



opensource / Linux

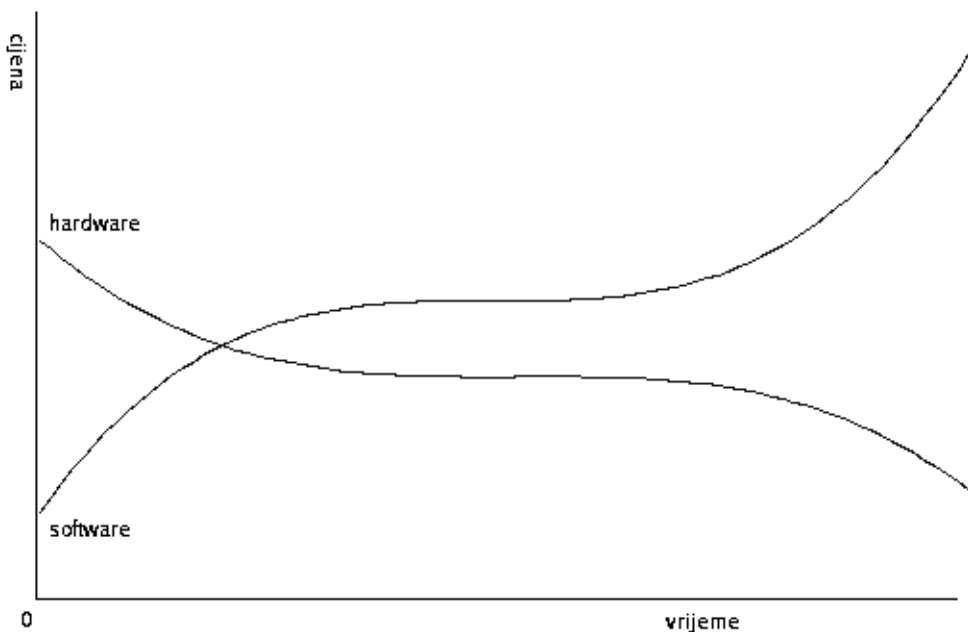
epizoda prva

Ana Stručić
Dalibor Karlović

Uvod
Unix
FSF
Linux
Pitanja i odgovori
Korisni resursi

Uvod

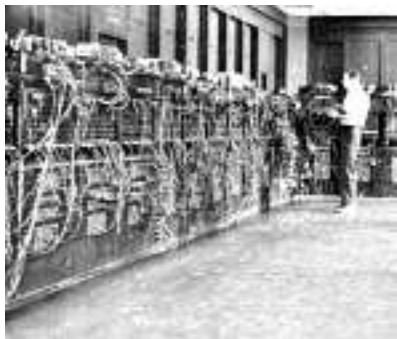
Kada bi netko pravio popis deset najvažnijih izuma dvadesetoga stoljeća, računala bi bila daleko na prvom mjestu. Ako samo na kratko pogledamo iz vlastite perspektive gdje sve računala nalaze svoju primjenu, možemo zaključiti da bi danas život bez njih bio u potpunom kaosu ili bi to bio, u najgorem mogućem slučaju, početak nekog novog srednjeg vijeka. Kako god bilo, činjenica jest da su informatički sustavi iznimno važan čimbenik modernog društva. Banke, poštanski uredi, škole, benzinske postaje, tvrtke, pa čak i običan dućan u susjedstvu gdje kupujete kruh i mlijeko koriste informatičku tehnologiju koja uvelike olakšava poslovanje, vođenje podataka te njihovu točnost.



Grafika 1: Odnos cijena koštanja hardvera i softvera u ukupnoj cijeni računala kroz povijest

Od samih početaka informatike pa do danas, stvari su napredovale velikom brzinom. U tridesetak godina, od prvih gigantskih računala koja su težila nekoliko tona došlo se do računala doslovno veličine ljudskog dlana. Takav napredak

tehničkog tj. hardverskog¹ sklopa, neminovno je vodio i prema napretku softvera², konkretno operacijskih sustava. Ono što se mijenjalo tijekom godina je upravo odnos cijene koštanja ove dvije komponente računala: dok je na početku razvoja računalne znanosti cijena hardvera nadvisivala cijenu prosječne kuće (kako cijenom, tako i fizičkom visinom), tako je kroz godine cijena softvera rasla, a cijena harvera padala. Procjenjuje se da na današnjem uredskom računalu sama cijena fizičkog računala odnosi samo jednu petinu ukupne cijene, dok se ostatak cijene odnosi na softver.



Fotografija 1: Primjer prvih računala, tzv. mainframe

je kreirala te koja ga održava. Kako bismo mogli pričati o opensource zajednici, opensource operacijskom sustavu, hakerskoj etici i sličnim temama, moramo definirati te pojmove.

