

Октобар 2012.

ЛИБРЕ!



Часопис о слободном софтверу

број

06



SLACKWARE

Историјат домаћих FLOSS заједница:

Slackware Србија

Представљамо:

Slackware 14



Ubuntu

18. октобар 2012.

Објављен је *Ubuntu*
12.10



Linux Mint

30. октобар 2012.

Отворен је *Linux*
Mint Store!



Creative Commons Ауторство-Некомерцијално-Делити под истим условима 3.0 Србија.

Нови број доноси

Дочекали смо и нову верзију *Slackware* линукса. То је појава која се не дешава тако често. Предходно издање, *Slackware 13.37*, објављено је скоро тачно пре годину и по дана (27. априла 2011.). Екипа ЛиБРЕ! часописа је имала среће да већ у 6. броју има рецензију и представљање ове дистрибуције у најновијем издању 14.

Slackware је сигурно једна од легендарних и једна од најстаријих дистрибуција. Прва верзија ове дистрибуције је објављена још 17. јула 1993 године. Вршњак је са *Debian* пројектом. За разлику од *Debian*-а који своју филозофију заснива на апсолутној стабилности, *Slackware* доноси идеалан однос између нових технологија и традиције са једне стране (*current* грана) али не занемарује и оне којима је неопходна апсолутна стабилност (*stable* грана). Он није дистрибуција за почетнике али је идеалан за корак даље у линукс свету. Конципиран је тако да натера свог корисника да научи више али није толико компли-

кован да би заинтересовани корисник одустао од будућег учења и истраживања. Уз мало труда корисник ће кроз коришћење ове дистрибуције боље схватити свој систем и боље га прилагодити својим потребама него што би то урадио у некој другој дистрибуцији која је више намењена почетницима као што су то *Ubuntu* или *Mint*.

Објављивање новог *Slackware 14* је и повод да у овом броју представимо и малу али врло активну заједницу *Slackware* Србија. Захваљујући њима сви који се одлуче на авантуру звану *Slackware* ће имати одличну помоћ и подршку па читав процес прилагођавања новој филозофији неће бити нимало трауматичан. Заједница егзистира већ 7 година и за то време је прикупљена завидна количина знања и документације који су на располагању свима.

Од новитета од овог броја вам нудимо серијал о серверским технологијама. Одлучили смо да за основу узмемо *Debian server* као један од најкоришћенијих (доста тога се може применити и на *Ubuntu server*). У овом броју почињемо са уводом.

Примамо све више писама са

питањем како се можете придружити и писати за часопис. Има више начина али један од најједноставнијих је да напишете текст за часопис и пошаљете на нашу адресу електронске поште libre@lugos.org. У прилогу додајте информације о вашој сфери интересовања (специјалности) и намеру да ли желите да се придружите као сталан или повремени сарадник часописа. Хвала за интересовање и радујемо се унапред сарадњи.

До читања
ЛиБРЕ! тим

Моћ слободног
софтвера:



Број: 05

Периодика излажења: месечник

Главни и одговорни уредник:
Никола Харди

Извршни уредник:
Александар Станисављевић

Главни лектор:
Жељко Шарић

Лектура:
Александар Станисављевић
Ромео Млинар
Владимир Попадић
Жељко Павков

Редакција:
Бојан Богдановић
Горан Мекић
Гаврило Продановић
Марко Матошевић
Стефан Ножинић
Жељко Попивода
Михајло Богдановић
Мишо Јовановић
Владимир Цицовић
Марко Костић
Милутин Гавриловић

Графичка обрада:
Златан Васовић
Дејан Маглов

Контакт:
IRC: #floss-magazin
на irc.freenode.org

Е-пошта:
libre@lugons.org

ЛиБРЕ! вести

стр. 6

Вести



Пулс слободе

стр. 8

Активизам:

Промоција Ubuntu 12.10
GNU/Linux оперативног
система у Зајечару

стр. 8



Историјат FLOSS заједница:

Slackware Србија заједница

стр. 12

Представљато

стр. 15

Slackware 14

стр. 15



Крајем септембра је изашла нова верзија Slackware линукса. Нова верзија носи ознаку 14.0 и доноси ажуриране верзије програма, неколико нових пакета и уобичајену стабилност. Slackware 14.0 је доступан за x86, x86_64 и

ARM платформе...

Како да?

стр. 20

Мала школа:

Scribus 1.4 (5. део)

стр. 20



Администрација Линукс система:

Праћење перформанси
система (2. део)

стр. 24

Сваки рачунар на свету има свог администратора система. Између осталих дужности, администратор треба да има увид у рад свог система. У наредном делу овог текста видећемо неколико основних наредби и алатки за надгледање и праћење рада и заузећа дискова и процесора.

Ослобађање

стр. 30

Conky -
Систем монитор

стр. 30

Слободни професионалац

стр. 35

Geogebra

стр. 35

Математика је јако занимљива наука али није редак случај да понекад немамо времена да решавамо нешто банално. Поента је усредсредити се на нешто комплексније а онај банални део који често замара и тера људе да одустану од проблема, решава наш

рачунар. Овај програм, који представљамо има баш ту намену.

Интернет, преже и комуникације стр. 37

Cyber криминал – (4. део) стр. 37

Настављамо серијал представљањем типова и примера cyber криминала.

**Криптографија/
Криптологија - (1. део)** стр. 40

Сервери стр. 45



Web Сервер (1. део) стр. 45

Сам свој тајстор стр. 48

**Мултимедија:
GNU/Linux DAW (5. део)** стр. 48

**Графика:
Фото уређивање и графички
дизајн на Линуксу (2. део):
Растерска графика и GIMP** стр. 50

У прошлом броју смо Вам препоручили слободне програме за све области графичког дизајна и уређивање фото-графија. У овом броју ћемо се бавити растерском графиком...

Тобилни кутак стр. 53

APK стр. 53



Забавне стране стр. 55



Ubuntu

октобар 2012.



Објављен је *Ubuntu 12.10* са опцијом резултата претраге са *Amazon-a*, *desktop remote* пријављивања...
(18. октобар)

Одабрано је кодно име за *Ubuntu 13.04* → *Raring Ringtail*.
(18. октобар)

Подршка за *Ubuntu 11.04*, *Natty Narwal*, прву верзију Ubuntu-а са *Unity* окружењем радне површи, је истекла 28. октобра 2012. *Ubuntu 11.04* се у финалној верзији појавио 28. априла 2011.
(28. октобар)

Корисни линкови:

<http://www.omgubuntu.co.uk/2012/10/ubuntu-12-10-released>

<http://www.omgubuntu.co.uk/2012/10/ubuntu-13-04-named-raring-ringtail>

<http://www.omgubuntu.co.uk/2012/10/ubuntu-11-04-support-ends-today>

Instagram

27. септембар 2012.



Instagram

Instagram desktop клијент за линукс под називом „*Friendly Photos*“ је у припреми. Аутор је најавио да ће апликација имати подједнако леп (а можда и лепши?) кориснички интерфејс.

Користан линк:

<http://www.omgubuntu.co.uk/2012/10/desktop-instagram-app-being-made-for-linux>

Linux Mint

30. октобар 2012.



Отворен је *Linux Mint Store*! Ова продавница ће имати све што и стара која се звала „*Partners*“ уз додатак новог партнера - *ThinkPenguin-a*. *Linux Mint 14* ће бити

објављен крајем новембра.

Користан линк:

<http://blog.linuxmint.com/?p=2194>

VirtualBox

31. октобар 2012.



VirtualBox је виртуализер опште намене за x86 хардвер који је намењен за сервере, десктоп рачунаре али и за *embedded-овану* употребу. *VirtualBox 4.2.4*.

81684 је најновија верзија која је објављена.

Користан линк:

<http://www.itsvet.com/arhiva/2012-10-31>

Firefox

9. октобар 2012.



Firefox 16.0.2 са собом доноси велики број нових карактеристика које обезбеђују брже, сигурније и прилагодљивије искуство када је реч о претраживању *Web-a*. *Firefox Smart Location bar* који је познат и

као *Awesome Bar* прилагођава се корисницима и њиховом начину коришћења и на тај начин обезбеђује информације које су у већој мери прилагођене сваком кориснику. Најновије издање популарног Web претраживача које је објављено носи ознаку *Firefox 16.0.2* и доноси закрпе за раније примећене слабе тачке али и одређени број других, минорних унапређења.

Користан линк:

<http://www.itsvet.com/arhiva/2012-10-31>

Thunderbird

9. октобар 2012.



Thunderbird 16.0.2 је одличан е-mail клијент који стиже од истог оног тима који стоји иза *Firefox* претраживача. *Thunderbird* пружа *IMAP/POP* подршку, уграђени

RSS читач, подршку за *HTML mail*, моћну претрагу, напредно филтрирање порука, могућност груписања порука, опцију управљања више-струким е-mail и newsgroup налога, између осталог. *Thunderbird* одликује и могућност архивирања слична оној која постоји у *Gmail-y*.

Користан линк:

<http://www.itsvet.com/arhiva/2012-11-01>

SeaMonkey

9. октобар 2012.



SeaMonkey је Web претраживач, напредни е-mail и newsgroup клијент, *IRC chat* клијент као и *HTML* едитор.

Најновија верзија која је објављена носи ознаку *SeaMonkey 2.13.2* и оно што је одликује јесте велики број сигурносних закрпа.

Користан линк:

<http://www.itsvet.com/arhiva/2012-10-31>

ЛИБРЕ! пријатељи



Активизат:

Промоција Ubuntu 12.10 GNU/Linux оперативног система у Зајечару

Аутори: Павле Радивојевић и Милан Рајчић

У организацији Омладинског центра Зајечар и *Ubuntu* заједнице Србије а на иницијативу Милана Рајчића (*nihil_enochian*), Павла Радивојевића (*Lateralus*) и Милана Бошковића (*mibol*), 27. октобра 2012. године, одржана је по други пут промоција најпопуларније дистрибуције *GNU/Linux* оперативног система *Ubuntu 12.10*. У прелепом простору Омладинског Центра Зајечар, заинтересованим посетиоцима, предности *Ubuntu*-а представили су чланови из врха *Ubuntu* заједнице Србије Бојан Богдановић (*bojše*), Ладислав Урошевић (*urosš*) и Срђан Удовички (*Rainmaker*).



За разлику од претходне, која је била



оријентисана на коришћење линукс оперативног система у пословним окружењима, на овој промоцији пажња је била усмерена на кућне кориснике као и на практично упознавање са новом верзијом *Ubuntu*-а и другим десктоп окружењима. У кратком уводу Бојан Богдановић је изнео главне предности и карактеристике линукс оперативних система у глобалу, да би Ладислав Урошевић у наставку предавања детаљно описао интерфејс најновије верзије Убунту 12.10 оперативног система као и нове технологије које нам доноси ова верзија. У јако интересантном излагању Урош је нагласио предност *keyboard-shortcut*-ова и *Unity* интерфејса због рапидног напредовања технологије која нам доноси све мање екране осетљиве на додир. Наравно било је речи о неизоставној оријентацији *Ubuntu* оперативног система ка интернету, где нам сам интерфејс омогућава да са два или

три клика пронађемо музику, садржаје или локације које одговарају нашим интересовањима. Занимљиву завршницу предавања дао је Срђан Удовички који је представио различита десктоп окружења која је могуће лако инсталирати у постојећи систем и тако потпуно променити изглед радног окружења које додатно можемо модификовати по сопственим потребама. Било је речи о *XFCE*, *LXDE* и *GNOME classic (GNOME fallback)* десктоп окружењима, као и изузетно “лаганом” по питању ресурса менаџеру прозора *E17 – Enlightenment*, којим можете лако свој давно напуштени стари рачунар вратити у живот.



По завршетку предавања, предавачи су били отворени за питања стидљиве али знатижељне публике а она су била углавном у вези са *dual boot*-ом и сличним питањима која највише интересују будуће кориснике линукс оперативних система. Заинтересованим посетиоцима су на располагању били дискови са дистрибуцијама

Ubuntu, *Kubuntu*, *Xubuntu* и *Lubuntu* у верзији 12.10, и мноштво налепница и тракица Убунту заједнице Србије. Иако су овог пута посетиоци били у нешто мањем броју (око тридесетак), за шта је мало била заслужна и киша која је падала скоро целог дана, није засметало да предавање прође у најбољем могућем реду.



У неформалном дружењу са предавачима, представници зајечарске линукс заједнице ЗаЛин, два Милана и Павле, разговарали су о организацији за будуће презентације, промоције и ширење свести о *Free* и *Open Source Software*-у у Зајечару, Тимочној крајини и другим мањим градовима и локалним заједницама.





ba

Prvi LUGoNS BarCamp održaće se u s

Pozivamo Vas na BarCamp, otvoreni skup sa radionicama i druženjem čiji sadržaj i proces će učesnici osmisliti na sastanku, a razvijati u nastavku. BarCamp služi za razmenu za diskusije, učenje i stvaranje, a proizvod tog rada može se videti već krajem manifestacije.

events.lugons.org • www.lugons.org



LUGONS barcamp

subotu 01. decembra u Novom Sadu

ma, predavanjima

a početku samog

menjivanje ideja,

ože delimično da

Ulaz je SLOBODAN!

Subota, 01. decembar 2012.

od 12.00 – 20.00h

Lokacija:

Hotel Aurora

Beogradski kej 49b

21000 Novi Sad

Web: www.hotelauroora.rs

LUGONS
GROUP OF NOVI SAD

Историјат домаћих FLOSS заједница:

Slackware Србија

Srpska Slackware zajednica



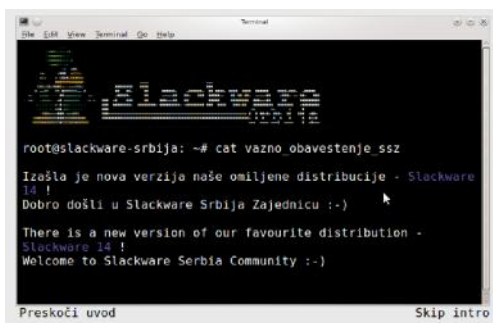
Аутор: Slackware Србија заједница

Негде у лето 2006. млади Димитрије Спасојевић, пун ентузијазма и гладан нових изазова оснива *Slackware Србија заједницу*. Тих година, Димитрије дипломира на факултету Информационих Технологија на универзитету Метрополитан у Београду али још у средњој школи налази своју љубав према линуксу и отвореном коду (енгл. *Open Source*). Као и сваки линукс корисник авантуристичког духа, креће да истражује линукс дистрибуције, од *Mandriva-e*, *Ubuntu-a*, *Debian-a* долази до *Slackware-a* где проналази оно што је тражио. Тада се рађа љубав према *Slackware-у* која је касније пренета и на заједницу.

Тешка времена су била за све тадашње *GNU/Linux* дистрибуције, па и за *Slackware*. Требало је подесити ту фамозну *dial-up* интернет конекцију из *Open Source* управљачких програма, јер је то представљало највећу препреку за повезивање на интернет и размену искустава са другим корисницима *GNU/Linux-a*. Након преласка ове препреке, креће потрага за истомишљеницима по српском сајбер простору. Прво упознаје *Hatebreeder-a* (Синишу Бобића) и са њим заједно креће у оснивање *Slackware Србија заједнице*. Убрзо се заједници при-

дружује Младен Исаковић који је већ имао доста знања о *Slackware* линуксу и тада заједница креће „пуном паром“ да ради на порталу и форуму.

Нешто касније, што због финансија што



због приватних послова (студије, породица...) долази до смањене активности чланова ССЗ и стагнације заједнице. У наредне две – три године заједница се одржала захваљујући ентузијазму Бранка (Бранко Грубић) који је у то веома тешко време за заједницу био један од најактивнијих чланова. Такво стање је трајало до краја 2011. године када је свим члановима био послат допис да је заједница пред гашењем и објављена је понуда да је неко преузме.

Међу члановима који су примили допис су била и два тадашња и један бивши уредник линукс подфорума на *MyCity*

форуму. Тамо започиње отворена дискусија како спасити *Slackware Србија заједницу*. Предраг Дамњановић Пеца, власник и администратор *MyCity* форума излази са великодушном понудом која се данас ретко среће. Наиме, Пеца нуди хостинг и домен од свом трошку и тако спасава једну од најстаријих заједница *GNU/Linux* корисника у Србији.

