

Jun 2015. Broj 37

LIBRE!

Časopis o slobodnom softveru



 FREE SOFTWARE
FOUNDATION



JOŠ IZDVAJAMO

Fedora 22 - vaš novi šešir!
Banjaluka Barkamp 0x01



Creative Commons Autorstvo-Nekomercijalno-Deliti pod istim uslovima

Reč urednika

Tri godine posle

LiBRE! časopis je napunio pune tri godine postojanja. Za te tri godine objavili smo trideset šest redovnih brojeva u standardnom ritmu jedan broj mesečno i bez dvobroja (letnjih pauza). Redovnost izlaženja smatramo našim najvećim uspehom. Pored toga smo objavili jedan probni broj (broj 00) i jedan specijal o Skribusu.

Obradili smo preko dvesta šezdeset različitih tema o slobodnom softveru u preko četiri stotine članaka. Preko pedeset autora su učestvovali u realizaciji ovih brojeva kao stalni ili povremeni saradnici. U vremenima kad koherentna srpska zajednica zapravo ne postoji, naročito smo ponosni na to da smo makar oko ovog projekta uspeli da okupimo veći broj ljudi da zajedno rade.

Povremeno smo naše čitaoce obaveštavali o našim problemima s ciljem održavanja vitalnosti ovog projekta. Hvala još jednom svima koji su pravilno shvatili naše apele. Naši alarmi nisu bili bezrazložni. Pomoć nam je bila potrebna; bez nje ovaj projekat bi opstao još neko vreme, ali je pitanje da li bi doživeo treću godišnjicu.

Iz godine u godinu se ponavlja ista situacija da su nam meseci maj i juni najteži period. To je period pred godišnji odmor, period najvažnijeg studentskog roka - junskog roka i kraja školske godine. Tad smo svi prilično zauzeti privatnim obavezama i izvinjavamo se



čitaocima zbog malo lošijeg kvaliteta i kašnjenja prošlog, trideset šestog broja.

Ovaj broj realizujemo u pomalo ubrzanom režimu i unapred se izvinjavamo ako nismo uspeli da postignemo naš standardni kvalitet. Ovim brojem ulazimo u našu četvrtu godinu. Imamo mnogo planova kako da unapredimo ovaj projekat. Još uvek se bojimo da glasno progovorimo o svim planovima da nam se ne bi izjalovili. Pokušaćemo da iskoristimo letnju pauzu da malo konkretizujemo ideje kako bismo mogli najesen da ih promovišemo.

Naš konkurs za nove saradnike je konstantno otvoren. Svaka pomoć je dobrodošla. Uspevamo da održavamo časopis, ali smo u manjku ljudi za održavanje sekundarnih delova projekta: veb-prezentacija, Fejsbuk strana, Tviter i Git projekti. Iako su to sporedni delovi projekta, vrlo su važni za njegovu prezentaciju. Ovako zapušteni ovi delovi projekta daju lošu sliku o samom časopisu, koji i sam počinje da izgleda zapušteno. Pozivamo sve zainteresovane koji misle da mogu da nam pomognu da se jave na našu već poznatu adresu elektronske pošte - [libre \[et\] lugons \[dot\] org](mailto:libre@lugons.org).

Do narednog broja,

LiBRE! tim

Sadržaj

Vesti

str. 6

Puls slobode

Banjaluka Barkamp 0x01 - Šta ste propustili?
HKLBGD - Beogradski hakerspejs

str. 9
str. 14

Predstavljamo

Fedora 22 - vaš novi šesir!

str. 20

Kako da...?

Vagrant (4. deo)

str. 28

Slobodni profesionalac

Vaš posao, open-sors posao (4. deo) - Manifest GNU-a
Ansibl (3. deo)

str. 33
str. 46

Sam svoj majstor

Butstrap - komponente

str. 51

Hardver

Biglbon Blek Rev C-Vodič od prvog dana
Biglbon Blek u službi veb-servera

str. 54

Moć slobodnog
softvera





LIBRE! prijatelji



REGIONALNI
LINUX PORTAL

linuxzasve.com



Grupa korisnika GNU/Linux operativnih sistema u Lovčencu

info i tutorijali na srpskom
lubunturs.wordpress.com



Broj: 37

Periodika izlaženja: mesečnik

Izvršni urednik: Stefan Nožinić

Glavni lektor:

Admir Halilkanović

Lektura:

Jelena Munčan

Saška Spišjak

Milena Beran

Milana Vojnović

Aleksandar Božinović

Aleksandra Ristović

Grafička obrada:

Dejan Maglov

Ivan Radeljić

Dizajn: White Circle Creative Team

Autori u ovom broju:

Momčilo Medić

Kriptopank

Nenad Marjanović

Zlatan Vasović

Jelena Georgijević Krasojević

Ostali saradnici u ovom broju:

Marko Novaković

Mihajlo Bogdanović

Počasni članovi redakcije:

Željko Popivoda

Željko Šarić

Vladimir Popadić

Aleksandar Stanisavljević

Kontakt:

IRC: #floss-magazin na irc.freenode.net

E-pošta: libre@lugons.org

Vesti

2. jun 2015.

Kanonikal i Be-Ku (BQ) pokreću novi telefon

Ove dve kompanije su najavile novi model telefona sa Ubuntu operativnim sistemom.

Koristan link: <http://t.co/m0mDkWLFB1>



4. jun 2015.

Plej-on-linuks koristi Java programski jezik od verzije 5

Izvorni kod ove platforme će biti refaktorisan i umesto Pajtona koristiće Javu.

Koristan link: <http://t.co/hk0GD0hCmb>



4. jun 2015.

Objavljen Open-si-vi (OpenCV) 3.0

Objavljena je nova verzija ove biblioteke, do sada najbrža.

Koristan link: <http://t.co/r4lqe7WMGg>





4. jun 2015.

Dragon-flaj BSD ažurira svoje AMD upravljačke programe

Dragon-flaj BSD (*DragonFlyBSD*) distribucija ažurira svoje AMD Radeon upravljačke programe.

Koristan link: <http://t.co/cSzVK9aBTB>



8. jun 2015.

Sistem-d prelazi na Github

Sistem-d (*systemd*) prebacuje svoju infrastrukturu na Github.

Koristan link: <http://t.co/8osggXLMk3>



8. jun 2015.

Nova verzija Linuks kernela

Objavljena je nova verzija Linuks kernela - verzija 4.1-rc7.

Koristan link: <http://t.co/SxEwYyzla5>



8. jun 2015.

Fedora objavila novu verziju svog programa za konfiguraciju

Fedora je objavila novu verziju svog programa za konfiguraciju koji je baziran na GTK3 biblioteci.

Koristan link: <http://t.co/DyRH9ddSAu>
<http://t.co/5RM3hGN2Fu>



Vesti

8. jun 2015.

Američka vlada prebacuje celu infrastrukturu na HTTPS

Američka vlada je najavila prebacivanje cele infrastrukture na HTTPS protokol.

Koristan link: <https://t.co/HF8RIGqVTW>



8. jun 2015.

Epl otvara svoj programski jezik

Ova kompanija je najavila da će otvoriti kod do kraja godine za svoj programski jezik Swift (*Swift*).

Koristan link: <http://t.co/PkFkuZKYAM>



11. jun 2015.

Mate 1.10 objavljen

Nakon petnaest meseci razvoja, verzija 1.10 okruženja radne površi Mate konačno je objavljena. Velika novost je to što Mate od ove verzije podržava GTK2 i GTK3. Međutim, navodi se da se i dalje eksperimentiše sa podrškom za GTK3.

Koristan link: <http://j.mp/1GRRk0G>





BANJA LUKA
barcamp
0x01

Šta ste propustili?

Autor: Jelena Georgijević Krasojević

U subotu, 6.6.2015. godine, održan je prvi banjalučki Barkamp pod nazivom „Barkamp BL 0x01” (BarCamp BL 0x01) na Elektrotehničkom fakultetu u Banjaluci. Sam cilj ovog prvog okupljanja je bio da se probudi i animira zajednica, te da se vrati dinamika dešavanja u Banjaluci. Sve zajednice iz okruženja su bile više nego dobrodošle.



Puls slobode

Takođe, cilj prvog banjalučkog Barkampa je povezivanje zajednica kako regionalno tako i internacionalno, zatim druženje, hakerisanje, učenje i umrežavanje. Poseban naglasak je na jačanju i širenju zajednice, kao i na stvaranju mogućnosti da kompletno okruženje ima mesto na kojem bi se razmenjivala novostečena znanja i iskustva. Nadamo se da smo ovom prilikom ostvarili naš cilj, pošto je „Barkamp BL 0x01“ okupio oko pedeset učesnika - što iz zemlje, tako i iz inostranstva.



Barkamp je zvanično započeo Nikola Nenadić predavanjem „Embedded real-time GNU/Linux“. Potom je usledio Marko Grumić predavanjem - Uvod u io.smashthetack.org - WarGames how-to. Jelena Georgijević Krasojević najavila je BalCon2k15 - Third Luck The Charm - Balkan Computer Congress, koji će se održati u Novom Sadu od 11. do 13. septembra u Muzeju savremene umetnosti Vojvodine.

Zatim je usledilo predavanje Daniela Kurtjaka - Pravljenje aplikacija sa grafičkim korisničkim interfejsom za ugrađene računarske sisteme. Predavanje je bilo vezano za razvoj aplikacija sa grafičkim korisničkim interfejsom za ugrađene računarske



sisteme kao što su Set Top Box i automobilski paneli. Demonstracija je odrađena na proizvodu razvijenom u R&D odeljenju RT-RK kompanije i odnosila se na pravljenje aplikacija sa grafičkim korisničkim interfejsom korištenjem prilagođenog razvojnog okruženja Eklipt (Eclipse). Grafički korisnički interfejs baziran je na Lua programskom jeziku i opisan putem Iks-em-el (XML) shema, koje su se kasnije izvršavale na simulatoru na ličnom računaru pisanom u programskom jeziku C++.

Posle Daniela Kurtjaka usledilo je veoma interesantno predavanje „Upravljanje vremenom“, koje je održao Goran Vukalović. Vreme – najvrednija stvar koju jedan čovek poseduje. Neki ga imaju dovoljno da završe sve obaveze i poslove, dok ih drugi prebacuju za neki naredni period, ili ih nikada ne dovrše. U čemu je razlika? Šta nam je potrebno da završimo sve obaveze



Puls slobode

na vreme? Vremenska mašina? Litri kofeinskih napitaka? Možda. Na ovom predavanju smo prošli kroz nekoliko jednostavnih ali korisnih tehnika za bolje raspoređivanje i efektivnije korištenje vremena.



Prvi banjalučki Barkamp imao je čast da zatvori Stefan Stojanović predavanjem How to Arduino hard.

Barkamp je posle nastavljen druženjem u kafiću Harmonija do kasno u noć. Organizatori prvog banjalučkog Barkampa su Ognjen Vidović, Marko Grumić i Marko Ivanović. Ovom prilikom im se zahvaljujemo na gostoprimstvu i na odličnoj organizaciji Barkampa u nadi da će u što skorije vreme organizovati neko novo dešavanje u Banjaluci.

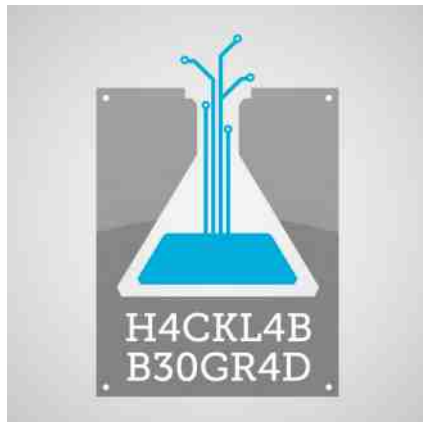
Oni koji su propustili Barkamp, detaljan raspored predavanja mogu pronaći na internetu: <https://events.lugons.org/?p=1856>.



Banjaluca Barkamp 0x01



HKLBGD - Beogradski hakerspejs



Autori: HKLBGD zajednica

„Hakerspejs“ (eng. *hackerspace*), „haklab“ ili „mejkerspejs“ (eng. *makerspace*), termini su za koje ste verovatno već čuli, ali možda ne znate šta označavaju i kakva se ideja krije iza njih. Zato ćemo vas upoznati sa jednom ovakvom oazom koja se krije iza imena „Haklab Beograd“ (HKLBGD) u centru Beograda (<http://oosm.org/>).