Izvorni kôd

Računala sama po sebi nemaju inteligencije i sve što znaju napraviti je ono što ih mi naučimo. Učimo ih tako da programiranjem napišemo program koji obavlja

¹ Hardver (eng. hardware – čvrsto oruđe) označava fizički dio računala. Informatičari obično govore da je hardver onaj dio računala koji možete šutnuti. U hrvatskom jeziku ispravan izraz je “očvrsje” iako nikad nije ušao u široku strukovnu upotrebu, osim kao predmet ismijavanja.

² Softver (eng. software – meko oruđe) je programski dio računala, u hrvatskom jeziku “omekšje”, napomena o upotrebljivosti ovdje također vrijedi.

neku operaciju, tek tada računala znaju obaviti tu operaciju. Da bismo ih naučili novu operaciju, moramo napisati novi program. Ono što računala znaju obavljati jesu najosnovnije manipulacije s podacima (nazvat ćemo te operacije **instrukcije**), npr. zbrajanje, ali ne onakvo zbrajanje na kakvo smo mi navikli već mnogo primitivnije. Čak da bismo računalo naučili zbrajati dva broja, moramo

<pre>#include <stdio.h> int main() { printf("Pozdrav svijetu\n"); return 0; }</pre>	2.)	<pre>0000:0000 7f 45 4c 46 01 0000:0010 02 00 03 00 01 0000:0020 60 0d 00 00 00 0000:0030 21 00 1e 00 06 0000:0040 34 80 04 08 e0 0000:0050 04 00 00 00 03 0000:0060 14 81 04 08 13 0000:0070 01 00 00 00 01 0000:0080 00 80 04 08 90</pre>
1.)		3.)

Grafika 2: Kompiliranje: 1.) izvorni kôd, 2.) kompiliranje, 3.) izvršni kôd

napisati program. Tekst programa koji mi pišemo zove se izvorni kôd kao na primjeru.

Sad slijedi novi problem: računalo je glupo. Ne možete napisati program i očekivati od računala da ga shvati već mu morate vaš program nekako prevesti na jezik koji razumije (tj. svesti sve operacije na osnovne instrukcije koje ono poznaje). Ovaj proces se zove kompiliranje³ ili prevođenje. Proces je donekle reverzibilan, ali uz velike probleme i nikad ne možete dobiti onaj izvorni kôd iz kojeg je izvršni potekao.

Zamislite da vi pričate samo hrvatski, vaš poznanik Melvin samo francuski, a vaša prijateljica Mirela oba jezika. Da bi vi mogli primiti neku poruku (npr. ponudu za posao) u Francuskoj od Melvina, morate razumijeti sadržaj. Jedini način je da vam Mirela prevede poruku s jednog na drugi jezik i tek je onda možete razumijeti. U ovom slučaju vi ste programer, računalo je Melvin, a Mirela je kompilator (prevoditelj), program koji vrši prevođenje iz jezika u jezik. Zanimljivo je da je i Mirela nekoć bila nerazumljiva Melvinu (nije pričala francuski, tj. bila je u obliku izvornog kôda). Drugi problem: želite razgovarati s Jorgeom koji zna samo

³ Kompiliranje (eng. compilation – sastavljanje), proces prevođenja izvornog kôda programa u izvršni oblik razumljiv računalu.

španjolski, valja vam se upoznati s Petrom koja zna španjolski i hrvatski (npr. Melvin je jedan iz serije popularnih Pentiuma, dok je Jorge PowerPC, tip procesora koji se koristi u Appleovim računalima). Ideja iza ovog načina rada je da vaš jednom na hrvatskom napisan uradak svaki od tih “sugovornika” dobije na svojem “materinjem jeziku”.

Operacijski sustav

Svima onima koji su rasli uz moderna računala, zavaljeni u svoje udobne stolice pritiskali tu i tamo koju tipku na tipkovnici, koristeći se mišem i šarenim sličicama na ekranu kao glavnim alatima, teško je zamisliva slika bušenja kartica u koje su se tim načinom upisivali programi. Još je teže zamislivo čekanje od najmanje 24 sata da bi dotični program dao neke rezultate, a da ne spominjemo mnogobrojne modifikacije koje je bilo potrebno učiniti svaki put kad bi isti stroj trebao koristiti neki drugi nadobudni korisnik. No ipak, da bi razumjeli kako su današnji operacijski sustavi postali to što jesu, moramo se nakratko vratiti u te davne “prehistorijske” početke.



Grafika 3: Shema računalnih slojeva

operacijski sustav je zadužen da tekst unesen tipkovnicom preda programu koji će ga obrađivati, ali isto tako da taj tekst pohrani na disk, prikaže na monitoru ili ispiše na pisaču prema naredbama programa za obradu teksta. Također, operacijski sustav programerima pruža temelj za izgradnju svojih djela jer im daje neke osnove koja bi oni sami morali (i morali znati) isprogramirati sami. Za to se koriste tzv. API⁵ mogućnosti Osa koje

Prije svega, trebalo bi reći što je to “operacijski sustav” koji se cijelo vrijeme spominje. OS⁴ možemo smatrati “srcem” računala koje nadgleda i usklađuje rad svih računalnih komponenti. Primjerice,

⁴ Operacijski sustav (eng. operating system), ispravan oblik jako često korištenog neispravnog oblika “operativni sustav”. Operacijski zato što dolazi od operacija koje provodi, a ne operativni (radeći, funkcionalan).