Bocke (Бојан Поповић), *Drummer* (Раде Јекић) и моја маленкост *Broker* (Михајло Богдановић) смо већ били чланови *Slackware Србија заједнице*, те преузимамо администрацију заједнице. Од старе екипе ту су били Млађа, (Младен Исаковић) који нас је сачекао, упутио, једном речју дао нам „кључеве“ за врата *Slackware Србија заједнице* и Димитрије, захваљујући коме је домен безбрижно пренет на нови хостинг.



У том тренутку је форум био прилично успаван и требало је доста тога урадити. У свој тој несрећи било је и среће. *Bocke* (Бојан Поповић) и *Drummer* (Раде Јекић) су били већ искусни *Slackware* корисници. *Bocke*, дугогодишњи корисник линукса, програмер аматер и *Drummer*, стручњак за хардвер и такође дугогодишњи *Slackware* корисник, удахнули су нови живот

у заједницу. Моја маленкост је била задужена за маркетинг или како би се то рекло војничким жаргоном "човек за везу". Убрзо се администрацији заједнице придружио и *Qki* (Александар Кукић), један од активнијих чланова заједнице у њеној прошлости.



Кренула је миграција заједнице на нови хостинг. Требало је реди-зајнирати постојећи форум и портал, формирати Вики странице, позабавити се са документацијом актуелне верзије *Slackware-a*, подићи *IRC* канал, урадити логотип заједнице, формирати *YouTube* канал и још много малих али не небитних ствари је ваљало или уредити или формирати, да би се на крају на порталу појавило обавештење, „Поново ради биоскоп“. То обавештење означава нову „страницу“ у историји *Slackware Србија заједнице*.

Пошто је *Slackware* јако захвална дистрибуција за детаљна подешавања и измене у складу са сопственим жељама, на форуму је формиран и одељак за наше *SlackBuild* пројекте. Овај део форума је под контролном палицом *Bocketa*, *Drummer-a* и *Qki-ja*. На заједници постоји неколико *Slack-Build* пакета за које се надамо да ће корисницима *Slackware-a* олакшати инсталацију програма који није до-

ступан из неких од најпопуларнијих *Slackware* репозиторијума.

Не би било правично не поменути уреднике на форуму, порталу и Вики страницама заједнице. Уредници су мотор заједнице. *Slackmuz* (Петар Милојевић) марљиво уређује Вики странице, *Stereo* (Жељко Шарић) такође вредно уређује портал заједнице. Ту је и *Sinisa* (Синиша Дробац) који се бави дизајном и већ је допринео са неколико занимљивих позадинских слика.

Поред набројаних ствари заједница узима учешће у свим актуелним дешавањима на *GNU/Linux* сцени у Србији. Прво појављивање представника обновљене заједнице се десило у Новом Саду у марту месецу, са нашим другарима из *LUGoNS*-а, потом скуп корисника и љубитеља линукса у Сава Центру у априлу месецу, па предавање на *Linux Day 2.0.* у Високој *ITC* школи у мају месецу и предавање на недавно одржаном Дану Слободе софтвера у септембру месецу.

Све ове активности су одјекнуле и ван граница Балкана. Поменути смо на Берлиосу (постављена је веза до CC3-а) као прилично активна заједница и перспективан пројекат у *Slackware* екосистему <http://slackworld.berlios.de/> што је велика част али и обавеза за будући рад.

Планови за будућност?

Не би пуно да откривамо, има их пуно, иде се полако али сигурно. Пројекат наших *SlackBuild* скрипти улази у нову фазу и у некој скорој будућности ће бити жртва мале реорганизације. Ту је и мала школа основе *C* програмског језика. Планирана је и ревизија

документације за *Slackware 14.0* који је изашао крајем септембра.

Slackware Србија Заједница је отворен и гостољубив интернет кутак. Уколико желите да нам се придружите и заједно са нама заједницу учините бољом, позивамо Вас да посетите прву станицу у *Slackware* авантури - наш форум на адреси <http://slackware-srbija.org/forum>. Одатле можете наставити на наш Вики који се налази на адреси <http://slackware-srbija.org/wiki/doku.php>. Када Бас читање документације и учење мало заморе, свратите и на наш *IRC* канал на *Freenode.org* - *#slackware-srb*. Ту ће Вас дочекати занимљива и увек расположена екипа линукс ентузијаста.

Корисни линкови:

- [1] <http://slackware-srbija.org/>
- [2] <http://slackware-srbija.org/forum/>
- [3] <http://slackware-srbija.org/wiki/doku.php>
- [4] <http://www.mycity.rs/Linux/Bocketu-Drummeru-Linuxo-i-ostalim-Slack-ovcima.html>
- [5] <http://slackworld.berlios.de/>
- [6] <http://www.slackware.com/>
- [7] <http://www.linuxquestions.org/questions/slackware-14/>

Slackware 14



Аутор: Бојан Поповић

Крајем септембра је изашла нова верзија *Slackware* линукса. Нова верзија носи ознаку 14.0 и доноси ажуриране верзије програма, неколико нових пакета и уобичајену стабилност. *Slackware* 14.0 је доступан за x86, x86_64 и ARM платформе.

Slackware 14 долази са више десктоп окружења. Примарно је *KDE*, којем је и посвећено највише пажње. *KDE* се налази у верзији 4.8.5 и долази са комплетним скупом програма за свакодневне потребе. Ту је такође и *Xfce*, којем је у овом издању посвећено мало више пажње него у претходним. *Xfce* пакети су сада добили свој директоријум на инсталационом DVD-у. За разлику од претходног када је *Xfce* био упакован у једном пакету,

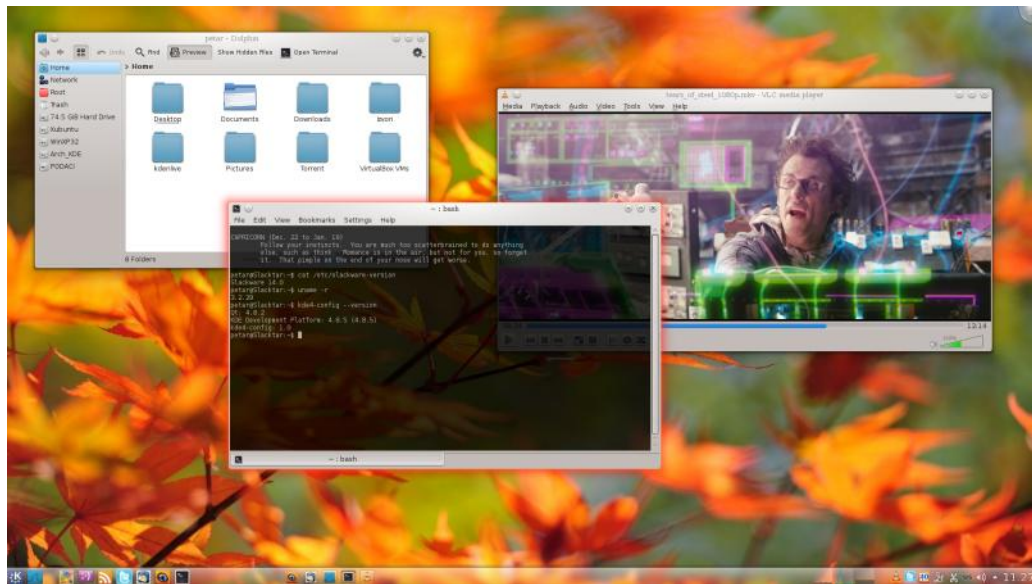
сада су компоненте упаковане одвојено. Новина за кориснике *Xfce*-а је и скок са 4.6 на верзију 4.10, као и укључивање неколико додатних аплета за *Xfce* панел. Од осталих, ту су *Fluxbox* 1.3.2, *Fvwm* 2.6.5 и *Window-Maker* 0.95.3. За носталгичаре, ту је и последња верзија *Blackbox*-а (0.70.1).

Од X11 десктоп програма, *Slackware* 14.0 доноси најновије верзије *Mozilla Firefox* (15.0.1), *Thunderbird* (15.0.1) и *Seamonkey* (2.12.1) пакета. Од интернет програма ту су и *Pidgin* (2.10.6), *Xchat* (2.8.8), *Pan usenet* читач (0.139), *gftp* (2.0.19) и стандардни скуп *KDE* интернет програма (*Konqueror*, *Kopete*, *Akgregator*, *Kmail*, *Knode*, *Blogilo*, *Ktorrent*, *Kget*,...). Ту је и *Gimp* у верзији 2.8.2. А укључен је и *Xpaint* у верзији 2.8.16, као и програми за преглед графичких датотека: *Geeqie* (1.1), *Gwenview* (2.8.5) и класични *XB* (3.10a)

са *Jumbo* закрпом. Укључено канцеларијско окружење је *Calligra* 2.4.3 (некадашњи *KOffice*). *Calligra* укључује стандардни скуп канцеларијских програма, као што су: *Calligra Words* -

- *Xmms* (1.2.11),
- *Aumix* (2.9.Amarok (2.6.0),

и стандардни скуп *KDE* мултимедијалних програма (*KAudioCreator*, *KPlayer*, *Juke*).



програм за обраду текста, *Calligra Sheets* - програм за табеларна израчунавања (*spreadsheets*), *Calligra Stage* - програм за прављење и приказ презентација, *Kexi* - програм за управљање базама података, *Plan* - програм за планирање и управљање пројектима, *Braindump* - програм за прављење мапа ума, *Calligra Flow* - програм за прављење дијаграма, као и неколико ситних алата. *Calligra* такође укључује и програме за обраду графике. За растерску графику ту је *Krita*, за векторску *Karbon*. Од мултимедијалних програма, ту су:

- *Audacious* (3.3.1),
- *Amarok* (2.6.0),
- *MPlayer* (1.1_20120701),
- *Dragon Player* (2.0),
- *K3B* (2.0.2), класични

Од *X11* едитора, укључени су:

- *Emacs* (24.2.1),
- *Vim* (7.3.645), као и *KDE*-ов
- *Kate* едитор.

Од програма за преглед докумената ту су:

- *Okular*,
- *Xpdf*,
- *GV* и
- *Xdvi*.

Slackware долази и са великим избором конзолних програма. То укључује и чак неколико едитора: поред горе поменутих (*Emacs*, *Vim*), ту су и

- *Jed* (0.99_19),
- *Joe* (3.7),
- *Jove* (4.16.0.73) и
- *Nano* (2.3.1).

Kao подразумевани *vi* едитор користи се *Elvis* (2.2) у минималној конфигурацији. *Slackware* укључује и неколико конзолних програма за пуштање аудио датотека:

- *moc* (2.5.0a4),
- *mpg123* (1.14.4),
- *Madplay* (0.15.26),
- класични *Amp* (0.7.6),
- програм за слушање интернет радио станица - *Soma* (2.8.5) и
- *Worboce* аудио *CD* плејер (2.40).

Укључена су и два аудио миксера: *au-mix* (2.9.1) и *Rexima* (1.4). Ту је и швајцарски нож за аудио датотеке - *Sox*, у верзији 14.4.0. Kao и *sc* (7.16) -

клијенти е-поште: *(Re-)Alpine* (2.02), *Elm* (2.5.8) и *Mutt* (1.5.21). *IRC* клијенти:

- *Epic5* (1.1.3) и
- *Irssi* (0.8.15).

Usenet читачи:

- *nn* (6.7.3),
- *slrn* (0.9.9.pl1), *trn* (3.6) и *tin* (1.8.3).

Ftp клијенти:

- *lftp* (4.3.8) и
- *ncftp* (3.2.5).

Програми за претраживање интернета:

- *Links* (2.7) и
- *Lynx* (2.8.7).



конзолни *spreadsheet* програм, *Midnight Commander* (4.8.4) - познати *GNU* клон *Norton Commander*-а и *seejpeg* - програм за преглед графичких датотека за линукс *framebuffer*.

Slackware доноси и велики избор конзолних интернет програма. Ту су

Kao и *mcabber* (0.10.1) - конзолни *Jabber* клијент и *Snownews* (1.5.12) - конзолни *RSS* читач.

Kад су у питању развојни алати новина је и укључивање *LLBM* компајлера са *Clang* фронтендом и *Glade* интерфејс дизајнера у основну дистрибуцију. *GCC* скуп компајлера је сада ажуриран

на верзију 4.7.1, *Python* је ажуриран на 2.7.3, *Perl* је у верзији 5.16.1, *Ruby* у верзији 1.9.3p194, *Tcl* у верзији 8.15.1, *Guile* у верзији 1.8.8, а *Common Lisp* у верзији 2.49. Ту су и различити системи за контролу верзија: *GNU RCS*, *CVS*, *SVN*, *GIT* и *Mercurial*. Као и *NASM*, *YASM*, *GNU* и *BSD make* и стандардни скуп *GNU* развојних алата.

Slackware 14.0 долази са *Apache web* сервером у верзији 2.4.3, *MySQL* 5.5.27 и *PHP* 5.4.7. Укључено је и неколико сервера е-поште: *Sendmail* 8.14.5, *Openwall Popa3d* 1.0.2, *Re-Alpine* (некадашњи *UW*) и *Imapd* 2.0.2. Ту су и



два *ftp* сервера: *ProFTPD* 1.3.4a и *vsftpd* 3.0.2. *Slackware* доноси и велики број мрежних алата за дијагностику (*nmap*, *iptraf-ng*, *netcat*, *mtr*,...), употребу и сервирање садржаја на различитим врстама мрежа (*SMB*, *NFS*, *Bluetooth*, *Wireless*,...). Једна од важних новина је и додатак *NetworkManager*-а који се може једноставно инсталирати и укључити избором у менију током инсталације.

Као што се види из горњих редова, *Slackware* 14.0 доноси пун скуп пакета

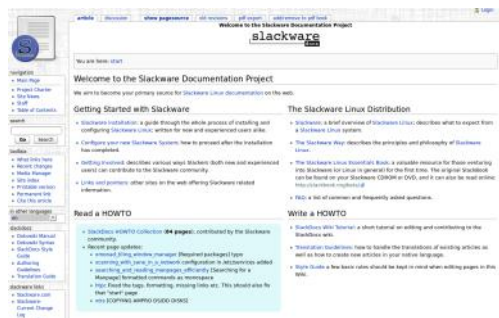
за десктоп, развојну или серверску употребу. Корисници који нису задовољни подразумеваним избором, неће наћи додатне пакете у званичном репозиторијуму - *Slackware DVD* садржи целокупну дистрибуцију. Додатни пакети су доступни из спољних извора. Препоручени извор додатног софтвера је интернет локација <http://slackbuilds.org/> (*SBo*). <http://slackbuilds.org/> је репозиторијум скрипти за изградњу додатног софтвера. *SBo* репозиторијум је доступан преко *rsync*, *GIT*-а, *ftp* или *http*.

Такође постоје и спољни алати за laku инсталацију додатних програма користећи *SBo* скрипте. Најпопуларнији и најкоришћенији у *Slackware* заједници је *sbopkg* <http://www.sbopkg.org/>. *Sbopkg* је могуће користити као командни или дијалог интерфејс за *SBo* репозиторијум, подржава задавање редоследа изградње и на основу задатих параметара аутоматски изграђује и инсталира пакете. Редоследи изградње (енгл. *queues*) се могу посматрати као неки вид управљања зависностима при изградњи програма. Mario Giachero одржава репозиторијум редоследа који је недавно ажуриран и за *Slackware* 14.0. Адресу за преузимање можете наћи на локацији <http://www.sbopkg.org/queues.php/>

У верзији 14.0, *SBo* репозиторијум је претрпео неколико мањих промена. Најзначајнија промена је додавање поља *requires* (захтева). Поље *requires* означава потребне пакете који нису укључени у основну *Slackware* дистрибуцију. Већ постоји неколико алата који користе ову могућност, од којих је најзанимљивији *sbodeps* (<http://www.iqius.org/sbodeps/>). *Sbodeps* је једноставан алат који је у могућности да аутоматски преузме и

изгради пакет укључујући све спољне зависности. Такође се може користити и само за једноставно излистивање редоследа изградње.

Оба поменута алата аутоматски изграђују и инсталирају пакете из изворног кода. Корисничка интервенција је минимална. Али изградња већих пакета може захтевати доста времена или преоптеретити рачунар. Без бриге, постоје и репозиторијуми бинарних пакета. Често се на



страницама чланова основног *Slackware* тима могу наћи додатни или ажурнији пакети. Тако *Eric Hameleers* (*AlienBOB*) и *Robby Workman* (*rworkman*) одржавају већи број пакета које је могуће једноставно инсталирати на постојећи систем. Ерик, међу осталим, одржава и ажурне дистрибуције *LibreOffice*-а и *FFmpeg*, док је са Робијеве странице могуће преузети *TexLive* и додатне *Xfce* програме и аплете. Ерик такође одржава блог који често садржи и опште новости везане за правац развоја и новотарије у *Slackware* линуксу. Постоје и други репозиторијуми бинарних пакета који овом приликом неће бити поменути.

