

Јануар 2015. Број 32

# ЛИБРЕ!

Часопис о слободном софтверу



Photo by Mitch Altman

ЈОШ ИЗДВАЈАМО

**Федора 21**  
**Hibernate ORM**



Creative Commons Ауторство-Некомерцијално-Делити под истим условима

**Реч уредника****„Ми” и „ви” - прича о партнерству са читаоцима**

Од раних дана овог пројекта смо желели да имамо комуникацију са вама читаоцима, мада то још увек не функционише онако како смо очекивали. Важно је да пре свега разјаснимо ко смо то „ми”, а ко сте „ви” у овом тексту. „Ми” се редовно чујемо на састанцима и на друге начине активно учествујемо у пројекту. Сви „ми” смо се заправо скупили из других, старијих заједница и склонили смо се са стране да бисмо у својем миру радили на часопису. Нисмо „ми” неки нови, другачији. И „ви” сте „ми”, нема разлике међу нама, само не причамо довољно често.

Изгледа нам као да постоји нека врста баријере између редакције, аутора и читалаца. Некако смо дошли у ситуацију да је добар део аутора и иначе ангажован у пројекту око других задатака. Међутим, једнако важан део ЛИБРЕ! заједнице сте и ви, читаоци. Да смо кукали, кукали смо. Сигурно смо већ и досадили. Стање у редакцији је сада приметно боље, а сада сте ви на реду за буђење.

Када би требало да се бранимо, могли бисмо да кажемо да смо „ми” свој део посла одрадили. У готово сваком броју нашле су се реченице као што су „помозите да постанемо још бољи”, „поделите своје мишљење” и „јавите се да...”. Такође, на страници сваког броја постоји простор за коментаре, стално је неко на IRC-у, има нас на Фејсбуку и Твитеру, а одавно одржавамо отворене састанке „уживо” на Мамблу. Часопис има своју званичну и-мејл адресу и то је место где је свака порука најбрже примећена, прочитана, и одакле се шаље одговор. Нисмо остали у потпуном мраку и тишини, али повратних информација још увек је недовољно. Знамо да нас читате, желимо да нам се и јавите.

Ако постоји уредништво, читаоци и канал за комуникацију, а комуникације нема - то значи да постоји проблем. Размислили смо и разговарали смо о томе шта може да буде проблем. Разговарали смо само међусобно унутар редакције на јавним састанцима и закључак је да проблем може да постоји у сваком од ова три дела наше заједнице. Можда смо се улењили па нисмо јасно поставили одговарајућа питања. Можда на нашим просторима још није довољно развијена култура узајамног учествовања у пројекту, јер колико је на нама да стварамо часопис,



толико је и на вама да га делите, хвалите и чините бољим. Трећи, а уједно и најједноставнији проблем је инфраструктура и сам начин како можемо да размењујемо идеје. Покушаћемо овим текстом да начнемо овај „наш“ део проблема.

Пројекат је пре скоро три године настао као идеја Александра Станисављевића на Убунту Србија форуму. Ишли смо полако, редом, ИРЦ, и-мејл, вики, Либератор... Све у свему, једна врло занимљива прича коју ћемо врло радо да поделимо са вама ако вас занима. Осим инфраструктуре, „расли“ смо и у погледу организације, броју тимова и чланова. Дизајн је доживео барем три велике ревизије, а посебно смо поносни и на наше лекторе. То је укратко о нама. Питајте нас ако вас занима још нешто, или желите да се детаљније представимо. А ко сте ви? После оволико година, ми још увек нисмо сигурни ко нас највише чита. Следи неколико конкретних питања око којих желимо да нам помогнете.

1. Да ли сте напредни корисник, хакер, програмер, редовни корисник, или почетник?
2. Да ли су вам текстови сувише далеки, сложени и незанимљиви? О чему желите да читате?
3. Желите ли свој пројекат да представите широј заједници путем нашег часописа?
4. Шта желите да поделите са осталим читаоцима? Како сте чули за часопис и слободан софтвер?

Други проблем, који треба да решимо, јесте да обезбедимо најједноставнији пут за комуникацију. Можда смо мало старомодни па очекујемо да и-мејл функционише и у 21. веку. Који је ваш предлог? Да ли бисте радије одговорили на анкету? Желите ли да преуредимо сајт тако да буде пријатнији за коментаре? Верујемо да је и ово проблем на чијем решењу треба да радимо, а ту су нам потребни ваши савети и мишљења. За почетак, најважније је да скупите храбрости и воље да нам се јавите на који год начин желите. Ако не можете да одлучите којим путем да нам се јавите - пошаљите нам и-мејл на добро познату адресу [libre \[at\] lugons \[dot\] org](mailto:libre[at]lugons[dot]org) да и „ви“ званично постанете „ми“.

Срдачан поздрав и хвала!

ЛИБРЕ! тим

# Садржај

## Вести

стр. 6

## Пулс слободе

*31C3 - a new dawn*

стр. 9

## Представљамо

Федора 21

стр. 19

## Како да...?

Увод у програмски језик C

стр. 24

## Ослобађање

У потрази за идеалном дистрибуцијом:  
Критеријуми избора према примарној  
функцији рачунара (5. део)

стр. 28

## Слободни професионалац

*Hibernate ORM* (1. део)

стр. 39

## Интернет мреже и комуникације

Енкриптована електронска пошта (5. део)

стр. 44

## Sam svoj majstor

*LaTeX презентација: Beamer* (3. део)

стр. 50

## Хардвер

*BagleBone Black Rev C*: Водич од првог дана (3. део)

стр. 56

Моћ слободног  
софтвера





Број: 32

Периодика излажења: месечник

Извршни уредник: Стефан Ножинић

Главни лектор:

Александар Божиновић

Лектура:

Јелена Мунђан Сашка Спишјак

Милена Беран Милана Војновић

Александра Ристовић

Адмир Халилкановић

Графичка обрада:

Дејан Маглов

Иван Радељић

Дизајн: White Circle Creative Team

Аутори у овом броју:

Ненад Марјановић

Момчило Медић

Петар Симовић

Дејан Чугаљ

Лука Герзић

Никола Харди

Остали сарадници у овом броју:

Марко Новаковић Михајло Богдановић

Почасни чланови редакције:

Жељко Попивода

Жељко Шарић

Владимир Попадич

Александар Станисављевић

Контакт:

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

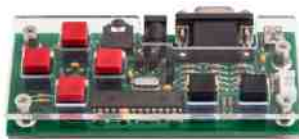
Е-пошта: libre@lugons.org

**Вести**

2. јануар 2015.

## Нови пројекат Воје Антонића

Воја Антонић, пројектант првог кућног рачунара направљеног у Југославији - „Галаксија”, има нови пројекат слободног хардвера. Ради се о *DIY* (ди-аи-вај, енг. *Do it yourself*) мини ретро 2D играчкој конзоли.



Користан линк: <http://t.co/WuLVwddftb>

---

3. јануар 2015.

## Линукс **AMD**® власнички управљачки програм за графику „уништи“ Виндоуз

Форониксови (*Phoronix*) тестови показују да је **AMD**-ов власнички управљачки програм на Линуксу показао боље перформансе у односу на исти на Виндоузу када су у питању неке могућности *OpenGL*-а (Опенциел).



Користан линк: <http://t.co/E3Oe3mGxk4>

---

5. јануар 2015.

## Пирати пишу свој закон

Пиратска партија Европе, која већ у два мандата има своје представнике у Европском парламенту, почела је израду сопственог закона о интелектуалној својини. Постојећи европски закон о интелектуалној својини има елементе ауторског монопола што Пиратска партија жели да измени.



Користан линк: <http://t.co/Uyl0ZBBlo0>

---



6. јануар 2015.

## Фондација Линукс добила три нова члана

Фондацији Линукс су се почетком ове године придружиле IIX, *Micron Technology* и *Planisys*.

Користан линк: <http://t.co/Ht9MEq3W7a>

---



7. јануар 2015.

## Флаксбокс 1.3.6

Флаксбокс (енгл. *Fluxbox*) верзија 1.3.6 је доступна за преузимање.

Користан линк: <http://t.co/fpsGgeEsAE>

---



FLUXBOX

---

7. јануар 2015.

## Сербиан 2015 Опенбокс

Домаћа дистрибуција Линукса, Сербиан, базирана на стабилној грани Дебијана (енгл. *Debian*), објавила је нову верзију са графичким окружењем Опенбокс (енгл. *OpenBox*). Ова верзија Србиана намењена је слабијим/старијим рачунарима.

Користан линк: <http://j.mp/15xR5xk>

---



Вести

9. јануар 2015.

## Ардуино освојио награду на CES 2015

Ардуино (енгл. *Arduino*) је освојио награду за иновацију на овогодишњем CES 2015 (си-и-ес, енгл. *Consumer Electronics Show*).

Користан линк: <http://t.co/cW0oqnaYid>



11. јануар 2015.

## Фајерфокс оперативни систем на телевизорима

Панасоник (енгл. *Panasonic*) је најавио телевизор који ће користити Фајерфокс оперативни систем (енгл. *Firefox OS*).

Користан линк: <http://t.co/K2GnTsVDTR>



24. јануар 2015.

## RailsGirls® радионице у Новом Саду

У Новом Саду су одржане радионице које су намењене женама које би желеле да науче програмирање. Радионице су трајале два дана, 24. и 25. јануара ове године.

Користан линк: <http://j.mp/1zwo1Bv>

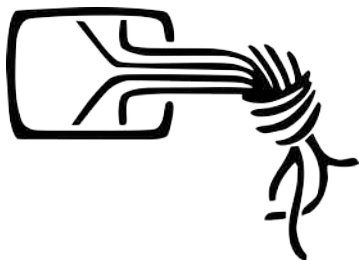






Аутор: Лука З. Герзић

## Шта је то 31C3?



31C3 је ознака тридесет првог *ССС* (енг. *Chaos Computer Club*) годишњег конгреса који се одржава у Хамбургу. То је највећи и најстарији европски хакерски конгрес на свету. Један од оснивача клуба је Вау Холанд (нем. *Wau Holland*), по коме је касније креирана и истоимена фондација. Клуб је основан давне 1981. године, док конгрес почиње да се одржава од 1984. године. *ССС*

клуб броји више од 5700 чланова. Званичан опис клуба, који организује конгрес, јесте: „Галактичко удружење живих форми, независно од старости, пола, расе или социјалне оријентације, које не познаје границе за слободу информације”.

Како бисте имали бољи увид у величину овог конгреса, даћемо неколико података са овогодишњег тридесет првог по реду конгреса.

За време трајања конгреса, који је отворен 24/7, било је више од 13.000 посетилаца и више од 1.000 волонтера. Постављено је преко 9km струјних каблова и 3200 утичница за посетиоце, а потрошено је 53 *MWh* струје за четири дана конгреса. Постојало је 125 *WiFi* приступних тачака (енг. *Access Point* — *AP*), те више од 2 *Aruba 7200* контролора са десетогигабитним ап-линковима (у шпицу је имао 7800 корисника - 68 клијената у просеку по приступној тачки) - и виђено

## Пулс слободе

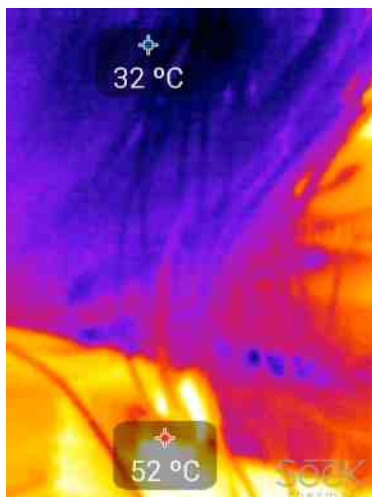
је преко 20.000 уникатних уређаја на WiFi мрежи. Већи део комуникационе опреме донирао је Џунипер (енг. *Juniper Networks*), тачније 1.2т (тоне) Џуниперове опреме (3 милиона евра укупне осигуране вредности). Поред свега тога, процесуирано је 80TB стрим видеа уживо, енкодвано и дистрибуирано са конгреса ка преко 90.000 посетилаца путем CCC CDN опреме (енг. *content delivery network*). Конгрес поседује сопствену GSM и DECT (енг. *Digital Enhanced Cordless Telecommunications*) мрежу (свако може затражити GSM SIM картицу или донети свој DECT телефон). За оне који воле ретро, ове године је поново био активан Зајденштрасе (нем. *Seidenstrasse*) (систем цеви за трансфер капсула кроз пластичне цеви уз помоћ вакуума).