⁵ Aplikacijsko sučelje za programiranje (eng. Application Programming Interface) je način na koji program kroz OS pristupa hardveru ili nekim mogućnostima koja mu OS pruža (npr. Iscrtavanje prozora).

dopuštaju da se npr. podrška za digitalnu video kameru ugradi samo na jednom mjestu (u OSu), a kasnije je programeri samo koriste na neki priređeni način.

U najranijim danima elektroničko - digitalnih naprava, svaka operacija se vršila na ogoljelom hardveru. Postojalo je tek nekoliko računala i u biti, sva su ona bila eksperimentalne prirode. Istraživači koji su se bavili razvojem računala u isto vrijeme bili su i programeri i korisnici, a radili su na čistom hardveru, bez ikakvog operacijskog sustava. Svoje programe pisali su u assembly⁶ jeziku, a pokrenut program imao je potpunu kontrolu nad cijelim računalom. Popravljanje i pronalaženje pogrešaka sastojalo se od popravaka i softvera i hardvera – bilo je potrebno izmijeniti dijelove kôda u samom programu isto kao što je bilo potrebno mijenjati i fizičke komponente računala.

Upravo nedostatak operacijskog sustava značio je da samo jedna osoba može upotrebljavati dotično računalo u nekom vremenu. Čak i u istraživačkim laboratorijima, znanstvenici su se borili tko će dobiti više vremena na raspolaganje. Prva mogućnost bila je rezervacija sistema kada su istraživači trebali unaprijed rezervirati svoje mjesto pred ekranom po principu kartica i slotova (usporedbe radi – kovanica i jukebox).

Vrlo visoka cijena prvih računala značila je da tih nekoliko naprava mora biti djelotvorno što je više moguće. Sistem rezervacije računala nije bio od velike koristi. Ako bi jedan od istraživača završio svoj posao prije nego što je planirano, drugi korisnik ne bi mogao koristiti računalo tako dugo dok zadano vrijeme ne istekne. Međutim, ako bi vrijeme isteklo prije nego što bi posao bio dovršen, znanstvenik bi morao “spakirati svoje stvari” ostavivši svoj rad nedovršen. Čak i kad su stvari išle savršeno, previše se vremena trošilo na otkrivanje pogrešaka u programu ili proučavanje rezultata.

Rješenje ovog problema ležalo je u pripremi materijala na nekom od ulaznih medija (bušene kartice, papirna vrpca, magnetska vrpca) nakon čega bi se rad

⁶ Assembly (eng. zbir, skup) je jezik računala, u našem primjeru s jezicima kao da vi naučite francuski umjesto da koristite Mirelu kao posrednika. Naučiti ovaj jezik je teže nego naučiti tri francuska. Svaki tip računala ima svoj dijalekt.

dalje prosljedio tzv. računalnom operateru koji bi bio zadužen za izvršavanje radova onim redom kojim su pristigli. Nakon što je računalo obavilo svoj dio posla, podaci bi se ponovno prebacili na neki izlazni medij (kartica, vrpca ili ispis na papiru), a rezultati bi se vratili natrag svom programeru koji bi u miru mogao proučiti gdje su se dogodile pogreške i što je pošlo krivo.

Kako se upotreba računala sve više povećavala, programeri su primijetili da bespotrebno ulažu dvostruke napore. Naime, svaki programer je pisao svoju proceduru kako će teći uzimanje i ispisivanje podataka s magnetnih vrpca. Imalo je više smisla napraviti jedinstvenu proceduru za svaki medij koju bi tada mogao koristiti svaki programer, umjesto da svatko svaki put iznova sastavlja svoje. I tako su programeri počeli pisati programe koji bi im omogućili lakši i jednostavniji rad s računalom, te što bolju optimizaciju zadanih poslova. Dodatna prednost je bila i ta da se jednom napisana funkcija mogla koristiti uz pretpostavku da je ispravna jer je mnogo puta testirana.

Ponovno, više je smisla imalo da se posao raspodjeli među programerima, nego da se svaki put izmišlja “topla voda” što je u konačnici i dovelo do bržeg razvoja programa.

Zajedničke funkcije i naredbe bile su organizirane u datoteke i mogle su se upotrijebiti za bilo koji program po potrebi. U duhu zajedničkog surađivanja među ranim programerima, te su datoteke bile besplatno distribuirane, što je prvi i najraniji primjer koliko otvoreni kôd (na kojemu se temelji i sam Linux) može doprinijeti razvoju softvera.

Hackeri

Najzlorabljениji termin vezan uz računarstvo svakako je naziv hacker. U javnosti i medijima se najčešće veže uz kriminalne radnje, tzv. računalni kriminal. U zbilji su stvari ponešto drugačije.