Slackware добија редовне сигурносне закрпе које можете инсталирати користећи *slackpkg*, алат који је укључен у *Slackware* од верзије 12.2.

Ако користите развојну верзију - *Slackware Current*, препоручено је да пратите промене у *Changelogu*. Она су доступна и у облику *RSS Feed*-а (<http://dev.slackware.it/rss/>).

Slackware је добио и нову званичну локацију са документацијом - *Slackware Documentation Project* (<http://docs.slackware.com/>). *Slackware Documentation Project* је пројекат покренут тик пред издавање верзије 14.0. Покренули су га *Niki Kovacs* и *Eric Hameleers*. *Slackware Documentation Project* садржи велики број чланака који су раније били раштрани широм интернета (*Slack Wiki*, *AlienBOB wiki*, *Slackware Essentials*,...), као и неколико оригиналних чланaka. *Slackware Documentation Project* је доступан у неколико језика, али то у овом тренутку не укључује српски.

Slackware је дистрибуција коју би свако требало да проба када стекне одређени ниво знања и пожели да крене даље. Захваљујући једноставности имплементације, администрације и прилагођавања, рад на *Slackware*-у пружа детаљнији увид у то како *GNU/Linux* систем у ствари функционише. *Slackware* се труди да кориснику не стоји на путу или намеће решења специфична за дистрибуцију или када то чини, оставља и могућност да се поступи и другачије. Са друге стране, *Slackware* доноси најбољи могући баланс између модерних технологија и традиције и познат је као једна од дистрибуција најближих оригиналној *Unix* филозофији.

Мала школа: **Scribus 1.4**

(5. део)

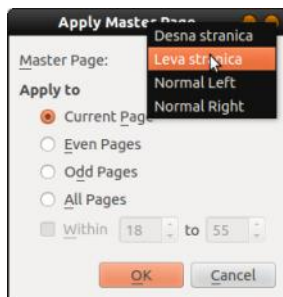


Аутор: Дејан Маглов

У прошлом броју креирали смо и сачували мастер странице за леву и десну страницу нашег *PDF* часописа. Сада идемо корак даље и примењујемо ове мастер странице на постојеће, већ отворене, другу и трећу страницу, а после ћемо применити и на све нове странице које будемо креирали.

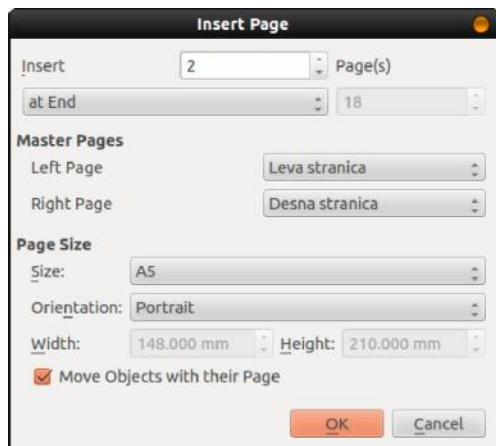
Са друге и треће странице обрисаћемо векторску линију коју смо нацртали у циљу креирања мастер странице и обрисаћемо текст и број странице из подножја странице. Уместо тога применићемо снимљену мастер страницу и то Леву на левој страници а Десну на десној страници. Најједноставнији начин за примену мастер странице на страници је кликом десним тастером миша на страницу и бирање из дијалога *Apply Master*

Page... У дијалогу бирамо жељену мастер страницу и мету примене. Као мета може се изабрати тренутно активна страница, све непарне странице, све парне странице или све странице од-до. Једноставним избором мастер странице (Леве или Десне коју смо креирали), избором *Current page* (тренутне активне странице) и кликом на дугме *OK* применићемо мастер страницу на странице, прво на страници 2 а затим на исти начин и на страници 3.



Додаћемо сада и две нове странице (4 и 5) и одмах при отварању применити мастер странице. Идемо на падајући мени *Page* → *Insert* и отвориће се нови дијалог. У том дијалогу прво треба уписати колико страница желимо да отворимо, у нашем случају две. После тога где желимо да отворимо нове странице. Опције су пре тренутно активне странице (*before Page*), после тренутно активне странице (*after Page*) или на крају документа (*at End*). Ми ћемо изабрати да се отворе на крају документа. Нас још занима да на тим страницама применимо мастер странице и бираћемо наше креиране мастер странице Леву за леве странице и Десну за десне странице. Поред овог постоји могућност да те странице промене формат али то нас не занима пошто желимо у целом документу да имамо исти формат страница па ћемо

само кликнути на *OK* да би креирали наше нове странице.



Приметићемо да је изглед страница 2 и 3 сада исти као и пре брисања оних елемената који су ушли у састав мастер странице али са том разликом да им је сада немогуће приступити и поново едитовати. Мастер страница се понаша као специјалан *layer*. Пре него што наставимо да мало појаснимо појам *layer-a*. У графичким програмима *layer* означава слој неког цртежа. Најчешће се тумачи и пореди са прозирним фолијама које постављамо на цртеж и затим цртамо по њима. Свака фолија се уређује за себе али се увек, осим ако другачије не наредимо, виде све фолије и то у редоследу по нивоима које смо одредили. Наравно да горњи *layer-i* имају приоритет и елементи графике из горњих *layer-a* заклањају елементе у доњим *layer-ima* па се мора водити рачуна о њиховом распореду. У *Scribus-u* као и у свим другим графичким програмима по отварању новог документа креиран је само један *layer*, он се у *Scribus-u* зове *Background* и видљив је у навигационој линији.

Мастер страница се понаша као *layer* који се налази испод прозирног *Background layer-a*. Једина разлика у односу на обичан *layer* је та што се не појављује у падајућем избору *layer-a* на навигационој линији. Једини начин да се приступи мастер страници и на њој нешто измени је да се отвори *editor* мастер странице. То се ради тако што одемо у падајући мени *Edit → Master Pages...*

Постављање главног текст оквира на страницу

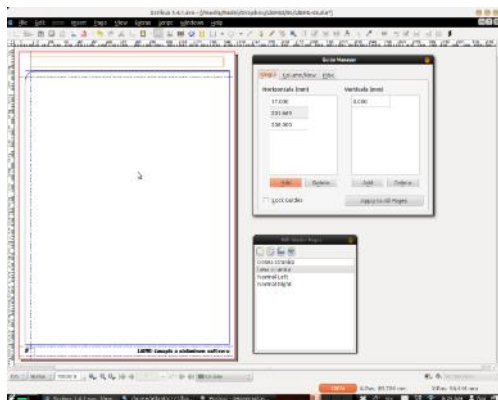
Свака књига, брошура, часопис има своју форму странице а у њој доминира текст оквир. Сви остали елементи су подређени тексту. Постоје различите методе слагања текста на страници. У књигама доминира текст у једној монолитној колони. За разлику од књига, часописи обично текст слажу у више ужих колона јер по неким истраживањима то доприноси бржем читању. Колоне у часопису су толико широке да око може једним погледом да обухвати цео ред без померања очне јабучице тако да читалац часописа несвесно само помера поглед у правцу горе-доле и тако добија на брзини читања.

Ово је главни разлог зашто и наш часопис треба да има 2 колоне иако је А5 релативно узан формат. Текст оквир ће и у нашем часопису бити доминантан елемент и обухватаће читав простор у главном делу странице који је ограничен маргинама. Имамо само један проблем а то је, да смо по линији маргине нацртали векторску линију раздвајања. Због тога ћемо морати направити две вођице које ће одвојити текст оквир за по 4 милиметра од линија раздвајања. Да не би имали

пуно водећих линија које би нам после сметале правићемо различите водеће линије за леве странице и посебно за десне странице. Користићемо особину мастер странице да водеће линије које су креиране тамо се појављују и активне су такође на свим страницама на којој је та мастер страница примењена. То значи да можемо нашу Леву мастер страницу да допунимо са водећим линијама које су тачно одмакнуте од векторских линија за 4 мм, то јесте вертикално на позицији 8 мм (4 мм до маргине + 4 мм одвајања) и хоризонтално на позицији 17 мм (13 мм до маргине + 4 мм одвајања). Десну мастер страницу допуњавамо водећим линијама вертикално на позицији 140 мм (148 мм ширина A5 – 4 мм до маргине – 4 мм одвајања) и хоризонтално на позицији 17 мм (13 мм до маргине + 4 мм одвајања).

Да подсетимо још једном на поступак:

1. отворимо едитор мастер странице (*Edit → Master Page... → Лева (Десна) страница*)
2. отворимо едитор водећих линија (*Page → Manage Guides...*)
3. додајемо (дугме *Add*) хоризонталне вођице на 177 мм и вертикалне 8 мм (Лева мастер страница) односно 140 мм (Десна мастер страница).

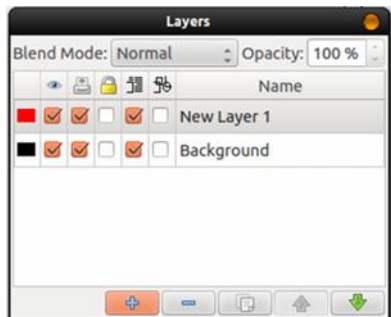


4. За завршетак само затворимо дијалог *Manage Guides* и на крају и дијалог *Master Pages*.

Приметићете да су се водеће линије појавиле на свим страницама на којима су мастер странице примењене. Сад је једноставно постављати текст оквире унутар постављених вођица са једне стране и маргина са друге стране. Сад је време да и то урадимо.

Пре него што креирамо наш текстуални оквир направићемо један нови *layer* само за то. То је добро јер онда *Background layer* можемо да користимо за креирање позадина испод убаченог текста и друге сличне ефекте. Дијалог за креирање *layer-a* је у падајућем менију *Windows → Layers* или једноставно *F6*. Једноставним кликом на „+“ додаје се нови *layer* са именом „*New Layer 1*“. Двокликом на име *layer-a* добијамо могућност да променимо име и ми ћемо то променити у „Текст“. Овде се врше и фина подешавања особина *layer-a*. Једноставним потврђивањем поља можете подесити да *layer* буде видљив или не, да буде штампан или не, можете га закључати да не би случајно направили нежељене промене на њему, можете подесити да текст из нижих *layer-a* „плута“ око елемената из тог *layer-a* (нарочито корисно ако *layer* садржи слике или табеле, онда ће текстови из нижих *layer-a* окруживати ове елементе и неће бити покривен) и на крају једна специфична особина да елементи из тог *layer-a* буду јасно означени обојеном оквирном линијом (нарочито корисно ако су у питању линкови, који се приказују као оквир области у којима су активни. Као такви нису превише уочљиви али ако је ова опција потврђена овај оквир ће бити означен јасном упадљивом бојом). Сада може-

мо да затворимо дијалог *Layers* и кренемо са креирањем текст оквира.



Проверимо у навигационој линији да ли је „Текст“ *layer* активан, бирамо алат за текст оквир из линије алата и једноставно кликом (леви тастер миша) у пресеку постављених вођица (горњи леви угао на Левој страници), развлачењем (држање тастера миша) до угла маргина (доњи десни угао на Левој страници) и пуштања тастера миша креирамо нови текст оквир.

Подешавање текстуалног оквира



Да би наш текст био у две колоне потребно је уредити креирани текст оквир. То се ради помоћу *Properties* дијалога који смо користили и у прошлом наставку мале школе. Тада смо га користили за форматирање текста а сада ћемо да применимо његове опције *Text* → *Columns & Text Distances*. Ту ћемо одабрати 2 за број колоне и 5 мм размака

између колоне (*Gap*). За сада остала подешавања одвајања текста од ивица нам нису интересантна и оставићемо их на 0. Сва подешавања у *Properties* дијалогу се одмах примењују па не постоји посебно дугме за прихватање промена. Након завршетка подешавања само угасимо дијалог да нам не смета.

Једном направљени текст бокс можемо да копирамо на све отворене странице. Најједноставније је да се то уради *copy/paste* методом, означимо текст оквир и копирамо га у *clipboard* (*CTRL+c*) а затим означимо нову страницу и налепимо (*CTRL+p*). На десним страницама текст оквир ће бити померен у десно за 4 мм па их треба просто померити у предвиђену позицију једноставним хватањем (мишем) оквира и померањем.

Ово је тренутак када би требало по први пут да сачувамо наш рад као *Template* (шаблон). Претпостављамо да ћемо овакву организацију часописа користити за сваки наредни број часописа тако да нам одговара да је без садржаја. Овакав шаблон није комплетан фали му још *layer-a*, предефинисане боје, стилови, насловна страница и још што шта али то можемо и накнадно додати у овај шаблон и снимити поново. Снимаћемо ову верзију са *File* → *Save as Template...* У отвореном дијалогу осим имена шаблона треба одабрати и категорију, у нашем случају „*Magazine*“ и потврдити да запамти примењене фонтове (*Include Fonts*). Подразумевано шаблон ће бити снимљен у директоријуму *~/scribus/templates* и то нећемо мењати.

Наставиће се.

Администрација GNU/Linux система

Праћење перформанси система

(2. део)

Сваки рачунар на свету има свог администратора система. Између осталих дужности, администратор треба да има увид у рад свог система. У наредном делу овог текста видећемо неколико основних наредби и алатки за надгледање и праћење рада и заузећа дискова и процесора.

Аутор: Михајло Богдановић

Поздрав Линери! Као што је обећано настављамо даље са програмима/алатима са којима можемо утврдити заузеће диска као и његове перформансе, такође у овом броју бавићемо се и заузећем процесора. Па да кренемо полако.

Заузеће диска и његове перформансе

Заузеће диска се лако може пратити командом **df** и **du**. Команда **df** приказује стање расположивог простора на диску док команда **du** приказује колико је простора заузето на диску.

Употребом опције **-k** команде **df** може се видети заузеће диска у килобајтима

```
bash-4.1$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
/dev/root        20641788 12602500   6990696   65% /
/dev/sda8        15481840  4701848   9993560   32% /media/root_ubuntu
/dev/sda9        42924248 28350532 12393264   70% /media/home_ubuntu
/dev/sda1         102396    46764    55632   46% /media/rsv
/dev/sda2        83779580 73083256 10696324   88% /media/windows
/dev/sda5        62918540 51342056 11576484   82% /media/hard1
tmpfs            1803556     116    1803440    1% /dev/shm
/dev/sda6        83884700 58033040 25851660   70% /media/hard2
```

У данашњим приликама извештај ће бити најчитљивији употребом опције **-h**

```
bash-4.1$ df -k
Filesystem      1K-blocks    Used Available Use% Mounted on
/dev/root        20641788 12602500   6990696   65% /
/dev/sda8        15481840  4701848   9993560   32% /media/root_ubuntu
/dev/sda9        42924248 28350532 12393264   70% /media/home_ubuntu
/dev/sda1         102396    46764    55632   46% /media/rsv
/dev/sda2        83779580 73083256 10696324   88% /media/windows
/dev/sda5        62918540 51342056 11576484   82% /media/hard1
tmpfs            1803556     116    1803440    1% /dev/shm
/dev/sda6        83884700 58033040 25851660   70% /media/hard2
```

Опција **-h** потпуно разјашњава колико имамо расположивог простора у сваком систему директоријума.

```
bash-4.1$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        20G   13G   6.7G   65% /
/dev/sda8        15G   4.5G   9.6G   32% /media/root_ubuntu
/dev/sda9        41G   28G   12G    70% /media/home_ubuntu
/dev/sda1       100M   46M   55M   46% /media/rsv
/dev/sda2        80G   70G   11G   88% /media/windows
/dev/sda5        61G   49G   12G   82% /media/hard1
tmpfs            1.8G  116K   1.8G    1% /dev/shm
/dev/sda6        80G   56G   25G   70% /media/hard2
```

Ако употребимо опцију **-l** добићемо само податке о простору на диску који се физички налази на систему али не и за повезане мрежне системских директоријума нити за дискове којима приступамо даљински.

