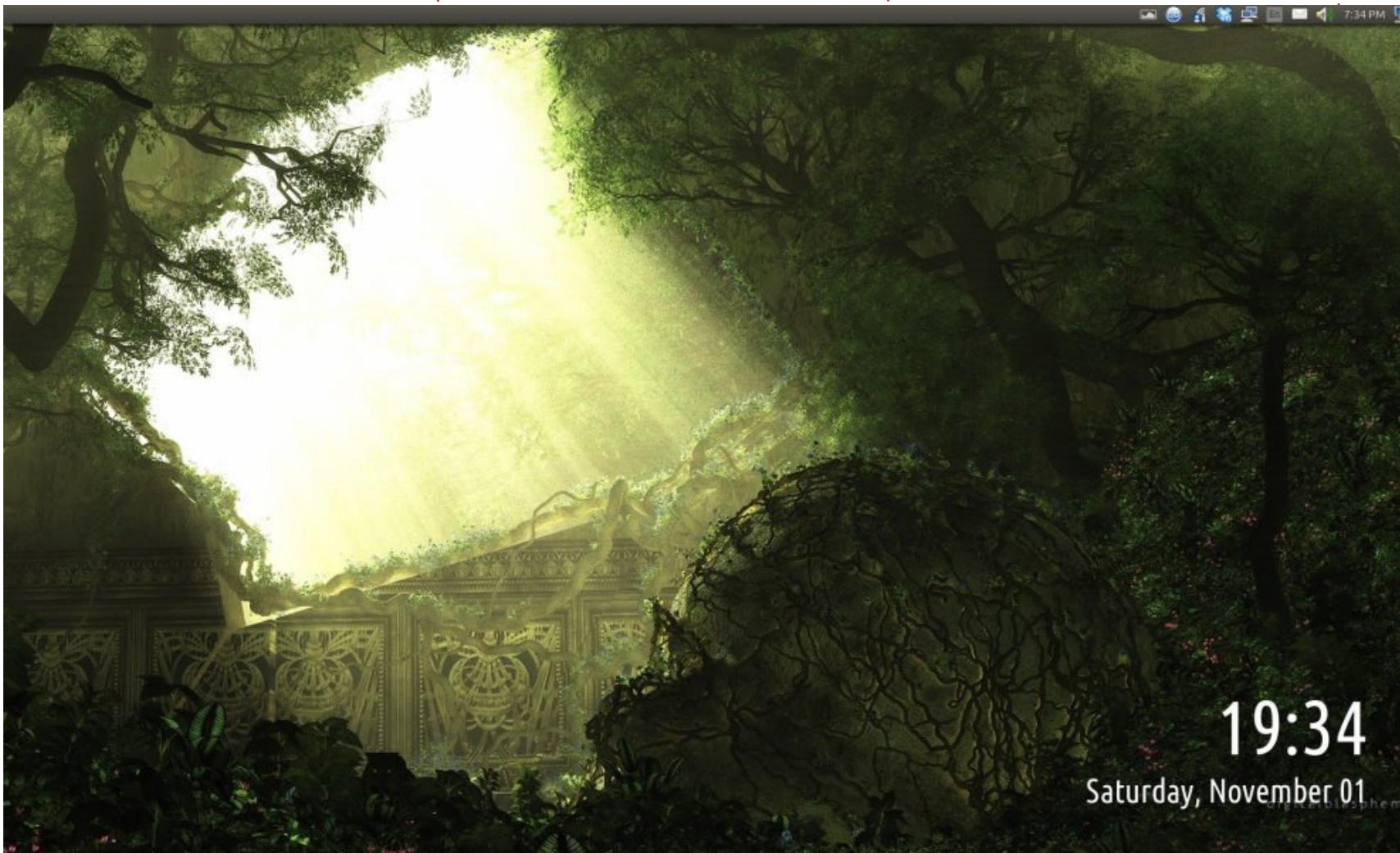




Az én asztalom

Fordította: Molnár Tibor

Itt az alkalom, hogy megmutasd a világnak az asztalodat (desktop) vagy PC-d. Küldj képernyőképeket és fényképeket a misc@fullcirclemagazine.org e-mail címre! Kérlek, mellékelj egy rövid szöveges leírást az asztalodról, a saját gépedről vagy az asztalod illetve a PC-d bármely egyéb érdekességeiről.



box add-in használatával, a Google Drive mappát az Insync add-in használatával szinkronizálom.