Hakerspejs je prostor u kojem se okuplja zajednica čija su interesovanja najviše u vezi sa računarima, tehnikom, tehnologijom, elektronikom, mašinama, ili jednostavno naukom i njenom primenom. To je mesto gde se ljudi sastaju zbog razmene ideja, organizuju radionice i rade na raznim projektima, ali ujedno se družu i kroz druženje razmenjuju veštine i znanje, ne samo iz kompjuterskih oblasti. Razgovara se i o ostalim temama iz svakodnevnog života.

HKLBGD se takođe može smatrati alternativnom obrazovnom institucijom, mestom gde se mogu steći konkretna i korisna znanja, od kojih neka i nedostaju klasičnim obrazovnim programima u školama i na fakultetima.



Iako su ovakvi prostori inicijalno nastali u zapadnim zemljama, beogradski Haklab, kao i drugi hakerski prostor bilo gde u svetu, a ima ih svuda - nije kopija nekog drugog laba, nego se ova ideja uvek realizuje na svojevrsan način zbog suptilnih idejnih i kulturnih različitosti koje postoje u ovim zajednicama.

U Haklabu su više nego dobrodošli i ljudi koji se bave umetnošću, filozofijom ili bilo kojom naukom, ljudi zainteresovani za obostranu razmenu znanja i veština. Otuda i saradnja sa ITS-Z1 (<http://its-z1.org/>), umetničkom platformom Dragana Ilića, u čijem stvaralaštvu tehnologija igra bitnu ulogu.



Takođe, u ovakvim prostorima se održavaju i razni regionalni događaji i okupljanja ljudi, kako iz domaćih, tako i inostranih zajednica, pa je poznavanje engleskog jezika zgodno, ali nije obavezno.

Najznačajniji događaj regionalnog karaktera je NSND (Ništa Se Neće Dogoditi).

Puls slobode

<http://www.nsnd.org/o-nista-se-nece-dogoditi/>.

Održava se kriptoparti (<http://goo.gl/T13KRt>), koji traje, po pravilu, nekoliko časova. Ideja je da se korisnicima interneta i digitalnih tehnologija (uglavnom, ali ne i isključivo), aktivistima i novinarima, skrene pažnja na dostupnu tehnologiju putem koje mogu obezbediti privatnost svojih komunikacija u najširem smislu reči. Kao opšte mesto zastupa se mišljenje da je privatnost vrednost koja je na udaru od strane država i njihovih administracija, kao i od velikih korporacija.

U velikoj meri se zagovara i ohrabruje korišćenje Linuks operativnog sistema, programa otvorenog koda i uopšte slobodnog softvera, ali možete lako primetiti i korisnike drugih operativnih sistema, najpre Vindouza i Meka, koji su u manjem broju i uglavnom kod gostiju ovog prostora.

Iako članstvo formalno ne postoji, član može biti svako ko se tako oseća i deli ideje na kojima ovakav prostor počiva.

Prostor koji ova zajednica koristi, a koji se nalazi u Daničarevoj ulici na Vračaru, ustupila je na korišćenje članica ove zajednice Željka Dezire Milošević 2012. godine, od kada datira i naziv „Haklab Beograd”. Zanimljivo je da je organizacija postojala i ranije pod nazivom „Razmena veština”, a članovi su se okupljali u obližnjim kafićima ili kulturnim centrima. Tada je nastala i Gugl grupa ili jednostano mejl-lista pod istim nazivom koja se do danas nije promenila i služi kao oglasna tabla za nova obaveštenja kako članovima, tako i posetiocima. (<https://groups.google.com/forum/?hl=en-US#!forum/razmenavestina/join>)

Što se tiče radionica, one su raspoređene po danima, termini nisu fiksni, a satnice su samo okvirne.





Puls slobode

Međutim, tačno se zna kojim danima su određene radionice i to se skoro nikada ne menja.

Ponedjeljom su radionice Pajton programskog jezika koje su okrenute praktičnim rešenjima i implementacijama u trenutnim realnim potrebama i eksperimentima.

Utorkom je radionica za Drupal CMS-a, a postojala je veb-radionica koja se bavila poznatim programskim jezicima za veb (HTML, CSS, PHP i Javaskript), a koja je trenutno neaktivna.

Sreda je Otvoreni dan ili Opendej/razmena_veština, kao dan bez obaveza za odmor, ćaskanje, razmenu novih ideja i planova.

Četvrtak je rezervisan za radionicu troleologije (eng. *trolleyology*) koja se bavi fundamentalnim etičkim pitanjima u vezi sa vrednošću života, i koja, iako se čini, samo kao filozofija ili suva teorija zapravo ima praktične primene između ostalog u pametnim vozilima koja mogu nezavisno sama sobom upravljati. A odnedavno, četvrtak je dobio još jednu radionicu - francuski jezik, koja se, kada interesovanja ima, situacija omogući, a ljudi dogovore - održi i još nekim danom u nedelji.

Petkom je FemViki, radionica tokom koje članice Haklaba uređuju Vikipediju, pišu i prevode tekstove o značajnim ženama koje su nepravedno izostavljene iz ove svetske enciklopedije i riznice svetskog znanja.

Subota je već tradicionalno rezervisana za najpraktičniju radionicu, radionicu elektronike i Arduina, koja ide ukorak sa idejama i realizacijama interesantnih uređaja sa sličnim prostorima iz celog sveta.

Nedelja je tu za radionicu logike, kako klasične tako i neklasične, posle koje sledi radionica kriptografije, koja je u početku bila više teorijska i pratila kurs sa Kursere (eng. *Coursera*), dok je sada više usmerena praktičnim upotrebama kriptografije kao odgovoru na masovno narušavanje privatnosti i zaštitu korisnika.

Iako su neke od radionica trenutno u stanju hibernacije, to ne treba pogrešno shvatiti, jer se one uglavnom prilagođavaju potrebama i željama učesnika. Po potrebi nastaju nove radionice, a neke trenutno ili trajno stagniraju.



Više informacija

- [Tviter](#)
- [Sajt](#)
- [Open-Strit-Map](#)
- [IRC](#)

Predstavljamo



Fedora 22

vaš novi šešir!

Autor: Momčilo Medić





Fedora 21 je uspostavila nove smernice koje su se odlično pokazale i uzete su kao osnova za poboljšanje u Fedori 22. Ova verzija dolazi bez drastičnih promena kao što je to bio slučaj sa prethodnom i priprema put za Fedoru 23, koja bi mogla da unese zaista značajne novine. Fedora 22 nije imala uobičajeno kašnjenje zato što je za ovo izdanje određen striktan raspored kako bi se ostavilo dovoljno vremena da razvoj Fedore 23 bude temeljan, a sam sistem detaljno testiran.



Dugo očekivani **dnf** je konačno spreman i zamenio je **yum** kao podrazumevani upravnik paketima u Fedori. Sintaksa je prilagođena navikama i očekivanjima korisnika, tako da je u većini slučajeva dovoljno samo zameniti imena komandi (npr. **sudo yum install nano** postaje **sudo dnf install nano**). Naravno, razlike postoje i dobro je upoznati se sa njima zarad lakšeg usvajanja (detaljne informacije možete naći na: http://dnf.readthedocs.org/en/latest/cli_vs_yum.html).

Nastavljajući trend prethodne verzije, Fedora 22 donosi tri proizvoda - Radnu stanicu (eng. *Workstation*), Server i Oblak (eng. *Cloud*), kao i izvedbe (eng. *Spins*)

Predstavljamo

KDE, Iksfejs, ARM i druge.

Ovo su neke od izmena specifičnih za određeni proizvod:

Fedora Radna stanica



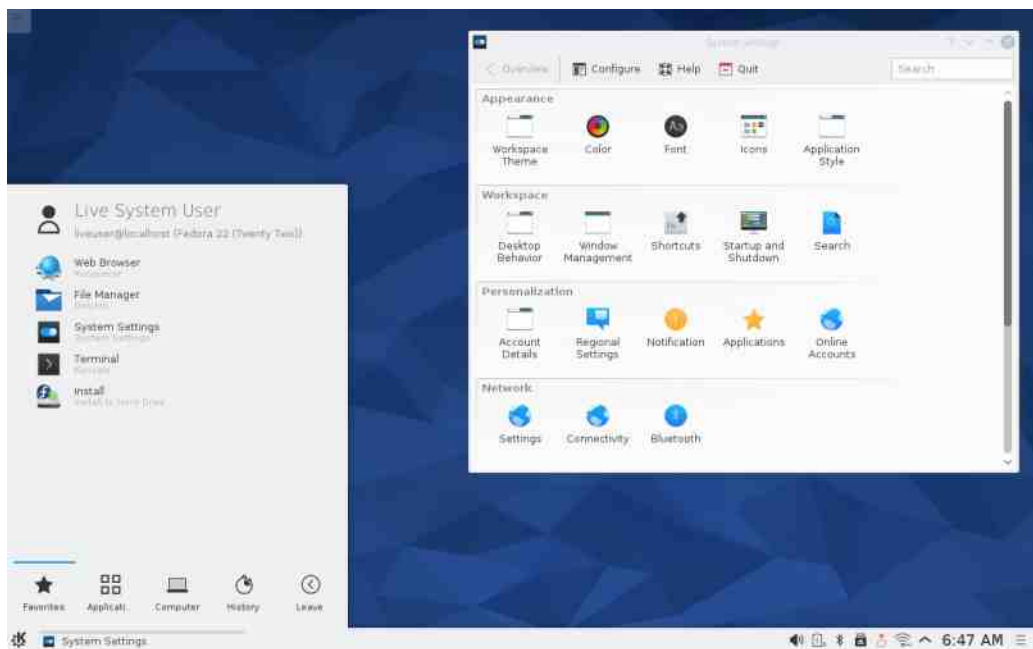
Pre nego što se potpuno prebaci na Vejland (eng. Wayland), Fedora je odlučila za početak da se samo ekran za prijavu iscrtava pomoću ovog novog grafičkog servera. Naravno, ovo je dobrodošla promena jer Vejland obećava povećanu sigurnost, stabilnost, kao i svež i jednostavniji kod. Međutim, problem će se pojaviti ako koristite vlasničke upravljačke programe za grafičku karticu. Naravno, ovo je predviđeno i u tim slučajevima će se ekran za prijavljivanje prikazivati upotrebom dobrog starog Iksa (eng. X). Pripremajući se za Vejland, **libinput** je zamenio **X.org input** u obradi ulaza sa tastature, miša, tačpeda i ostalog.



Svaka nova Fedora donosi i novi Gnom. Ovaj put u verziji 3.16, Gnom donosi mnoge izmene od kojih je svakako najveća premeštanje sistemskih poruka sa donjeg dela ekrana u gornji gde se do sada nalazio samo kalendar.

Obaveštenja se sada pojavljuju u vrhu ekrana, tik ispod časovnika, a po sakrivanju se mogu naći sa leve strane kalendara, poređane po redosledu pojavljivanja.

Vredno pomena je da i „traj“ (notifikaciona zona) više ne postoji, nego su ikonice programa koje zahtevaju ovakvo ponašanje smeštene u poseban panel koji se sam skriva, a nalazi se u donjem levom uglu ekrana. Ova drastična promena je uvedena zato što ne postoji objedinjeni standard za programe koji koriste ovakav vid funkcionalnosti i održavanje takvog sistema više nije u interesu Gnom razvojnog tima i planira se potpuno uklanjanje.