Када желимо да утврдимо колики део простора на диску се користи, употребимо команду **du**. Њена основна синтакса је

```
du [opcija] datoteka
```

```
sysadmin@server ~/backup # du -h
7.7G .
```

```
sysadmin@server ~/backup # du -k
8015940 .
```

```
sysadmin@server ~/backup # du -m
7829 .
```

```
sysadmin@server ~/backup # du -s
8015940 .
```

- h** - приказује извештај у јединицама које су најчитљивији кориснику
- k** - приказује резултате у килобајтима
- m** - приказује резултате у мегабајтима
- s** - приказује само збирне вредности

Улазно-излазне операције са диском обично су уско грло система. Оперативна меморија и процесор раде више хиљада пута брже од најбржег диска. Срећом, језгро оперативног система и савремени чврсти дискови користе софтверску и хардверску меморијску оставу, међумеморију за предходно учитавање и одложено записивање на хард диск да би прикрили и смањили поменућу разлику у брзини рада. Ипак, као администратор система и даље би требали знати да утврдите проблеме и изборите се са њима када се појаве на улазно-излазним операцијама са диском.

Корисно је да се упознате са следећим фразама које се често користе за описивање перформанси система:

- **Средње време приступања** (*Average seek time*) - процена просечног трајања операције приступања
- **Максимално време приступања** (*Maximum seek time*) - трајање кретања главе

- диска између две стазе на диску које су међусобно највише удаљене
- **Минимално време приступања** (*Minimum seek time*) - трајање кретања главе диска између две суседне стазе
- **Кашњење лоцирања** (*Queuing delay*) - време потребно контролеру диска да утврди где на диску треба да прочита/упише податке
- **Ротационо кашњење** (*Rotational latency*) - време потребно да се право место на диску доведе испод главе за читање/уписивање, изражено бројем обрта у минути (RPM)
- **Кашњење приступања** (*Seek latency*) - време потребно глави за читање/уписивање да се позиционира на одређени цилиндар диска
- **Време приступања** (*Seek time*) - Време потребно глави за читање/уписивање да пређе са једне стазе диска на другу

За утврђивање проблема са перформансама улазно-излазних операција најчешће се користи команда *iostat*. Слично команди *vmstat* и команда *iostat* се најједноставније задаје на следећи начин

```
iostat [интервал] [број]
```

Интервал дефинише број секунди између два узастопна извештаја, а број представља укупан број извештаја.

```
bash-4.2$ iostat 5 5
Linux 3.2.29 (brok4)    10/20/2012    _x86_64_    (4 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           4.63    0.20   1.51   1.83    0.00   91.83

Device:            tps    Blk_read/s    Blk_wrtn/s    Blk_read    Blk_wrtn
sda                 14.41         546.93        295.56    1454984    786272

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           1.51    0.00   0.90   0.35    0.00   97.24

Device:            tps    Blk_read/s    Blk_wrtn/s    Blk_read    Blk_wrtn
sda                 1.20         4.80         24.00         24         120

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           1.45    0.00   0.75   0.15    0.00   97.64

Device:            tps    Blk_read/s    Blk_wrtn/s    Blk_read    Blk_wrtn
sda                 0.40         0.00         4.00         0         24

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           1.36    0.00   0.75   0.00    0.00   97.89

Device:            tps    Blk_read/s    Blk_wrtn/s    Blk_read    Blk_wrtn
sda                 0.00         0.00         0.00         0         0

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           3.21    0.00   1.21   0.10    0.00   95.48

Device:            tps    Blk_read/s    Blk_wrtn/s    Blk_read    Blk_wrtn
sda                 0.40         0.00        19.20         0         96
```

tps - број операција преноса података (или улазно-излазних захтева за прекид) у секунди послатих диску

Blk_read/s - број блокова података учитаних са уређаја у секунди
Blk_wrtn/s - број блокова података уписаних на уређај у секунди
Blk_read - укупан број учитаних података
Blk_wrtn - укупан број уписаних података

Ако детаљније желимо да испитамо улазно-излазне перформансе диска, употребићемо опцију **-x**, задајући и диск односно партицију која нас занима. Опција **-x** омогућава детаљнију статистику улазно-излазних операција задатих дискова или свих дискова ако не задамо диск.

Наредна команда приказује снимак активности диска `/dev/sda` током превођења језгра оперативног система (такође [интервал=5] [број=5]). Због опције **-d** команда `iostat` не приказује збирни извештај о коришћену процесора

```
bash-4.2# iostat -d -x /dev/sda 5 5
Linux 3.2.29 (brok4)      10/20/2012      _x86_64_      (4 CPU)
```

Device:	rrqm/s	wrqm/s	r/s	w/s	rsec/s	wsec/s	avgrq-sz	avgqu-sz	await	svctm	%util
sda	5.40	2.59	6.29	1.97	298.26	172.95	57.05	0.28	33.61	4.66	3.85

Device:	rrqm/s	wrqm/s	r/s	w/s	rsec/s	wsec/s	avgrq-sz	avgqu-sz	await	svctm	%util
sda	0.00	0.40	0.00	0.60	0.00	8.00	13.33	0.01	14.67	14.67	0.88

Device:	rrqm/s	wrqm/s	r/s	w/s	rsec/s	wsec/s	avgrq-sz	avgqu-sz	await	svctm	%util
sda	0.00	1.00	0.00	6.80	0.00	62.40	9.18	0.10	15.38	0.68	0.46

Device:	rrqm/s	wrqm/s	r/s	w/s	rsec/s	wsec/s	avgrq-sz	avgqu-sz	await	svctm	%util
sda	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Device:	rrqm/s	wrqm/s	r/s	w/s	rsec/s	wsec/s	avgrq-sz	avgqu-sz	await	svctm	%util
sda	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

rrqm/s - број спојених захтева за читање података у секунди
wrqm/s - број спојених захтева за уписивање података у секунди
r/s - број захтева за читање података у секунди
w/s - број захтева за уписивање података у секунди
rsec/s - број учитаних сектора у секунди
wsec/s - број уписаних сектора у секунди
avgrq-sz - просечна величина захтева у секторима
avgqu-sz - просечна дужина реда за чекање захтева
await - просечно време чекања улазно-излазне операције на извршење изражено у милисекундама
svctm - просечно време извршавања улазно-излазне операције изражено у милисекундама
%util - проценат заузећа процесора захтевима за улазно-излазне операције

Коришћење процесора

Најпознатији услужни програм за праћење коришћења процесора јесте `top`, који приказује најзахтевније процесе. Програм `top` прати рад процесора у реалном времену, што значи да приказује листу активних процеса који га највише користе. Приказ се подразумевано ажурира на сваке три секунде, мада је интервал ажурирања променљив у току извршавања команде `top`, са **d** или **s** и задатом вредношћу.

Пример замене интервала ажурирања на 6 секунди

```
top - 12:47:36 up 205 days, 8:46, 1 user, load average: 0.27, 0.17, 0.11
Tasks: 216 total, 1 running, 214 sleeping, 0 stopped, 1 zombie
Cpu(s): 1.7%us, 0.4%sy, 0.0%ni, 97.8%id, 0.0%wa, 0.0%hi, 0.1%si, 0.0%st
Mem: 8190820k total, 7765524k used, 425296k free, 431288k buffers
Swap: 2102456k total, 2176k used, 2100280k free, 6097212k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1476	mysql	20	0	457m	79m	4292	S	6	1.0	4197.19	mysqld
9591	www-data	20	0	220m	20m	3800	S	6	0.3	0:00.24	apache2
9638	www-data	20	0	220m	19m	3688	S	5	0.2	0:00.16	apache2
27	root	20	0	0	0	0	S	0	0.0	3:02.47	events/0
9329	www-data	20	0	215m	14m	4028	S	0	0.2	0:00.17	apache2
1	root	20	0	8352	648	604	S	0	0.0	1:27.95	init
2	root	20	0	0	0	0	S	0	0.0	0:00.00	kthreadd
3	root	RT	0	0	0	0	S	0	0.0	0:11.89	migration/0
4	root	20	0	0	0	0	S	0	0.0	0:18.38	ksoftirqd/0
5	root	RT	0	0	0	0	S	0	0.0	0:00.00	watchdog/0
6	root	RT	0	0	0	0	S	0	0.0	0:00.16	migration/1
7	root	20	0	0	0	0	S	0	0.0	0:24.77	ksoftirqd/1
8	root	RT	0	0	0	0	S	0	0.0	0:00.00	watchdog/1
9	root	RT	0	0	0	0	S	0	0.0	0:00.05	migration/2
10	root	20	0	0	0	0	S	0	0.0	0:13.11	ksoftirqd/2
11	root	RT	0	0	0	0	S	0	0.0	0:00.00	watchdog/2
12	root	RT	0	0	0	0	S	0	0.0	0:00.01	migration/3

Приказ команде top

```
sysadmin@server ~ # top
```

```
top
```

```
top - 13:06:16 up 205 days, 9:05, 1 user, load average: 0.01, 0.11, 0.11
Tasks: 213 total, 2 running, 210 sleeping, 0 stopped, 1 zombie
Cpu(s): 0.0%us, 0.0%sy, 0.0%ni, 100.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 8190820k total, 7911116k used, 279704k free, 433660k buffers
Swap: 2102456k total, 2176k used, 2100280k free, 6196160k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
10853	root	20	0	19064	1440	1004	R	138	0.0	0:00.06	top
1	root	20	0	8352	648	604	S	0	0.0	1:27.96	init
2	root	20	0	0	0	0	S	0	0.0	0:00.00	kthreadd
3	root	RT	0	0	0	0	S	0	0.0	0:11.89	migration/0
4	root	20	0	0	0	0	S	0	0.0	0:18.38	ksoftirqd/0
5	root	RT	0	0	0	0	S	0	0.0	0:00.00	watchdog/0
6	root	RT	0	0	0	0	S	0	0.0	0:00.16	migration/1
7	root	20	0	0	0	0	S	0	0.0	0:24.78	ksoftirqd/1
8	root	RT	0	0	0	0	S	0	0.0	0:00.00	watchdog/1
9	root	RT	0	0	0	0	S	0	0.0	0:00.05	migration/2
10	root	20	0	0	0	0	S	0	0.0	0:13.11	ksoftirqd/2
11	root	RT	0	0	0	0	S	0	0.0	0:00.00	watchdog/2
12	root	RT	0	0	0	0	S	0	0.0	0:00.01	migration/3
13	root	20	0	0	0	0	S	0	0.0	0:09.00	ksoftirqd/3
14	root	RT	0	0	0	0	S	0	0.0	0:00.00	watchdog/3
15	root	RT	0	0	0	0	S	0	0.0	0:01.16	migration/4
16	root	20	0	0	0	0	S	0	0.0	0:20.14	ksoftirqd/4

Први ред приказује укупно време рада система, слично резултату команде **uptime**. Следећа два реда резултата команде **top** сумирају коришћење процесора (у апсолутним бројевима, односно у процентима) приказујући активне процесе, процесе који су привремено заустављени, зомбије и трајно заустављене процесе. Зомби (енгл. *zombie*) је процес који се завршио пре него што је његов родитељски процес успео да то сазна. Они се зову зомби-процеси јер су „мртви“ и не заузимају ресурсе (осим одреднице у табели процеса језгра оперативног

система), али се ипак појављују при излистивању процеса. У четвртм и петом реду резултата сумирано је актуелно коришћење меморије, на начин као што то приказује команда *free*. Шести ред је у ствари заглавље за преостале податке - процесе који највише оптерећују процесор, подразумевано приказане по опадајућем редоследу према заузећу процесора.

PID - идентификатор процеса

USER - име корисника који је покренуо процес

PRI - приоритет извршавања процеса

NI - измена приоритета процеса (*nice*)

SIZE - величина меморије потребна за извршавање процеса

RSS - величина физичке меморије коју процес заузима

SHARE или **SHR** - величина дељене меморије коју процес користи

STAT или **S** - актуелно стање процеса

%CPU - проценат укупног процесорског времена који користи процес

%MEM - проценат укупне физичке меморије коју користи процес

TIME - укупно потрошено време процесора на извршење процеса

COMMAND - команда којом је процес покренут, приказана у заградама ако је процес одложен у виртуелну меморију

Жељени процес можемо „убити“ у самом приказу команде *top*, са тастером **k** или **K** уписујући *PID* (идентификатор процеса).

Приказ

```
top - 13:027:16 up 205 days, 9:05, 1 user, load average: 0.01, 0.11, 0.11
Tasks: 213 total, 2 running, 210 sleeping, 0 stopped, 1 zombie
Cpu(s): 0.0%us, 0.0%sy, 0.0%ni,100.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 8190820k total, 7911116k used, 279704k free, 433660k buffers
Swap: 2102456k total, 2176k used, 2100280k free, 6196160k cached
PID to kill:
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
7773 www-data 20 0 218m 19m 4336 S 8 0.2 0:00.48 apache2
1488 mysql 20 0 455m 76m 4328 S 2 1.0 1033:17 mysqld
7839 www-data 20 0 211m 12m 3828 S 2 0.2 0:00.06 apache2
7610 www-data 20 0 218m 19m 4328 S 1 0.2 0:00.58 apache2
7658 www-data 20 0 226m 20m 5192 S 1 0.3 0:00.22 apache2
```

Стање процеса може да буде **D** (привремено заустављен на одређено време), **R** (извршава се), **S** (привремено заустављен), **T** (заустављен или праћен) или **Z** (зомби). Ако се иза ознаке стања процеса налази знак **,** процес има негативну измену приоритета, а ако је знак **N**, његова измена приоритета је позитивна. Уколико ту стоји знак **W**, процес је одложен у виртуелну меморију. Иначе, редослед приказаних процеса можемо да променимо тако што ћемо команди *top* дати налог да осим заузећа процесора узме у обзир и неки други критеријум. Док програм *top* ради притисните тастер **o** или **O** а затим тастер у приказу листе испод да би приказ био уређен по жељеном критеријуму.

a или **A** - редослед покретања

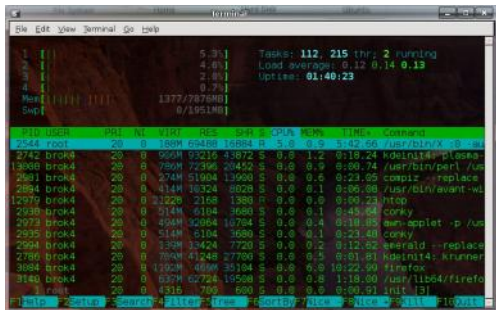
m или **M** - заузеће физичке меморије

n или **N** - бројчана вредност идентификатора

r или **P** – заузеће процесора
t или **T** – утрошено процесорско време

Од конзолних програма за надгледање перформанси система сличан програму *top* је *htop*.

Приказ



<http://www.dodaj.rs/f/22/Qh/YadS-MKZ/prikaz.jpg>

Толико за овај број.

Не пропустите у наредном броју још један важан сегмент у администрацији линукс система а то је надгледање рада у мрежи.

Наставиће се...

Conky - Систем монитор

Conky је лагани system monitor, дизајниран да може да пружи детаљан увид у рад система. Conky је могуће инсталирати на сваку GNU/Linux дистрибуцију. Програм је слободан за коришћење и не захтева никакву финансијску надонкаду.

Аутор: Жељко Шарић

Највеће предности овог, наизглед скромног, програма су:

- мало искоришћење ресурса рачунара,
- велике могућности конфигурисања,
- могућност позивања специфичних команди из терминала и читавање добијених вредности,
- са *LUA* подршком можете имати врло модеран *Conky* изглед и
- лакоћа преношења конфигурационе датотеке на било коју *GNU/Linux* дистрибуцију.

Conky је доста прегледан а за његово коришћење неопходно га је инсталирати и пронаћи примере конфигурационих датотека и поставити у Ваш */home* директоријум. Готово свака линукс заједница на

свету има, на својим форумима, одељак који се бави подешавањем и изгледом *Conky*-ја.

У *Ubuntu/Mint/Debian* дистрибуцијама је довољно укуцати

```
sudo apt-get install conky
```

и пакет ће бити инсталиран. На Вама остаје само да поставите скривену датотеку у `/home` партицију и покренете *conky*.

Покретање се ради кроз било који емулатор конзоле тако што ће те укуцати

```
conky
```

Conky можете конфигурисати у току рада програма, едитовањем `.conkyrc` датотеке неким од омиљених текст едитора и мењати подешавања која већ постоје. Чим снимите промене, мења се и *conky* прозор. Првобитни *conky* прозор који будете видели приликом покретања програма није визуелно занимљив. Срећом, могућности подешавања су изузетно велике па самим тим можете мењати велики број опција као и додавати нове.