Variety –

<http://peterlevi.com/variety>

Digital Blasphemy –

<http://www.digitalblasphemy.com>

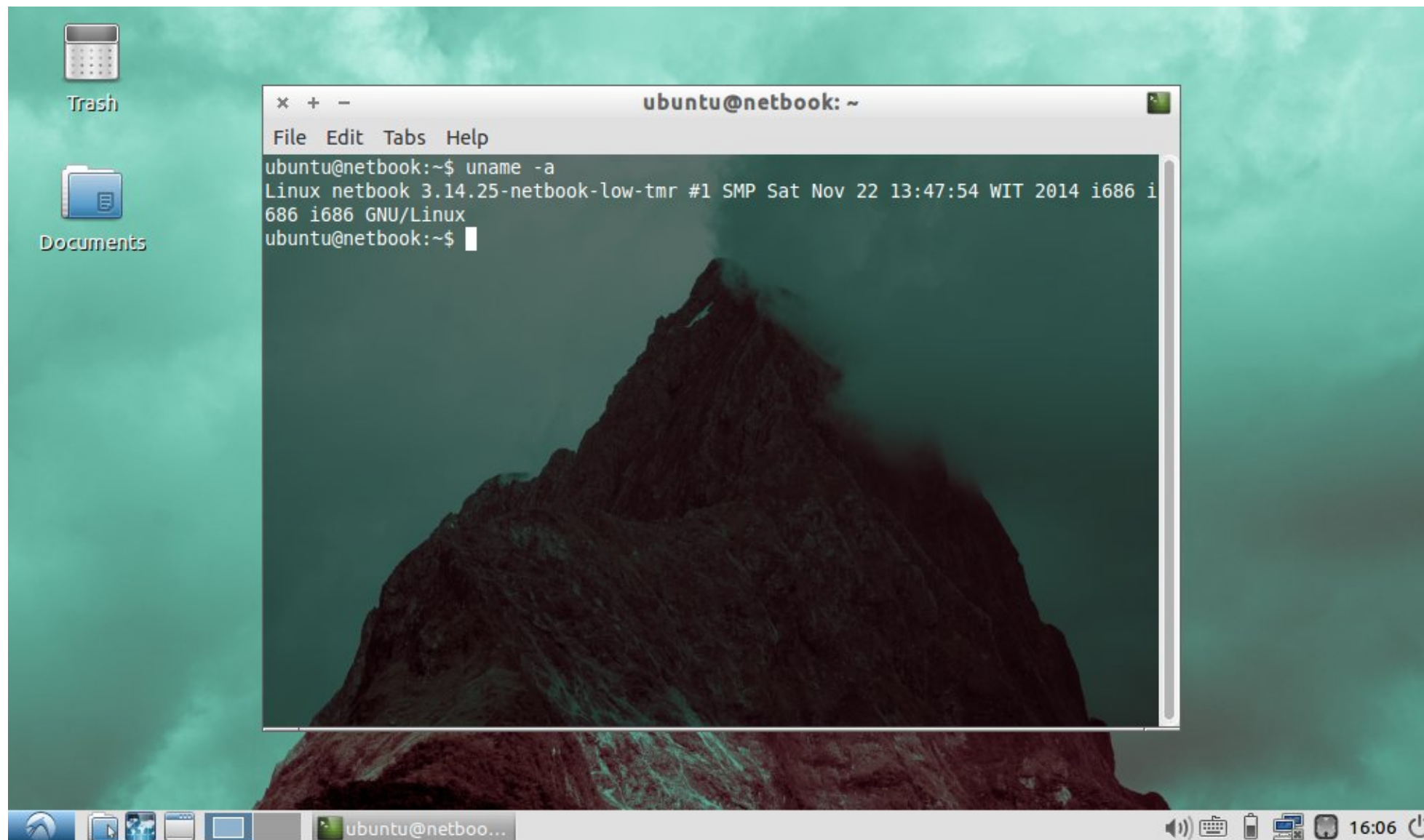
Gar Nelson

Az én rendszerem egy régi Shuttle Glamor XPC, melybe dupla grafikus kártya van telepítve. A két monitor egy-egy 24"-os Samsung, amelyek mindegyike