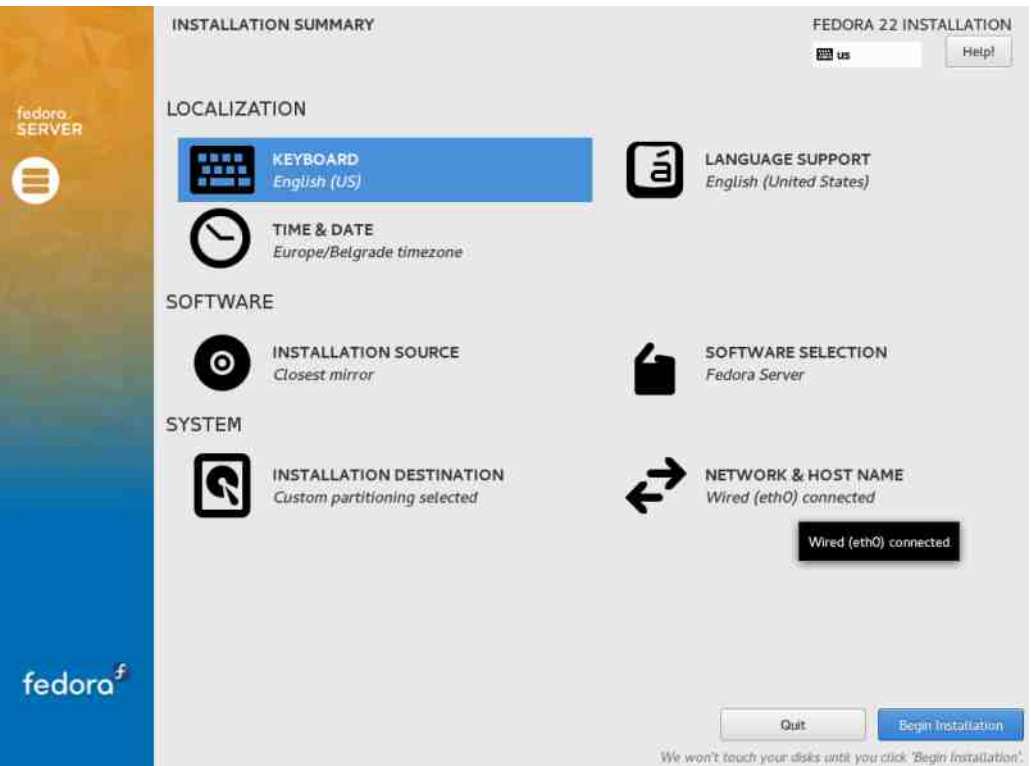
Između ostalih promena izdvajamo vizuelne promene na menadžeru datoteka

Predstavljamo

Nautilusu, pregledaču dokumenata Ivins (eng. *Evince*), i slika Oko Gnoma (eng. *Eye Of Gnome*), kao i nekoliko novih aplikacija: Kalendar (eng. *Calendar*), Knjige (eng. *Books*), IDE bilder (eng. *Builder IDE*), koji je sponzorisan od strane zajednice.

Preporučujemo da probate i Nantius (eng. *Nuntius*), program koji će prenositi vaše notifikacije sa Androida direktno u Gnom (potrebno je da imate Android verzije 4.3 ili novije).

Fedora Server



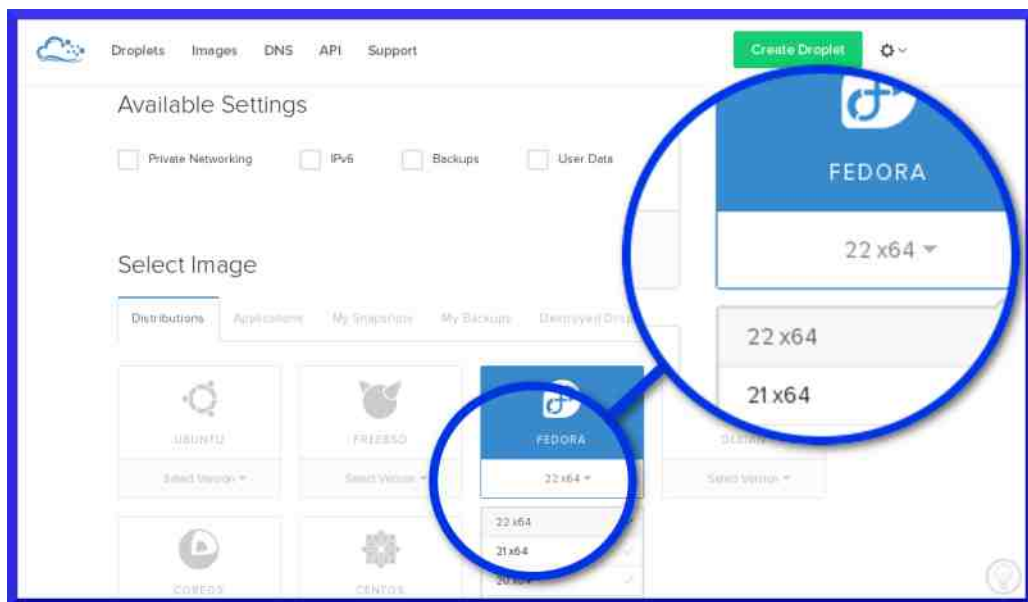
Uloge Servera su proširene i sada možete odrediti i ulogu baze podataka. U verziji



21, Fedora Server je uvela sistem „dodeljivanja uloga” svojim serverima i tada je bila dostupna uloga domen kontrolera upotrebom Fri-aj-pi-eija (eng. *FreeIPA*), a sada nova uloga koristi Postgreskuel (eng. *PostgreSQL*) da bi izvršila pripremu operativnog sistema za ulogu servera baza podataka.

Podrazumevani sistem datoteka je sada **XFS** na **LVM**-u za sve tačke montiranja, osim za **/boot**, koji zbog tehničkih ograničenja Graba (eng. *GRUB*) pokretačkog programa mora biti izvan **LVM**-a i sa **ext4** sistemom datoteka.

Fedora Oblak



Za korisnike Digitalnog okeana (eng. *DigitalOcean*) oblak provajdera, dostupna je slika za postavljanje Fedore 22 na instancama.

Pored raznih poboljšanja (osveženih Docker (eng. *Docker*) slika, unapređenog Atomik (eng. *Atomic*) domaćina...), najveća novina u „oblaku” je uvođenje

Predstavljamo

zvaničnih Vagrant slika (eng. *vagrant image*) kako bi programeri mogli lako uspostaviti razvojno okruženje pod Linux, Mek OS Iks ili Vindouz operativnim sistemima koristeći **libvirt** ili Virtualboks (eng. *VirtualBox*). Oni mogu izabrati Vagrant boks za Fedora 22 Atomik host ili osnovnu ediciju Fedora oblak 22.

Ako razmišljate o isprobavanju nove distribucije, ili tek pravite prve korake u svetu slobodnog softvera, Fedora 22 će zasigurno podržati vaše zahteve.

Da li je Fedora za vas?



Mnoštvo raznih distribucija GNU/Linux i odabir prave je česta tema među početnicima, pa čak i iskusnijim hakerima. Fedora Fondesh (temelji Fedore) su četiri osnovne vodilje koje služe u upravljanju odlukama i postavljanju ciljeva u okviru razvoja i održavanja ovog slobodnog operativnog sistema. Upoređivanjem ovih temelja sa vašim interesovanjima i prioritetima možete dobiti sliku o tome da li je Fedora pravi sistem za vas.

Sloboda - sav softver koji se nalazi u Fedori je slobodan i otvorenog koda. Strogim pravilima i revizijom se osigurava da programi nisu osakaćeni softverskim patentima i licencama koje ograničavaju slobodu korisnika. Izuzetak je jedino firmver u jezgri, koji je neophodan za određene uređaje. U današnje vreme mnogi žrtvuju slobodu zarad lagodnosti, ali i zbog pomoći korisnicima potpuno zatvorenih operativnih sistema da pređu na nešto slobodnije. Fedora je svesna toga i usko saraduje sa riznicama vlasničkih programa (upravljački progami za Invidija grafičke karte, razni kodeci, Stim (eng. *Steam*) i slično) kako bi instalacija i podešavanje istih ipak bilo moguće. Naravno, od presudnog značaja je da to bude dodatni korak kako bi ljudi bili svesni da programi u tim riznicama ne poštuju slobodu korisnika.



Prijatelji - ovo je možda neobičan temelj za operativni sistem, ali itekako važan. Fedora Projektom upravlja zajednica, a prave zajednice nema bez prijateljstva. Česta okupljanja na raznim mestima, što uživo - što preko interneta, omogućavaju da se svi umešani u razvoj, promociju i upotrebu Fedore osećaju dobrodošlim i među prijateljima. Pored formalnih, održavaju se i neformalna druženja sa namenom da se održi prijateljstvo u vidu žurki povodom izdanja i slično. Bliski odnosi među razvojnim timovima pružaju dodatni nivo kvaliteta, požrtvovanosti i odanosti jednih prema drugima, samim tim je krajnji proizvod kompletniji, uniformisan i pravo je zadovoljstvo učestvovati u njegovoj izgradnji.

Mogućnosti - moderan operativni sistem može ostati konkurentan i upotrebljiv samo ako se konstantno unapređuje i proširuje. Uzevši ovo kao temelj razvoja, jasno je da Fedora Projekat želi da bude u toku dešavanja, da bude aktuelan i adekvatan za svaku potrebu. Svaki slobodan operativni sistem vam pruža mogućnost proširenja i podešavanja do najsitnijih detalja, ali se Fedora posebno zauzela da taj proces ne bude mukotrpan, nego jednostavan i podržan tako da možete izvesti skoro sve što zamislite i opet zatražiti podršku od zajednice.

Biti prvi - slobodan softver je često mesto inovacija i fantastičnih novih tehnologija. Upravo usvajanje ovih tehnologija je ono što im daje održivost, kvalitet i unapređenja. Razvojni timovi u Fedori su veoma sposobni kada je reč o prepoznavanju vizije novih tehnologija kao i njihovom usvajanju. Ponekad je potrebno napraviti nepopularne izbore i promeniti ustaljeni pravac u korist napretka, a u mnogim slučajevima usvajanje tehnologije se pokazalo kao pravi korak, te su i ostale distribucije kasnije prihvatile istu promenu.

Ako se slažete sa četiri F (*Freedom, Friends, Features, First* - Sloboda, Prijatelji, Mogućnosti, Biti prvi) i mislite da je to veoma zdrava i dobro osmišljena osnova za inovaciju, razvoj i unapređenje operativnog sistema - onda je Fedora zaista skrojena po vašoj meri.

Kako da...?



VAGRANT (4. deo)

Autor: Ivan Radeljić

Vagrant konfiguraciona datoteka i početna tačka svakog Vagrant projekta je osnovna. Njena sintaksa je pisana u Rubi programskom jeziku i sada ćete videti koliko može biti moćna upotrebom promenljivih.

Podešavanja virtuelne mašine

oznaka: **config.vm**

Podešavanja sa ovom oznakom menjaju konfiguraciju mašine kojom Vagrant upravlja. Ovoj oznaci pripada velik broj promenljivih koje možemo koristiti. Ovde bismo naveli samo one promenljive koje smo koristili prilikom izrade Vagrant boksa za Liberator projekat. Pa da počnemo:

- `config.vm.box`

Ovim podešavamo koji boks će biti podignut na mašini. Vrednost ove promenljive mora biti ime instaliranog boksa ili kraće - ime boksa na [HashiCorp's Atlas](#) izvoru. U našem primeru za Liberator imamo slučaj:

```
config.vm.box = "debian"
```

- `config.vm.box_url`

Veb-adresa na kojoj se nalazi konfigurisan boks. Vrednost ove promenljive može biti bilo koja validna veb-adresa ili skraćeno ime boksa na [HashiCorp's Atlas](#)-u.



Takođe se može koristiti i niz adresa, kao i putanja do lokalne datoteke na disku. U našem primeru imamo da je vrednost ove promenljive veb-adresa.

```
if VAGRANT_ARCH == "x64"
    liberator.vm.box_url = "https://ftp.lugons.org/vagrant/
    debian-7.6.0-x86_64.box"
else
    liberator.vm.box_url = "https://dl.dropboxusercontent.com/
    s/60pv9an852jx9y0/Debian-7.3.0-i386b.box?dl=1"
end
```

Ovde imamo jedan složeniji slučaj gde na osnovu vrednosti konstante VAGRANT_ARCH određujemo koji ćemo boks koristiti. U zavisnosti od toga šta smo izabrali, to može biti boks za x86 ili x64 arhitekturu procesora.