Конфигурациона датотека се састоји из два дела:

- први део је везан за основна подешавања у смислу позиције, величине *conky* прозора и фонтова;
- други део је везан за конфигурацију приказа у *conky* прозору.

Conky прозор најчешће стоји са леве или десне стране екрана али се може поставити и у горњем или доњем делу екрана. Могуће је покренути и неколико `.conkyrc` конфигурационих датотека и наместити више *conky* прозора да стоје на екрану корисника. Прозор *conky* програма је у већини

подешавања провидан тако да лепо изгледа на Вашем екрану.

Conky се може искористити да прикаже и разне *widget*-е. То се ради ако је *conky* инсталиран са *LUA* подршком. *LUA* подршка се инсталира заједно са додатним пакетима и не долази са самим *conky* пакетом. Могућности које се добијају још више приказују могућност и снагу овог сјајног малог програма.

Неке од ствари које можете поставити да приказује *conky* су:

- искоришћење процесора (без обзира на број језгра – чак и свако језгро појединачно),
- искоришћење *RAM* меморије,
- искоришћење *swap* партиције,
- искоришћење `/`, `/home` и свих осталих партиција на систему,
- приказ *upload/download* брзине Ваше мреже,
- приказ *IP* адресе уколико је неопходно,
- температуре процесора, графичке картице, тврдих дискова, *SSD*,...

Conky је способан да прикаже највећи број битних ствари приликом праћења рада и оптерећености система. Могућност да извршава команде из терминала, читање тих вредности и приказ у *conky* прозору су још један плус. На тај начин можемо исчитати и приказати, на пример, првих 5 процеса који имају највеће искоришћење процесора и *RAM* меморије.

На следећем примеру ћемо видети како изгледа *conky* када се инсталира на систем и како изгледа када се подеси по потребама корисника.

```
stereo - Linux 3

Uptime: 1d 14h 35m
Frequency (in MHz): 2812
Frequency (in GHz): 2.81
RAM Usage: 1.21GiB/5.82GiB - 20%
Swap Usage: 80.1MiB/1.86GiB - 4%
CPU Usage: 11%
Processes: 128 Running: 0

File systems:
/ 8.29GiB/13.9GiB
Networking:
Up: 78B - Down: 68B

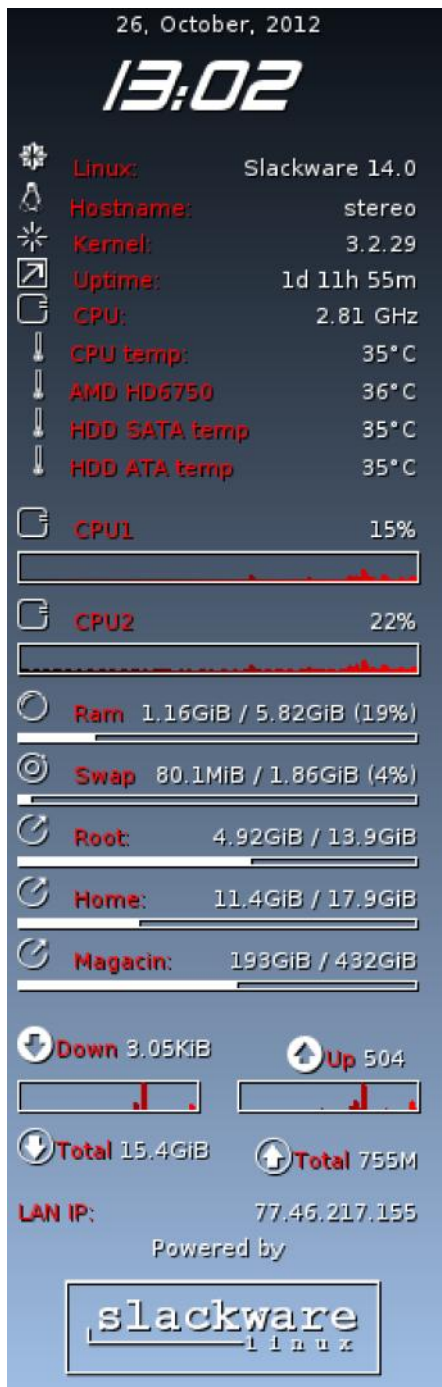
Name          PID   CPU%  MEM%
plugin-containe 2301   6.00   3.03
compiz         1963   0.50   0.71
dbus-daemon    1791   0.50   0.03
X              1770   0.50   1.83
```

SLIKA 1: Conky1.png

Врло ружно, зар не. Након што видите како други корисници сређују и подешавају свој *conky*, просто пожелите да и Ви имате нешто лепо на екрану а уз то и јако корисно. Ево примера како може изгледати Ваш *conky*.

SLIKA 2: Conky2.png

Као што видите ово сад изгледа далеко пристојније. Да би овако изгледао потребно је унети измене у *conky* конфигурациону датотеку. Конфигурациона датотека за мој *conky* захтева и инсталацију неколико фонтова који не долазе заједно са системом али је то, наравно, једноставно решиво. Само инсталирајте додатне датотеке.



Ово је садржај моје *.conkyrc* датотеке:

```
background no
font Sans:size=8
#xftfont Sans:size=10
use_xft yes
xftalpha 0.9
update_interval 3.0
total_run_times 0
own_window yes
own_window_type override
own_window_transparent yes
own_window_hints decorated,be-
low,sticky,skip_taskbar,skip_pager
double_buffer yes
minimum_size 220 5
maximum_width 200
draw_shades yes
draw_outline no
draw_borders no
draw_graph_borders yes
default_color white
default_shade_color black
default_outline_color green
alignment top_left
gap_x 12
gap_y 35
no_buffers yes
uppercase no
cpu_avg_samples 2
override_utf8_locale yes
```

TEXT

```
${alignc}${time %d, %B, %Y}
${font Radio Space Bold Ital-
ic:size=30}${time %H:%M}${font}

${font OpenLogos:size=14}E${font}
${color red}$sysname: ${col-
or}${alignr}${exec cat /etc/slackware-
version}
${font OpenLogos:size=14}t${font}
${color red}Hostname: ${col-
or}${alignr}$nodename
${font StyleBats:size=14}q${font}
${color red}Kernel: ${col-
or}${alignr}$kernel
```

```
${font StyleBats:size=14}v${font}
${color red}Uptime: ${col-
or}${alignr}$uptime
${font StyleBats:size=14}A${font}
${color red}CPU: ${col-
or}${alignr}${freq_g} GHz
${font Weather:size=14}x
${font}${color red} CPU temp: ${col-
or}${alignr}${execi 10 sensors |
grep 'Temp' | cut -c17-18}°C
${font Weather:size=14}x
${font}${color red} AMD HD6750
${color}${alignr}${execi 10 aticon-
fig --adapter=0 --od-gettemperature
| grep Temp | cut -c37-38}°C
${font Weather:size=14}x
${font}${color red} HDD SATA temp
${color}${alignr}${execi 10 sudo
/usr/sbin/hddtemp /dev/sda | cut
-c55-56}°C
${font Weather:size=14}x
${font}${color red} HDD ATA temp
${color}${alignr}${execi 10 sudo
/usr/sbin/hddtemp /dev/sdb | cut
-c55-56}°C

${font StyleBats:size=14}A${font}
${color red}CPU1 ${col-
or}${alignr}${cpu cpu1}%
${cpugraph cpu1 15,200 000000
ff0000}
${font StyleBats:size=14}A${font}
${color red}CPU2 ${col-
or}${alignr}${cpu cpu2}%
${cpugraph cpu2 15,200 000000
ff0000}
${font StyleBats:size=14}g${font}
${color red}Ram ${col-
or}${alignr}$mem / $memmax ($mem-
perc%)
lignr}${fs_free /} / ${fs_size /}
${fs_bar 4 /}
${font StyleBats:size=14}o${font}
${color red}Home: ${col-
or}${alignr}${fs_free /home} /
${fs_size /home}
${fs_bar 4 /home}
```



```
{font StyleBats:size=14}o${font}
${color red}Magacin: ${col-
or}${alignr}${fs_free /mnt/magacin}
/ ${fs_size /mnt/magacin}
${fs_bar 4 /mnt/magacin}
```

```
{font PizzaDude Bul-
lets:size=14}U${font}${color
red}Down ${color}${downspeed eth0}
${alignr}${font PizzaDude
Bullets:size=14}O${font}${color
red}Up ${color}${upspeed eth0}
${downspeedgraph eth0 15,90 000000
ff0000} ${alignr}${upspeedgraph eth0
15,90 000000 ff0000}
${font PizzaDude Bul-
lets:size=14}T${font}${color
red}Total ${color}${totaldown eth0}
${alignr}${font PizzaDude
Bullets:size=14}N${font}${color
red}Total ${color}${totalup eth0}
```

```
${color red}LAN IP: ${col-
or}${alignr}${execi 200 curl ifcon-
fig.me}
${voffset 5}${alignc }Powered by
#${font Radio Space Bold Ital-
ic:size=16}${alignc }POWERED
BY${font}
${voffset -20}${font OpenLo-
gos:size=100}${alignc -20}q${font}
```

Conky садржи преко 300 уграђених објеката, на Вама остаје да видите бар део тих могућности а ако Вам буде занимљиво наставите са истраживањем јер *conky* једноставно даје невероватне могућности у приказу система и незаменљив је алат за брзо праћење перформанси система, без битнијег оптерећења система.

Срећно подешавање!

Интернет презентација:
<http://conky.sourceforge.net/>

deviantART:
<http://browse.deviantart.com/?qh=§ion=&q=conky/>

Слободан софтвер у образовању:

GeoGebra

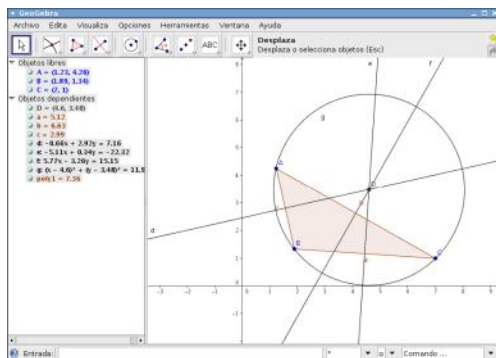
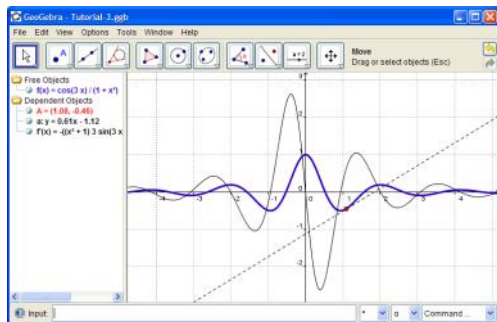
Аутор: Стефан Ножинић

Ако сте прави *geek* онда засигурно не може да прође добар осећај када решите један математички проблем. Математика је јако занимљива наука али није редак случај да понекад немамо времена да решавамо нешто банално. Поента је усредсредити се на нешто комплексније а онај банални део који често замара и тера људе да одустану од проблема, решава наш рачунар. Овај програм, који представљамо има баш ту намену. У питању је слободан програм за геометријска цртања, проучавање графика функција и још много тога. *GeoGebra* се може инсталирати и покренути на линуксу и неким врстама *UNIX*-а, на *MS Windows* и на *Mac OSX* оперативном систему. Такође, можемо је користити и *online* ако желимо. Писана је у *Java* програмском језику тако да је могуће портовати на све платформе који подржавају *java runtime*.

Математика је јако занимљива наука али није редак случај да понекад немамо времена да решавамо нешто банално. Поента је усредсредити се на нешто комплексније а онај банални део који често замара и тера људе да одустану од проблема, решава наш рачунар. Овај програм, који представљамо има баш ту намену.

Могућности

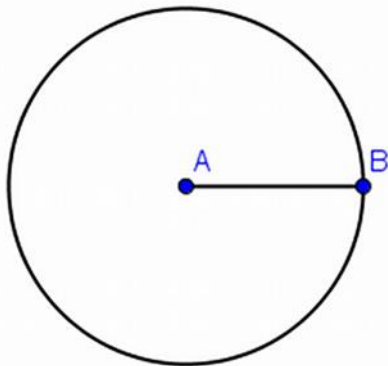
GeoGebra може да обрађује геометријске функције као што су цртање тачака, линија, троуглова, кругова, углова и још много тога. Такође може и да нацрта график неке функције што доста може помоћи у математичкој анализи. Можемо да видимо, на пример, у којој тачки две функције имају исту вредност, можемо да конструишемо свашта и још много других ствари.



Ми наводимо један пример коришћења *GeoGebra*-е.

Конструкција једнакостраничног троугла:

Корак 1: Прво када покренемо *Geo-*



Gebra-у видимо Декартов координатни систем. Пошто нам он за ово није потребан можемо га искључити путем опције *View->Axes*.

Корак 2: Изаберемо алатку „*Segment between two points*“ и изаберемо неке 2 тачке. Добијамо дуж **AB**

Корак 3: Изаберемо алатку „*Circle with center through point*“ и кликнемо на тачку **A** па онда на тачку **B**. Добијамо следеће:

Корак 4: Сада са истом алатком кликнемо на тачку **B** па онда на тачку **A**. Сада имамо 2 кружнице које се секу у 2 тачке.

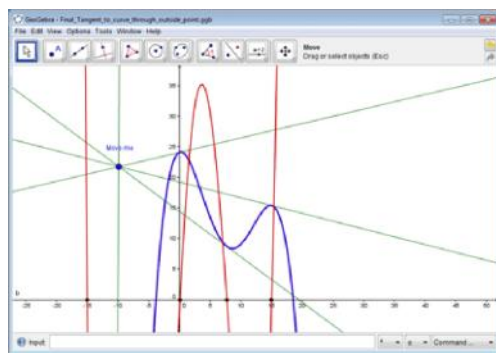
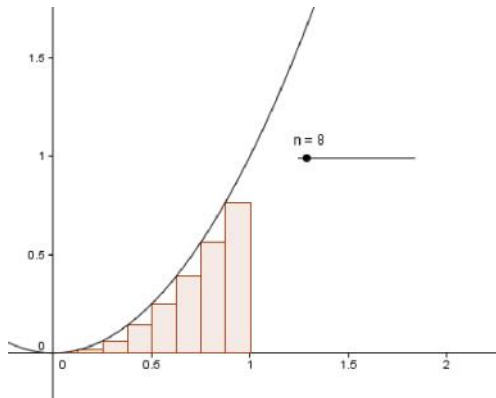
Корак 5: да обележимо те 2 тачке користимо алатку „*Intersect two objects*“. Кликнемо на једну па онда на другу кружницу. Добијамо тачке **C** и **D** које су тачке пресека.

Корак 6: Сада ћемо конструисати

троугао који нам треба. Кликнемо на „*Polygon*“ и потом кликнемо на тачке **A B C A** редом. Дobili смо троугао.

Корак 7: кликнемо на опцију „*Angle*“ и потом кликнемо на тачке редом: **C A B A B C B C A** и тиме смо добили приказ сва 3 унутрашња угла овог троугла. Примећујемо да су сва 3 једнака и то по 60 степени. Сада Кликнемо на алатку „*Move*“ и покушамо померати тачке, примећујемо да се углови не мењају.

Ето то је био један пример коришћења овог фантастичног програма који препоручујемо.