Наша екипа имала је ту привилегију да погледа како сам „Колокејшн”<sup>1</sup> (енг. *Colocation*) део (где можете донети своју опрему) изгледа, као и да буде део тима

---

<sup>1</sup> **Colocation** - део рачунског центра где свако може да донесе своју машину и да је повеже на интернет.



од свега осам људи који су знали где је прави NOC (енг. Network Operations Center) у коме је смештена главна комуникациона опрема конгреса (конгрес поседује две лажне локације опреме). С обзиром на то да смо поседовали малу термалну камеру, снимили смо за вас колико су врући core („кор“) рутери конгреса. Уверите се и сами.

Мада, када говоримо о овом конгресу, требало би уствари да размишљамо о једном великом хакерспејсу. Заправо је то један огроман простор који контролише целокупна заједница, где људи са заједничким интересима о компјутерима, мрежама, машинству, науци, дигиталној

и/или електронској уметности размењују мишљења, сарађују или се такмиче. Суштина је у колаборацији, учењу и дружењу. Када говоримо о хакерспејсу, морамо поменути да се први и највећи хакерспејс налази управо у Немачкој, а зове се C-base (Си-бејс).



**Пулс слободе****Припреме за одлазак и улазак на конгрес**

План за одлазак на овакав догађај подразумева пре свега набављање авионских карти и хотелског смештаја барем три месеца унапред. Иако се конгрес одржава у склопу *CCH* (енг. *Congress Center Hamburg*) поред којег је одмах хотел „Радисон Блу“ (енг. *Radisson Blue*), већ два месеца пред конгрес хотел је комплетно букиран, као и већина оближњих мањих хотела и хостела. Одлазак на овакав догађај препоручили бисмо да буде у тиму људи сличних интересовања. Без познавања немачког језика, и без познанстава, у соло варијанти, све ово једноставно неће деловати на вас на исти начин.

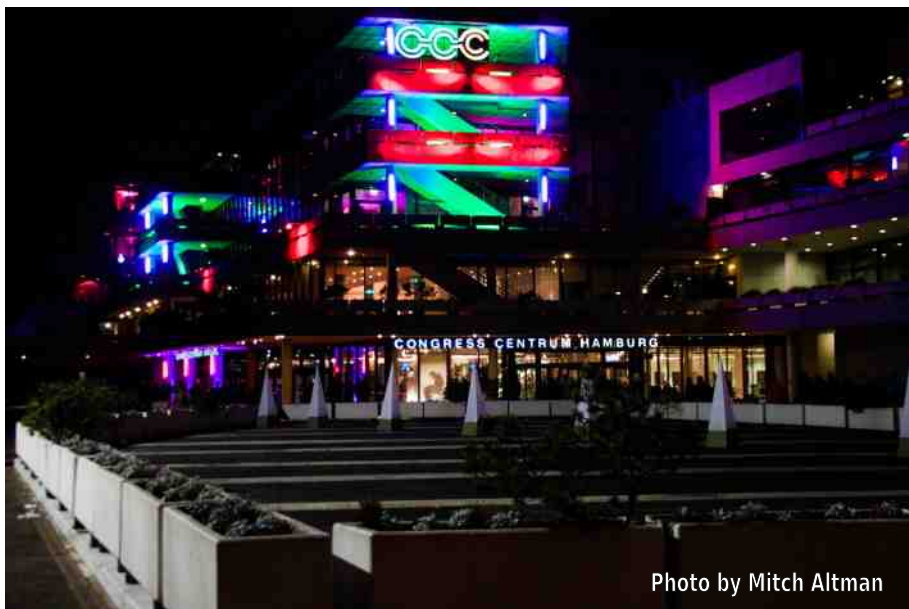


Photo by Mitch Altman

Препоручили бисмо вам да понесете следеће ствари: ноутбук са Линуксом (по могућству новији са *802.11n/ac WiFi* картицом уз додатну батерију), *DECT* телефон и/или стари *GSM* телефон, неколико радио станица/воки-токија, батерије за пуњење уређаја преко *USB*-а (што јаче то боље, више од *8000 mAh*), етернет каблове (2-5м дужине), продужни гајтан (2-5м), *USB3* тврде дискове (више од *2TB*), различите верзије *USB* каблова, *USB2/3* стикове, квалитетне слушалице, *VPN gateway* (рецимо *TL3020* са *openwrt*-ом), *3G WiFi dongle*, различите врсте адаптера, термос за кафу, воду или сок, и потребне лекове. Пре поласка, обавезно урадите



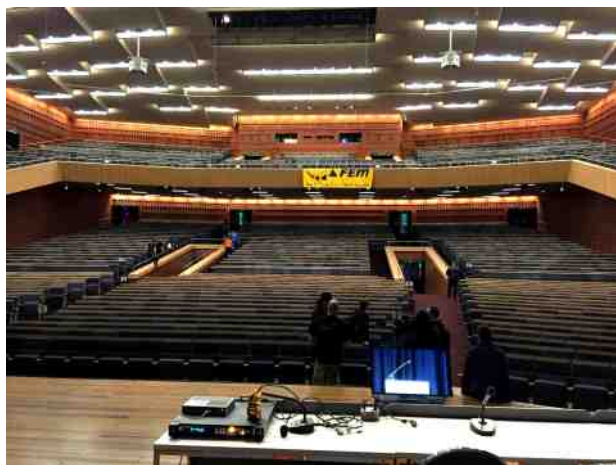
комплетан бекап својих уређаја, подигните заштитни зид (енг. *firewall*) и водите рачуна да на својим уређајима немате поверљивих датотека. На свим уређајима искључите све непотребне сервисе, подесите телефон на мануелни избор *GSM* провајдера, а ниједан уређај никад немојте остављати без надзора.



Топло препоручујемо долазак један дан раније или барем вече пред конгрес, уз обавезну регистрацију то вече (уз регистрацију *DECT* телефона и/или набавку *GSM* картице која кошта 2 евра), зато што вас на дан конгреса ред за узимање наруквице може „коштати“ неколико сати до уласка, јер су редови огромни. Овогодишња улазница за конгрес је коштала 100 евра за све дане.

Обавезно преузмите апликацију за *iPhone®* или Андроид која ће вам олакшати преглед дешавања и сам распоред конгреса.

## Простор и сам догађај



Поред великог броја предавања, која наравно можете пратити и уживо путем стрима, ту су и друга врло интересантна дешавања као и дружење са пријатељима. Сва предавања су била превођена на немачки и/или енглески језик како титлом на екрану, тако и преко *DECT/GSM* мреже, наравно у реалном времену. Можда је најбољи

осећај онај да сте своји међу својима, а ту мислимо међу људима који се баве истим или сличним послом као и ви, а који слично или исто размишљају.

**Пулс слободе**

Photo by Mitch Altman

Сам простор, тачније просторије конгреса, феноменално су осмишљене и украшене. Много труда и новца уложено је у сјајне идеје како за осветљење тако и за уређење самог конгресног простора. Дефинитивно најбоље украшен био је велики ходник који је изгледао као ходник свемирског брода, украшен великим бројем пројектора, лед диода и рефлектора, уз врло специфично амбијентално осветљење. Овај ходник повезивао је главни хол са просторијом где су биле смештене радионице за лемљење и *CTF* (енг. *capture the flag*) екипе.



Од интересантних дешавања ту је наравно *CTF* догађај (*Jeopardy style*) који се традиционално организује. Различити нивои тежине задатака као и разноврсност покривених



области и ове године били су на нивоу задатка. CTF активне категорије биле су: *pwn*, *crypto*, *reversing*, *malware*, *signals* и *web*. Било је регистровано преко 1000 тимова, а победник ове године била је група *Oxffa*. Више детаља о CTF-у можете погледати овде <https://31c3ctf.aachen.ccc.de> или на <https://ctftime.org/>.

Затим, можете се опробати у обијању брава (енг. *lockpicking*) на неколико радионица у оквиру конгреса. Кад спомињемо радионице, велики део конгреса био је посвећен и изради сопствених ручно прављених CNC (енг. *Computer Numerical Control*) машина као и 3D (тродимензионалних) штампача. Размењивала су се искуства у креирању комплексних 3D објеката као и креирање сопствених 3D штампача различитих намена.

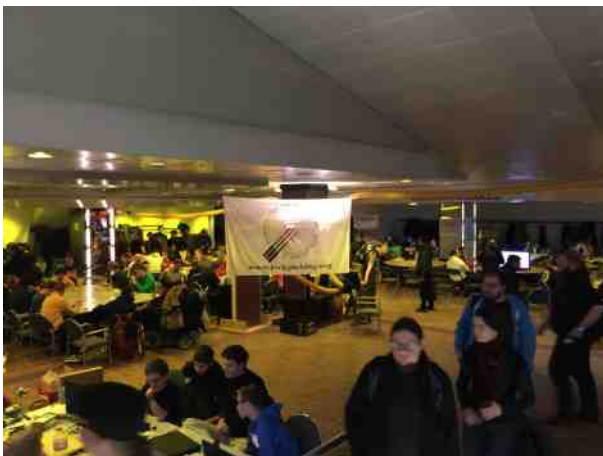


Photo by Mitch Altman

## Пулс слободе

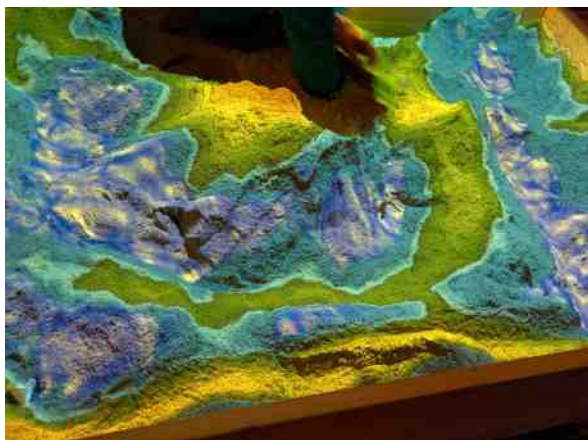
Неколико радионица за лемљење, од оних конципираних за почетнике, до оних напредних било је актуелно током читавог конгреса. За све оне који су желели да купе готове китове било је заиста великог избора.

За оне који су били фрустрирани опремом, или су једноставно желели да униште перманентно неку компоненту (или цео рачунар), била је инсталирана велика хидраулична преса која би тај део опреме претворила у гомилу ситног ђубрета. Одличан начин да перманентно уништите, рецимо, свој стари хард диск, зар не?

На самом конгресу могуће је креирати сопствену мајицу/дуксерицу, коју вам могу одштампати или извести директно на вашој гардероби. Поред тога било је могуће ласером гравирати било шта од пластике или алуминијума на неколико ласерских гравера који су били присутни на догађају. За оне који су желели да купе мајице, дуксерице и остали рекламни материјал, цене су се кретале од 25 до 45 евра.

На конгресу је могуће креирати *ID* (идентификациону) карту са подацима које ви изаберете. Све што је било потребно да урадите јесте да се прошетате до дела за фотографисање, направите своју фотографију, и попуните жељене податке. Лажна лична карта делује врло уверљиво, али сигурно неће проћи проверу немачке полиције, за остале не бисмо били тако сигурни.

Посебну пажњу посветили су деци. Поред пуне играонице, где је била смештена велика количина лево коцкица и других играчака, био је креиран сегмент са песком изнад којег је био монтиран 3D скенер који је у зависности од висине или дубине песка контролисао пројектор који је осветљавао рељеф песка (реке, језера, равнице, брда и планине) у реалном времену. Била је, такође, поставка свих могућих ретро игрица на старим конзолама. За мало старију децу креиране су посебне радионице за прве кораке у електроници.