- config.vm.network

Vrši podešavanje mreža na mašini. To može biti preusmeravanje portova, konektovanje na javnu mrežu ili kreiranje privatne mreže. Ova promenljiva može biti pozvana više puta i na taj način možemo imati više mreža koje mogu imati zasebna podešavanja. U našem primeru imamo slučaj:

```
liberator.vm.network :private_network, ip: "192.168.66.6"
```

Ovim smo definisali statičku IP adresu za našu mašinu. Bitno je da se IP adresa ne podudara sa IP adresom neke druge mašine u mreži i da bude iz reda rezervisanih privatnih adresa koje nisu javno dostupne.

- config.vm.boot_timeout

Vreme u sekundama koje će Vagrant čekati pre nego što mašina postane dostupna. Kod nas je to slučaj:

```
config.vm.boot_timeout = 600
```

- config.vm.provision

Ova postavka definiše snabdevače (eng. provisioners) na mašini koji će automatski instalirati i konfigurirati softver kada mašina bude kreirana. Naravno, u Liberatoru je za

Kako da...?

snabdevača korišćen Ansibl

```
liberator.vm.provision :ansible do |ansible|
  ansible.playbook = "provision/site.yml"
  ansible.host_key_checking = false
  ansible.groups = {
    "vagrant" => ["liberator"],
  }
end
```

Kao što smo već pomenuli, Ansibl koristi plejbuk datoteke za konfiguraciju Vagrant mašine. Šta je zapravo plejbuk? Plejbuk je JAML (YAML) dokument koji obuhvata neophodne korake koji će se preduzeti na Vagrant mašinama. Za više saznanja ispratite serijal tekstova o Ansiblu u našem časopisu.

Podešavanja za SSH

oznaka: **config.ssh**

Podešavanja koja nose ovu oznaku odnose se na konfigurisanje kako će Vagrant pristupati vašoj mašine preko SSH protokola. Ovde ćemo napomenuti neka od najvažnijih:

- config.ssh.username

Korisničko ime koje će Vagrant koristiti prilikom SSH pristupa. Podrazumevana vrednost je „vagrant“.

- config.ssh.password

Lozinka koju će Vagrant koristiti prilikom provere SSH korisnika. Treba napomenuti da Vagrant preporučuje da se koristi autentifikacija pomoću ključeva.

- config.ssh.host

IP adresa ili naziv računara (eng. hostname) na koji pristupamo preko SSH protokola.



- `config.ssh.port`

Port za SSH protokol. Podrazumevana vrednost je 22.

- `config.ssh.guest_port`

Port na gostujućoj mašini na kojoj SSH radi. Ovo je korisno za neke provajdere da detektuju preusmerene portove. Na primer, ako ova promenljiva ima podrazumevanu vrednost 22 i ako Vagrant detektuje da je port 4567 na domaćinu preusmeren na port 22 na gostujućoj mašini, Vagrant će pokušati da koristi port 4567 kako bi komunicirao sa gostujućom mašinom ako ne postoji druga opcija.

- `config.ssh.private_key_path`

Putanja do privatnog ključa koji se koristi za pristup gostujućoj mašini preko SSH protokola. Obično je ovo nesiguran ključ koji dolazi zajedno sa Vagrantom, pošto njega koriste svi javni boksovi. Ako napravite svoj boks sa svojim ključem, onda treba da podesite da vrednost ove promenljive pokazuje do tog ključa. Vrednost ove promenljiva može biti i niz koji će pokazivati na više ključeva.

- `config.ssh.insert_key`

Kao što smo već rekli, Vagrant automatski ubacuje nesigurni par ključeva. Ako koristite vaše privatne ključeve, onda ima smisla podesiti ovu vrednost kao "false".

Postoji još podešavanja vezanih za SSH. Ovde smo naveli neke od najbitnijih. Za ostale pogledajte zvaničnu dokumentaciju.

Podešavanja za Vagrant

oznaka: **`config.vagrant`**

Ova podešavanja se odnose na sam Vagrant.

- `config.vagrant.host`

Ova promenljiva podešava tip domaćina koji „vrti” Vagrant. Podrazumevana

Kako da...?

vrednost je :detect, što prouzrokuje da Vagrant automatski detektuje domaćina. Vagrantu je potrebno da zna ovu informaciju da bi mogao da izvršava neke zadatke koji su specifični za domaćina, kao što su pripremanje NFS direktorijuma ako su omogućeni. Ako Vagrant ne uspe da detektuje domaćina, onda je potrebno manuelno podesiti ovu vrednost.

Za kraj

Nadamo se da smo vam ovim serijalom približili Vagrant i način na koji funkcioniše. Nemojte da vas uplaše sva ova podešavanja i mnoštvo promenljivih kojima Vagrant raspolaže. To je samo dokaz da je Vagrant veoma moćan alat i da benefit od njega ima ceo razvojni tim. Naš savet je da krenete postepeno sa instalacijom javnog boksa i sa podrazumevanim podešavanjima. Kasnije po potrebi ili iz radoznalosti možete menjati vrednosti promenljivih i uvideti kako utiču na vaš projekat. Zbog izuzetne moći skaliranja Vagrant će sigurno naći primenu u svakom projektu.

Pregled popularnosti GNU/Linuks i BSD distribucija za mesec jun

Distrowatch

1	Mint	3063<
2	Debian	1611<
3	Ubuntu	1419>
4	Fedora	1290<
5	Mageia	1282>
6	openSUSE	1041<
7	Manjaro	901<
8	CentOS	881<
9	Arch	706<
10	LXLE	660<
11	Mangaka	657>
12	Peppermint	620>
13	elementary	599<
14	PCLinuxOS	591<
15	Android-x86	581<
16	Sabayon	576>
17	antiX	555>
18	Puppy	504<
19	Lubuntu	476>
20	Kali	471<
21	wattOS	450=
22	deepin	448<
23	Black Lab	428>
24	Zorin	426<
25	KaOS	419>

Pad <
 Porast >
 Isti rejting =
 (Korišćeni podaci sa Distrovoča)



Vaš posao, open-sors posao (4. deo)

Manifest GNU-a

Autor: Dejan Maglov

Objavljajući do sada nastavke ovog serijala, osetili smo da u priči nešto nedostaje. Priča o profitu koji je vezan za softver otvorenog koda nepotpuna je ako se ne vratimo korenima i još jednom ne razjasnimo osnovne pojmove.

Ovom prilikom koristimo tridesetogodišnjicu Manifesta GNU-a da još jednom skrenemo pažnju na taj vrlo važan dokument. Dugo smo razmišljali da li samo da postavimo link ka ovom dokumentu, ili da ga objavimo i mi u celosti i tako doprinesemo da ovaj dokument, koji je temelj postojanja i ovog časopisa, pročita što više ljudi. Podsetićemo starije korisnike slobodnog softvera da ovo nije prvi put da se objavljuje Manifest GNU-a u časopisu na srpskom jeziku. Ekipa Gnuzile je već objavila ovaj dokument u aprilu 2006. godine u svom specijalnom izdanju objavljenom povodom posete Ričarda Stolmana Beogradu (<https://dl.dropboxusercontent.com/u/844096/GNUzilla/GNUzillaRMS.pdf>).

Tridesetogodišnja vremenska distanca je vrlo malo nagrizla osnovne postulate ovog dokumenta što daje dodatnu vrednost istom. Zato vam toplo preporučujemo da ponovo pročitate ovaj dokument ako ste ga već ranije čitali, ili ga pročitate prvi put da bi vam bilo jasno zašto slobodni softver treba da postoji.

* * *

Slobodni profesionalac

Ričard Stolman

prevod na srpski jezik: Strahinja Radić

Manifest GNU-a

Donji manifest GNU-a je napisao Ričard Stolman na početku projekta GNU (1985. godina) da bi obezbedio učešće i podršku projektu. Prvih nekoliko godina Manifest je neznatno ažuriran kako bi izrazio razvoj projekta, ali od 1987. godine nije menjan jer ga je već videlo mnogo ljudi u takvom obliku.

Izmene na izvornom tekstu su izvršene prvenstveno jer su uobičajeni nesporazumi, koji su mogli biti izbegnuti drugačijim izborom reči. Fusnote koje su dodate 1993. godine pomažu da se određeni delovi razjasne.

Aktuelne informacije o dostupnom GNU-ovom softveru pogledajte na našem veb-serveru, <http://www.gnu.org>. Za informacije o načinu na koji nam možete pomoći, pogledajte <http://www.gnu.org/help/help.html>.

Šta je GNU? Gnu nije Juniks!

GNU, što znači Gnu nije Juniks, naziv je potpunog softverskog sistema kompatibilnog sa Juniksom, koji pišem da bih ga besplatno dao svima koji mogu da ga koriste.⁽¹⁾ Pomaže mi nekolicina drugih dobrovoljaca. Prilozi u vremenu, novcu, programima i opremi su nam veoma potrebni.

Za sada imamo uređivač teksta Emacs (Emacs) sa Lispom (Lisp) za pisanje naredbi uređivača, korektor izvornog koda, generator raščlanjivača saglasan sa Jakom (yacc), linker i oko trideset pet pomoćnih programa. Šel (interpretator naredbi) je skoro dovršen. Novi prenosivi optimizujući prevodilac za Ce (C) je preveo sam sebe i možda će biti objavljen ove godine (1985). Početno jezgro postoji, ali je potrebno još dosta

¹ Ovdje je izbor reči bio nepažljiv. Namera je bila da niko neće morati da plati za dozvolu da koristi sistem GNU-a. Međutim, reč „free” (slobodno, besplatno) ovo ne stavlja do znanja, i ljudi ih često interpretiraju kao da neće morati, ili će morati veoma malo da plate za primerke GNU-a. Ovo nikad nije bila željena zamisao. Kasnije, manifest pominje mogućnost postojanja firmi koje nude uslugu distribucije zarad profita. Kasnije sam naučio da pažljivo razlikujem „slobodno” (eng. free) u smislu slobode i „besplatno” (engl. free) u smislu cene. Slobodni softver je softver koji korisnici mogu da raspodeljuju i menjaju. Neki korisnici mogu da dobiju primerke besplatno, dok drugi plaćaju za dobijanje primeraka, i ako plaćanje pomaže unapređenju softvera, tim bolje. Bitno je da svako ko poseduje primerak ima slobodu da saraduje u njegovom korišćenju sa drugima.



odlika da bi se imitirao Juniks. Čim jezgro i prevodilac budu dovršeni, biće moguće raspodeljivati sistem GNU pogodan za razvoj programa. Koristićemo Teh (TeX) za formatiranje teksta, ali se radi i na enrofu (nroff). Koristićemo i slobodni, prenosivi prozorski sistem Iks (X). Posle ovoga dodaćemo prenosivi Opšti Lisp, igru Carstvo (Empire), program za tabelarno računanje (spreadsheet) i stotine drugih stvari, zajedno sa dokumentacijom. Nadamo se da ćemo na kraju obezbediti sve one korisne stvari koje se obično isporučuju sa sistemom Juniks, i više od toga.

GNU će moći da pokreće programe za Juniks, ali neće biti istovetan Juniksu. Ugrađićemo sva poboljšanja koja su se pokazala kao praktična na osnovu našeg iskustva sa drugim operativnim sistemima. Konkretno, planiramo da imamo duga imena datoteka (više od osam karaktera), brojeve verzija datoteka, neprobojan sistem datoteka, možda dopunu naziva datoteka, podršku za prikaz nezavisan od terminala i možda prozorski sistem zasnovan na Lispu kroz koji nekolicina programa u Lispu i obični programi za Juniks mogu da dele ekran. I Ce i Lisp će biti dostupni kao sistemski programski jezici. Pokušaćemo da podržimo UUCP (UUCP), MIT-ov Haosnet (Chaosnet) i Internetske protokole za komuniciranje.

GNU je prevashodno namenjen mašinama sa virtualnom memorijom u klasi 68000/16000 jer su one najlakše za prilagođavanje. Dodatni trud da se on prilagodi manjim mašinama je ostavljen nekome ko želi da ga koristi na njima.

Kako bismo izbegli strašnu zabunu, molim vas da slovo „G” u reči „GNU” izgovarate kao „G” (ne kao „dži”), kada se ta reč odnosi na naš projekat.[1]

Zašto moram da napravim GNU?