Субер криминал

(4. део)

Типови и примери субер криминала

Autor: Vladimir Cicović

Типови и примери *cyber* криминала:

- индустријска шпијунажа
- обавештајна шпијунажа / саботаже
- интелектуална крађа
- финансијске крађе
- крађа идентитета
- класичне преваре
- ситуациони криминал
- финансијска добит уз помоћ вируса (*ботнет*)
- DDOS напад
- педофилија

Индустријска шпијунажа

Најочитији пример јесте Кина - САД и САД - остатак света, где је утврђено да је америчка страна неколико пута користила легитимна средства (шпијунажу) како би обезбедила конкурентност компанија из САД. 1999. године тим независних сигурносних експерата је пронашао такозване *NSA KEY* у *Windows*-у *NT*, који се користио за различите намене. Енкрипција која се користила је била *RSA*, где имате јавни/тајни кључ. Јавни кључ је био у *Windows NT* а тајни на страни (вероватно) *NSA*. Овим је било могуће слати информације са *Windows NT* иако *Microsoft* и *NSA* поричу повезаност, стручњаци су нашли назив *_NSAKEY*

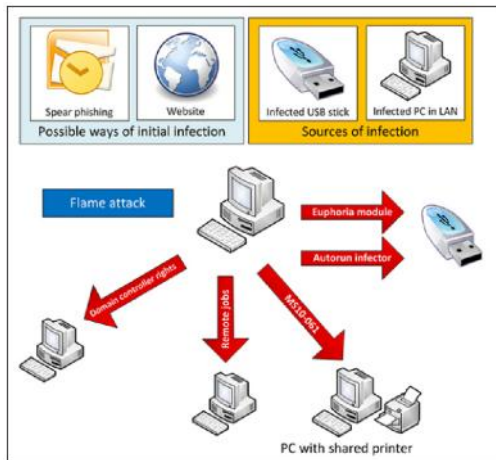
унутар бинарног фајла. што је очигледно и повезано са *NSA*. Могуће је да су овом приликом неке компаније у ЕУ зони изгубиле послове од неколико милијарди у корист компанија из Америке. Један од последњих трикова из куће *cyber* криминала јесте „губљење“ *usb flash* диска на паркингу неке фирме. И при том се на *usb/flash* дисковима налази вирус који ће се даље ширити. Врло лукаво.

Обавештајна шпијунажа / саботаже

Последњих 10-ак година се користе програмске алатке (вируси, *spyware* и слично) како би се дошло до обавештајних података. Тако је у нападу *MOSAD*-а на *Mahmoud Al-Mabhouh*-а, коришћено праћење *e-mail* комуникације жртве. Починиоци су знали када ће доћи, то јест знали су комплетну агенду жртве и самим тим себи обезбедили пут извршења атентата. Постојањем *Stuxnet* вируса (примећен је у јуну 2010.), који је нападао само одређене техничке делове (*Siemens* програмабилни логички контролери) који су потребни за нуклеарке и рад нуклеарних постројења довео је у сумњу директно САД, пошто су они осим Израела били окренути нападу на Иранска постројења. *Duqu* вирус (примећен је септембра 2011.), према *Symantec*-у је очито дело истог аутора. Оба вируса су користили до тада непознате *exploit*-е за придобијање већих привилегија на *Windows* оперативном систему. *Flame* (detektovan 28. маја 2012.) је идентификован искључиво на машинама у Азији - Иран, Израел, Сирија... Овај вирус је у могућности да прислушкује *Skype* и остале програме за гласовну комуникацију. Вирус се

користи сертификатом *Microsoft*-а како би се непримећено инсталирао на рачунаре. Сертификат је разбијен такозваном Колизижом *MD5 hash*-а.

Flame botnet - путеви инфекције



Сви ови вируси су направљени са једним циљем: уништењем нуклеарног програма Ирана. Треба напоменути да је пре неколико година из Кине било проваљено у немачку владу, владу САД и друге институције при чему су украдени гигабајти података и службених преписки на највишем нивоу.

Интелектуална крађа

Свако дело копирања или неовлашћене дистрибуције информација, без пристанка аутора, сматра се интелектуалном крађом. У свету, овај вид *cyber* криминала је заступљен у скоро свим земљама. Али по бројности одскачу неке афричке земље, земље бившег СССР - заправо свуда где не постоји жеља за реализацијом законских норми за овај вид криминала. У нашој земљи се манифестује тако што се већина научних и семинарских радова копирају са интернета.

Наравно академска заједница је решила овај проблем навођењем извора информација али не користе сви овај вид протекције.

Финансијске крађе

Већина људи користи кредитне картице као вид плаћања услуга. У последње време, све више људи користи их за плаћање путем интернета. Остављајући број картице и осталих информација на одређеним сајтовима - сами постају мета у неком периоду или ако куцају са рачунара инфицираним рачунарским вирусом - њихове информације се злоупотребљавају веома брзо.

Вирус може да мења „отисак“. У бази антивируса постоји идентификација одређених вируса, међутим, аутори вирус могу да направе тако да се вирус ажурира одмах по препознавању. Такође, могу да произведу сваких десет минута по један облик односно непознат „отисак“ вируса. Ово им оставља могућност да упадају у финансијске институције као што су банке, мењачнице, берзе.

Највећа заштита *online* плаћања јесте *two-way* идентификација, односно идентификација клијента-банке је обједињена у такозвани *Man in The Browser* - где имате *Java script* који користи пропуст у модерним читачима и шаље нападачу потребне информације. Такође, пре него што жртва „приступи“ (овде се лажира сајт банке), нападач припреми одређене информације које се појављују жртви - она мисли да је приступила сајту банке а заправо, када укуца идентификацију, нападач преузима *online* банковни рачун и прави трансакцију према одређеном рачуну. Први пут када је

изведено оштећена страна је имала око 400.000 долара губитка, то се десило 2006. у САД.

Нажалост не постоји проверени начин да се утврди верификација информација које су у *browser-y*, па самим тим је могуће манипулисати *web* страницом неке банке.

Овде треба напоменути и токене који су проваљени на основу криптоанализе и рада токена. Са милијарде могућности - криптоанализом сведено је на неколико могућности то јест токена који омогућују приступ *on line* банкарским услугама.

Нажалост, наша јавност није упозната са овим врстама напада а банкарски сектор би претрпео велики трошак мењања постојећих токена, сајтова и осталог (слање информација да се токен промени, долазак странке у просторије банке и слично).

Наставиће се...

Преглед популарности GNU/Linux/BSD дистрибуција за месец октобар

Distrowatch

1	Mint	3604>
2	Mageia	2795>
3	Ubuntu	2548<
4	Fedora	1489>
5	Arch	1464>
6	Debian	1423<
7	openSUSE	1290<
8	Puppy	1051>
9	PCLinuxOS	1048>
10	Zorin	920>
11	Lubuntu	869>
12	Snowlinux	817>
13	CentOS	782=
14	Slackware	738<
15	ROSA	730<
16	Pear	705>
17	CrunchBang	685<
18	SolusOS	657>
19	Xubuntu	654>
20	FreeBSD	623>
21	Lite	599>
22	Zenwalk	586>
23	Chakra	574>
24	Kubuntu	557>
25	Ubuntu Studio	526=

Пад <

Пораст >

Исти рејтинг =

(коришћени подаци са Distrowatch-a)

Криптографија/Криптологија

(1. део)

Писати о историји криптографије/криптологије је врло незахвално. Почећемо са мањим делом, како бисмо касније увели читаоце у део ове науке о тајном писању (kriptos - тајно, grafein - писање, крипто-графија). Иначе, историја о криптологији је заиста пуна тајни. Разлог томе је врло једноставан - људи су чували тајне а криптографија им је помагала. Битни догађаји људске историје су увек под неким велом тајни.

Аутор: Владимир Цицовић

Историја криптологије

Велики император Јулије Цезар (Јули 100 п.н.е - Март 44 п.н.е), настојао је да сачува своју империју. Знао је да сваког тренутка мора имати контролу над својом империјом. У том погледу најважнији део су биле информације које је слао и примао од својих војсковођа. Те информације су биле шифроване уз помоћ шифарског система који данас зовемо "Цезарова шифра". Цезар је узео алфавет отвореног текста и померио за 3 места и добио алфавет шифрованог текста.

Пример:

Отворени Текст: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Шифровани Текст: D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

Тако ће реч *NAPAD* постати *QDSDG* (где је $N \Rightarrow Q$, $A \Rightarrow D...$). Ово је могао само да дешифрује онај ко је познавао овај шифарски систем. Прва справа за помоћ при шифровању је била скитала. Овде их спомињем јер је то први пут да људи користе справу како би шифровали. Касније ће доћи и до израде механичких справа као што су Лорензова машина, Енигма и многе друге.

Криптографија се највише развијала током 2. светског рата. У том периоду, поред развијања математике, телекомуникација и сличних области, дошло је до експлозије употребе шифарских система. Самим тим су се развијале области везане за употребу, слање и коришћење шифарских система.

Хладни рат је донео употребу сателитских система који су били заштићени *One time pad* шифарским системом. Такође, дошло је до појаве такозваних симетричних крипто-система и асиметричних крипто-система.

Иначе, криптографију која користи рачунаре, то јест шифарске системе (овде можемо рећи и алгоритам за рачунар) настале на рачунарима и прилагођене раду рачунара зовемо “Модерном криптографијом” а све раније шифарске системе стављамо под “Класична криптографија”. Да разјаснимо: “криптологија” и “криптографија” - криптологија је наука о тајном, гдје имамо области:

- криптографија - наука о тајном писању,
- криптофонија - наука о заштити звука,
- криптовизија - наука о заштити слике.

Другим речима , криптологија обухвата криптографију.

Терминологија Криптографије

У криптографији имамо две врсте информација: *отворени текст* и *шифровани текст*. Операција која отворени текст претвара у шифровани текст зове се *шифарски систем*. У модерној криптографији то се зове *алгоритам*. Сваки легалан процес обрнут од шифровања се зове *дешифровање*. Сваки процес који није легалан, односно користи непознавање шифарског система да би се дошло до текста, зове се *декриптовање*. Процес анализе, класификовање и друге радње над шифрованим текстом од стране треће стране се зове *криптоанализа*. Сваки знак отвореног текста има своју замену у виду *шифарске замене* или *шифрованог знака*.

Облици комуникације могу бити:

- уговорени облик комуникација (Београд - Дубровник);
- тајни облик комуникације (Брод креће са угљем - где је Брод - артиљерија, угаљ - гранатирање);
- шифровани облик комуникације (шифровани текст: АДФ ФДА - отворени текст: ВАН НАВ);
- невидљиви облик комуникације (невидљива мастила).

Ми ћемо се обазрети на тему шифрованог облика комуникације.

Поједини шифарски системи користе кључ. Кључ за шифровање може бити:

Према дужини:

- коначни
- бесконачни

Према типу кључа:

- логички (београд, сарајево...)
- нелогички (МАКОС12гооце=)
- мнемонични (НАПАД = 62723 , ако узмемо бројеве са моб. телефона)

Криптографски Системи

Генерална подела:

- 1) Систем премештања
- 2) Систем замењивања
- 3) Комбинацијске шифре (премештање + замењивање)

Сви шифарски системи могу да имају АЛФАБЕТ: сређени или несређени (АБЦД.., ДОЈА..)

Шифарски системи премештања:

- обично премештање
- премештање кључем
- премештање решеткама
- двоструко премештање

Шифарски систем замењивања се дели на:

- 1) Шифре просте замене
- 2) Шифре сложене замене

1 - Шифре просте замене (МОНОАЛФАБЕТСКЕ) се деле на:

- алфаветске шифре
- биграмске, Триграмске и полигамске шифре
- кодне таблице
- кодови
- шифре рашчлањивањем слова

2 - Шифре сложене замене (ПОЛИАЛФАБЕТСКЕ)

- шифре са сређеним алфабетом
- шифре са несређеним алфабетом

Шта су то шифре премештања ?

Сваки отворен текст над којим се изврши одређено премештање тако да други људи не могу да препознају - зове се шифарски систем премештања. Својствено је шифрама премештања да им се оргинална слова представљају баш онаква каква јесу и пре премештања, тако да, ако узмемо фреквенцију појављивања (причаћемо о лингвистичкој статистици касније) и премештени текст - лако уочавамо да је у питању шифарски систем премештања.

Шта су то шифарски системи са решеткама?

Просто речено, узмете квадрат 5x5 и напишете неку реченицу вертикално, а узмете слова хоризонтално. То што сте узели је само познато ономе ко поседује идеју да је то 5x5 (наравно, може се урадити крипто анализа и за мање од 10 секунди се разбија). Ако 5 унутрашњих поља означите са црним, то јест не пишете у њих - и онда испишете - то већ постаје проблем за некога да уради јасну криптоанализу текста, то јест даје тежину за криптоанализу. Сада ово што пишемо зове се решетка - текст уписан у коцку - односно било који облик који има унутар себе коцке (може бити чак и пирамида - набитније да прималац има исту решетку

испред себе).

Како обични шифарски систем може бити “појачан”?

Најбоља ствар су такозвани “лажни” знаци. Две стране се морају договорити око лажних знакова - како би их препознали. Наравно да постоје начини да се препознају лажни знаци, шифарски елементи који су само заблуда за криптоаналитичара. Али, такође постоје начини да се они препознају. Пример употребе лажних знакова: *NAPAD* - *CBRBS* - *C13BR44B5S*; овде смо додали бројеве као лажне знакове.

Шта су то шифре замењивања ?

Шифарски систем замењивања јесте онај у коме најмање један знак отвореног текста постаје своја замена у шифрованом тексту. Такође, где има замена: један знак отвореног текста = један знак шифрованог текста, зове се моноалфабетска или проста замена. Шифарски систем замене где један знак отвореног текста може да постане најмање две замене шифрованог текста, зове се шифарски систем сложене замене.

Другим речима: шифре просте замене су оне где се користи један алфавет отвореног текста и један алфавет или један скуп знакова шифарског система. Шифре сложене замене су оне које користе један алфавет отвореног текста и више алфабета шифарске замене.

Пример просте замене: $A = C, B = D...$ (Цезарова шифра) Пример сложене замене: $A = \{B, C, D, ..A\}$ - (Вигнерова шифра); $AC = DB, CA = BD$ (Плауфаиров шифарски систем).

Шта су то комбинацијске шифре?

Шифарски системи који комбинују шифре премештања и шифре замене зову се комбинацијски шифарски системи. Пример: *NAPAD*, шифра премештања: *PAAND*, шифра замене: *CVVXJ* (Цезарова замена). Мала напомена: савремени крипто алгоритми (модерна криптографија) користе комбинацијске шифре, на пример *AEC* - има премештање такозвано *C-Box* и онда шифарске замене.

Цезарова шифра, ROT13

Једноставнији облик шифарског система јесте Цезар шифарски систем. То је обични супституцијски шифарски систем где је једно слово отвореног текста једнако слову шифарског текста.

Пример 1. (узимамо слова алфавета који се користи на рачунарима):

отв. текст: *ABCDEFGHIJKLMNOPQRSTUVWXYZ*

шиф. текст: *FGHIJKLMNOPQRSTUVWXYZABCDE*

Овде имамо сређени алфабет (гледа се шифарска замена у односу на отворени текст) а може бити и несређени облик алфавета (слова су у произвољном низу постављена).

Пример 2.

отв. текст: *ABCDEFGHIJKLMNOPQRSTUVWXYZ*

шиф. текст: *VEKUYNQLJTROABWXPCHFGZIMS*

Сада ћемо направити трансформацију текста у шифровану замену.

Алфабет из примера 1. :

отв. текст: *MI ZELIMO NAZAD*

шиф. текст: *RO EJKQRT SFEFI*

Урадимо кратку анализу Цезаровог шифровања које смо управо извршили.

Математички приказ ове трансформације је: $F(x) \Rightarrow F(y)$, где имамо из првог скупа (отворени текст) копирање на други скуп (шифрована замена) - у којем нема таквог елемента првог скупа, који може имати више од једне замене у другом. Такође, приметимо да је $F(x) \Rightarrow F(y)$, исто као и $F(y) \Rightarrow F(x)$. Другим речима, све особине отвореног текста добија и шифарска замена. Практично то значи да неко ко познаје особине језика врло брзо може да “разбије” шифарску замену.

Под особинама језика спада:

- фреквенција појављивања слова
- фреквенција биграма, триграма, полиграма
- специфичност језика (рецимо, -АНА наставци у реченицама или -ОВА, -СКА и слично)
- број појављивања глагола, придева, именица и сл.
- просечна величина речи, реченице
- удаљеност слова (*NAPAD*, где је *A* удаљено од другог *A* тек једно место или *N* од *DZ* и слично)
- диференцијалност текста (текстови исте дужине, где имају поклапања или мимоилажења)
- јединственост речи (ускоро, следеће поглавље ...)

Ово спада под криптоанализу, о томе ћемо у наредном броју.

ROT13 има врло занимљиву историју. У ери интернета, постојао је обичај да ако желите да нешто ружно или неприкладно кажете на *news* групама, једноставно реч која не би била фина да се куца/чита, шифрујете *ROT13* (енгл. rotate by 13 places). Такође, он је супституцијски шифарски систем који користи сређени алфабет.

Напоменућу вам да *Windows XP*, користи *ROT13* да би “сакрио” у *registry*-ју ваше активности.

Наставиће се...