Hacker (ili pohrvaćena verzija “haker”) je osoba najčešće natprosječne inteligencije, iznimne znatiželje te osobita poznavanja visoke tehnologije. Uglavnom se vezuju uz računala jer uz složenost sustava dolazi i veća

vjerojatnost pogreške koja se tada može iskoristiti, ali hackeri su svoju karijeru započeli s prvim telekomunikacijskim kanalima. Možemo reći da je osoba koja je potkupila telegrafista da pošalje neku lažnu vijest bila prvi hacker, a pravi hackeri su nastajali kako je rasla upotreba telefona. U SADu su telefonske govornice omogućavale nazivanje ukoliko im se poslao kontrolni zvuk određene frekvencije. Poznata je stvar da je Steve Jobs⁷ financirao svoje prve projekte upravo izradom uređaja koji su ovo omogućavali.

Hackeri su kao subkultura umjereni anarhisti, tj. bore se protiv “sustava”, ali ne batinama i kamenjem nego znanjem. Najčešći primjeri takvog ponašanja su provaljivanje u računalne sustave velikih kompanija, banaka i sl, ali ne zbog činjenja štete na istima, već zbog samog zadovoljstva uspjeha i učenja o sustavu. Ljudi koji održavaju takve sustave i oni koji ih napadaju su dvije strane iste medalje: obje strane su hackeri, samo što su prvi u javnosti ocrtani u pozitivnom svjetlu. Često se događa da hacker, nakon što obavi svoju provalu na sustav, svom “protivniku” (administratoru sustava) pošalje upozorenje o takvoj mogućnosti, kao i o načinima popravka.

Drugi zadaci kojih se hackeri rado prihvaćaju su poboljšavanje postojećih sustava kako bi radio bolje, brže, učinkovitije. Ovdje se jasno vidi primjer koji su zauzeli ljudi iz GNU zajednice koji sa ponosom mogu nositi ime hackera. Kako bi se čak i u nekoj javnosti diferenciralo hackera kriminalca od hackera stvaratelja, potonjima se pridjeljuje titula kernel⁸ hackera. Svakako najpoznatiji kernel hacker je Linus Torvalds, otac Linuxa.

Kako su sposobnosti koje su potrebne da bi se bilo hacker prilično rijetke, činjenica je da poneki hackeri iskorištavaju svoje znanje i sposobnosti u komercijalne svrhe kao što su industrijska špijunaža, krađa ili mijenjanje podataka, prisluškivanje, korištenje tuđeg sustava za svoje potrebe (squatting⁹ ili freeriding¹⁰). Ovi ljudi su kriminalci osim što su hackeri i ne valja cijelu subkulturu

⁷ Generalni direktor (eng. CEO – Chief Executive Officer) računalnog diva Apple

⁸ Objašnjeno dalje u tekstu.

⁹ Squatting (eng.) neovlašteno zauzimanje tuđih resursa

¹⁰ Freeriding (eng.) besplatna vožnja, sinonim za squatting

ocrniti kao kriminalce zbog loših pojedinaca članova subkulture.

Osim hackera, poznajemo i crackere i script kiddie. Cracker je osoba koja pokušava pokvariti nešto što radi, npr. kod eliminiranja dijela programa koji osigurava da ga platite (odatle i popularni naziv “crack”) dok su drugi hacker wannabe koji na Mreži pronalaze gotove programe napisane od pravih hackera i tada ih koriste kao igračku, bez razumijevanja kako rade, kao i želje za razumijevanjem. Niti jedan od ova dva termina nema pozitivno značenje.

Unix

Bilo kakav tekst o Linuxu ne može biti potpun ukoliko se ne spomene njegov “djed” i uzor, Unix. Nastao u '70ima na američkim sveučilištima, i danas se obilato koristi, iako neke mnogo naprednije verzije.

Početak

Dok je veličina nekadašnjih računala uzrokovala velike probleme, kasnije se nametnula još jedna gora stvar: svako računalo radilo je s drugim (primitivnim) operacijskim sustavom. Softver je uvijek bio prilagođen jednoj jedinoj namjeni, a najveći nedostatak bio je taj što softver s jednog računala nije uopće radio na drugome. Značajna je još jedna stvar: operacije su se vrlo razlikovale i to što je netko savršeno dobro poznao “svoj” operacijski sustav nije odmah značilo da će znati raditi i na susjedovom. Takvi problemi uzrokovali su velike poteškoće kako korisnicima, tako i sistem administratorima. No, nagovještaj rješenja nazirao se već sredinom '60-ih godina.

Godine 1965. računalni stručnjaci iz Bell Labs i General Electrica udružili su snage i započeli rad na projektu MULTICS¹¹. Zamisao je bila napraviti jedan jedinstven operacijski sustav, kompatibilan sa različitim konfiguracijama računala. No, projekt nije dobro napredovao i Bell Labs je otkazao suradnju 1969.