Smatram da je zlatno pravilo da, ukoliko mi se program sviđa, moram da ga podelim i sa drugima kojima se sviđa. Prodavci softvera žele da podele korisnike i onda da vladaju njima primoravanjem svakog korisnika da se složi da ga ne deli sa drugima. Ja odbijam da ovako onemogućim solidarnost sa drugim korisnicima. Ne mogu čiste svesti da potpišem ugovor o neobjavlivanju ili ugovor o softverskoj licenci. Godinama sam radio u Laboratoriji za veštačku inteligenciju (eng. *Artificial Intelligence Lab* - AIL) kako bih se odupirao takvim tendencijama i drugim negostoljubivostima, ali na kraju su preterali: nisam mogao da ostanem u instituciji gde mi se tako nešto nameće.

Kako bih mogao da nastavim da koristim računare čistog obraza, odlučio sam da sastavim dovoljan korpus slobodnog softvera kako bih bio u mogućnosti da ne koristim uopšte softver koji nije slobodan. Napustio sam AIL kako bih uskratio MIT-u svaki pravni izgovor da me spreče da distribuiram GNU.

¹ Ova napomena se odnosi na englesku verziju. [SR]

Slobodni profesionalac

Zašto će GNU biti kompatibilan sa Juniksom?

Juniks nije za mene idealan sistem, ali nije ni tako loš. Osnovne osobine Juniksa se čine dobrim i mislim da bih mogao da popunim nedostatke u Juniksu bez njihovog kvarenja. Osim toga, sistem kompatibilan sa Juniksom bi mnogi prihvatili.

Kako će GNU biti dostupan?

GNU nije u javnom vlasništvu. Svakom će biti dozvoljeno da menja i distribuira GNU, ali nijednom distributeru neće biti dozvoljeno da spreči dalju distribuciju. Takoreći, vlasničke izmene neće biti dozvoljene. Hoću da osiguram da sve verzije GNU-a ostanu slobodne.

Zašto mnogi programeri žele da pomognu?

Našao sam mnogo drugih programera koji su zagrejani za GNU i žele da pomognu.

Mnogo programera je nesrećno zbog komercijalizacije sistemskog softvera. Ona im možda omogućava da više zarađuju, ali i zahteva da se osećaju kao da su u sukobu sa drugim programerima umesto da ih doživljavaju kao prijatelje. Osnovni čin prijateljstva među programerima je deljenje programa. Današnji uobičajeni marketinški ugovori iz korena zabranjuju programerima da se odnose jedni prema drugima kao prema prijateljima. Kupac softvera mora da izabere između prijateljstva i poštovanja zakona. Prirodno, mnogi odlučuju da je prijateljstvo važnije, ali oni koji veruju u zakon se često ne osećaju lagodno ni sa jednim izborom. Oni postaju cinični i počinju da misle da je programiranje samo način za sticanje novca.

Radeći na GNU-u i koristeći ga, a ne vlasničke programe, možemo biti gostoljubivi prema svima i poštovati zakon. Uz sve to, GNU služi kao primer za ugled i barjak koji okuplja druge da nam se pridruže u deljenju. Ovo nam može pružiti osećaj harmonije koji je nemoguć ako koristimo neslobodni softver. Za polovinu programera sa kojima razgovaram, ovo je važna vrsta sreće koju novac ne može da zameni.

Kako možete pomoći?

U današnje vreme, da biste videli spisak zadataka na kojima možete raditi, pogledajte <http://fsf.org/campaigns/priority-projects> i http://savannah.gnu.org/people/?type_id=1. Za druge načine na koje možete pomoći, pogledajte <http://www.gnu.org/help/help.html>.

Molim proizvođače računara za priloge u mašinama i novcu. Molim pojedince za priloge u programima i radu.

Ukoliko priložite računare, možete očekivati da će GNU na njima pre proraditi. Mašine bi trebalo da budu potpuni sistemi, spremni za upotrebu, sa dozvolom za kućnu upotrebu i bez potrebe za sofisticiranim hlađenjem ili napajanjem.



Našao sam veoma mnogo programera voljnih da povremeno rade na GNU-u. Za većinu projekata takav povremeni razdeljeni rad bi bio veoma težak za upravljanje. Nezavisno napisani delovi ne bi mogli da međusobno funkcionišu. Međutim, kod konkretnog posla zamene Juniksa, ovaj problem ne postoji. Potpun sistem Juniksa sadrži stotine pomoćnih programa, od kojih je svaki pisan odvojeno. Većina propisa o sprezi je ograničena kompatibilnošću sa Junixsom. Ako svaki donator napiše kompatibilnu zamenu za jedan Junixsov pomoćni program i učini da ona na pravi način zamenjuje original na sistemu Juniksa, onda će ovi pomoćni programi funkcionisati na pravi način kada se sastave. Čak i kada bi Marfi stvorio nekoliko neočekivanih problema, spajanje ovih delova će biti izvodljiv zadatak. (Jezgro će zahtevati bližu saradnju i njega će razvijati manja, kompaktnija grupa.)

Ukoliko dobijem priloge u novcu, biću u mogućnosti da zaposlim nekoliko ljudi na stalnom ili povremenom radu. Plata neće biti visoka po programerskim standardima, ali ja tražim ljude za koje je izgradnja duha zajedništva isto toliko važna kao i zarađivanje. Ovo vidim kao način da omogućim posvećenim ljudima da svoju punu pažnju posvete radu na GNU-u i tako ih poštedim potrebe da zarađuju za život na drugi način.

Zašto će svi korisnici računara imati koristi?

Kada se jednom napravi GNU, svako će moći da koristi dobar sistemski softver slobodno, baš kao što diše vazduh.⁽²⁾

Ovo znači mnogo više od izbegavanja cene za Junixsovu licencu. Ovo znači da će se izbeći mnogo nepotrebnog udvostručavanja truda u sistemskom programiranju. Ovaj trud može da se umesto toga usmeri na unapređivanje mogućnosti softvera.

Potpuni izvorni kod sistema će biti dostupan svima. Kao posledica ovoga, korisnik kome su potrebne promene u sistemu će uvek imati slobodu da ih sam načini, ili da zaposli programera ili firmu da ih načine umesto njega. Korisnici više neće biti prepušteni na milost i nemilost jednom programeru ili firmi koja poseduje izvorni kod i koja je jedina u mogućnosti da menja softver.

Škole će moći da obezbede bolje obrazovno okruženje koje će podsticati sve učenike da proučavaju i poboljšavaju sistemski kod. U računarskoj laboratoriji na Harvardu je važno pravilo da se nijedan program ne može instalirati na sistem ako mu izvorni kod nije dostupan javnosti i oni su se pridržavali tog pravila, odbijajući da instaliraju određene programe. To me je veoma inspirisalo.

² Ovo je drugo mesto na kome nisam pažljivo napravio razliku između dva različita značenja reči „free“ (engl. free = „slobodno“ ili „besplatno“). Rečenica takva kakva je nije pogrešna — možete da nabavite primerke GNU-ovog softvera besplatno, od vaših prijatelja ili preko mreže. Ali ona navodi na pogrešan zaključak.

Slobodni profesionalac

Na kraju, prestaće suvišna razmatranja o tome ko poseduje sistemski softver, i šta sa njim sme a šta ne sme da se radi.

Ugovori koji primoravaju ljude da plate za korišćenje programa, uključujući licenciranje primeraka, uvek nose ogromnu cenu po društvo kroz nezgrapne mehanizme koji su potrebni da se ustanovi koliko (to jest, koje programe) neko mora da plati. Samo policijska država može da primora svakog da se pokori. Uzmimo za primer svemirsku stanicu gde se vazduh mora proizvoditi po visokoj ceni: naplaćujući svakom ko diše po litru vazduha može da bude pošteno, ali nošenje gas maske sa meračem po ceo dan i noć se ne može tolerisati čak iako svi mogu da priušte da plate račun za vazduh. I sveprisutne TV kamere koje nadgledaju da možda ne skidate maske su sramota. Bolje je podržati fabriku vazduha porezom po glavi stanovnika i otarasiti se maski.

Kopiranje celog ili delova programa je za programera isto toliko prirodno kao i disanje i isto toliko produktivno. Moralo bi da bude isto toliko slobodno.

Neki prigovori ciljevima GNU-a koji se sa lakoćom opovrgavaju

„Niko ga neće koristiti ako je besplatan, jer to znači da se ne može pouzdati u podršku.”

„Mora se platiti za program da bi se platilo obezbeđivanje podrške.”

Ako bi ljudi radije platili za GNU i podršku nego nabavili GNU besplatno bez podrške, firma koja obezbeđuje samo podršku ljudima koji su nabavili GNU besplatno bi trebalo da bude profitabilna.⁽³⁾

Moramo da napravimo razliku između podrške u obliku stvarnog programiranja i običnog održavanja. Ovo prvo je nešto na šta se ne može osloniti kod prodavaca softvera. Ako vaš problem ne muči dovoljno ljudi, prodavac će vam reći da se gubite.

Ako vaše poslovanje mora da se osloni na podršku, jedini način je da posedujete sav potreban izvorni kod i alate. Tada možete zaposliti bilo koju osobu da vam reši problem. Vi niste prepušteni na milost i nemilost jednoj osobi. Sa Junksom, zbog cene izvornog koda, većina firmi o ovome ne vredni ni da razmišlja. Sa GNU-om, ovo će biti jednostavno. Još uvek će biti moguće da ne postoji kompetentna osoba, ali za ovaj problem nije kriv način raspodele. GNU ne rešava sve svetske probleme, već samo neke od njih.

U međuvremenu, korisnicima koji ne znaju ništa o računarima je potrebno održavanje: obavljanje poslova koje bi oni lako uradili, samo što ne znaju kako.

Takve usluge bi mogle da obezbede firme koje prodaju samo usluge održavanja i

³ Danas postoji nekoliko takvih firmi.



popravke. Ukoliko je tačno da će korisnici radije utrošiti novac za proizvod sa podrškom, onda će oni isto toliko biti voljni da kupe podršku a da proizvod dobiju besplatno. Firme za podršku će se nadmetati u kvalitetu i ceni. Korisnici neće biti vezani ni za jednu određenu firmu. Za to vreme, oni oni kojima podrška nije potrebna će moći da koriste program bez potrebe da plaćaju podršku.

„Ne možete da zainteresujete mnogo ljudi bez reklame, a da biste to omogućili, morate da naplaćujete za program.”

„Ne vredni reklamirati program koji ljudi mogu da dobiju besplatno.”

Postoje razni oblici besplatnog ili veoma jeftinog publiciteta koji mogu da se iskoriste za informisanje velikog broja korisnika računara o nečemu kao što je GNU. Verovatno se može dopreti do većeg broja korisnika računara reklamom. Ako je ovo stvarno tako, posao koji reklamira uslugu umnožavanja i slanja GNU-a poštom za novac bi trebalo da bude dovoljno uspešan da isplati reklamiranje i stvori zaradu. Na taj način će ga plaćati samo korisnici koji imaju koristi od reklamiranja.

Sa druge strane, ukoliko mnogo ljudi dobije GNU od svojih prijatelja, i takve firme ne uspeju, ovo će pokazati da reklama nije bila stvarno potrebna za širenje GNU-a. Zašto zagovornici slobodnog tržišta ne puste slobodno tržište da o ovome samo odluči?(4)

„Mojoj firmi je za ostvarivanje prednosti nad konkurencijom potreban vlasnički operativni sistem.”

GNU će izbaciti softver operativnih sistema iz sveta konkurencije. Nećete biti u mogućnosti da ostvarite prednost u ovoj oblasti, ali ni vaši konkurenti neće moći da ostvare prednost nad vama. Vi i oni ćete se takmičiti u drugim oblastima, zajednički napredujući u ovoj. Ako je vaša delatnost prodaja operativnog sistema, GNU vam se neće sviđati, ali to je vaš problem. Ako je vaša delatnost nešto drugo, GNU vas može spasti od uvlačenja u skupi posao prodaje operativnih sistema.

Voleo bih da vidim kako se izgradnja GNU-a podržava poklonima velikog broja proizvođača i korisnika, smanjujući troškove za sve.(5)

„Zar programeri ne zaslužuju nagradu za kreativnost?”

Ukoliko išta zaslužuje nagradu, onda je to doprinos društvu. Kreativnost može da predstavlja doprinos društvu, ali samo ako je društvu dopušteno da koristi njene rezultate. Ako programeri zaslužuju nagradu zbog stvaranja inovativnih programa, po nekoj analogiji zaslužuju i da budu kažnjeni ako sprečavaju korišćenje tih programa.