Web сервери

(1. део)

Аутор: Милутин Гавриловић

Желимо да представимо пар решења мапираних *http* сервера уз могућности измене портова, *high-load*, као и измену читаве логике функционисања и комбиновања два или више сервера. Овај начин рада омогућује и високу флексибилност као и рад са читавим системом кроз *http*. Предност оваквог решења је у мапирању конфигурационих датотека и чувања истих, као и самог садржаја. Решење је скоро самоадминистрирајуће јер се приликом сваког паљења рачунара опоравља мапа, дозволе, власништва (уколико је до тога неким случајем дошло)...

Но добро, да се вратимо на садржај чланка:

„Састојци“:

- Оперативни систем - *Debian (Crunchbang)* или *Debian based (Ubuntu, Mint, ...)*
- *Http* сервер програми - *apache2, nginx, SymplePythonHttpServer*.
- *Ftp* сервер програми: *vsftpd...*
- Скриптинг: *php5, perl, python...*
- Интерфејси за „скриптовање“: *cgi, php-fastcgi, perl-fastcgi*.
- Протоколи: *http, https, ftp, irc...*
- Апликативни слој за рад са базама података: *mysql (phpmyadmin), pgsql (phppgadmin)*.

Резиме:

Пошто је тема преопширна и требаће пар месеци да се разради цела прича и напише (да се не попуни цео број),



рашчланићемо ово све на више целина:

1. *LAMP (Linux, Apache, Mysql, Php, Python, Perl)*
2. *LEMP (Linux, [E]Nginx, Mysql, Php, Python, Perl)*
3. *LAEMPP (Linux, Apache, [E]Nginx, Mysql, Postgresql, Php, Python, Perl)*
4. *Route-X-LAEMPPP* (Све заједно + *custom* портовање и лично мапирана архитектура)
5. *IRC* сервер (*Unrealircd*)
6. *File* сервер (*SAMBA*)

Стартујмо!

Пре првог дела (бр. 1 - *LAMP*) да покушамо најједноставније да објаснимо ста су технологије којима ћемо се служити: *Debian* је линукс дистрибуција настала 1993., која садржи искључиво слободни софтвер. Конфигурисана је за 11 различитих архитектура. Поред система за инсталирање и ажурирање програма *APT*, *Debian* садржи и многе алате за сигурносне аспекте система и његово уређење. Често се користи као дистрибуција за сервере али је популарна и за радне станице (*desktop*). Нуди огроман број програм и програмских пакета. *Debian* је потпуно демократски организован, у рукама

заједнице и све одлуке се доносе у јавности.

Извор: *Wikipedia*



Web сервер *Apache* је web сервер отвореног кода за *unix*-олике оперативне системе, *MS Windows*, *Novel...* *Apache* је један од најкоришћенијих web сервера на интернету. Лако је подесив за пријављивање на системе база података и рад са доста скриптних језика... Такође је подржан од стране више графичких корисничких окружења која имају једноставнији начин подешавања сервера. *Apache* је развијен од стране отворене заједнице програмера под вођством фондације „*Apache software*“. Данас више од 55% од укупно свих web сервера користе *Apache software*.
Извор: *Wikipedia*



Web сервер *Nginx* (*motor-x :: engine-x*) је бесплатан *HTTP* сервер отвореног кода, високих перформанси као и „*reverse proxy*“, подједнако добар и као *IMAP/POP3 proxy* сервер. *Igor Sysoev* је започео развој *nginxa* 2002. године, са првом званичном верзијом 2004. године. На *ngxin* се тренутно „*servira*“ 12,18% (22 200 000 рачунара) активних

интернет презентација свих домена. *Nginx* је познат због његових високих перформанси, стабилности, великог сета опција, једноставне конфигурације и заузимања веома мало ресурса.

Мало појашњење реверсног проксија: *Reverse proxy* ради у име мрежног сервера. Најчешће коришћење реверсног проксија је да се заштити web сервер. Када корисник на интернету шаље упит web серверу заштићеног реверсним проксијем, *reverse proxy* пресреће захтев и проверава да ли су подаци садржани у упиту прихватљиви и да не садрже неке не-*HTTP* податке или неке малициозне *HTTP* команде. Ако су подаци прихватљиви, *reverse proxy* ће примити захтевани садржај од web сервера и проследити га до оригиналног корисника и на тај начин, корисници на интернету никада не приступају директно вашем web серверу.

Извор: *Nginx*

Иначе, најпростије објашњење *Proxy* уопште је једноставно кеширање страница. Акценат стављамо на *proxy* из разлога јер би се *Nginx* у једном од случајева користио као *reverse proxy* и као канал за мапирање портова...



Php (*Hypertext Preprocessor*) је објектно-оријентиран програмски језик намењен првенствено програмирању

динамичних *web* страница. Прве верзије су се звале *PHP/FI* (*Personal Home Page Tools/Forms Interpreter*) и биле су скуп *perl* скрипти, које је развио *Rasmus Lerdorf* за бројање посета на својој приватној *web* страници. То је било негде око 1995. године. После, када је настала потреба за више функција развио је нову верзију у програмском језику *C*, која је могла радити с базама података и омогућила је корисницима да програмирају једноставне динамичке *web* странице. *Rasmus* је одлучио да објави *PHP* као програм отвореног кода, тако да га свако може побољшати. Данас је *PHP* један од најзаступљенијих програмских језика за програмирање *web* базираних програма. Врлине су му јер је јако сличан *C*-у, лако се памти, преносив је...

Извор: Wikipedia



Python је превођени програмски језик развијен од стране *Guido van Rossum*-а 1990 године. По аутоматској меморијској алокацији, *Python* је сличан програмским језицима као сто су *Perl*, *Ruby*, *Smalltalk*... *Python* дозвољава програмерима коришћење неколико стилова програмирања. Објектно-оријентисано, структурно и аспектно-оријентисано програмирање су стилови дозвољени коришћењем *Pythona*. Не треба напомињати да ова флексибилност чини *Python* програмски језик све популарнијим. *Python* се највише користи на линуксу али постоје и верзије за друге оперативне

системе.

Извор: Wikipedia



MySQL је вишенитни, вишекориснички систем за управљањем базама података. *MySQL* је отвореног кода и бесплатан је за употребу. Систем ради као сервер обезбеђујући вишекориснички интерфејс за приступ бази података.

Извор: Wikipedia

PostgreSQL је објектно-релациони систем за управљање базама података, базиран на *POSTGRES* верзији 4.21, развијеног на Универзитету Калифорнија у одељењу за рачунарске науке *Berkeley*. *POSTGRES* је био пионир у неким концептима које су постале доступне у комерцијалним базама тек доста касније. *PostgreSQL* је *open-source* изданак оригиналног *Berkeley* кода. Подржава велики део *SQL* стандарда и обезбеђује многе савремене карактеристике:

- комплексне упите
- стране кључеве (*foreign keys*)
- *trigere*
- прегледе (*views*)
- трансакциони интегритет

PostgreSQL може бити слободно коришћен, модификован и дистрибуиран од стране сваког корисника за било какву употребу, била она приватна, комерцијална или академска.

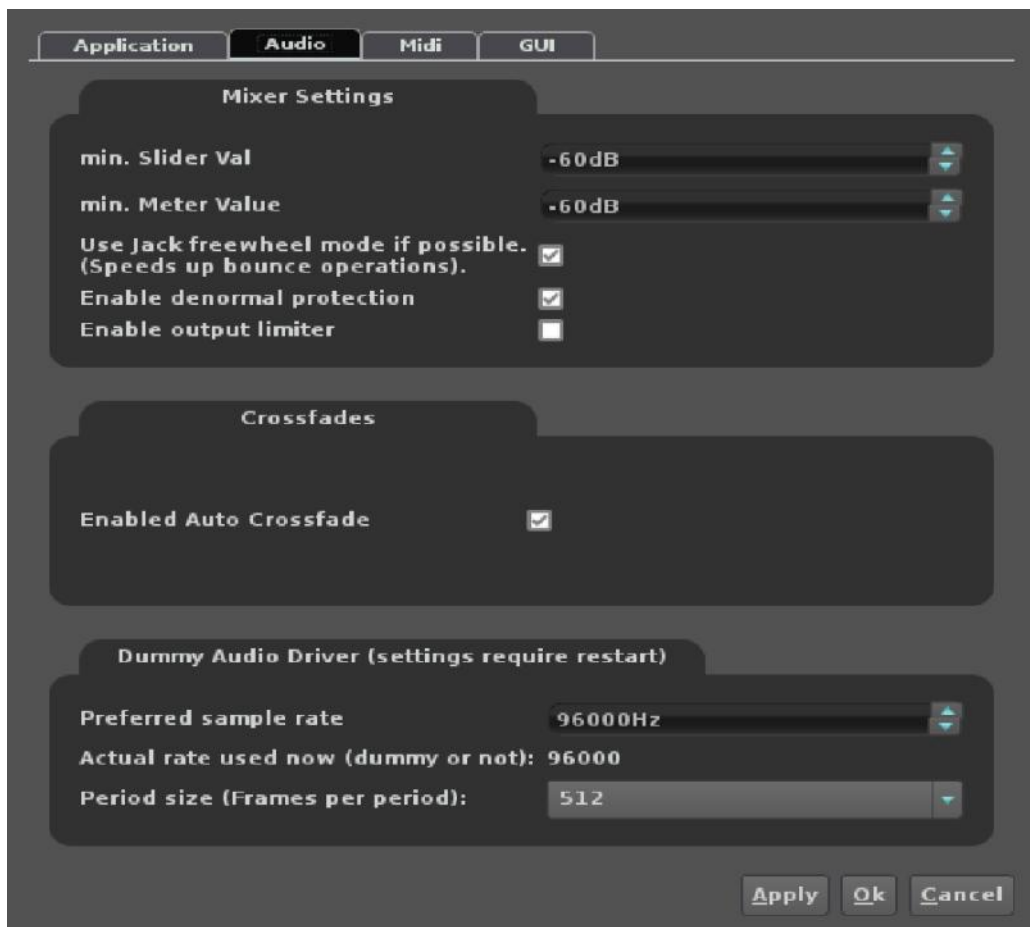
Извор: Wikipedia

GPU/Linux ДАШ Завршни радови

(5. део)

Аутор: Горан Мекић

Снимили сте своје гитаре, снимили сте глас, накуцали бубњеве и снимили или накуцали бас. Оно што желите да урадите је да добијете аудио фајл који можете пустити у аудио плејеру. За почетак, све *MIDI* траке треба да пребаците у аудио. За сваку *MIDI* траку, направите по једну аудио траку. Њене улазе спојите на излазе одговарајућих програма (на пример, *LinuxSampler* ако сте га користили за бубњеве). Спремите поменуте аудио траке за снимање кликом на *r* на траци и кликните на црвено *r* у доњој палети алата. Када пустите песму, свака од



трака би требала да снима са свог улаза и да на тај начин добијете аудио еквивалент сваке од *MIDI* траке. Сада мутирајте све *MIDI* траке и пустите песму још једном, да проверите да ли је све спремно. Оно што обично желите да урадите је да убаците мало *reverb*-а на бубњеве како би добили на ширини.

Све је спремно, цела песма је по Вашем укусу и време је за финални *mixdown*. Песму треба пребацити у аудио формат а то је обично *WAV* датотека. Треба да снимате у *WAV* датотеку као што сте снимали аудио траку. Ово је сувише дуготрајна операција за нешто што се сведе на сабирање сигнала. Из главног менија одаберите *Settings -> Global Settings*. Појавиће се прозор са неколико табова. Одаберите *Audio* таб. Опција *Use Jack freewheel mode if possible* треба да је активна. Кликните на *OK* и поставите заставице (користе се за *loop*, *punch in* и *punch out*) тако да захватају целу песму. Кликните на *r* на траци под именом *Master*. Отвориће се прозор у коме, између осталог, можете да подесите како ће се звати излазна датотека. Након још једног клика на *OK* и пар секунди колико треба *OpenOctave* програму да уради *mixdown* добићете *WAV* датотеку.

Ако мало изанализирате добијену датотеку, увидећете две ствари. Прва, да је Ваша песма нешто тиша него већина музике коју имате на свом рачунару. Начин да се „изборите“ са овим је нормализација. Процес нормализације је у основи прост: скенира се цела датотека и проналази најгласнији део, израчуна се колико је он тиши у односу на најгласнији могући сигнал и за ту разлику се појача комплетна датотека. У пракси, овај процес захтева много компликованију функцију, гледа се *RMS* снага, а не проста амплитуда и слично.

Оно што у већини случајева даје добре резултате у мом искуству је следећа команда:

```
# normalize -a -6dB pesma.wav
```

Да не бисте морали да радите *mixdown* из почетка, пре нормализације сачувајте оригиналну датотеку.

Друга ствар коју сте можда приметили је да фреквенција учесталости датотеке одговара фреквенцији на којој ради *JACK*. То може а и не мора да Вам одговара. Ако имате ситуацију да *JACK* ради на *96kHz*, желећете на крају да пребаците на *44.1kHz*, пошто је то стандард за *CD* а и велики број аудио датотека је баш у овом формату. Команда која ће урадити конверзију, уколико је потребна, је:

```
# sox 'pesma.wav' -r 44100  
'pesma44k1.wav'
```

Датотека под именом *pesma44k1.wav* је резултат Вашег целокупног рада, међутим, постоји пар савета за крај који могу бити корисни. За почетак, уколико сте ентузијаста да поделите Ваш рад са другима, *OGG* формат је много згоднији за слање, пошто је неупоредиво мањи. Команда која ће направити *OGG* од *WAV* датотеке је:

```
# oggenc 'pesma44k1.wav'
```

Добићете датотеку *pesma44k1.ogg* коју можете слати и електронском поштом, уколико није предугачка песма. И за крај, два савета који сам ја добио када сам почео да се бавим аудио обрадом: нека ти гитаре увек имају мањи *gain* него што мислиш да треба и нека су гитаре увек тише у односу на остатак док их слушаш у програму за *DAW*. Разлози су једноставни. Мањи *gain* чини гитаре јаснијим, а нормализација која је описана ће издићи ниво гитара.

Са срећом, аудио ентузијаста!

Фото уређивање и графички дизајн на Линуксу

Растерска графика: GIMP

(2. део)

Аутор: Златан Васовић

У прошлом броју смо Вам препоручили слободне програме за све области графичког дизајна и уређивање фотографија. У овом броју ћемо се бавити растерском графиком и обрадом, ретуширањем и манипулацијом фотографија. Даћемо Вам и основне информације о *GIMP*-у као најважнијем алату за едитовање растерске графике.

Растерска графика: Основни појмови

Дефиниција: Растер или битмапа је структура података представљена у правоугаоној мрежи пиксела (обојених тачака), видљивих помоћу монитора, одштампаних на папиру или приказаних на неком другом медију. Растерска графика у дигиталном свету је природна интерпретација слика која слику рашчлањује на тачке тачно одређене нијансе боје и интензитета. Нијанса и интензитет сваке тачке (пиксела) је одређена бројном вредношћу. На основу те бројне вредности монитор интерпретира слику дајући свакој тачки тачно одређену количину енергије као би се добила потребна нијанса боје и интензитет осветљења. На сличан начин раде и штампачи само што они уместо осветљавања тачке користе пигмент.

Као што смо рекли тачка, пиксел (*px*), је основна јединица слике и одређена је бројном вредношћу. Комплетну слику одређује број тачака по хоризонтали и вертикали те матрице. Постоји још једна врло важна особина растерске слике а то је резолуција. Резолуција је квалитет слике која се огледа у густини тачака по јединици дужине. Најчешће се, у свету дигиталног дизаја, користи јединица за резолуцију *DPI*. *DPI* је јединица за густину броја тачака по инчу. За илустрацију ћемо вам поменути да је уобичајена резолуција за слике намењене за интернет презентације *72dpi*. Ова резолуција је довољно добра да слике буду јасне а довољно мала да датотеке са сликама не буду „тешке“ (у бајтовима) за приказивање на интернету. За штампу се користи већа резолуција, *300dpi*. Неки штампачи могу да постигну и резолуције од *600dpi* или више. Прича о растерској графици не би била потпуна да не поменемо и колорне шеме. Пошто је пиксел одређен само бројном вредношћу различити уређаји би на различите начине интерпретирали боје. Пиксел монитора је састављен из 3 дела од којих сваки део интерпретира једну од 3 основне боје, црвену, зелену и плаву (*RGB*). Ови пиксели се чак на мониторима и ТВ апаратима са катодном цеви могу видети и голим оком или уз помоћ лупе. Интензитетом ових боја добијају се све нијансе *RGB* колорне шеме. *RGB* колорна шема је

стандардизована па сви произвођачи калибришу своје уређаје тако да приближно сви једнако интерпретирају боје из овог спектра. Насупрот мониторима штампачи за интерпретирање нијанси боја користе 4 основне боје: цијан, магента, жута и црна (СМЈК). Мешањем и разблаживањем ових боја добијају се све нијансе спектра те колорне шеме. *RGB* и *СМЈК* се не поклапају у потпуности и због тога треба водити рачуна. Приликом припреме материјала за штампу се употребљава одговарајућа (СМЈК) колорна шема.