Ken Thompson i Dennis Ritchie, obojica računalni stručnjaci iz Bell Labs-a

¹¹ MULTiplexed Information and Computing Service (eng. mnogostruki servis za informacije i računanja)



Fotografija 2: Očevi Unixa i Ca, Thompson i Ritchie '70ih

modificirali su MULTICS da bi na računalu PDP – 7 (Programmed Data Processor, računalu s tada ogromnih 4KB radne memorije) mogli igrati “Space Travel”. Rezultat takvih modifikacija od milja su nazvali UNICS¹² što je u biti igra riječi s originalnim nazivom. No, nakon raskida suradnje ozbiljno su počeli razmatrati mogućnost razvoja operacijskog sustava koji bi bio

alternativa MULTICSu. Ono što su željeli bilo je razviti sustav na boljoj programskoj podlozi od assemblya koji bi bio dostupan svima. Tako je iz želje za igranjem dvojice dokonih programera nastala osnova modernog računarstva.

Prvi koraci

Prva verzija Unix-a ipak je bila napisana u assemblyu iako se od samih početaka težilo nekom višem programskom jeziku.

Prvo je 1971. godine Ken Thompson pokušao Unix “prepisati” pomoću Fortrana¹³ no zbog mnogobrojnih komplikacija odmah je odustao. To mu je bio poticaj da smisli jednostavan programski jezik BCPL¹⁴. U skladu s potrebama smanjenja troškova, Ritchie je napisao skraćenu verziju BCPLa i nazvao ga B. Godine 1972. Ritchie je poboljšao B i rezultat nazvao C¹⁵. Thompson je stvorio koncept tijeka¹⁶, tj. mogućnosti da se izlaz jednog programa pretvori u ulaz drugog

¹² UNiplexed Information and Computing Service (eng. jednostruki servis za informacije i računanja)

¹³ FORmula TRANslator (eng. prevoditelj formula) – jedan od prvih pravih programskih jezika

¹⁴ Basic Combined Programming Language (eng.), osnovni kombinirani programski jezik

¹⁵ C je jedan od najpopularnijih programskih jezika današnjice, izrazito je robustan, producira brz i efikasan kôd. Ogroman dio Windows OSa pisan je u Cu.

¹⁶ Tijek (eng. pipe – cijev) je koncept pomoću kojeg se izlaz iz jednog programa preusmjerava u ulaz u drugi program.

programa. To je omogućilo razvoj Unixove filozofije “kutije s alatom”, tj. svaki pojedini alat ima samo jednu funkciju koju mora obavljati dobro i brzo. Ako se metodom tijeka upari nekoliko alata raznih namjena, može se obaviti proizvoljno težak zadatak. Odavde dolazi i poznati KISS¹⁷ akronim koji je postao moto svakog Unix, a kasnije i GNU programera.



Fotografija 3: Ritchie i Joy

prilagoditi kernel novom okolišu, a sve ostalo je radilo kao i dosad. Svojstvo mogućnost prijenosa OSa ili aplikacije na neku drugu računalnu arhitekturu naziva se portabilnost (prenosivost) i osnovni je pokazatelj dobrog dizajna aplikacije.

U ljeto '74e Thompson je otišao predavati na Berkeleyju, a na njegovo mjesto došao je Bill Joy. Zajedno s Ritchiejem nastavio je razvoj Unixa.

Naslijeđe

Nakon 1979. povijest Unix-a postoje pomalo zbrkana. Naime, te godine izašla je sedma verzija (v7) za koju se popularno kaže da je *...grandfather of all extant Unix systems*¹⁸. I dok je sveučilište u Berkleyu razvijalo svoju varijantu sustava

¹⁷ KISS (eng. Keep It Simple, Stupid – Što jednostavnije, budalo) je koncept koji podsjeća programera da ne pokušava dodati milijun opcija u svoj program nego neka riješi jedan jednostavan problem, ali neka ga riješi dobro.

¹⁸ (eng.), djed svih nadolazećih Unix sustava

koji je nazvan BSD (Berkley Software Distribution), stručnjaci iz Bell Labs nastavili su razvijati Unix pod imenom System III, a kasnije "System V". Krajem '80ih i početkom '90ih bjesnili su ratovi između te dvije strane iako je svaka varijanta usvojila brojne osobine one druge.

Komercijalno gledano, većina prodavača hardvera priklonila se "Sistemu V", no budući da je u njega bilo ugrađeno mnogo inovacija iz BSDa, zaraćene strane odlučile su zakopati ratnu sjekiru i udružiti snage. Rezultat toga su tri BSD verzije Unixa (bazirane na originalnom sedmom izdanju – v7): FreeBSD, NetBSD i Open BSD. Zanimljivo je spomenuti da je svoju verziju Unixa razvijao i Microsoft, XENIX.