⁴ Zadužbina za slobodni softver se deset godina uglavnom finansira iz službe za distribuciju, iako je ona fondacija, a ne firma. Možete i da naručujete stvari od ZSS-a.

⁵ Nekoliko računarskih firmi je 1991. sponzorisalo održavanje GNU-ovog prevodioca za Ce.

Slobodni profesionalac

„Zar programeru ne bi trebalo dozvoliti da traži nagradu za svoju kreativnost?“

Nema ništa loše u želji da se rad isplati, ili traženja da se uveća nečiji prihod - sve dok se ne koriste destruktivna sredstva. Ali sredstva koja su danas uobičajena na polju softvera su zasnovana na destrukciji.

Izvlačenje novca od korisnika programa sprečavajući ih da ga koriste je destruktivno, jer ograničenja smanjuju dužinu korišćenja i broj načina na koji program može da se koristi. Ovim se smanjuje količina dobra koje čovečanstvo dobija od programa. Kada je namerno odlučeno da se ograniči, štetne posledice te odluke predstavljaju namernu destrukciju.

Razlog zbog kojeg dobar građanin ne koristi takva destruktivna sredstva da bi postao bogatiji je, što ako bi svi tako činili, svi bismo postali siromašniji usled međusobne destruktivnosti. Ovo je kantovska etika ili Zlatno pravilo. Kako mi se ne sviđaju posledice koje bi usledile ukoliko bi svako nagomilavao informacije, moram da to shvatim kao pogrešno. Izuzetno, želja da se bude nagrađen za kreativnost ne opravdava lišavanje sveta u celini te kreativnosti ili njenog dela.

„Neće li programeri umreti od gladi?“

Mogao bih da odgovorim da niko nije primoran da bude programer. Većina nas ne može da zaradi novac za stajanje na ulici i kreveljenje. Ali, zauzvrat, mi nismo osuđeni da provedemo živote u stajanju na ulici i kreveljenju, i gladovanju. Jednostavno, radimo nešto drugo.

Ali to je pogrešan odgovor jer prihvata skrivenu pretpostavku onoga koji je postavio to pitanje da programerima neće biti isplaćen ni dinar bez vlasništva nad softverom. Sve ili ništa.

Međutim, pravi razlog zašto programeri neće umreti od gladi je što će još uvek biti moguće da budu plaćeni za programiranje, samo ne toliko kao sada.

Sprečavanje kopiranja nije jedina osnova za poslovanje zasnovano na softveru. To je najčešća osnova jer donosi najviše novca. Kad bi bilo zabranjeno ili odbijeno od strane kupaca, softverski posao bi prešao na druge načine organizacije koji se sada ređe koriste. Uvek postoje razni načini da se organizuje bilo koji posao.

Verovatno programiranje neće biti toliko unosno na novim osnovama kao što je sada, ali to nije argument protiv promene. Ne smatra se nepravdom visina sadašnjih plata prodavaca. Kada bi programeri isto toliko zarađivali, i to bi bila nepravda. (U praksi bi oni i dalje značajno više zarađivali.)

„Zar ljudi nemaju pravo da kontrolišu kako im se koristi kreativnost?“



„Kontrola nad korišćenjem ideja” zapravo predstavlja kontrolu nad životima drugih ljudi, i obično se koristi da im zagorča život.

Ljudi koji su pažljivo proučavali pitanje prava intelektualne svojine⁽⁶⁾ (naprimer, pravnici) kažu da ne postoje suštinska prava na intelektualnu svojinu. Vrste takozvanih prava na intelektualnu svojinu koje propisuje vlada su stvorene posebnim pravnim aktima u konkretne svrhe.

Naprimer, sistem patenata je uspostavljen kako bi podstakao pronalazače da otkriju detalje svojih izuma. Njegov cilj je da pomogne društvu, a ne pronalazačima. Tada je životni vek patenta od 17 godina bio kratak kada se uporedi sa brzinom napretka tehnologije. Kako su patenti od značaja samo za proizvođače, za koje su cena i namera ugovora o licenciranju mali kada se uporede sa uspostavljanjem proizvodnje, patenti često ne čine mnogo štete. Oni ne smetaju većini pojedinaca koji koriste patentirane proizvode.

Ideja o autorskim pravima nije postojala u stara vremena, kada su tvorci često kopirali druge u neintelektualnim delima. Ova praksa je bila korisna, i bila je jedini način da njihova dela prežive, pa makar i delimično. Sistem autorskih prava je stvoren isključivo za svrhu podsticanja stvaranja. U domenu za koji je zamišljen — knjige, koje su ekonomično mogle da se kopiraju samo preko štamparske prese — on je činio malo štete, i nije predstavljao prepreku većini pojedinaca koji čitaju knjige.

Sva prava intelektualne svojine su samo licence koje daruje društvo, jer se mislilo, ispravno ili pogrešno, da će društvo kao celina od njih imati koristi. Ali u konkretnoj situaciji moramo da se zapitamo: da li je za nas bolje da dozvolimo takvo licenciranje? Za šta ovlašćujemo određenu osobu?

Današnji programi su veoma različiti od knjiga pre stotinu godina. Činjenica da je najlakši način da se program umnoži umnožavanje programa između bližnjih, činjenica da program poseduje i izvorni kod i objektni kod koji se razlikuju, i činjenica da se program koristi, a ne čita i ne uživa, se kombinuju, stvarajući situaciju u kojoj osoba koja nameće autorska prava šteti društvu kao celini i materijalno i duhovno, a to ne bi smela da radi bez obzira na to da li joj zakon to dozvoljava ili ne.

⁶ Osamdesetih još uvek nisam shvatao koliko je zbunjujuće govoriti o „pitanju” „intelektualne svojine”. Taj izraz je očito nastao iz predrasude. Činjenica da on izjednačava mnoge različite zakone, koji pokreću sasvim različita pitanja, je tananija. Sada svima tražim da u potpunosti odbace izraz „intelektualna svojina”, kako ne bi nekoga naveli da zaključie da ovi zakoni čine jedno celovito pitanje. Način na koji se može biti jasniji je da se o patentima, autorskim pravima i robnim markama raspravlja odvojeno. Pogledajte dalje objašnjenje načina na koji ovaj izraz širi konfuziju i dvosmislenost.

Slobodni profesionalac

„Uz konkurenciju se bolje radi.“

Paradigma konkurencije je trka: nagrađivanjem pobjednika, podstičemo svakoga da trči brže. Kad kapitalizam stvarno ovako funkcioniše, to je dobro, ali njegovi branioci greše u pretpostavci da on uvek ovako funkcioniše. Ako trkači zaborave za šta se nudi nagrada i namere da pobjede po svaku cenu, oni mogu da razviju drugačije strategije, kao što je napadanje drugih trkača. Ako se trkači potuku, svi će zakasnuti.

Vlasnički i softver zatvorenog koda su moralni ekvivalenti trkača koji se tuku. Tužno je što je to tako, ali jedini sudija koji postoji se ne protivi tuči — on je samo propisuje („Na svakih deset metara protivnika možete da udarite jednom“). A trebalo bi da ih razdvoji i kazni za sam pokušaj tuče.

„Neće li svi prestati da programiraju bez novčanog podsticaja?“

Ustvari, mnogi ljudi će programirati bez apsolutno ikakvog novčanog podsticaja. Programiranje nekima predstavlja neodoljivu fascinaciju, i to obično onima koji su u njemu najbolji. Ne manjka profesionalnih muzičara koji se bave muzikom, iako ne postoji nada da tako zarade za život.

Ali, ustvari, ovo pitanje, iako često postavljano, nije primereno situaciji. Plaćanje programerima neće nestati, već će se samo smanjiti. Dakle, pravo pitanje je da li će iko programirati sa smanjenim novčanim podsticajem? Moje iskustvo pokazuje da hoće.

Preko deset godina mnogi od najboljih programera u svetu su radili u Laboratoriji za veštačku inteligenciju, za mnogo manje novca nego što su mogli da zarade bilo gde drugde. Ali su zaradili mnogo vrsta nenovčanih nagrada: naprimer, slavu i zahvalnost. Osim toga, kreativnost je i zabavna, što je samo po sebi nagrada.

Ali, kada im je ponuđena prilika da obavljaju isti interesantan posao za mnogo novca, većina njih je napustila posao.

Ono što pokazuje činjenice je da će ljudi programirati i iz drugih razloga osim bogatstva - ali ako im se omogući i da puno zarade, to će očekivati i zahtevati. Organizacije sa niskim zaradama lošije prolaze od onih sa visokim zaradama, ali se to ne bi dešavalo ukoliko bi one sa većim zaradama bile zabranjene.

„Očajnički su nam potrebni programeri. Ukoliko oni zahtevaju da prestanemo da pomažemo bližnjima, moramo da ih poslušamo.“

Nikada nećete biti toliko očajni da poslušate ovakvu vrstu zahteva. Setite se: „Milione za odbranu, ali ni dinara za harač!“^[2]

² Poznata izjava Tomasa Džefersona. [SR]



„Programeri moraju nekako da zarađuju za život.“

Gledano na kratke staze, ovo je tačno. Međutim, postoje brojni načini na koje programeri mogu da zarade za život bez prodavanja prava na korišćenje programa. Taj način je sada uobičajen jer programerima i biznismenima donosi najviše novca, a ne zato što je jedini način da se zarađuje za život. Lako je naći druge načine ukoliko želite da ih nađete. Evo nekoliko primera.

Proizvođač koji napravi novu vrstu računara će platiti za prilagođavanje operativnog sistema novom hardveru.

Prodaja usluga obuke i održavanja može takođe da uposli programere.

Ljudi sa novim idejama bi mogli da raspodeljuju programe kao friver(7), tražeći priloge od zadovoljnih korisnika, ili prodajući usluge održavanja. Sretao sam ljude koji već uspešno rade na takav način.

Korisnici sa sličnim potrebama mogu da stvore korisničke grupe i plaćaju članarinu. Grupa bi sklopila ugovor sa programerskim firmama da pišu programe koje bi članovi grupe želeli da koriste.

Sve vrste razvoja softvera mogu se finansirati softverskim porezom

Pretpostavimo da je svako ko kupi računar obavezan da plati x procenata cene za račun softverskog poreza. Vlada bi ovo davala agenciji kao što je NZS (Nacionalna zadužbina za softver) za pokrivanje troškova razvoja softvera.

Ali, ako kupac računara sam priloži za izgradnju softvera, bio bi izuzet iz plaćanja poreza. Mogao bi da uloži u projekat po svom izboru — često birajući ga jer se nada da će koristiti njegov rezultat kada on bude završen. Može da bude zaslužan za bilo koji iznos priloga do punog iznosa poreza koji je morao da plati.

Ukupni porez bi mogao da bude izabran glasanjem oporezovanih, i izvagan u skladu sa iznosom koji će im biti naplaćen.

Posledice bi bile sledeće:

- Zajednica korisnika računara bi podržavala izgradnju softvera.
- Ova zajednica bi odlučivala o tome koji nivo podrške joj je potreban.
- Korisnici koji posvećuju posebnu pažnju tome na koje projekte je potrošen njihov deo mogu ovo sami da izaberu.

⁷ Naknadno smo naučili da razlikujemo „slobodan softver“ i „friver“. Izraz „friver“ označava softver koji možete da ponovo distribuirate, ali obično vam nije dozvoljeno da ga proučavate i menjate njegov izvorni kod, pa većina frivera nije slobodan softver.

Slobodni profesionalac

Dugoročno gledano, oslobađanje programa je korak napred prema postrestriktivnom svetu, u kome niko neće morati da naporno radi samo da bi zaradio za život. Ljudi će moći da se posvete aktivnostima koje su zabavne, kao što je programiranje, posle provođenja neophodnih deset sati nedeljno na neophodnim poslovima kao što su zakonodavstvo, porodično savetovanje, opravka robota i istraživanje asteroida. Neće biti potrebe da se zarađuje za život od programiranja.