# Федора 21

**Аутор:** Момчило Медић

Након годину дана стигла је Федора 21.



Овај пут без кодног имена, Федора 21 долази са много новина које ће привући велики број љубитеља слободног софтвера. Пре него што почнемо са уобичајеним набрајањима новитета и верзијама програма, потребно је објаснити „вечито“ кашњење Федоре које се често помиње у круговима слободног софтвера. Наиме, велики број дистрибуција ГНУ/Ликукса се могу поделити према начину издавања. Постоје дистрибуције које се објављују као ролајућа издања (енгл. *rolling release*), којима заиста и нису потребна посебна издања јер њихови корисници преко ажурирања добијају најновије верзије актуелних програма. Ово гарантује

## Представљамо

најновије верзије програма, али носи одређени ризик по стабилност система. Постоје едиције које имају стриктно задато време издавања без синхронизација са другим пакетима. Они ће издати своју едицију у задато време без обзира на број отворених програмских грешака у систему и укупне недовршености система. Са друге стране, имамо дистрибуције које немају задат никакав посебан датум или период објављивања, него се ради на новој верзији док год се не буде сматрала потпуно спремно и стабилном. Овакав начин обезбеђује поуздан систем, али због тога „трпе“ иновације. Федора се одлучила за златну средину са одређеним периодом издавања као циљем, али тек ако је систем у потпуности тестиран. Зато се Федора сматра веома стабилном дистрибуцијом која у себи садржи најбоље и најновије из света слободног софтвера. Такође, Федора се планира паралелно са будућим већим издањима програма или групама програма као нпр. Гном (енгл. *Gnome*) и сл.



Након свега тога, потребно је рећи да се чекање исплатило и да Федора 21 има потенцијал да буде најбоља едиција ове дистрибуције до сада. Али, зар само 21? Где су она чудна и интересантна кодна имена? Федора тим је одлучио да су кодна имена, која су досад била у употреби, потпуно бескорисна. Аутор овог текста је волео та имена и чинило се да оперативни систем некако има душу, односно неку шаљиву страну, али питање „која верзија Федоре је носила кодни



назив Бифи Миракл (енгл. *Beefy Miracle*)” убедило га је да су она заиста бескорисна. Не само да се едисије нису могле повезати са именима, него је требало из верзије у верзију ажурирати како број тако и кодно име у датотекама, програмима, сликама и осталом.

Федора 21 прави велики скок са Гнома 3.10 из „двадесетице” на Гном 3.14. Но, нећемо упоређивати временски веома удаљене верзије Федоре, него ћемо видети шта то Гном 3.14 доноси са собом. Нове приметне промене анимација за мени и прозоре су суптилне али естетски веома угодне. Очекујте да ће вас нови Гном обавестити ако бежична мрежа на коју сте се повезали тражи да се поново пријавите. Ако сте корисник Гуглових сервиса, бићете пријатно изненађени подршком за слике у апликацији Фотографије (енгл. *Photos*). Апликација Време (енгл. *Weather*) која приказује локално време, иако са оскудним бројем градова у Србији након инсталације, веома је квалитетна и атрактивна, те се вреди помучити и подесити је и за мања места. Наравно, Гном не би био такав какав је, да нема побољшања и унапређења у подршци за екране осетљиве на додир, мултитакилно осетљиве екране (енгл. *multi-touch*) и Вејленд (енгл. *Wayland*). Међу осталим новинама су још и израда навигационих рута у апликацији Мапе (енгл. *Maps*), Гном софтвер подршка додацима за програме и тзв. „*GTK+Inspector*” за програмере.



fedora  
SERVER



fedora  
WORKSTATION



fedora  
CLOUD

Ступила је на снагу тзв. „*Fedora.next*” иницијатива у којој имамо три различите верзије: Федора радна станица, Федора сервер и Федора облак (енгл. *Cloud*). Федора радна станица је намењена кућним рачунарима који углавном служе за рад са канцеларијским пакетима, развој апликација, припрему мултимедије, забаву и слично. За потребе кућних сервера, као и сервера који захтевају подршку за најновије технологије - ту имамо Федору сервер. Ова верзија никако не покушава да буде конкуренција ЦентОС-у или Ред хет ентерпрајз линуксу, него да попуни место сервера који садржи иновације. И на крају, чињеници колико рачунари постају јефтине сведоче и све приступачнији системи облака. У ту сврху, Федора је обезбедила Федору облак за припремање Докер (енгл.

## Представљамо

*Docker*<sup>1</sup>) инсталационих слика као и других решења за рачунарство у облаку. За оне који желе да испробају Вејленд, у Федори 21 ће то моћи да ураде једноставно и лако са подршком за „Гном на Вејленду“. Морамо напоменути да, иако је у плану да Вејленд у Федориној верзији 22 или каснијој буде подразумевани графички сервер, за сада је још увек у експерименталној фази и не препоручује се испробавање, осим ако нисте спремни да загребете мало дубље. *DevAssistant* је алатка корисна великом броју програмера, а од сада се налази као подразумевани програм у Федори. Ту је и *Logs* апликација која уме да буде помало збуњујућа за кориснике терминала, али је у сваком случају добар додаток за кориснике који преферирају графичко окружење. Овај чланак се не би могао сматрати комплетним без спомињања кокпита (енгл. *Cockpit*). Кокпит је апликација за мониторинг и подешавање сервера и рачунара. Веома је једноставан за инсталацију и покретање, након чега ће вас оставити без речи својим изгледом и могућностима. Потребно је напоменути да је ово софтвер који је тек у развоју и као такав не би требало да се користи на системима у продукцији.



Наравно, као и свака верзија Федоре, и 21 доноси многе изведбе (енгл. *Spins*) за кориснике ове дистрибуције који траже нешто другачији доживљај. Можда у виду другачијег подразумеваног окружења: KDE, ИксФЦЕ, Мате, или у виду система предвиђеног за музичаре - *Fedora Jam-KDE*, за научнике - *Fedora Scientific*

<sup>1</sup> <http://spins-test.fedoraproject.org/sr/docker/>



KDE, или за оне који воле да тестирају сигурност - *Fedora Security*. Ово су само неки од изведби који су доступни за Федору. Препоручујемо да се упознате са њима и видите постоји ли нешто „само за вас“ на <http://spins.fedoraproject.org>.

Желели бисмо да вам скренемо пажњу на велику подршку људи иза Федора пројекта и на општу ангажованост. Ако знате енглески, можете потражити помоћ на сваком кораку. Ту је Федорин форум, где ретко које питање остане без одговора. Уколико желите систем одговора и да заголицате нечији ранкинг, на располагању вам је [ask.fedoraproject.org](http://ask.fedoraproject.org). Наравно, ниједан систем подршке не може да се мери по брзини са добрим, старим IRC-ом, где ће вам се понудити одговор у року од неколико минута на серверу Фринод (енгл. *Freenode*) у соби *Fedora*. Надамо се да ће овај кратки чланак бити иницијална каписла вашег интересовања за фантастичну дистрибуцију - Федору 21.

Корисни линкови:

- [1] <http://fedoraproject.org/en/>
- [2] <http://webchat.freenode.net/?channels=#fedora>
- [3] <https://ask.fedoraproject.org/>
- [4] <http://www.fedoraforum.org/>

Преглед популарности *GNU/Linux* /*BSD* дистрибуција за месец јануар

## Distrowatch

1	Mint	2451>
2	Ubuntu	1562>
3	Debian	1460>
4	openSUSE	1274<
5	Android-x86	1153>
6	PCLinuxOS	1129>
7	Fedora	1096<
8	elementary	981>
9	Deepin	971<
10	CentOS	948>
11	Arch	926=
12	Mageia	890>
13	Manjaro	783>
14	SteamOS	727>
15	Puppy	677>
16	KaOS	655>
17	LXLE	638=
18	Lubuntu	581>
19	SparkyLinux	579<
20	Zorin	577<
21	Simplicity	561>
22	NixOS	539>
23	Tails	501>
24	ROSA	495>
25	ZevenOS	490=

Пад <  
 Пораст >  
 Исти рејтинг =  
 (Коришћени подаци са Дистровоча)

**Како да...?**

# Увод у програмски језик C

## (8. део)

**Аутор:** Никола Харди

У претходним бројевима сте имали прилику да читате о C програмском језику. Надамо се да смо успели да представимо довољан део овог програмског језика да бисте могли да почнете стварати своје програме, или да лакше савладате напредније ствари. Наредним текстовима ће бити завршен серијал и у њему можете да прочитате неке ствари које смо приметили да недостају у претходним текстовима. Такође, даћемо и неколико смерница за даље истраживање, учење и играње. Почећемо од функција.

У историји развоја програмских језика постоји један тренутак када су тадашњи програмери приметили да неке делове кода користе више пута. У то време су се програми писали на бушеним картицама, а „математичка библиотека” је био термин који је означавао скуп картица са корисним кодовима за математику, увезаних машином. Функције, или потпрограми, како их неки још зову, јесу делови кода које је могуће по потреби позвати. Након извршеног кода, који та функција има, процесор наставља да извршава код од места на којем је функција била позвана.

Пре него што пређемо на детаљан опис како се декларишу, дефинишу и позивају функције у C-у, биће објашњено зашто су оне толико важан механизам. Основна сврха је да се код који се понавља издвоји на једно место и само позива. Тако је омогућено да тај део кода искористите на неком другом својем пројекту или га поделите са другима. Други разлог је једноставније одржавање пројеката. Софтвер увек има грешке и то се не доводи у питање. Наравно, примећујете да је много елегантније исправити грешку на једном месту (унутар функције) него на сваком месту где би се тај код појавио (када се позивају те функције).

Функције могу да обухвате и код који није дословно исти. Код можемо да „параметризујемо”. То значи да при сваком позиву функције можемо да подесимо





неке делове кода. На пример, ако имамо функцију за испис имена, можемо да јој поставимо параметар које име желимо да буде исписано. То се постиже путем параметара или аргумената. У овом тексту нећемо правити разлику између параметара и аргумената иако она формално постоји.

Последња ствар коју треба да поменемо када причамо о идеји функција је да оне могу, а не морају, да имају резултат. Строго речено, потпрограми који имају резултат су функције, а потпрограми који немају резултат су процедуре. Методе су јако сличне функцијама, али се углавном помињу у контексту објектно оријентисаног програмирања. Нико вам неће судити ако функцију назовете процедуром или обрнуто, али лепо је знати шта је шта. Најчешће ћете ипак чути за израз функције или методе, па ћемо и ми ове термине користити већом слободом.

Као и код променљивих, функција мора да буде декларисана пре него што је позовемо. Декларација функције се назива још и њеним „потписом” и састоји се од типа повратне вредности, имена потпрограма и листе параметара, завршно са знаком тачка-зарез. Уколико уз декларацију функције желимо и да је дефинишемо, то може да буде урађено тако што се изостави знак тачка-зарез и у наставку потписа се напише тело потпрограма.

Пример процедуре без параметара:

```
void procedure()  
{  
    puts("Hello world!");  
}
```

Пример функције са два параметра:

```
int sum(int a, int b)  
{  
    int c;  
    c = a + b;  
    return c;  
}
```

## Како да...?

Примећујете да параметри имају свој тип и назив. Понашају се слично променљивим. Унутар функције могу бити декларисане додатне променљиве. Резултат функције, који се још назива и повратна вредност, „враћа“ се наредбом *return*. Важно је рећи да се извршавањем наредбе *return* прекида извршавање целе функције, што је илустровано следећим примером.

Пример прекида извршавања функције наредбом *return*:

```
int min(int a, int b)
{
    if(a < b)
        return a;
    else if(b < a)
        return b;
}
```

Уколико је број *a* мањи од броја *b*, извршавање ће се завршити већ у другој линији ове функције, а до друге провере неће ни доћи. Занимљиво је приметити и да ова функција има грешку. Шта се догађа уколико су бројеви једнаки?

Функције имају сврху само ако их користимо, односно „позивамо“. Функције се позивају по имену, а од променљивих се разликују по томе што се у позиву функције пише и пар заграда, унутар којих су вредности параметара када је то потребно.