1920x1080 felbontásra van állítva, összességében 3840x1080. (DVI-t HDMI-be átalakító kábelt használok.) Az operációs rendszer Ubuntu 14.04 LTS, és Varietyt futtatok, ami

egy órát jelenít meg, és véletlenszerű idézeteket a Digital Blasphemy háttérkép-sorozaton. Oxygen ikontémát használlok, és redglass kurzort. A Dropbox mappát a Drop-





Sziasztok! Én Ubuntu felhasználó vagyok. Általános iskolás korom óta Ubuntut használok. A netbookra optimalizált szokásos kernellel tettem a Lubuntut még kényelmesebbé a netbookom számára.

mára.

A netbookom adatai:

Proc: Intel Atom N270 1,6 GHz
RAM: 1 GB DDR2

Háttértár: Sandisk Cruzer Blade 8 GB
Lubuntu 14.04-et használlok.

Faqih Jakha Juantomo



Közreműködnél?

A FULL CIRCLE-nek szüksége van rád!

Egy magazin, ahogy a Full Circle is, nem magazin cikkek nélkül. Szükségünk van játékok, programok és hardverek áttekintő leírására, ezenkívül bármire, amit elmondanátok a *buntu felhasználóknak. A cikkeiketek küldjétek a következő címre: articles@fullcirclemagazine.org

Folyamatosan keressük a cikkeket a magazinba. Segítségül nézzétek meg a **Hivatalos Full Circle Stílus Útmutatót**: <http://url.fullcirclemagazine.org/75d471>

Véleményed és Linuxos tapasztalataidat a letters@fullcirclemagazine.org címre, Hardver és szoftver **elemzéseket** a reviews@fullcirclemagazine.org címre, **Kérdéseket** a „Kávét” rovatba a questions@fullcirclemagazine.org címre, **Képernyőképeket** a misc@fullcirclemagazine.org címre küldhetsz, ... vagy látogasd meg a **fórumunkat** a fullcirclemagazine.org címen.



FCM 94. szám

Lapzártá:

2015. február 8-a, vasárnap

Kiadás:

2015. február 27-e, péntek



A Full Circle Csapat



Szerkesztő – Ronnie Tucker
ronnie@fullcirclemagazine.org

Webmester – Rob Kerfia
admin@fullcirclemagazine.org

Podcast – Les Pounder & Co.
podcast@fullcirclemagazine.org

Szerkesztők és Korrektorok

Mike Kennedy, Gord Campbell, Robert Orsino, Josh Hertel, Bert Jerred, Jim Dyer és Emily Gonyer

Köszönet a Canonical-nek, a fordítócsapatoknak a világban és **Thorsten Wilms**-nek az FCM logóért.



A Full Circle Magazin beszerezhető:

EPUB - Az utóbbi kiadások megtalálhatók epub formátumban a letöltési oldalon. Ha bármi problémád lenne az epub fájljal, küldj e-mailt a mobile@fullcirclemagazine.org címre.



Issuu - Olvashatod a Full Circle magazint online az Issuu-n: <http://issuu.com/fullcirclemagazine>. Oszd meg és értékelj a magazint, hogy minél többen tudjanak a magazintról és az Ubuntu Linuxról.



Google Play – Már olvashatod a Full Circle magazint a Google Play/Books oldalán. Keresd a „full circle magazin”-t, vagy kattints ide: <https://play.google.com/store/books/author?id=Ronnie+Tucker>



Full Circle Magazin Magyar Fordítócsapat

Koordinátor:
Pércsy Kornél

Fordítók:

Bozóki András
Horváth Albert
Jancsek Árpád
Meskó Balázs
Molnár Tibor

Palotás Anna
Sipos Zoltán
Takács László
Veres László

Lektorok:

Almási István

Veres László

Szerkesztő:
Kiss László

Korrektor:
Heim Tibor