Mi smo već sada veoma smanjili obim posla koji celo društvo mora da obavi za celokupnu produktivnost, ali samo mali deo ovoga se pretvorio u slobodno vreme za radnike jer je neophodno mnogo neproduktivnih uz produktivne aktivnosti. Glavni uzroci ovoga su birokratija i izometrične borbe protiv konkurencije. Slobodni softver će veoma smanjiti ove otežavajuće okolnosti u proizvodnji softvera. Mi to moramo da uradimo, kako bi se tehnički dobiti u produktivnosti pretvorili u smanjenje posla koji moramo da obavimo.

* * *

Manifest GNU-a je uspešno započeo projekat koji i danas živi. Jedino što možemo da primetimo da nije zaživeo čisti GNU OS nego GNU/Linux što ni malo ne umanjuje značaj ovog projekta i njegove prvobitne ciljeve. Naprotiv, ovakav ishod samo potvrđuje da je Manifest uspeo u nameri da obezbedi podršku i učešće programera iz celog sveta.

Pre trideset godina smo imali samo vlasnički komercijalni i besplatni vlasnički softver takozvani friver (eng. *freeware*). Od Manifesta se razvija i slobodni softver koji u sebi objedinjuje sledeće osobine:

1. to je softver otvorenog koda,
2. objavljen je pod licencom koja dozvoljava slobodno korišćenje u bilo koju svrhu, proučavanje koda, izmenu koda i redistribuciju koda,
3. licenca slobodnog softvera zabranjuje da izmenjeni slobodni softver može da postane vlasnički softver (zatvorenog koda).

Manifest samo ovakav softver smatra potpuno slobodnim.

Od objave manifesta najveća zabuna u vezi s ovim dokumentom leži u činjenici da se često, slučajno ili namerno, pogrešno tumači reč sloboda (eng. *free*). Istina engleska reč za slobodu je višeznačna i može se tumačiti i kao besplatno. Autor manifesta u više navrata naglašava da je ovde reč o slobodi a ne isključivo o



nečemu što je besplatno.

Ako se ovakva sloboda dà softveru, on praktično postaje besplatan, naročito u internetskoj eri, ili može biti samo jednokratno prodat. Manifest zato propisuje šta je moralno naplatiti u slobodnom softveru da bi on ostao potpuno slobodan. Mogu se naplatiti sve sekundarne aktivnosti oko slobodnog softvera kao što su distribuiranje slobodnog softvera (Fondacija za slobodan softver), podrška i održavanje (Red Hat, SUSE Enterprajz), edukacije, školovanje i sertifikacija (Red Hat), proizvodi koji nastaju kao plod rada na slobodnom softveru (Piksel), razvoj specifičnih softvera po narudžbi za slobodni softver i reklamiranje (Gugl).

Slobodan softver očigledno sužava prostor mogućnost zarade na softveru. Ovo je osnovni razlog zašto se nije ostvarilo predviđanje iz Manifesta da će slobodni softver, nakon što bude razvijen, „ukinuti” softver zatvorenog koda. Interes da vlasnički softver opstane imaju kompanije i većina programera koji ne žele da se odreknu visokih primanja. Zasada nije promenjeno zakonodavstvo koje bi ukinulo ili ograničilo kopirajt (isključivo pravo na kopiranje softvera); i dok sami korisnici ne budu odbili da kupuju vlasnički softver, paralelno će egzistirati slobodni i vlasnički softver.

Između ove dve sasvim suprotstavljene struje (vlasnički i slobodni softver) nastala je i pomirljiva struja softvera otvorenog koda. Softver otvorenog koda (eng. *open-source software* - OSS) treba da pomiri dve suprotstavljene struje i ima ideju da je koegzistencija moguća. OSS jeste otvorenog koda, ali ne mora biti potpuno slobodan softver jer svojom licencom ograničava prava korisnika na korišćenje u bilo koju svrhu, redistribuciju, izmenu koda i slična ograničenja. OSS može da bude komercijalni softver, ali i ne mora. Kada želimo da objedinimo slobodni softver i softver otvorenog koda, koristimo skraćenicu FLOSS.

Činjenica je da skoro svaka komercijalizacija slobodnog softvera dovodi do određenih ograničenja i gubitka dela slobode softvera. Zbog toga u ovom serijalu pričamo upravo o OSS-u i samo eventualno pominjemo ograničenja koja komercijalizacija nameće, čime takav softver prestaje da bude potpuno slobodan. Pošto smo još jednom razjasnili osnovne pojmove, možemo od narednog broja da nastavimo da istražujemo OSS modele poslovanja.

Slobodni profesionalac



3. deo

Autor: Stefan Nožinić

U prethodnom delu ste mogli pročitati kako se podešava inventar i tu ste videli kako možemo održavati više servera te njima proslediti određene parametre i klasifikovati ih u grupe na osnovu njihove namene. Sada je vreme da pređemo na pravu stvar i da pokažemo šta još Ansibl može da radi. U ovom tekstu dajemo realne primere koje ćemo objasniti i koji se često koriste. Na kraju dajemo i neke opšte korisne savete iz prakse kojih se treba pridržavati kako bi vam kod bio uredan i kako biste mogli lako održavati svoju konfiguraciju.

Ako pratite naš serijal o Vagrantu, onda sigurno znate da Ansibl možete primeniti i kroz Vagrant prilikom kreiranja svoje virtuelne mašine.

Najpre počinjemo od osnovnih pojmova, a potom prelazimo na kompleksnije stvari i savete.

Plejbukovi

Moglo bi se reći da tek ovde počinje magija i čarolija ovog alata. Sve do sada se moglo izvesti i bez Ansibla uz malo više kuckanja, ali na ovom delu se vidi koliko Ansibl olakšava proces (prim. aut.). Isto tako, u početku je možda svakome čudan ovaj koncept, ali to je normalno ako se prvi put susrećete sa sistemima orkestracije servera. Zamislimo da morate da održavate više prodavnica. Sada zamislimo da imate robote (module) od kojih su neki zaduženi za čišćenje, neki za pakovanje robe, a neki za otvaranje i zatvaranje prodavnice.



U tom slučaju biste vi morali da napišete pravila po kojima će se roboti ponašati. Recimo, želite da se sve prodavnice prvo zatvore i potom počiste, da se spakuje roba, a posle pakovanja da se prodavnice ponovo očiste. Sve dok prodavnica nije čista, ona ne sme da se otvori.

Sada ovo možemo prevesti na naš glavni problem – nama su prodavnice računari koji mogu biti udaljeni serveri, virtuelne mašine, ili vaš lokalni sistem. Roboti su nam moduli Ansibla, a naš plan koji kucamo je zapravo plejbuk (eng. playbook).

Dakle, mi pišemo plan u obliku tekstualnog fajla, a potom to prosledimo Ansiblu. On treba da na osnovu te naše specifikacije željenog stanja konfigurise sistem. Kasnije ćete videti kako se ovi planovi razdvajaju u posebne fajlove radi lakše organizacije i održavanja.

Ovaj pristup je mnogo pogodan za velike sisteme jer vam omogućava da držite vaše specifikacije u sistemu kontrole verzija kao što su to git ili svn.

Specifikacije Ansibla, plejbukovi, pišu se u YAML sintaksi. Ovaj format je intuitivan i nije ga potrebno učiti posebno. Lako se čita i iz koda se jasno vidi šta se želi postići. Ove specifikacije su upravo to – dakle, nisu skripte sa raznim petljama i slično. Kod pisanja specifikacija se teži velikoj preciznosti. Organizovane su tako da ih je lako razumeti. Razume ih čak i onaj ko nema iskustva sa Ansiblom.

Počinjemo sada od trivijalnog primera. Imamo definisane veb-servere i želimo podesiti Apači na njima. U tom slučaju bi naša specifikacija izgledala ovako:

```
---
- hosts: webservers
  vars:
    http_port: 80
    max_clients: 200
    remote_user: root
  tasks:
    - name: ensure apache is at the latest version
      yum:
        pkg: httpd
        state: latest
    - name: write the apache config file
      template:
```

Slobodni profesionalac

```
src: /srv/httpd.j2
dest: /etc/httpd.conf
notify:
- restart apache
- name: ensure apache is running
service:
  name: httpd
  state: started
handlers:
- name: restart apache
  service:
    name: httpd
    state: restarted
```

Kao što vidite, specifikacija je veoma intuitivna i lako se čita. Na početku definišemo na koje servere se odnosi specifikacija kao i promenljive koje prosleđujemo (ovo će se koristiti u obrascima - eng. templates). Definišemo i korisnika preko kog će se Ansibl logovati i na kojem će se izvršavati komande. U grupi sa zadacima se definišu sledeći zadaci: prvo proveravamo da li je Apači instaliran, a kada to uradimo, podešavamo konfiguracionu datoteku iz obrasca koji se nalazi na lokaciji /srv/httpd.j2 - zatim na kraju pokrećemo servis. Imamo i jedan upravljač (eng. handler), koji se pokreće posle konfiguracije i koji je zadužen za restartovanje servisa.

Jako je važno da udvajate kod kao u primeru jer se u suprotnom specifikacija neće ponašati očekivano. Pored toga, udvajanje je jedan od načina dobre prakse i dobre organizacije koda. Na ovaj način ćete vaš kod moći da razumete i posle nekoliko meseci bez razvoja i rada na istom.

Specifikacija je tako sastavljena i Ansibl tako funkcioniše da će izvršiti promene na sistemu samo ako su one potrebne. Ovo znači da specifikaciju možete izvršiti više puta, a ona će promene napraviti samo kad mora da bi sistem dovela u željeno stanje.

Za više primera preporučujemo da pogledate zvaničnu stranicu dokumentacije, gde možete pronaći riznicu sa dosta primera za različite metode i koncepte. Ako budete imali teškoća i nejasnoća, preporučujemo da kontaktirate s nama na našu adresu elektronske pošte.



Pokretanje odnosno izvršavanje specifikacije postiže se upotrebom komande `ansible-playbook` na sledeći način:

```
ansible-playbook specifikacija.yml -f 10
```

Potrebno je primetiti da smo zadali i nivo paralelizma odnosno broj forkovanih procesa (10 u našem slučaju).

Dobra praksa i organizacija koda

Dokumentacija kaže da koristite role. Mi to isto kažemo. U nastavku se to jasnije vidi.

Direktorijum treba imati ovakvu strukturu:

```
production - inventar za produkciju
site.yml - glavna specifikacija
group_vars/
  grupa1
  grupa2
host_vars/
  server1
  server2
roles/
  common/
    tasks/
      main.yml
    handlers/
      main.yml
    templates/
      ntp.conf.j2
      httpd.conf.j2
    files/
      bar.txt
      foo.txt
  vars/
    main.yml
defaults/
```

Slobodni profesionalac

```
main.yml
meta/
  main.yml
web/
  ... isto kao i za common
python/
  ...
  ...
```

Evo kako bi izgledao site.yml:

```
- hosts: webservers
roles:
- common
- web
```

Sada možemo pokrenuti

```
ansible-playbook site.yml -i production
```

koji će izvršiti celu konfiguraciju.

U našem primeru imamo samo jedan inventar, ali ih možemo imati više ukoliko, recimo, želimo razdvojiti servere za produkciju i za test.

Za više detalja i više primera preporučujemo da pročitate dokumentaciju Ansibla. Tamo možete naći reference za sve module koje možete koristiti.

Ovim tekstom završavamo serijal. Ako mislite da smo nešto važno propustili, ili mislite da smo nešto loše objasnili, pišite nam na adresu naše elektronske pošte.



ANSIBLE



B Bootstrap

Komponente

Autor: Zlatan Vasović

Nakon upoznavanja sa Bootstrapom (eng. *Bootstrap*), možemo krenuti na sledeći korak, koji uopšte nije lak - izrada veb-sajta.

Izbor komponenti CSS-a

Prva stvar koju treba uraditi pre izrade sajta je izbor komponenti. Veoma je važno osmisliti izgled i funkciju sajta, te odatle odabrati potrebne CSS komponente. Potrebno je pročitati dokumentaciju za Bootstrap o svim komponentama (<http://getbootstrap.com/css> i <http://getbootstrap.com/components/>) kako bismo makar okvirno znali čemu svaka komponenta služi.