Пример позива функције *sum()*:

```
int zbir = sum(2, 3);
```

Последња ствар која је остала недоречена је да се параметри у функцију могу пренети по вредности или по адреси. Практична употреба и детаљи о разликама су ипак можда превише за један уводни курс о C-у. Најважније ствари које треба имати на уму су да, када параметре користите као у нашим примерима функција *sum()* и *min()*, тада се они преносе по вредности и практично се копирају. То значи да ако доделимо нове вредности параметрима, те вредности ће важити само унутар функције, а не и на месту где је функција позвана. Да би такве доделе биле видљиве и ван функције, користи се пренос параметара по адреси.



То заправо значи да се при навођењу параметара у декларацији користе показивачи, а при позиву функције се користе адресе (показивачи или адресе променљивих).

Пример преноса параметара по адреси:

```
void swap(int *a, int *b)
{
    int tmp;
    tmp = *a;
    *a = *b;
    *b = tmp;
}
```

Позив:

```
int a = 5, b = 7;
swap(&a, &b);
```

Требало би да овај текст покрије основне ствари о руковању потпрограмима. Уколико вам је потребна инспирација како да увежбате рад са функцијама, ево лепог проблема:

Покушајте да напишете функцију за испис елемената низа на стандардни излаз. Затим напишите функцију за сортирање низа у којој ћете позивати функцију за испис низа и посматрајте како елементи низа мењају своја места.

Уколико вас занима нешто више, или нам се поткрала нека грешка, молимо вас да нам се јавите. Срећно!

# Learn C Programming

## У потрази за идеалном дистрибуцијом:

# Критеријуми избора према примарној функцији рачунара (5. део)

**Аутор:** Дејан Маглов

Код оперативног система Виндоуз прилагођавање рачунара према његовој примарној функцији се своди само на прави избор потребних апликација (апликативног софтвера). Апликације за Виндоуз су у већини случајева добре, али врло захтевне по питању хардверских ресурса. Ако томе додамо поприличну захтевност самог оперативног система, долазимо до закључка да добра продуктивна машина са Виндоузом мора да буде прилично „јака“. Дубљом анализом самих процеса могли бисмо закључити да добар део хардверске „снаге“ одлази на многобројне споредне функције на које корисник само делимично може да утиче или чак не може никако да утиче.

Филозофија ГНУ/Линукса по питању креирања продуктивне машине је другачија. Захваљујући отвореном коду, прилагођавање рачунара одређеној функцији се не своди само на избор апликација, него је могуће и оптимизовати сам оперативни систем (ОС) тако да штеди хардверске ресурсе и максимално их користи за примарну функцију. Није једнака оптимизација система за примарну функцију великог броја аритметичких прорачуна, за баратање великом количином података коришћењем Булове алгебре, за обраду слика, звука и слично. Свака од тих функција захтева посебну оптимизацију система.

Иако је могуће од било које ГНУ/Линукс дистрибуције направити продуктивну машину за било коју функцију додавањем потребних апликација, на истом хардверу крајњи резултат ће бити различит по питању ефикасности и продуктивности. Различита оптимизација самог система је проузроковала те



разлике.

## Пријатељски настројен систем као критеријум за избор

За оперативни систем кажемо да је пријатељски настројен (енг. *User Friendly*) према кориснику ако корисник може лако да прилагоди тај систем својим потребама по питању изгледа и примарне функционалности. Виндоуз је поставио стандард за појам „пријатељски настројени оперативни систем” и по њему то је поглед на оперативни систем из угла мање искусних корисника рачунара. Подразумева да систем за корисника одради што више подешавања аутоматски, а за остала подешавања има графичке алате тако да корисник са неколико кликова миша може мало да прилагоди систем или његов изглед својим потребама.

У свету ГНУ/Линукса, на тај појам се може двојако гледати. Може се гледати из угла искусног корисника и из угла почетника. У овом поглављу ћемо вам показати да су то два потпуно супротна угла гледања.

Објективно најбољи, најлакши и најприроднији начин да се ГНУ/Линукс оптимизује и прилагоди кориснику је преко командне линије. Кроз низ команди или чак скрипту команди корисник може најбрже да оптимизује и прилагоди свој ГНУ/Линукс систем и све ГНУ апликације. Добром скриптом може чак и да одржава велики број рачунара. Из тог разлога искусни корисници ГНУ/Линукса за пријатељски настројен ОС сматрају онај који има најједноставније команде и који не ради ништа аутоматски него кориснику оставља на вољу да свој систем сам оптимизује до најмањих ситуација.

Са друге стране, имамо почетнике у ГНУ/Линуксу који се још прилагођавају новом ОС-у, тек уче филозофију слободе софтвера и не познају рад у командној линији. За њих важи Виндоузова дефиниција за пријатељски настројен оперативни систем, иако то није најприроднији начин оптимизације ГНУ/Линукса. Аутоматика при оптимизацији Линукса као и помоћни графички алати за оптимизацију само непотребно оптерећују систем, а резултат је ретко идеалан. Ипак, по жељи корисника пројектоване су и такве дистрибуције намењене почетницима. На **слици 1** је приказан дијаграм фамилија ГНУ/Линукса од пријатељски настројених до сувише тешких из угла ГНУ/Линукс почетника:

## Ослобађање



слика 1

Из угла искуснијих корисника, овај дијаграм пријатељски настројених система је управо обрнут. Џенту нуди потпуну оптимизацију оперативног система. Ништа се не инсталира аутоматски, а за сваку оптимизацију постоји избор за корисника. Команде су довољно једноставне да тај процес креирања идеалне дистрибуције буде удобан за кориснике који знају шта хоће на крају да добију. То је за почетнике можда наизглед мазохистички приступ изградње оперативног система, али за искусне кориснике је опет мазохистички оптимизовати већ аутоматски одрађен Убунту систем. Толико би промена требало извршити у Убунтуу да се таква операција не исплати.

Објективно савршен ГНУ/Линукс систем се налази негде на десној страни дијаграма. Субјективно може бити било где на том дијаграму и то зависи само од корисника.

## Примарна функција рачунара као критеријум за избор

Већина корисника нема ни времена, ни знања, ни воље да ГНУ/Линукс систем гради из саставних делова, а ни да сваки део система идеално оптимизује према сопственом хардверу. Лепо је ако је могуће ОС идеално подесити али то није примарни циљ. Најважније је да ОС буде једноставан за употребу, да не



## У потрази за идеалном дистрибуцијом

оптерећује хардвер и свог корисника.

Из гомиле ГНУ/Линукс дистрибуција је могуће наћи већ довољно оптимизовану дистрибуцију за већину намена. У наставку ћемо препоручити неке од њих.

### Десктоп

Десктоп је неоправдано суверено царство Виндоузовог оперативног система. Представља систем намењен обичном свакодневном коришћењу рачунара које обухвата лако повезивање на интернет, претраживање интернета, комуникације (друштвене мреже, VoIP, инстант поруке), манипулацију мултимедијалним садржајем (слике, видео, звучни записи), рад са канцеларијским пакетом програма. Овај ОС првенствено је био намењен стационарном персоналном рачунару, али са унапређењем снаге мобилних рачунара (лаптоп и таблет рачунара) представља основни ОС и за те машине.

Када смо рекли да је ово неоправдано суверено царство Виндоуза, то смо и мислили. Одговорно тврдимо да ГНУ/Линукс има исто тако квалитетна решења за ову намену рачунара. Не само да су ГНУ/Линукс оперативни системи једнако квалитетни као и Виндоуз за ову намену рачунара, већ у многим сегментима су чак квалитетнији. Мање оптерећују хардвер, лакше се повезује са интернетом, отпоран је на малициозни софтвер за Виндоуз, није оптерећен антивирусним софтвером, лакше манипулише великом количином мултимедијалног садржаја, природно је предодређен да буде вишекориснички систем, боље чува приватност корисника, преинсталиран је сав потребан апликативни софтвер одмах по инсталацији система на рачунар, није потребно инсталирати покретачке програме (енг. *drivers*) и тако даље.

Иако свака ГНУ/Линукс дистрибуција може да задовољи ове потребе корисника, за ову намену ми препоручујемо дистрибуције које су једноставне за инсталацију, поседују графичке алате за подешавање система и велику базу апликативног софтера у својој ризници која се лако претражује и инсталира нови софтвер. За такве системе смо склони да кажемо да су то дистрибуције за почетнике. Заправо, ако корисник нема веће захтеве од оперативног система, ове дистрибуције могу да буду коначни избор и за врло искусне кориснике.

Ове критеријуме првенствено испуњавају Убунту и његови деривати, Дебијан и његови деривати, а за нешто искусније кориснике ГНУ/Линукса препоручујемо, такође, Арч Линукс и његове деривате. Конкретно, ако би препоручивали

## Ослобађање

дистрибуције за десктоп по критеријуму популарности по Дистровочу (енг. *Distrowatch*), убедљиво би на првом месту био Линукс Минт, а затим Убунту и његови деривати - Елементери (*elementery*) Линукс, Зорин, *LXLE*, Дипен (*Deepin*), Боди (*Bodhi*) Линукс, затим Манцаро (енг. *Manjaro* - дериват Арча), Дебијан, и на крају Арч Линукс.

Препоруке су поређане по популарности и од пријатељски настројених ка компликованијим.

Посебна препорука за почетнике је **Линукс Минт**. За старије рачунаре препоручујемо: **LXLE**, **Боди**, **Елементери** и **Дипен**. За искусније којима је потребна повећана стабилност препоручујемо **Дебијан**. За искусне ентузијасте који воле све сами да подесе препоручујемо **Арч Линукс**.



### Гејмерске машине

То је посебна подгрупа десктоп машина које захтевају јак хардвер. У принципу, ГНУ/Линукс није прављен да буде платформа за игре, тако да у овој области





## У потрази за идеалном дистрибуцијом

доста каска за Виндоузом. У последње време, првенствено захваљујући компанији Валв (енг. *Valve*), која је природно портовала игре на ГНУ/Линукс стање, у овом сегменту употребе ГНУ/Линукс се поправља. Иако је могуће на ГНУ/Линуксу покренути неке игре за Виндоуз преко програма Вајн (енг. *Wine*), за играчку машину ћемо ипак сматрати ону која нуди добру подршку за Валв Стим (енг. *Valve Steam*). Пошто је Валв подршку за ГНУ/Линукс прилагодио првенствено Убунтуу, наша препорука за ову намену рачунара су Убунту и његови деривати.

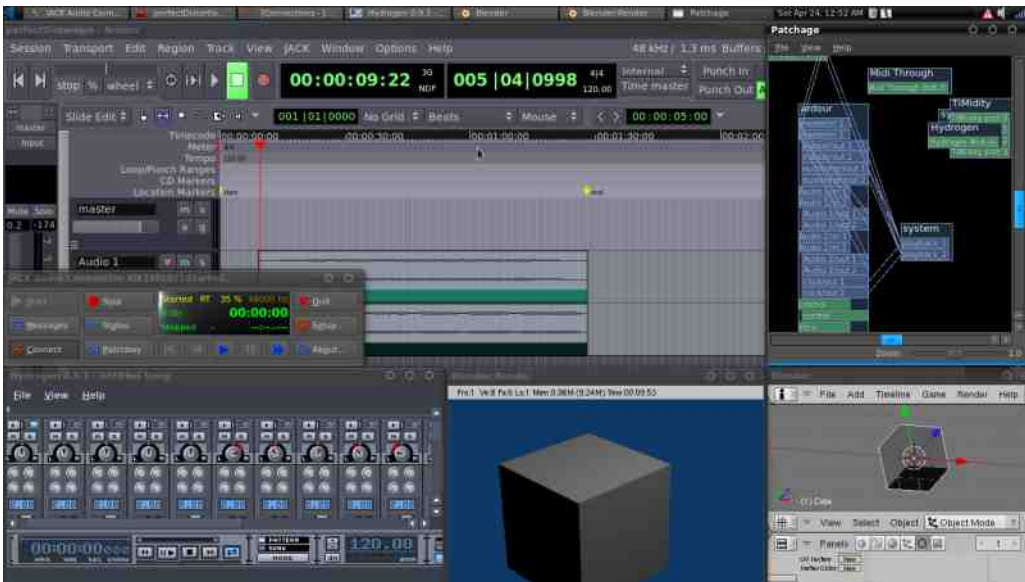


## Аудио и видео обрада

Ово је област која је суверено царство Епл Мек ОС десет (енг. *Apple Mac OS X*). То не значи да и у тој области ГНУ/Линукс нема добра решења. За ову намену постоји велики број апликација, намењених обради аудио и видео садржаја, од којих су најпознатије: Лајтворк (енг. *Lightworks* - власнички програм), Одесити (енг. *Audacity*), Ардур (енг. *Ardour*), и друге. Не само да постоје одговарајуће апликације, за ову намену су развијене и посебне ГНУ/Линукс дистрибуције. Овог пута вам препоручујемо три таква оперативна система: **Убунту студио** (енг. *Ubuntu Studio*), **Ка-икс-студио** (енг. *KXStudio*) и **Авелинукс** (енг. *AVLinux*).