Grafika 4: BSD logo

FSF

Kako to obično biva, početak nove ere biva potaknut najmanjim i najbeznačajnijim stvarima.

Početak

Godine 1971. izvjesni Richard Stallman započeo je rad u MIT¹⁹ovim laboratorijima kao suradnik za umjetnu inteligenciju. U '70ima slobodan pristup izvornim kôdu operacijskih sustava bio je raširen i standardan. Međutim, stvari su se počele mijenjati u '80ima. Računalni sistemi postajali su sve zatvoreniji, a izvorni kôd softvera koji se na njima pokretao postao je nedostupan. Kap koja je prelila čašu bio je ni više ni manje nego pisač. Stallman je imao problema s jednim od pisača u svom labosu i želio je sam modificirati upravljački program za njega. Zamolio je autora za izvorni kôd kako bi ga on sam mogao modificirati na način koji bi mu najviše odgovarao. Međutim, dotični gospodin ga je doslovno istjerao iz ureda i Stallman tada odlučuje pokrenuti projekt GNU. Cilj je bio

¹⁹ MIT (eng. Massachusetts Institute of Technology), Institut za tehnologiju sveučilišta Massachusetts, rasadnik visoke tehnologije.



Fotografija 4: Richard Stallman, vođa GNU pokreta

napraviti operacijski sustav kompatibilan s Unixom, a čiji bi izvorni kôd bio dostupan svim programerima koji bi tada imali pravo koristiti ga i modificirati. Godine 1985. osnovan je Free Software Foundation²⁰ koji je pod svoje uzeo GNU pokret.

Danas je FSF organizacija koja promiče slobodu softvera, slobodu upotrebe softvera na bilo koji način, kao i slobodu daljnje distribucije, kako svojeg softvera, tako i softvera treće strane koja ga je odlučila osloboditi.

GNU

GNU pokret Stallman započinje svojim manifestom koji je 27. rujna '83. poslao na Usenet grupu net.unix-wizards,net.usoft:

Starting this Thanksgiving I am going to write a complete Unix-compatible software system called GNU (for Gnu's Not Unix), and give it away free²¹ to everyone who can use it. Contributions of time, money, programs and equipment are greatly needed.

(Počevši ovog Dana zahvalnosti, napisat ću kompletni Unix-kompatibilni softverski sustav nazvan GNU (za Gnu Nije Unix) i dijeliti ga slobodno svakome tko ga može koristiti. Donacije u vremenu, novcu, programima i opremi su vrlo potrebne.)

U manifestu su objašnjeni kako etički, tako i političko – ekonomski razlozi zašto je takav sustav potreban. No, sve bi to ostalo na lijepo sročenom tekstu da čovjek

²⁰ FSF (eng. Free Software Foundation – Zaklada slobodnog softvera) je, iako nastao nakon GNUa, postao glavni pokretač i koordinator GNU pokreta.

²¹ Ovo je prvi i najveći problem cijele stvari: riječ “free” u engleskom ima značenje “slobodan”, ali i “besplatan”, zato se često koristi **free as in free beer** (besplatan kao pivo) i **free as in free speech** (slobodan kao govor). Hrvatski jezik nema ovih poteškoća.

uistinu nije napravio ono što je obećao: napisao je alate koji su potrebni da se emulira Unix sustav. Isprva je počeo sam, ali kako je vrijeme protjecalo, tako su se projektu pridruživali razni drugi programeri.

Osnovni problem ovako velikog projekta je upravo koordinacija: kako će komad kôda kojeg piše čovjek u Kini raditi s onim kojeg piše čovjek u Francuskoj kad se konačno stave na jedno mjesto? U pomoć je došao stari dobri Unix: kako su pojedini autori samo pokušavali iznova napisati sve alate koje ima Unix (a kojih su imali samo izvršne verzije), tada je dovoljno da savršeno oponašaju te alate i njihovo ponašanje. Jedino što je falilo je bio kernel.

Uloženo je mnogo napornog rada da se izradi egzaktna replika Unix kernela (projekt pod imenom GNU/Hurd), ali je zbog kompleksnosti kao i problema s koordinacijom (ovdje su autori morali jako blisko surađivati jer svi rade na istom projektu) posao jako sporo napredovao.

Zajednica



Grafika 5: Logo GNU pokreta

kôd ukoliko ga želi, kao i da navedete odakle vama taj softver.

Zajednica se od svih pravnih zavrzlama kojih u svijetu u kojem vam pokušavaju prodati vašu vlastitu dušu ima mnogo brani licencama koje su razvili. Najpoznatija je svakako GNU GPL²². Ukratko, kaže da sav softver koji je pod tom licencom

²² GPL (eng. General Public Licence) – opća javna licenca

mora biti distribuiran uz svoj izvorni kôd kao i da kupac ili osoba koja ga koristi ima pravo na sve već navedene slobode uz taj softver.