The screenshot shows the Bootstrap website's 'Components' page. At the top, there's a navigation bar with links for 'Bootstrap', 'Getting started', 'CSS', 'Components', 'JavaScript', and 'Customize'. The main heading is 'Components', followed by a sub-heading: 'Over a dozen reusable components built to provide iconography, dropdowns, input groups, navigation, alerts, and much more.' Below this is a section for 'Glyphicons' with the sub-heading 'Available glyphs'. It includes a grid of icons and their class names, such as 'glyphicon-search', 'glyphicon-heart', and 'glyphicon-star'. A sidebar on the right lists various Bootstrap components like 'Glyphicons', 'Dropdowns', 'Button groups', 'Input groups', 'Alerts', 'Modals', 'Navigation', 'Progress', 'List groups', 'Badges', 'Jumbotron', and 'Page header'.

Sam svoj majstor

Zamislimo da pravimo jednostavnu stranicu za prijavu. Za nju će nam trebati stilovi tipografije (eng. *Typography*) i formi (eng. *Forms*). Dakle, te komponente obavezno biramo.

Za blog su nam obavezno potrebni tipografija i obeležavanje stranica (eng. *Pagination* ili *Pager*).

Primeri postoje mnogo, ali, kao što smo prethodno rekli, izbor komponenti zavisi od želja i potreba pojedinca.

Izbor komponenti Javaskripta

Pored CSS komponenti, Bootstrap sadrži i Javaskript (eng. *JavaScript*) komponente. Te komponente su uglavnom korisne, a pritom ne ometaju korisnika pri pregledanju sajta. Neke od njih su upitni dijalozi (eng. *Modal*), padajući meni (eng. *Dropdown*) i *Scrollspy* (komponenta koja prati skrolovanje dok prelazimo sa jedne oblasti na drugu). Ipak ne treba birati nepotrebne komponente Javaskripta, jer ne rade sve na starijim internetskim pregledačima i usporavaju učitavanje sajta. Neki efekti mogu izgledati privlačno, ali nisu uvek neophodni.

Spajanje u celinu

Nakon što smo smislili koje nam komponente trebaju, prelazimo na sastavljanje delova u slagalicu.

Potrebno je otvoriti Bootstrap uređivač (eng. *Bootstrap Customizer*) - <http://getbootstrap.com/customize/>.

Prvo što vidimo je da su nam ponuđene pojedinačne (eng. *Less*) komponente. Funkciju svake od njih možemo pronaći u dokumentaciji, osim dve - *Print media styles* i *Component animations*. Namena prve je da prilagodi sajt za štampanje, što može biti vrlo korisno ako je sajt namenjen poslovnim korisnicima. Namena druge je da objedini stilove koji su neophodni za sve Javaskript komponente, pa je obavezno odaberite ako nameravate da odaberete i neku Javaskript komponentu.

Druga oblast su Javaskript komponente. Obavezno biramo tranzicije (eng. *Transitions*), jer su neophodne za animacije, a samim tim i za većinu komponenti. Takođe, ne zaboravimo da je J-kveri (eng. *jQuery*) neophodan za sve komponente



Javaskripta.

Bootstrap Getting started CSS Components JavaScript Customize Edit Blog

Customize and download

Customize Bootstrap's components, Less variables, and jQuery plugins to get your very own version.

Heads up! The default value for `@brand-primary` has changed in the latest release to improve contrast and accessibility. Please double check your compiled CSS when upgrading.

Have an existing configuration? Upload your `config.json` to import it.
 Drag and drop here, or manually upload.

Don't have one? That's okay—just start customizing the fields below.

Toggle all

Less files

Choose which Less files to compile into your custom build of Bootstrap. Not sure which files to use? Read through the [CSS](#) and [Components](#) pages in the docs.

Common CSS	Components	JavaScript components
<input checked="" type="checkbox"/> Print media styles	<input checked="" type="checkbox"/> Glyphicons	<input checked="" type="checkbox"/> Component animations (for 20)
<input checked="" type="checkbox"/> Typography	<input checked="" type="checkbox"/> Button groups	<input checked="" type="checkbox"/> Dropdowns

Naredna oblast je vezana za prilagođavanje CSS stilova. Objašnjeno je čemu služe određene promenljive, ali je i pored toga preporučljivo pogledati dokumentaciju pre nego promenimo vrednost određene promenljive.

Kada smo sve sredili za svoje potrebe, kliknemo na *Compile and Download*. Sve će biti zapakovano u ZIP arhivu po direktorijumima zajedno sa konfiguracijom datoteka (*config.json*), koju možemo koristiti da postavimo ista podešavanja kasnije.

Spremni smo za, možda, najteži korak - pisanje koda veb-sajta.

BeagleBone Black Rev C

Vodič od prvog dana (7. deo) - Biglbon Blek u službi veb-servera

Autor: Nenad Marjanović



beaglebone

U prošlosti većina korisnika bila je skoncentrisana na korišćenje veb-servera Apači (*Apache*) iako se još tada znalo da je teško održati resursno zahtevan servis na mašinama koje su sve više bile virtualizovane korišćenjem programa kao što je VMver (*VMware*). Prva verzija veb-servera Hijavata (*Hiawatha*) objavljena je 2002. godine i do danas se ovaj program vodi ideologijom otvorenog koda. Da iza uspešnog projekta

može stajati samo jedan čovek, dokazao je i njen autor Hugo Lesnik. Hijavata je kreirana na jedinstvenom konceptu i ideji, a to je sigurnost. Ovaj projekat danas pored gospodina Huga, pre svake izmene koda, proverava i kompanija koju sačinjavaju eksperti iz domena programiranja, čime je autor dokazao da ne želi odustajati od svog koncepta i motiva - sigurnost na prvom mestu.

Što se tiče resursa, sam server koristi oko *160MB* (u našem radnom okruženju) memorije, ali zato zauzvrat dobijemo što ni jedan od konkurenata danas nema, a to su integrisana antihackerska rešenja. Da li su ova rešenja savršena? Nisu. Međutim, ovo je jedini projekat koji je izvršio integraciju funkcionalnosti koje očigledno drugima nikad nisu bile na listi prioriteta, ili su dodavane u vidu modula koji bi samo još više usporili aktuelno korišćenje sistema. Instalacija za korisnike Debijana može se izvršiti upotrebom Debijan paketa, dodavanjem izvora u



`/etc/apt/sources.list` ili kompajliranjem izvornog koda. Mi ćemo opisati standardnu instalaciju korišćenjem `.deb` paketa. Prvo preuzimamo potreban paket.

```
wget https://files.tuxhelp.org/hiawatha/hiawatha_9.13_amd64.deb
```

Nakon ovoga instaliramo pakete od kojih Hijavata zavisi.

```
apt-get install libxslt1.1
```

Proces završavamo instalacijom `.deb` paketa.

```
dpkg -i hiawatha_*
```

Ukoliko želimo da uklonimo instalaciju u bilo kom trenutku, koristimo `dpkg` komandu.

```
dpkg -P hiawatha*
```

Konfiguracioni fajl nakon instalacije možemo naći na sledećoj lokaciji:

```
/etc/hiawatha/hiawatha.conf
```

Pokretanje i zaustavljanje servera vršimo komandama:

```
service hiawatha start  
service hiawatha stop
```

Da bismo testirali instalacioni proces, otvaramo internetski pretraživač koji koristimo - Fajerfoks, Kroum ili neki drugi. **http://localhost**

Rezultat koji dobijamo je sledeći:

Hardver

Installation successful

Congratulations! The Hiawatha webserver has successfully been installed on this computer. For more information about this webserver, visit the [Hiawatha website](#).

Please, give some feedback about your Hiawatha experiences at the [Hiawatha forum](#).

**Hiawatha Monitor**

Use the [Hiawatha Monitor](#) to keep track of all your websites served by Hiawatha.

Banshee PHP framework

About to create a new website? Use the [Banshee PHP framework](#) for improved website security.

Podešavanja sajta započinjemo integracijom *vhost* parametra u konfiguracionom fajlu (*/etc/hiawatha/hiawatha.conf*), po principu ostalih veb-servera.

```
VirtualHost {
  Hostname = www.my-domain.com
  WebsiteRoot = /var/www/my-domain/public
  StartFile = index.php
  AccessLogfile = /var/www/my-domain/log/access.log
  ErrorLogfile = /var/www/my-domain/log/error.log
  TimeForCGI = 20
  UseFastCGI = PHP5
  UseToolkit = banshee
}
```

Hostname predstavlja domen koji želimo koristiti, ali u slučaju da radimo testiranje, možemo koristiti i Aj-pi (*IP*) adresu našeg servera. *WebsiteRoot* je putanja, odnosno lokacija na našem serveru iz koje se pokreću HTML ili PHP fajlovi. *StartFile* možemo definisati po potrebi. Koristimo HTML za statički sadržaj, *index.html*, a u slučaju da želimo dinamički interfejs i funkcionalnosti, koristimo PHP, *index.php*. Hijavata podržava i druga jezička rešenja kao što su Rubi i Pajton. *AccessLogfile* je lokacija za čuvanje podataka o pristupu sajtu.



ErrorLogfile nam olakšava analizu grešaka sa kojima se suočavamo pri testiranju koda. *UseToolkit* je niz pravila za definisanje učitavanja linkova na sajtu po standardima koji spadaju u *SEO*. Ova pravila su definisana u funkciji *UriToolkit*. Za razliku od *Apacija* i *Endžiniksa* (*nginx*), dosta prekonfigurisanih pravila možemo naći na zvaničnom *Hijavatinom* veb sajtu. Ukoliko želimo koristiti *PHP* za pokretanje sajta, potrebno je instalirati paket *PHP* i nekoliko modula koji nisu integrisani.

```
apt-get install php5 php-pear php5-curl php5-mysql php5-fpm
```

Sigurnost na prvom mestu

Integrisana rešenja nam mogu pomoći da se zaštitimo od nekoliko najpopularnijih napada na veb aplikacije. Ako uzmemo za primer *XSS* (*Cross-site scripting*) napade, dodavanjem pravila u *vhost* podešavanja možemo blokirati većinu poznatih šablona (*paterna*) za ovaj tip napada.

```
PreventXSS = yes|no  
Primer: PreventXSS = yes
```

Jednostavnost integracije ne znači kompletnu zaštitu servera. Naprotiv, korišćenjem ranjivih aplikacija i dalje se izlažemo riziku od napada. Zaštita od napada *SQL injection* integriše se linijom koda.

```
PreventSQLi = yes|no  
Cross-site Request Forgery (CSRF)  
PreventCSRF = yes|no
```

Za one koji žele otići korak dalje, postoji opcija za kreiranje *hash* (heš) zaštitnog fajla. To znači da korišćenjem *wigwam* (*vigvam*) alata koji je integrisan u *Hijavatu*, preuzimamo *SHA256* vrednosti fajlova u folderu u kome se nalazi naš sajt (npr. */var/www/nassajt*) i nakon kreiranja *hash.txt* fajla ostavljamo ga na serveru na zaštićenoj lokaciji. U slučaju da neko proba (pa čak i mi sami) da izmeni fajlove (eng. *Site deface*), umesto uređene strane dobiće stranicu 404, grešku (*error*).

Hardver

```
wigwam -s hash.txt | mv hash.txt /var/www/hashfajl/ | chmod 600  
/var/www/hashfajl/hash.txt
```

U *vhost* podešavanja dodati:

```
FileHashes = /var/www/hashfajl/hash.txt
```

Više informacija o *wigwam* opcijama možete naći na sledećoj lokaciji:

<https://www.hiwatha-webserver.org/manpages/wigwam>

Ovo može biti previše sigurnosti ukoliko svakodnevno objavljujemo novi sadržaj, pa je zbog toga potrebno izmeniti fajl *hiawatha.conf* svaki put pre dodavanja novog članka i ostalog sadržaja. Takođe, obavezno je ponovno pokretanje veb-servera.

Na kraju možemo zaključiti da je Hijavata stabilno i sigurnosno rešenje koje može ponuditi opcije koje ne možemo naći kod drugih veb-server aplikacija. Ostaje nam da se posvetimo daljem čitanju detaljnih uputstva na sajtu autora i uživanje u sigurnosti koja je danas potrebna svakom ozbiljnom projektu. U narednom broju ćemo pisati kako jednostavno podesiti OpenVPN server na našem Biglbon uređaju.





**Tri godine sa
vama**

Hvala što nam verujete!