## Ослобађање

Специфичност ових дистрибуција је да осим што имају прединсталиран сав потребан софтвер, оне су у потпуности оптимизоване за обраду видео и аудио материјала. Ови системи штеде своје ресурсе и употребљавају их максимално за процесе аудио и видео обраде.



## Системи за тестирање сигурности мрежа

Свака ГНУ/Линукс дистрибуција се може користити за тестирање сигурности мрежа ако се инсталирају одговарајуће апликације за тестирање. Међутим, постоји и посебно развијена дистрибуција која је у потпуности прилагођена за ову функцију. Реч је о **Кали Линуксу**. Кали је базиран на Убунтуу, поседује три стотине прединсталираних пенетрационих алата намењених тестирању мрежа. Ово га чини једним од најбољих безбедносно оријентисаних система на тржишту. Кали Линукс је намењен да симулира напад на мрежи, што инжињерима за безбедност омогућава да уоче небезбедна места и шта је потребно урадити да би се повећала сигурност важних података на мрежи.



## У потрази за идеалном дистрибуцијом



### Програмирање

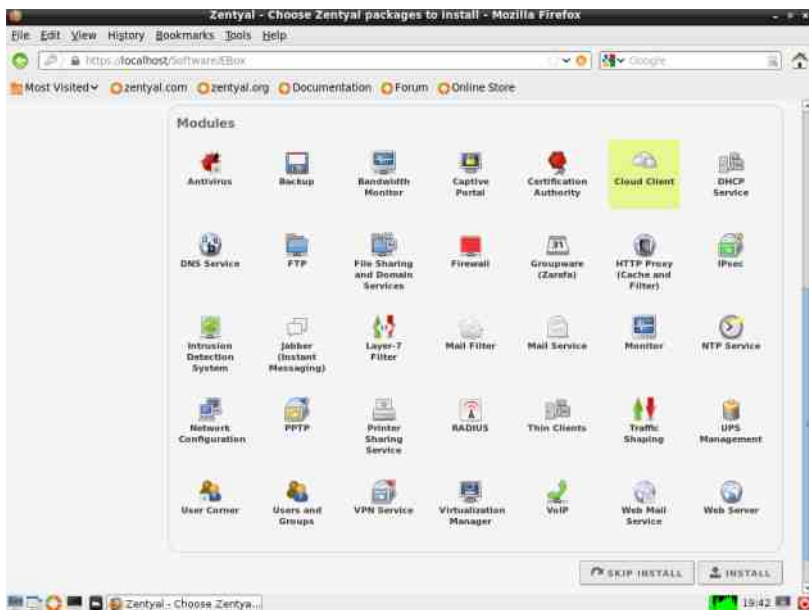
Одмах можете претпоставити да је оперативном систему који је отвореног кода, какав је ГНУ/Линукс, програмирање и развој софтвера једна од главних функција. Све дистрибуције Линукса су добро опремљене алатима за програмирање и развој софтвера. Ипак, програмерима је од великог значаја да раде на што свежијим алатима за програмирање. Алати се као и дистрибуције допуњавају временом новим функцијама које олакшавају програмерима рад. Због тога се за програмирање препоручују дистрибуције са цикличним надоградњама (енг. *rolling release*). Препоручујемо за ову примарну функционалност по критеријуму популарности: **Опенсусе** (енг. *OpenSUSE*), **Арч Линукс**, **Сабајон** (енг. *Sabayon*) **Линукс** и **Центу**.

### Сервери за мала предузећа

За већа предузећа и компаније ГНУ/Линукс нуди комерцијална решења као што су СУСЕ Линукс ентерпрајз и Ред хет ентерпрајз Линукс. Оно што се наплаћује код тих дистрибуција јесу подршка, одржавање, обука и слично. Поред комерцијалних решења за мања предузећа, ГНУ/Линукс нуди две бесплатне

## Ослобађање

дистрибуције намењене предузећима којима би плаћено одржавање било превелики издатак. Реч је о **Центосу** (енг. *CentOS*), бесплатној верзија Ред хет ентерпрајз Линукса, без плаћене професионалне подршке, као и **Зентајл**, Убунту базираном серверу који представља одличну замену за Виндоузов сервер.



## Решавање проблема у оперативним системима



Ово су наменске ГНУ/Линукс „живе” дистрибуције са прединсталираним дијагностичким софтвером и намењене дијагностици хардвера и спасавању вредних података са система који је у проблему. Може да се користи и на системима који су под Виндоузом. ГНУ/Линукс системе који су у проблему ове дистрибуције могу да опораве, док Виндоуз системе може хардверски да дијагностикује, опорави хардвер и спаси вредне податке. За ову сврху препоручујемо **Нопикс** (енг. *KNOPPIX*) „живи” CD (или USB) са прединсталираним великим бројем дијагностичког софтвера. За озбиљнију



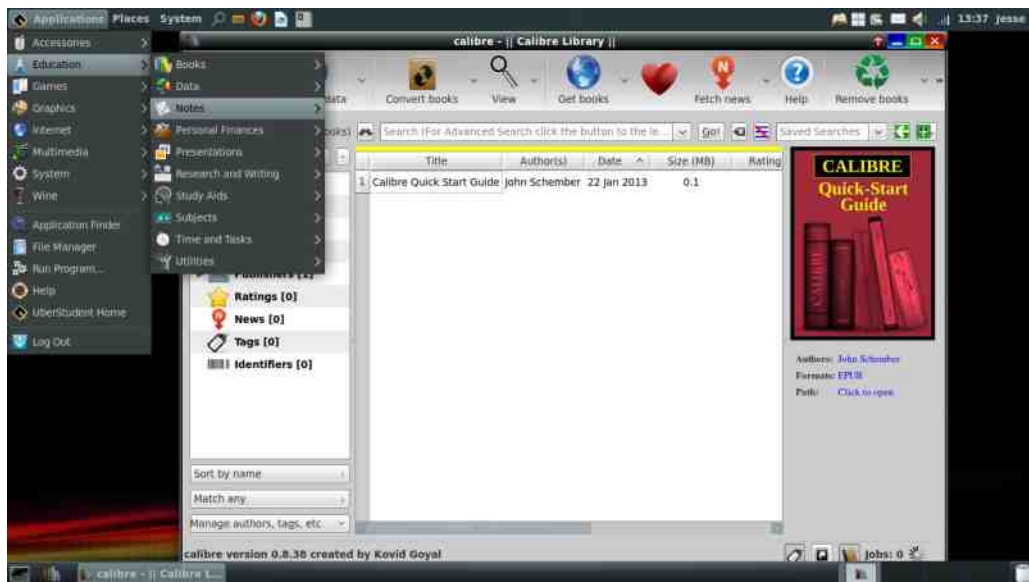
## У потрази за идеалном дистрибуцијом

дијагностику хардвера (од меморије, преко процесора и периферних уређаја) препоручујемо **Ултимејт бут ЦД** (енг. *Ultimate Boot CD*). Овај алат је одличан за опоравак и спасавање података са оштећених партиција тврдог диска. Вредан пажње је и **Систем рескју ЦД** (енг. *SystemRescueCD*) који поседује много алата усмерених на опоравак система и података.

### Образовање

ГНУ/Линук не запоставља ни образовање. Постоји мноштво слободних апликација намењених образовању деце од предшколског узраста, преко основношколског и средњошколског узраста, па све до студената. Неке дистрибуције су се фокусирале управо на овај образаони софтвер.

Овом приликом можемо да препоручимо **Едубунту** (енг. *Edubuntu*), Убунту верзију намењену деци предшколског и основношколског узраста. За нешто старије ту је **Уберстудент** (енг. *Uberstudent*) намењен образовању средњошколаца и студената.

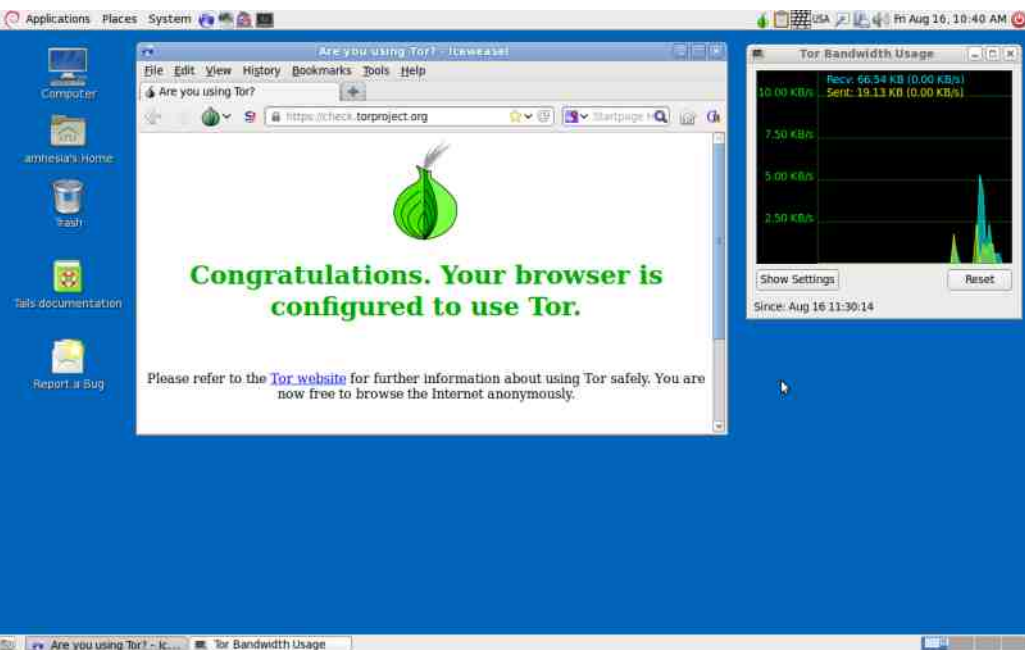


### Анонимност на мрежи

Још бисмо поменули дистрибуцију која се максимално труди да заштити

## Ослобађање

приватност свог корисника. За ову употребу препоручујемо Дебијанов дериват **Тејлс** (енг. *Tails*). Ова дистрибуција се труди да не оставља траг присуства свог корисника на интернету. Користи криптографију за шифровање свих података који напуштају систем и одлазе на мрежу како би заштитио приватност корисника и поверљивост његове комуникације.



## За крај

Ово је само део могућих примарних функција рачунара за које ГНУ/Линукс већ има припремљена решења. Има још дистрибуција са специфичном наменом као што је **Клонзила** (енг. *Clonezilla*) о којој смо већ писали, а која служи искључиво за прављење резервних копија података са тврдог диска. Ретке су функције које нису покривене решењима у слободном софтверу. Неке од њих су прецизно цртање (CAD - *dwg* формат је власнички), добра подршка за штампу (недостатак подршке за СМУК палету боја - власнички формат) и сличне. Све те функције су углавном условљене коришћењем неких општеприхваћених формата за одређену област коришћења рачунара, а који су заштићени власничким лиценцама.