GNU zajednicu čine programeri, grafičari, lingvisti, 3D modelari i ini stručnjaci u svojem polju širom svijeta, nerijetko su ti pojedinci vrhunski plaćeni da bi isti posao koji za GNU zajednicu obavljaju besplatno u svoje slobodno vrijeme obavljali u svojoj kompaniji. Jedino što ih spaja je želja da poboljšaju postojeći sustav (i time spadaju u definiciju hackera, vidi poglavlje Hackeri). Ukoliko ne znate programirati, ne ide vam grafika niti zvuk, ne znate skoro ništa, onda sigurno znate svoj materinji jezik, barem do granice da možete otprilike prevesti neke jednostavne fraze. Lokalizacija softvera je jedan od trenutnih svjetskih trendova i GNU zajednica nosi perjanicu istog.

U zajednici nastaje nekoliko smjerova, tako razlikujemo Open Source Software²³ zajednicu i FSF zajednicu. Razlike između usmjerenja su u nekim finesima oko etičkih pitanja, kao i neki detalji oko stremljenja i upravljanja pa ih nećemo navoditi. Zajednicama je opći cilj uglavnom isti i jako često se potpomažu i područje djelovanja im se isprepliće tako da problema nema.

Ono što ljude najčešće zbunjuje je prodaja slobodnog softvera (na engleskom je to u jednom obliku “prodaja besplatnog softvera” pa možete zamisliti njihovu zbunjenost). Činjenica je da se samim programiranjem slobodnog softvera ne zarađuje, a da proces programiranja košta. Da bi se omogućio daljnji razvoj, zajednica podržava prodaju medija na kojima se nalazi slobodni softver. Novci tako zarađeni vraćaju se zajednici i njime se omogućava nastavak razvoja upravo tog softvera koji se koristi.

Da rezimiramo: glavna misao koja pokreće GNU zajednicu je sloboda i razmjena dobara, ovdje su dobra znanje koje pojedinac ima. Ovakav način razmišljanja je jasno suprotstavljen kapitalističkom ustrojstvu pa nije ni čudno da se pokret nerijetko opisuje kao komunistički ili antiamerički. Možda je upravo to ono što zajednici daje dodatnu draž.

²³ Open Source Software (eng.) Softver otvorenog (pristupačnog, dostupnog) izvornog kôda

Linux

Pretpremijera

Godina je 1991., a u svijetu operacijskih sustava DOS još uvijek ima veliku prevlast, gledajući sad već golemo carstvo osobnih računala. Glasine kažu da ga je Bill Gates kupio od nekog hackera iz Seattlea za 50.000\$ i vrlo mudrom marketinškom strategijom DOS se zavukao u svaki kutak svijeta. Korisnici baš i nisu imali nekog izbora – Appleov Mac bio je bolji izbor, ali astronomska cijena koju si nitko nije mogao priuštiti davala je golemu prednost ogoljelom DOSu.

S druge strane, tu je bio i Unix, ali i on je bio daleko preskup za običnog korisnika budući da su se prodavači visokim cijenama pobrinuli da upravo takve korisnike maknu što dalje (nisu baš imali smisla za marketing). Izvorni kôd UNIX-a bio je strogo čuvan i sve to zajedno frustriralo je PC korisnike diljem svijeta budući da veliki igrači na tržištu softvera nisu donosili nikakvo rješenje.

Tračak nade pojavio se zajedno s MINIX-om. Njega je napisao profesor Andrew S. Tanenbaum koji je u praksi želio pokazati svojim studentima na koji način funkcionira jedan operacijski sustav.

MINIX nije bio baš pun mogućnosti, ali imao je tu prednost da je izvorni kôd bio dostupan svima. Svatko tko bi se dočepao Tanenbaumove knjige 'Operacijski sustavi' imao je pred sobom 12 000 linija izvornog kôda napisanih u assemblyju ili Cu. Ono čega su se prodavači softvera najviše bojali, postalo je početak stvarnosti – svaki programer dobio je uvid u izvorni kôd operacijskog sustava. Andrew Tanenbaum svesrdno je poticao svoje najbistrije studente da što temeljitije i pažljivije prouče način rada OSa – jedan od njih bio je i Linus Torvalds.



Fotografija 5:
Linus Torvalds u
mlađim danima

Novi dečko u gradu

Godine 1991. **Linus Benedict Torvalds** bio je student druge godine računalnih znanosti na sveučilištu u Helsinkiju. 21-godišnji momak volio je eksperimentirati s računalima i često je znao testirati do kojih granica neki program može ići, istraživajući usput način rada do najsitnijih detalja.