Растерски графички формати

У растерској графици постоји много формата а најкоришћенији су *PNG*, *JPG*, *GIF*, *BMP*, *WMF*, *TIFF*, *PICT*, *EPS*, *PSD*...

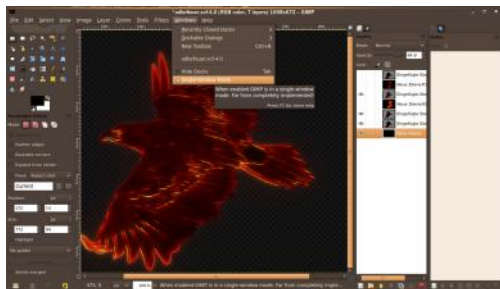
Проблем што сваки пиксел треба дефинисати да би слика била исцртана доводи до тога да би формат који би све дефинисао био огроман. Зато постоји више формата који врше компресију и поједностављење дефиниције слике. Пошто ни један формат не садржи потпуну информацију користимо формате који су прилагођени одређеним наменама. *JPG* и *GIF* су формати намењени *WEB*-у. Њихова компресија података о слици је таква да даје малу, лаку датотеку погодну за дељење на интернету. *JPG* је то решио губитком квалитета оригиналне слике а *GIF* употребом само 256 боја. *PNG* је унапређена *GIF* компресија која дозвољава већи број боја и композитне боје. *TIFF* је професионални формат намењен размени слика између „graphic design“ програма са малим губитком информација.

GIMP

Gimp је програм за растерско цртање и обраду фотографија који представља *Adobe Photoshop* у свету слободних програма.

GIMP је скраћеница од „*GNU Image Manipulation Program*“ - *GNU* програм за манипулацију (обраду) слика. Из самог назива се наслућује намена овог програма. Другим речима он је едитор растерске графике.

GIMP је лиценциран под *GNU* општом јавном лиценцом.



Прва верзија *GIMP*-а објављена је 1996 године. До данас је доживео 11 стабилних верзија. Тренутно актуелна стабилна верзија је 2.8.2 међутим и даље је најпопуларнија верзија 2.6 због великог броја додатака који још увек нису сви компатибилни са најновијом стабилном верзијом. Пројекат је у пуном развоју и већ су у тест верзији објављене верзије 2.10 и 3.0.

Нова стабилна верзија 2.8 доноси редизајниран *save/export* мени, додата је подршка за *JPEG 2000* и *PDF*, *GEGL* (*Generic Graphics Library* - Генеричка графичка библиотека) је у првој стабилној верзији (0.1)...

GIMP је најчешће међу преинста-

лираним програмима већине већих дистрибуција а ако случајно није, обавезно се налази у ризницама свих дистрибуција. *GIMP* је осим за линукс доступан и за *MS Windows* и за *MacOS X* платформу. Осим основног програма *GIMP* кресе и бројни додаци који могу да унапреде рад са овим програмом. Све званичне додатке можемо наћи у *Gimp Plugin Registry*-у (<http://registry.gimp.org/>). Овај регистар plug-in/-ова садржи и форум и документацију који нам могу бити од помоћи.

Додатне четкице можемо наћи на *GimpBrush*-у (<http://gimpbrush.com/>).

Као и код *Photoshop*-а само је машта ограничавајући фактор за овај алат. Не постоји радња у обради растерске слике која се не може урадити са овим едитором. У списак задатака који могу да се обаве овим алатом спада: конвертовање растерских формата, сликање (цртање), ретуширање (коректура) фотографија, фотомонтажа, припрема за штампу, припрема слика за објављивање на интернету...

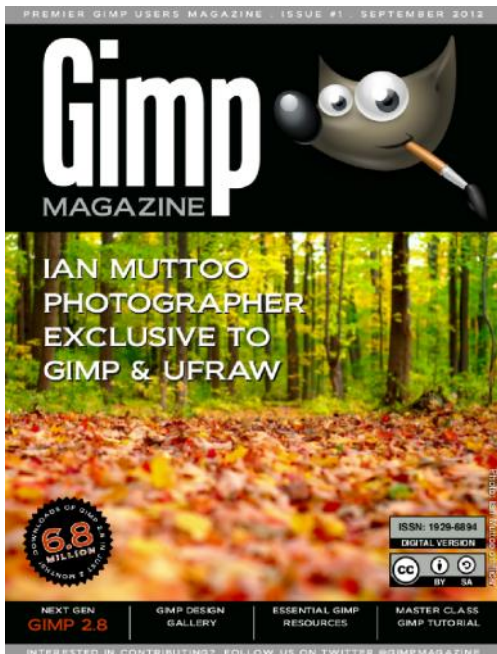
Интернет је препун туторијала у писаном и видео облику за рад у *GIMP*-у. Један од најсвеобухватнијих блогова са гомилом видео туторијала је <http://blog.meetthegimp.org/>

Креативне друштвене мреже

Када будете научили основе, екпериментисали (то је слободна воља) и нацтрали своје прве радове, слободно их поставите на неку од друштвених мрежа за креативност (креативне друштвене мреже). *OpenClipArt* није директно везан за *Gimp*, зато што преферира *SVG* формат.

Креативне друштвене мреже:

- [1] <http://www.colourlovers.com/>
- [2] <http://www.deviantart.com/>
- [3] <http://openclipart.org/>



Корисни линкови:

- [1] <http://gimpmagazine.org/>
- [2] <http://www.gimp.org/docs/>
- [3] <http://gimpforums.com/>
- [4] <http://gug.criticalhit.dk/>

Наставиће се...

APK - Android Package



APK је скраћеница од Android Package file - Android пакет... али не поштански, већ са програмом који (не)желите да инсталирате...

Аутор: Златан Васовић

Најједноставније речено, **APK** датотеке су *ZIP file* форматиран пакети базирани на **JAR (Java ARchive)** формату са ознаком *.apk* на крају. Ово не чуди, јер је *Android* програмиран у *Java-и*. *MIME type (Internet media type - Интернет медија тип)* ове архиве је ***application/vnd.android.package-archive***. **APK** садржи сав код програма (као и *.dex* датотеке), ресурсе, средства, сертификате и манифест датотеку.

Садржај пакета

APK датотека обично садржи следеће директоријуме:

- **META-INF** директоријум:
 - **MANIFEST.MF**: Манифест датотека
 - **CERT.RSA**: сертификат програма - *RSA* формат *PKCS#1*
 - **CERT.SF**: листа ресурса и *SHA-1*

преглед; на пример:

```
Signature-Version: 0.2
Created-By: 0.2 (Android)
SHA1-Digest-Manifest:
wxqnEAI0UA5nOhu78CGMwjkgGWE=
...
Name: res/layout/exchange_
component_back_bottom.xml
SHA1-Digest:
eACjMjESj77hg0cBFTZ0nqWrt7w=
...
Name: res/drawable-
hdpi/icon.png
SHA1-Digest:
DGEqylP8W0n0iV/ZzBx3MW7XgCA=
```

- **lib**: директоријум који садржи компајлирани код који је специфичан за процесор програмског слоја; подељен је у више директорујума у оквиру њега:
 - **armeabi**: компајлирани код само за све *arm* базирани процесоре
 - **armeabi-v7a**: компајлирани код само за све *armv7* и јаче процесоре
 - **x86**: компајлирани код само за *x86* процесоре
 - **mips**: компајлирани код само за *mips* процесоре
- **res**: директоријум који садржи ресурсе који нису компајлирани у *resources.arsc* (видите доле)
- **AndroidManifest.xml**: додатна *Android* манифест датотека која описује

име, верзију, права приступа и референциране датотеке библиотека апликације; може бити у *Android* бинарном (*Binary*) *XML*-у који може бити конвертован у *XML* текст читљив човеку уз помоћ алата као што су *apktool*, *Androguard*, *AXML-Printer2*...

- **classes.dex:** *class*-ови компајлирани у *dex* формату датотека разумљив *Dalvik* виртуелној машини
- **resources.arsc:** датотека која садржи унапред компајлиране ресурсе, као што је бинарни *XML* на пример.

Неки пакети садрже и додатне датотеке и директоријуме који се највише могу приметити у пакетима сложенијих програма (на пример неки *web shop*, продавница програма и слично).

Инсталирање

APK датотеке се могу додати на два начина: преузимањем са интернета и преузимањем са *Play Store*-а. Инсталери програма су управо *APK Installer* (део система) и *Play Store* кога нећете наћи на свим уређајима на којима се извршава *Android*.

Пример процеса инсталације: преузимање програма са интернета -> програм је преузет -> отварање пакета -> приказ захтева програма -> потврда инсталације -> инсталирање уз помоћ *APK Installer*-а -> инсталирање је завршено и можете покренути програм.

Дали смо пример за *APK Installer* зато што је таква инсталација потпунија. *Play Store* процес инсталације делом сакрива али се 90% тог процеса може приметити. Праћење процеса до краја у потпуности је врло битно да бисте

одмах открили грешку (*bug*) приликом инсталације и брзо је решили.

Сличности са осталим пакетима

APK по својој садржини подсећа на инсталационе пакете *desktop* оперативних система (*RPM*, *DEB*...). Највише сличности има баш са инсталационим пакетима за линукс као што су малопре споменути *RPM* и *DEB* због компајлирања, идентификације и слично. Има и сличности са *TAR.xx* (на пример *TAR.GZ*) односно *TAR.x* (на пример *TAR.Z*) пакетима у којима није могуће паковати корисничке датотеке (слике, документа, музика...) са телефона/таблета, већ искључиво системске (конфигурационе, *boot* и сличне датотеке).

Идентификација уз помоћ *RSA* кључева је такође примењена на десктоп линукс дистрибуцијама.

Највише сличности има са *ZIP* и *JAR* форматима од којих је настао. Ипак, највећи недостатак *APK*-а, а и самог *Androida* као оперативног система, јесте недостатак више формата за паковање корисничких датотека где је *ZIP* тренутно подразумеван. На линукс дистрибуцијама за десктоп је ово сређено са форматима из *TAR* и *BZIP* фамилија.

Уредник рубрике: Марко Костић

Жаргон Датотека – Прича:

Прича о Мелу

[Ову причу је на *Usenet* поставио Ed Надер 21. маја 1983 године.]

Скорашњи чланак посвећен мачо страни програмирања је садржао храбру и занимљиву изјаву:

Прави програмери пишу у *Fortran*-у.

Можда је то случај данас, у овој декадентној ери светлог пива, ручних калкулатора и „пријатељски настројеног програма“ али у Старим Добрим Временима када је појам програм звучао смешно и када су Прави Рачунари били састављени од бубњева и вакумских цеви, Прави Програмери су писали машински код. Не *Fortran*. Не *RETFOR*. Не чак ни *assembly*. Машински код. Сирови, нетакнути хексадецимални бројеви. Непосредно.

Како долази нова генерација програмера несвесна ове славне прошлости, осећам се обавезан да опишем најбоље што могу због генерацијског јаза како је Прави Програмер писао код. Зваћу га Мел јер је то било његово име.

Упознао сам Мела када сам се запослио у *Royal McBee Computer Cor-*

poration, фирми која је сад део фирме за производњу писаћих машина. Фирма је правила *LGP-30*, мали, јефтин (мерилима тог времена) рачунар са бубањ-меморијом и тек је почела да производи *RPC-4000*, побољшани, већи и бржи рачунар са бубањ-меморијом. Језгра су била скупа али је то било неважно јер нису дуго трајала ионако. (Због тога нисте чули за фирму и за рачунар).

Мене су запослили да бих написао *Fortran* компајлер за овај нови рачунар и Мел је био мој водич кроз мистичне делове овог рачунара. Мел није одобравао компајлере.

„Ако програм не може променити свој код“, рече Мел, „каква је корист од њега?“.

Мел је написао, у машинском коду, најпопуларнији програм у фирми. Радио је на *LGP-30* рачунару и играо ајнц са муштеријама на сајмовима рачунарске технике. Значај програма је био велики. *LGP-30* је продаван на сваком сајму и *IBM* продавци су били унаоколо и шапутали између себе. Да ли је велика продаја *LGP-30* рачунара била највише због ајнц програма, никад Мела нисам питао.

(Наставиће се...)

Жаргон Датотека – појам:

црни шешир

1. [честа употреба међу специјалистима за безбедност] *Cracker* односно особа која жели да провали у систем који се штити. Супротност белом шеширу који је реткост. Појам сиви шешир означава људе који имају хакерске способности у границама закона, на пример, анализа безбедности система. Сва три термина потичу од честог начина облачења у Вестернима где лоши момци носе црне а добри беле шешире.

2. [одстрањивање непожељног садржаја] „Црни шешир“, „бели шешир“ и „сиви шешир“ се користе да означе труд достављача интернет услуга. Црношеширски достављач „гаји“ спаме и не труди се да их одстрани. Белошеширски достављач их уклања при првом пријављивању а сивошеширски достављачи их уклањају повремено. Употреба овог појма је довело до такозване провере боје шешира достављача.

демонска књига: г.

Књига са називом „Дизајн и имплементација 4.3 *BSD Unix* оперативног система“. Тако је названа због насловне стране на којој се налази ђаволчић у патикама који држи виљушку (алудирање на основну могућност *Unix*-а на *fork(2)* системски позив). Прим.уред: *fork* на енглеском значи виљушка.

Ускршње јаје: г.

[потиче од обичаја ловљења ускршњих јаја у САД и неким деловима Европе]

1. Шала скривена у објектном коду програма која је намењена онима који прегледају код или који растављају објектни код.

2. Порука, слика или звучни ефекат који програм прикаже као одговор на неке недокументоване наредбе или притиске тастера. Најчешће се приказује нека шала или списак људи који су направили програм. Једно од најпознатијих ускршњих јаја је ускршње јаје које се може пронаћи на пар система где на наредбу *make love* („водити љубав“) систем одговори са *//no war* (не рат). Многи лични рачунари имају сложенија ускршња јаја похрањена у *ROM* чипу рачунара. Најчешће су то имена развијача *ROM*-а, политичка мишљења, делови песама и (у једном случају) слика целог развојног тима *ROM*-а.

Стрип:

the BIZARRE CATHEDRAL

BY
MERC + CRIMPERMAN

ШТО СИ
ТОЛИКО
СРЕЋАН?



ОСВЕТА ЈЕ
СЛАТКА КАД СЕ
СЛУЖИ ХЛАДНА



ЗАТО ШТО ТИ ЈЕ
ЗАЛЕПТИО
ЛАПТОП? ШТА
СИ УРАДИО?



СТО МУ...!

НИШТА ПОСЕБНО.
ОСТАВИО САМ
САМО ЈЕДАН ЦД..



ОВО
ЈЕ
НЕЧУВЕНО!!!



ФЕДОРИН
ЖИВИ ЦД!



<http://www.freesoftwaremagazine.com>

(c) Copyright 2008 Ryan Cartwright CC: By-NC-SA

LIBRE! PREPORUČUJE OKLOP

WEB HOSTING, REGISTRACIJA DOMENA, IZRADA SAJTA



Priušтите sebi pouzdan i povoljan Web Hosting kao i najjeftinija Registracija domena i Izrada sajta u Crnoj Gori i Srbiji!



52 besplatne skripte
4,500 besplatnih šablona



Tiket podrška
Podrška uživo

*Svi koji aktivno pišu članke o slobodnom software-u mogu nas kontaktirati za **BESPLATAN HOSTING!***

Neki od njih su i: ivanblago.info kao i otvorenikod.info



info@oklop.me



+382 68 500 322



oklop.me

“Sa nama ste sigurni kao pod Oklopom”

www.oklop.rs ili www.oklop.me

facebook

twitter

myspace

WIKIPEDIA

deviantART

YouTube

Google