## HIBERNATE ORM (1. део)

**Аутор:** Дејан Чугаљ

Један патент, који је пријављен још 1884. године, оставио је јединствен печат и начин пројектовања *IT* апликација садашњим и будућим нараштајима. Херман Холерит (*Herman Holerith*) патентирао је начин аутоматске обраде података пописа америчког становништва, што је била претеча савременог управљања базама података. То је био патент за који у то време нико није могао ни наслутити да ће тек, скоро седамдесет пет година касније, доживети прави процват – појавом првих рачунара. Модерне базе података су се популаризовале доласком модела исплативих рачунара шездесетих година 20. века, који су се првенствено користили за складиштење података великих компанија и пројеката. У самом почетку развоја база података примењивани су системи засновани на датотекама.

Седамдесетих година 20. века базе података добијају комерцијалну вредност појавом првих система за управљање базама података. Сложене структуре података великих компанија и пројеката су нам дале мрежни и хијерархијски модел као прве системе управљања базама података. Такве базе података су биле троме при извршавању захтева и красила их је велика сложеност приликом измена логичких целина у организацији података. Научним радом објављеним 1970. године Едгар Франк „Тед“ Код (*Edgar Frank „Ted“ Codd*) ( <http://goo.gl/12EqFp> ) је увидео мане постојећег модела базе података и своје решење проблема је засновао на математичким темељима, теоријом скупова и предикатском логиком.

## Слободни професионалац

### SQL



Као основна идеја и замисао научног рада Е. Ф. Кода је омогућавање техничко необразованим корисницима смештање, претраживање и манипулацију великим количинама података. Наиме, Код је замислио систем у коме корисници приступају подацима у базама података коришћењем команди сличним речима енглеског језика, док су сами подаци смештени у табелама.

Наравно да горе споменуте „команде” нису дошле саме по себи, већ је компанија *IBM* почела да развија систем за управљање базама података, који су назвали „*System R*”. Истраживачка група која је радила на овом пројекту је осмислила *SQL* (енг. *Structured Query English Language*) који је касније скраћен у *SQL* (енг. *Structured Query Language*). Овај језик се показао довољно моћним и практичним, тако да је 1978/1979. године почела и његова прва комерцијална примена како у књиговодственим организацијама, тако и у аеронаутичким компанијама.

Након ових примена језик из *IBM*-а прелази у *ANSI* (енг. *American National Standards Institution*) 1986. године, која га прихвата као језик за рад са релационим базама података. Полазећи од постојећих имплементација дефинисан је *IBM* стандардом (X3.135, <http://goo.gl/J3S6qQ>) којим се и одређује какав би *SQL* требало да буде. Године 1987. га такође прихвата и Међународна организација за стандарде - *ISO* (енг. *International Standards Organization*).

### Апликативни приступ повезивања са базама података

Експанзија развоја апликативног софтвера деведесетих година 20. века указала је потребу за једноставним начином повезивања и комуницирања са базама података. Некомпатибилност, нескалабилност и неинтероперабилност су довеле до тога да су ту привилегију комуницирања са базама података имали до тада само најiskusнији програмери, и то уз огроман уложен труд и време. Како стандардизација на свим пољима доприноси продуктивности, тако је и у информационим технологијама стандардизација *SQL* језика довела до појаве једног од првих *API*-ја (енг. *Application Programming Interface*), под називом *ODBC*





(енг. *Open DataBase Connectivity*), који је развио Мајкрософт у сарадњи са „Симба технологијом“ (енг. *Simba Technologies*) 1992. године.

Овај програмерски интерфејс је први пут омогућио да програмери-клијенти имају могућност употребе основних операција над подацима унутар база података из самих апликација тако што се постављају упити над базама података користећи се сада већ стандардизованим *SQL* језиком. Мада је ово омогућило лакши, бржи, поузданији приступ подацима у базама података – остали су неки од проблема који су се појављивали и пре увођења *ODBC*-а, а то су компатибилност између различитих система, трајност података, неподударност парадигми итд.

## Неподударност парадигми

Тежећи ка бољем објашњењу проблема неподударности парадигми, најбоље је да прво укратко објаснимо основне разлике објектне и релационе парадигме, и на који начин свака од њих руководи и управља подацима.

## Трајност података

Једна од основних идеја увођења и коришћења база података је управо у трајности података (енг. *persistence*), и ефикасном приступу истим. Другим речима, подаци једноставно морају да надживе било коју апликацију која их је створила. Многе сложене апликације, као што су информациони системи, били би потпуно бескорисни уколико подаци не би остали трајни након „истека рока трајања апликације“, или уколико се апликација замени новом верзијом. Сами подаци у апликацијама нису трајни, већ привремени (енг. *transient*). Потреба за очувањем података на основу којих свака апликација функционише, захтева и одређен систем који управља истим - и то су, у највећем броју случајева данашњих решења, релациони модели, тј. релационе базе података.

Друга страна „медаље“, која је засјала, је увођење *OOP* (енг. *Object Oriented Programming*) модела програмирања, који је скоро заменио структурне језике, али је исто тако довела програмере-клијенте у положај да скоро трећина написаног кода се управо односи на приступ, управљање и ажурирања података у базама преко *SQL*-а и *DBC*-а. Ови проблеми су постали у толикој мери изражени да би се могло рећи да, ако се сагледају ови статистички подаци продуктивности, *OOP* парадигма губи сваки смисао.

## Слободни професионалац

Смисао је, такође, изгубљен и у самој идеји *ООР* програмирања, тј. да се сви подаци којима се манипулише у апликацијама представљају преко објеката. Пошто се као систем за управљање подацима користи релациони модел, који само прима примитивне типове података, управо то представља највећу неподударност парадигми.

### Релациона парадигма

Релациона парадигма је заснована на математичким принципима, теорији скупова и предикатној логици. Релације имају моћне операторе сродне природном језику, док се манипулација подацима заснива на релационој алгебри. Тежњом ка што бољим перформансама и правилнијем уписом података, табеле се обично нормализују; програмерским жаргоном речено - врши се декомпозиција. Укратко, ако је нешто засновано на математичкој основи, готово са сигурношћу можемо тврдити да ће он као такав остати исти, у било ком „временском оквиру“, и то је један од разлога зашто су се релациони модели одржали у овом експоненцијалном развоју информационог технологија. Такође, можемо из овог закључити да је ово једна од парадигми која, барем за сада остаје, и доказано одлично обавља посао који јој је намењен.

### Објектно оријентисана парадигма

У *ООР* програмирању све своди на објекте. Објекти имају своје стање, понашање и својства. То је идеја која се искристалисала и дефинисала пут којим се одвија планирање, имплементација и израда једне апликације. Према објектној парадигми, два објекта су иста ако и само ако су им референце на објекат исте, тј. два објекта су једнака ако имају иста стања поља којима су описани. Пример из стварног света би отприлике био да, рецимо, имамо овцу и њеног клона. Са становишта ДНК анализе, то су идентични објекти, међутим то су две различите инстанце једног објекта (нпр. немају исто понашање). *ООР* као за свој главни циљ дефинише једноставност руковања подацима у самој апликацији.

### Могуће решење неподударности

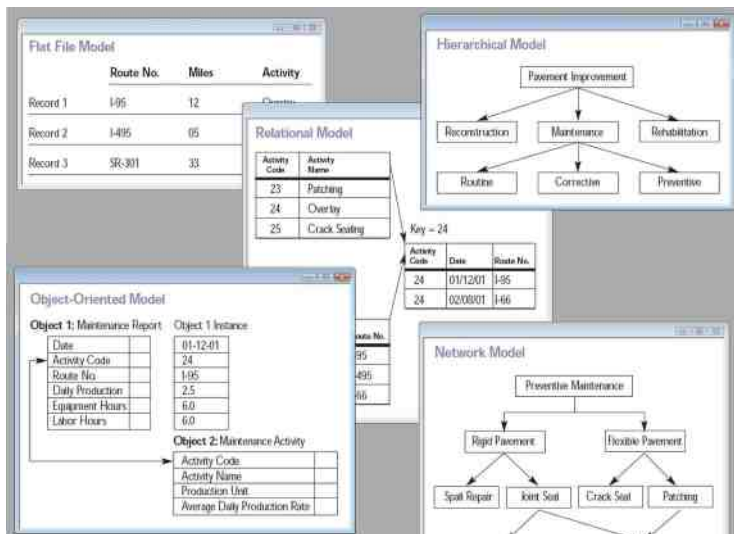
Релациони модели су засновани на математичким односима и широко су прихваћени. Уз сав могућ напредак информационог технологија, оне ће суштински радити исто. Упоређивањем ових парадигми лако се може увидети да



су разлике огромне и да се разилазе по самим основама, али то је тренутно најбољи модел који имамо, и треба да коегзистира са осталим моделима, па тако и са *OOP* парадигмом.

Како се проблем продубљује, релациони модел је добар у апликацијама које нису превелике ни сувише компликоване. Ако се ово узме у обзир, то доводи до проблема непродуктивности, трошења времена и комерцијалне неисплативости развијања апликација програмера-клијената у већим пројектним захтевима. Осим тога, одбојност ка новим технологијама још је више изражена ако се односе на већ устаљена и добро имплементирана решења, која за собом носе и стручњаке које су их савладали (овде се мисли на релациони модел, који се толико „одоматио“ да је постао „слепа“ водиља које се држе постојећи администратори база података и „оштро“ одбијају да се суоче са неким, можда „супериорнијим“, моделима и решењима, која се по самој својој природи намећу).

Тренутно, решење које се нуди је *OPM* (енг. *Object Relation Mapping*), који *OOP* програмере и администраторе база података с једне, и *OOP* парадигму и релациону парадигму с друге стране, спаја у својству међуслоја, те задовољава и „ублажава јаз“ неподударajuћих парадигми.



Наставиће се.

## Енкриптована електронска пошта (5. део)

**Аутор:** Петар Симиовић

Пошто смо сад већ анонимни и-мејл хакери, не само да знамо да шифрујемо и потписујемо пошту користећи криптографију, асиметричним јавним и тајним кључевима, већ можемо да сакријемо и идентитет пошиљаоца поруке помоћу анонимне мреже римејлера, сличне Тору (*Tor*). У овом последњем (петом) делу даћемо сигурносне савете и прећи ћемо, рецимо, теоријски део. Упутићемо кориснике у разлике између доброг и лошег сигурносног софтвера, разјаснићемо кључне речи на које треба обратити пажњу приликом одлучивања за одређено сигурносно решење, бар кад је у питању сигурна размена поште. Циљ овог прилога је да се корисници донекле оспособе да сами донесу исправну одлуку при избору сигурносног софтвера, како међу онима који тренутно постоје тако и међу неким који ће се тек појавити. Међутим, како ће се увек наћи неки софтвер који није поменут, корисно је знати како се најчешће траже и оцењују сигурносне апликације блиске теми овог текста.



Почнимо од нових вести из Гуглове (*Google*) кухиње. Овај информатички гигант је недавно започео рад на апликацији званој Енд-Ту-Енд (енгл. *End-To-End*), која није ништа друго до још један додаток његовом веб претраживачу који помаже при креирању, употреби и управ-

љању асиметричним (*PGP*) кључевима. Ново је то да је програм сада пребачен на Гитхаб (*GitHub*) (другим речима, *open-source*, линк: <http://goo.gl/9tlyVL>) са <http://goo.gl/sSsZFc>, али је још увек у алфа фази развоја. Међутим, никада не треба веровати компанијама чији су основни циљеви економско-политичке



## Енкриптована електронска пошта

природе, па тако ни овој. Већ се у првим редовима описа апликације на Гитхабу помиње да је заснована на Јаваскрипту (*java-script*), што у безбедносном бодовању никако не може бити позитивно. Као друго, сама чињеница да се ради о екстензији за претраживач не иде у њену корист, јер безбедност саме апликације не зависи само од ње већ и од претраживача на чији рад се екстензија ослања.

Као што смо већ напоменули у претходним деловима (другом и трећем), онлајн програми, као и екстензије претраживача за креирање, употребу и чување јавних и тајних кључева, веома су лоша идеја - јер корисник нема контролу над њима када је у питању скривени бекдор (*backdoor*) који потајно некоме одаје ваш приватни кључ. Чак и када су њихови кодови доступни, овакве апликације треба, уколико је могуће, избегавати. Што се екстензија тиче, ситуација је мало блажа у односу на онлајн креирање и чување кључева - јер ако их треће лице (компанија) зна, онда они нису тајни. Само неки примери су већ поменути: Веб Пе-Ге (*WebPG*) (екстензија), Мејлвелоуп (*Mailvelope*) онлајн и сада Енд-Ту-Енд.

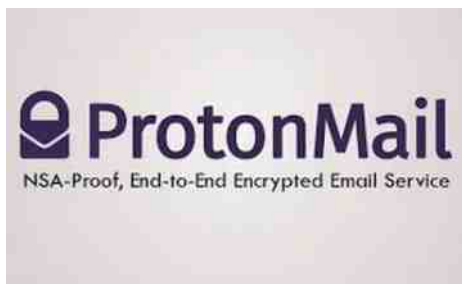
**1. Савет:** Избегавајте поверавање тајни апликацијама у „облаку” и програмима који раде само из веб претраживача.

**2. Савет:** Увек користите апликације са доступним изворним кодом. У овим апликацијама је теже да се нађе сигурносни пропуст, јер их може проверити свако ко поседује довољно знање, па их тако многи људи из целог света свакодневно испитују и поправљају.

Сада бисмо се осврнули на „сигурне” и-мејл сервисе новије генерације, о којима смо писали у првом броју, попут Протонмејла (*ProtonMail*) и досад још неспомињаног Скриптмејла (*Scriptmail*) који као главни адут сигурности истичу нечување лозинки корисника. Они на овај начин кориснике штите од својих запослених, а и себе и кориснике од националних безбедносно-информативних агенција које би желеле да лако добију шифре корисника. Сервиси овог типа не користе шифровање јавним и тајним кључем, већ корисник за сваку поруку задаје независну шифру, а затим добија линк до шифроване поруке која се налази на серверима тог сервиса и не напушта их. Тада корисник мора да примаоцу поруке пошаље линк преко кога може доћи до поруке као и до шифре за откључавање. Надамо се да сте већ сви уочили једну ману - особи којој пишете морате да саопштите кључ. Овде се поставља питање практичне примене оваквих метода, које су у суштини сигурне, ако сем ове мане преноса тајног кључа преко мреже заборавимо и злослутну јава скрипту који се обично врти у позадини или испод хаубе. Текст о сигурносном пропусту као и демо његове експлоатације која је

## Интернет мреже и комуникације

убрзо потом закрпљена, можете наћи овде: <http://goo.gl/kBEwn0> .



**3. Савет:** Овакви сервиси заиста нису сигурни, јер се заснивају на поверењу корисника према сервису да се порука отвореног текста (енг. *plaintext*) не чува код њих негде са стране. Такође постоји и проблем преноса и саопштавања тајног кључа примаоцу поруке. Ови сервиси су тек пионири у оваквој сигурносној техници, тако да није препоручљиво користити их за размене поверљивих информација.

Напредност сигурности и-мејл клијента: Приликом свакодневног рада, промовисања, едукације и разговора са људима о криптографији, испливао је веома реалан проблем у вези са недостатком заштите приватности личних података корисника и-мејла на рачунарима које дели више особа, односно уколико исти делите са још неким и обоје имате администраторске привилегије при заштитити садржаја ваших порука које сте примили од оне друге особе. Једно, привидно, решење је да направите и користите други (кориснички налог), јер ваши подаци и даље нису заштићени од администратора. Друга опција, мало тежа за спровођење, била би да направите шифровану партиципу у оквиру које ће функционисати ваш и-мејл налог. Међутим, постоји и боље, али не и савршено, решење: самом налогу се приступа преко шифре. Ову опцију је могуће укључити из Тандербирда (*Thunderbird*), ако одете на опцију *Edit » Preferences » Advanced » Config Editor... » (I will be careful!)* » онда укуцате *mail.password\_protect\_local\_cache* и двапут кликнете на ово подешавање да му се вредност промени у *True*, затим се подеси „*master password*” из *Edit » Preferences » Security » Passwords » Change master password* и унесете жељену шифру. Међутим, ово решење је могуће реализовати само ако један корисник користи Тандербирд на вишекорисничком рачунару.

**4. Савет:** Уколико желите висок ниво сигурности, најбоље би било да користите само свој рачунар.



## Енкриптована електронска пошта

У претходном прилогу смо помињали и *нут* римејлере који се практично ослањају на рад римејлера прве категорије, Сајберпанк римејлери (*Cyberpunk remailers*) и друге категорије као што је Миксмастер (*Mixmaster*), или функционишу по принципу јавне листе шифрованих порука. Подсетимо се укратко како они функционишу. Први тип представљају псеудонимни римејлери, или краће *Нут*, где је сервер, једноставно речено, мејл прокси адреса. *Нут* једноставно од пошиљаоца добијају његову и адресу примаоца поруке, а затим те податке бришу из поруке (како прималац не би сазнао стварну адресу и идентитет пошиљаоца) и замењују стварну адресу такозваном „псеудо-адресом”. Псеудо-адресу корисник (пошиљалац) сам региструје на *нут* серверу, као при регистрацији Ци мејл (*Gmail*) или Јаху (*Yahoo*) мејл адресе, а затим *нут* примљену поруку прослеђује примаоцу, са ваше *нут* адресе. Када прималац одговори на добијену поруку на вашу *нут* адресу, сервер је једноставно проследи на вашу стварну Ци мејл, Јаху или било коју другу адресу. Други тип је већ поменута листа на Гугл групи *alt.anonymous.messages*, где се људи једноставно дописују на једној листи шифрованим и потписаним порукама, а посматрач не може да зна ко пише поруку, коме је она намењена, нити који је садржај поруке. Једини проблем је техничке природе, а то је да ни корисници не знају која порука је намењена њима, а која не, па сваки пут кад желе да провере да ли им је неко писао морају да скину све поруке са листе и покушају да дешифрују једну по једну својим приватним кључем, а то уме да потраје. Група се налази на овој адреси <http://goo.gl/f3eNwA>, а програми за коришћење овог вида тајне комуникације су много бројнији за Виндоуз (*Windows*) него Гну Линукс (*Gnu/Linux*), нажалост.

**Напомена:** Сам *GPG-PGP* протокол није намењен анонимности корисника већ прикривању самог садржаја поруке. Адресе пошиљаоца и примаоца остају јавне, уколико не користите римејлере или *нут* сервисе.

Као што је назначено у напомени, *gpg* не сакрива идентитет већ само садржај поруке, па се не може сматрати софтвером за анонимизацију.

**5. Савет:** Користите *PFS* (енг. *Perfect Forward Secrecy*) протокол кад год је то могуће.

Иако *PFS* технологија није подржана у већини програма, могуће је имплементирати је, што није најпрактичније решење - јер би морали да за сваку поруку правите нове кључеве.

**6. Савет:** Користите мејл унутар анонимних мрежа.

Ово решење је најбоље и по питању анонимности и по питању сигурности. Цео

## Интернет мреже и комуникације

садржај је увек шифрован, рутира се кроз неколико рачунара и додатно меша са осталим подацима других корисника. На овај начин се избегавају различите напредне методе напада временске и количинске корелације саобраћаја како би се деанонимизовали корисници. Најпопуларније анонимне мреже су сигурно чувени Тор (енг. *Tor - The Onion Router*), са својим скривеним сервисима, па затим I2P (енг. *Invisible Internet Protocol*), са Сусимејлом (*Susimail*), и наравно Фринет (*Freenet*) са Фримејл (*Freemail*) сервисом. Овде намерно није поменут ниједан мејл сервис на Тору, јер их има много и најбоље је да сами потражите онај који би вам одговарао.

На Редиту (*Reddit*) постоји секција коју препоручујемо за примену шифрованог дописивања на овом сервису у секцији [/r/GPGpractice/](#).



Иако постоји озбиљан проблем са јавним серверима јавних кључева, све су чешћи покушаји појединаца да се и овај проблем реши или бар заобиђе. Један од њих је,





## Енкриптована електронска пошта

на пример, иницијатива на сајту [pgpasc.org](http://pgpasc.org) која има занимљиву идеју да појединци чувају своје јавне кључеве на својим серверима, на универзалном линку облика <https://url.com/pgp.asc>, како би се лако могли заменити новим кључем када то буде било потребно, јер јавни сервери јавних кључева не подржавају опцију брисања кључева. Наравно, ни овај пројекат није много бољи, јер корисници који желе да шифрују своје поруке не морају обавезно да имају и своје домене и сервере да би шифровали поруке. Али, позитивно је то што људи размишљају о томе како да реше постојећи проблем. Можда је будућност PGP-а у децентрализацији или у примени тзв. Блокчејн (*Blockchain*) технологије, на којој су засноване дигиталне валуте попут познатог Биткоина (*Bitcoin*).

GPG је, такође, недавно објавио нову верзију (од верзије 2.0 па надаље) са подршком за ECC (*Elliptic Curve Cryptography*), која би требало да пружа већи степен сигурности кључева исте дужине, него што би иначе, без ње.

Symmetric Key Size (bits)	RSA and Diffie-Hellman Key Size (bits)	Elliptic Curve Key Size (bits)
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

Table 1: NIST Recommended Key Sizes

Не можемо никако рећи да је будућност шифровања електронске поште мрачна, а и свест о сигурности је свакако подигнута од како је Сноуден изнео информације о интензивном шпијунирању безбедносних агенција - америчке NSA-е и британске GCHQ-а. Овде се завршава сага о шифровању електронске поште, али већ у следећем броју почињемо са прилозима о инстант месџинг (енг. *instant messaging*) шифровању.

## Сам свој мајстор

**L<sup>A</sup>T<sub>E</sub>X** презентација:

### **Beamer** (3. део)

**Аутор:** Никола Харди

У претходним бројевима сте могли да прочитате више о томе зашто волимо Бимер (енг. *Beamer*) и зашто ћете га вероватно и ви волети и радо користити. У претходном броју су већ представљени фрејмови, односно слајдови. Поред креирања слајдова, описан је и рад са математичким изразима, оно у чему је *LaTeX* (Латех)<sup>1</sup> најбољи. У овом броју ће бити настављено представљање основних функционалности Бимера. Најпре ћемо се подсетити употребе „домаћих” слова, а потом ће бити описано додавање насловне стране, руковање одељцима и садржајем, додавање илустрација и приказ кода.

### **LaTeX** и **UTF**

При раду на било којем садржају на нашем језику сусрећемо се са знаковима који нису обухваћени *ASCII* стандардом и због тога с времена на време морамо да се замислимо. У серијалу текстова о *LaTeX*-у је понуђено једно решење (погледајте ЛИБРЕ! 18). Ово је прави тренутак да представимо још један, можда и елегантнији начин.

Све што је потребно јесте да се на почетак документа дода следећи код:

```
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
```

---

<sup>1</sup>*LaTeX* се изговара Латех (/ˈlɑːtɛx/). То је скраћеница од Лампорт (Лесли Лампорт, амерички доктор рачунарских наука) и Тех (*TeX* - систем за обраду и прелом текста на рачунару).



Након овога би требало све магично да проради. Следи креирање правога слајда у облику насловне стране.

## Насловна страна

Један *LaTeX* документ може да има своје атрибуте као што су наслов, аутор, установа или институт и друго. На основу ових података се попуњава садржај заглавља на сваком слајду и може да се креира насловна страница. Следи пример.

Креирање насловне странице:

```
\title{Ovo je naslov}
\subtitle{Ovo je podnaslov \newline u dva reda}
\author[Nikola Hardi]{Twitter: @Atlantic777}
\institute{FTN - Fakultet Tehničkih Nauka - Novi Sad}

\begin{document}
  \begin{frame}
    \titlepage
  \end{frame}
\end{document}
```

## Одељци и садржај

Презентације су често подељене на одељке. Када нису подељене, треба их поделити (прим. аут.). Када желимо да означимо почетак новог одељка, за то можемо да користимо ознаке *section*, *subsection* и *subsubsection*, исто као у редовним *LaTeX* документима. То изгледа овако:

```
\section{Početak novog odeljka}
\begin{frame}
  \frametitle{Naslov slajda}
\end{frame}
```

За генерисање садржаја презентације унутар слајда се користи наредба `\tableofcontents`.

## Сам свој мајстор

### Илустрације

Илустрације је могуће укључити у садржај слајда на више начина. То је могуће урадити као у редовном *LaTeX* документу помоћу наредбе `\includegraphics` или помоћу додатних пакета, као што је *PGF*. Следи пример додавања илустрације помоћу наредбе `\includegraphics`:

```
\begin{frame}
  \frametitle{Илустрација}
  \includegraphics[height=\paperheight]{images/naziv_slike.png}
\end{frame}
```

### Кôд

Врло често постоји потреба за приказивањем кôда или посебно форматираног текста. У томе могу да помогну окружења *verbatim* и *lstlisting*. Овде постоје одређене значајније разлике у односу на уобичајене *LaTeX* документе.

Када се кôд додаје помоћу окружења *lstlisting*, потребно је укључити пакет *listings* и кôд дефинисати наредбом *defverbatim* пре употребе. Потом је тај кôд могуће додати на слајд. Ово је најједноставније илустровати примером.

```
\usepackage{listings}

\defverbatim[colored]\nazivkoda{
  \begin{lstlisting}
    int a = 5;
  \end{lstlisting}
}

\begin{frame}
  \nazivkoda
\end{frame}
```

Када се користи окружење *verbatim*, неопходно је то нагласити и при креирању тог слајда додавањем опције *containsverbatim*.

```
\begin{frame}[containsverbatim]
  \begin{verbatim}
```



```
Ovaj
    tekst je
    strogo formatiran.
\end{verbatim}
\end{frame}
```

## Теме

Иако је прављење нових тема за Бимер презентације врло сложено чак и за напредне кориснике, теме се често користе и важно је знати да их има много и да се лако бирају. Све што је потребно је на почетку документа изабрати тему наредбом **usetheme**. Занимљиво је споменути да су теме добијале називе по градовима у којима су одржавана презентације па су тако неке од најпознатијих тема назване Берлин (енг. *Berlin*) и Варшава (енг. *Warsaw*).

```
\usetheme{Berlin}
\usecolortheme{beaver}
```

Осим избора теме, можемо да изаберемо и шему боја помоћу наредбе **usecolortheme**. Могуће је изменити и боје појединачних елемената. За више детаља погледајте званично Бимерово упутство или неки од бројних туторијала.

Врло користан сајт за преглед тема за Бимер и шема боја је:  
<http://www.hartwork.org/beamer-theme-matrix/>

## За крај епизоде

За крај ове епизоде дајемо вам комплетан пример састављен од елемената које смо представили у овом броју:

```
\documentclass{beamer}

\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{listings}
\usetheme{Berlin}
\usecolortheme{beaver}
```

## Сам свој мајстор

```
\title{Ovo je naslov}
\subtitle{Ovo je podnaslov}
\newline u dva reda}
\author[Nikola Hardi]{Twitter:
@Atlantic777}
\institute{FTN - Fakultet
Техничких Наука - Novi Sad}

\begin{document}
  \begin{frame}
    \titlepage
  \end{frame}

  \begin{frame}
    \tableofcontents
  \end{frame}

  \section{Početak 1. odeljka}
  \begin{frame}
    \frametitle{Naslov slajda 1}

    \includegraphics[height=180px]{images/naziv_slike.png}
```

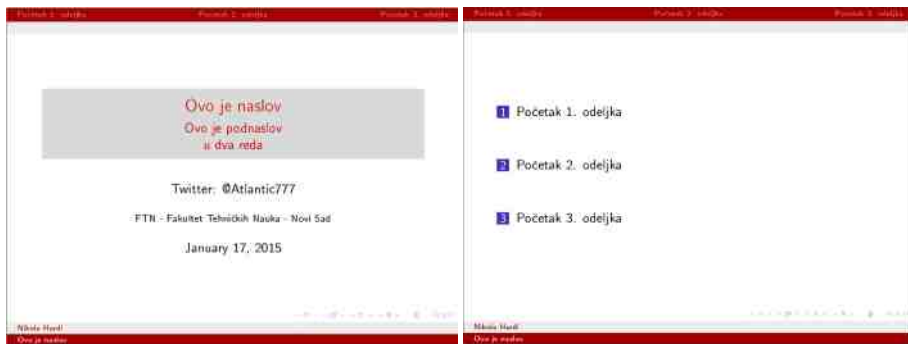
```
\end{frame}

\defverbatim[colored]\nazivkoda{
  \begin{lstlisting}
    int a = 5;
  \end{lstlisting}
}

\section{Početak 2. odeljka}
\begin{frame}
  \frametitle{Naslov slajda 2}
  \nazivkoda
\end{frame}

\section{Početak 3. odeljka}
\begin{frame}[containsverbatim]
  \frametitle{Naslov slajda 3}
  \begin{verbatim}
    Ovak
      tekst je
        strogo formatiran.
  \end{verbatim}
\end{frame}
\end{document}
```

Овај код ће дати следећи изглед презентацији:





Paralik 1. slika Paralik 2. slika Paralik 3. slika

Naslov slajda 1



**slika**

Naslov Hradi  
Ovaj je naslov

Paralik 1. slika Paralik 2. slika Paralik 3. slika

Naslov slajda 2

```
int a = 5;
```

Naslov Hradi  
Ovaj je naslov

Paralik 1. slika Paralik 2. slika Paralik 3. slika

Naslov slajda 3

```
Ovaj  
tekst je  
strog formatiran.
```

Naslov Hradi  
Ovaj je naslov

У следећем броју ћемо представити напредније елементе на слајдовима (табеле, набрајања и теореме), распоређивање елемената на слајду (рад са више колона), анимације и прелазе.

# L<sup>A</sup>T<sub>E</sub>X

**Хардвер****BeagleBone Black Rev C****Водич од првог дана (3. део) -  
подешавање IP адресе****Аутор:** Ненад Марјановић

Да би подесили фиксну IP (ај-пи) адресу потребно је познавање основа на пољу умрежавања рачунара. Како бисмо поједноставили процес, почнимо од термина „зида“ (енг. *wall*). Ако нам је професор информатике у школи био довољно добар, сигурно је објаснио да при конструкцији мреже у нашим стамбеним објектима увек треба

кренути од утичнице преко које имамо приступ интернету. Одатле и потиче метод „зид“. Ово у пракси значи следеће:

1. Познавање наше јавне IP адресе (нпр. 95.123.125.0)
2. Познавање наше локалне IP адресе (нпр. 192.168.0.1)

У Европи већина корисника за локалну IP адресу користе 192.168.0.1, такозвани локалхоуст (енг. *localhost*). Ова адреса заправо представља адресу вашег рутера, односно модема преко којег се обавља размена информација са уређајима у нашим домовима. Адреса зависи од самог интернет провајдера и опреме. У појединим случајевима ова вредност може бити промењена од стране корисника. Сваком уређају који се повеже са рутером, интегрисани IP - DHCP клијент даје на коришћење једну локалну IP адресу. Без обзира на подешавања, тај уређај никад не може користити адресу која је већ у употреби. Ако у овом случају прикључимо BBB Rev C са нашим рутером, у администрационом панелу рутера можемо видети да је и њему доделио једну локалну IP адресу, рецимо 192.168.0.10. За приступ рутеру користимо локалну IP адресу и корисничке податке које смо добили од провајдера. Време је да се позабавимо конфигурацијом. Ако смо већ укључили





рутер и BBB Rev C и записали адресу коју користи наш уређај потребно је извршити подешавање Дебијан (енг. *Debian*) оперативног система. За овај водич користимо неколико Линукс команди, као што су **ifconfig**, **ifup**, **nano**, **grep**, **cp**. Куцањем команде **ifconfig** добићемо резултате сличне следећим:

```
root@beaglebone:/# ifconfig

eth0      Link encap:Ethernet  HWaddr C8:A0:30:AB:32:3A
          inet addr:192.168.0.100  Bcast:192.168.0.255
          Mask:255.255.255.0

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host

usb0      Link encap:Ethernet  HWaddr A2:CB:A9:A8:0B:F4
          inet addr:192.168.7.2  Bcast:192.168.7.3  Mask:255.255.255.252
```

Сва три интерфејса *eth0*, *lo* и *usb0* имају одређену намену, али ми ћемо се позабавити параметрима *eth0* интерфејса. За овај корак потребно је приступити измени **/etc/network/interfaces** фајла. Користимо Нано (енг. *nano*) едитор, али ви можете користити и *Vi* (Ви-ај) едитор уколико сте навикли на њега.

Златно Линукс правило је чување копије фајла чији садржај мењамо:

```
cp /etc/network/interfaces /etc/network/interfaces.bak
```

Затим приступамо измени фајла коришћењем **nano** команде:

```
nano /etc/network/interfaces
```

У овом фајлу можемо видети различита подешавања, али ми ћемо обратити пажњу на *IP* и *MAC* (ем-еи-си) адресе нашег уређаја.

```
# The primary network interface
auto eth0
iface eth0 inet dhcp
# Example to keep MAC address between reboots
#hwaddress ether DE:AD:BE:EF:CA:FE
```

## Хардвер

Прво ћемо уметнути симбол # испред две линије које нећемо више користити, а онда на крају *interfaces* фајла додајемо:

```
auto eth0
iface eth0 inet static
    address 192.168.0.10
    netmask 255.255.255.0
    network 192.168.0.255
    gateway 192.168.0.1
```

Уколико нисте сигурни која је *IP* адреса и *MAC* адреса *eth0* интерфејса, користите следећу команду:

```
ifconfig | grep -w 'HWaddr\|inet addr' eth0      Link encap:Ethernet
HWaddr d0:39:72:3b:f1:44
    inet addr:192.168.0.10 Bcast:192.168.0.255
Mask:255.255.255.0
    inet addr:127.0.0.1 Mask:255.0.0.0

usb0      Link encap:Ethernet HWaddr 96:6f:36:10:10:45
    inet addr:192.168.7.2 Bcast:192.168.7.3 Mask:255.255.255.252
```

Остаје нам да додамо *BBB Rev C* *MAC* адресу у *interfaces* фајл:

```
hwaddress ether d0:39:72:3b:f1:44
```

Сачувајте фајл и унесите команду за покретање *eth0* интерфејса:

```
ifup eth0
```

Овим смо заокружили процес подешавања *BBB Rev C* уређаја и остаје нам повезивање односно пренос параметара на рутер. Ова процедура се разликује од произвођача до произвођача, тако да уколико имате проблема са моделом са којим располажете, пошаљите питање редакцији ЛиБРЕ! часописа или извршите консултацију документације коју сте добили при куповини рутера. Такође можете пронаћи већину објашњења за овај тип производа на интернету.



Пример подешавања:

Navigation: Général | WAN | DynDNS | DNS | **DHCP** | NAT | Route | Filtrage

### Serveur DHCP

Activation:  activé  désactivé

Première adresse: 192 . 168 . 0 . 5

Dernière adresse: 192 . 168 . 0 . 100

Bail (en secondes): 86400

Valider

### Adresses statiques

Adresse IP	Adresse MAC
192 . 168 . 0 . 10	d0 : 39 : 72 : 3b : f1 : 44

За сам крај можемо напоменути да су ова подешавања од великог значаја уколико желите да користите *BeagleBone Black Rev C* као веб сервер, *dns* (ди-ен-ес) сервер, Тор релеј (енг. *Tor relay*), за приступ и контролу уређаја са удаљеног рачунара и још много тога.

У следећем броју ЛиБРЕ! часописа писаћемо о подешавању Тор релеј сервера. Циљ овог чланка је упознавање са Тор мрежом и значај сваког корисника и администратора Тор система.

# ЛИБРЕ!

Часопис о слободном софтверу



Позива читаоце часописа, активисте и симпатизере слободног софтвера да дођу на нови Мамбл канал који је отворио ЛУГОНС на свом серверу.

Адреса канала је: [mumble.lugons.org](https://mumble.lugons.org) port 64738

Дођите да се дружимо!