Potkraj ožujka '91. Linus je počeo istraživati MINIX i odmah je osmislio kako da ga prilagodi da bi mogao čitati Usenet vijesti sa sveučilišta kod kuće na svom računalu (386DX, 33Mhz, 4MB RAM, 40 MB HDD – zvuči pomalo prethistorijski za ova današnja vremena, ali u tadašnje doba bila je to prava 'pila'). No, Linus se nije zaustavio tek na čitanju. Ubrzo je počeo smišljati kako da još više proširi mogućnosti MINIXa, a namjeravao je napisati program koji bi omogućio preuzimanje datoteka s Mreže i spremanje istih na tvrdi disk, Linus je upotrijebio MINIX kao osnovu na kojoj je izradio program, međutim bilo je nekih problema i Linus tada dolazi na ideju da napravi svoj vlastiti operativni sustav. Pomoć za to odlučio je potražiti na Usenet grupi `comp.os.minix` 25. kolovoza '91.:

Hello everybody out there using minix - I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones.

(Pozdrav svima koji koristite Minix – radim (slobodan) operacijski sustav (samo hobi, neće biti velik i profesionalan kao GNU²⁴) za 386 (486) AT klonove).

Ubrzo mu se javio čovjek također zainteresiran za operacijske sustave i dao mu je pristup FTPu²⁵ gdje je kreirao direktorij `/pub/os/Linux` posvećen Linusovom projektu. **Linux** je bio radni naziv za projekt međutim Linus se nije slagao s tim nazivom misleći da će ga ljudi smatrati nekom vrstom egomanijaka. OS se prema njegovoj zamisli trebao zvati FREAX²⁶, međutim ranije spomenuti čovjek nikako se nije mogao složiti s tim imenom i tako je ostao naziv Linux. Do dana današnjeg.

²⁴ Linus i nije bio prepun entuzijazma.

²⁵ FTP (eng. File Transfer Protocol), najstariji mrežni protokol za prijenos datoteka.

²⁶ Kako vidimo, Linus nije imao niti ukusa za imena.

U to rano vrijeme Linux je bio (i još uvijek jest) samo kernel, a samo sa kernelom čovjek ne može mnogo (iskreno, ne može ništa). Tu uskače Stallman sa svojim GNU pokretom koji je osigurao sve potrebne alate za daljni razvoj Linuxa, a Linux je zamijenio njihov neadekvatan kernel GNU/Hurd pa Linux postaje jedna varijanta GNU sustava, **GNU/Linux**, što je ispravan naziv za cijeli sustav. Kernelu je ime još uvijek samo Linux.



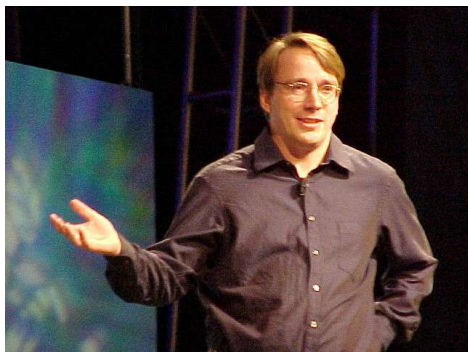
Grafika 6: Linux logo

vrsta veliki kraljevski pingvin. Na daljnje upite napokon se otvorio i napokon opisao nešto što bi njemu bilo simpatično kao logo Linuxa: pomalo tusti i pomalo tromi pingvin koji sjedi i zadovoljno se smješka. Pingvin, a posebno veliki kraljevski pingvin se smješka samo kod dvije stvari, ukoliko se upravo pario ili ukoliko se upravo natrpao ribe. Kako je prva opcija odbačena zbog očitih razloga, a u igri je bila i pomalo tusta figura pingvina, dobili smo logo kakav imamo i danas.

Sazrijevanje

Uz to što je Linux nesumnjivo velika informatička prekretnica, vrlo je zanimljiv i kao socijalni fenomen. Pokret slobodnog softvera kroz pojam Linuxa dobio je veliku pažnju, čak i na marketinškom tržištu gdje je okarakteriziran kao novi brand u softverskoj ponudi – open source.

Ono što je zbunjujuće za mnoge je pitanje kako je moguće da nešto tako uspješno poput Linuxa razvije gomila međusobno neorganiziranih entuzijasta isključivo kao hobi u slobodno vrijeme.



Fotografija 6: Linus Torvals

Najveći faktor uspjeha leži u dostupnosti cijelog kôda, plus GPL licenca koja dozvoljava modifikacije i promjene te objavljivanje istih. Kada neki sustav među svojim korisnicima ima mnogo programera, ako se nađe na problem, on je relativno lako rješiv.

Oko Linuxa se okupilo dovoljno programera-korisnika koji su postupno radili neke manje promjene i poboljšanja. No tada se dogodilo nešto poput efekta lavine – mnogo naizgled sitnih poboljšanja, ubrzalo je razvoj do nevjerojatnih granica, što je rezultiralo time da je privuklo još više entuzijasta programera – korisnika, i tako ukrug.

Linux danas

Dokazavši da su sva upozorenja i zlokobna proročanstva o neuspjehu bila pogrešna, Linux je proslavio desetu obljetnicu svog postojanja i razvoja. Možemo reći da je Linux definitivno najbrže rastući operacijski sustav u povijesti informatičkih tehnologija. Od '91. godine kada je za njega znalo tek nekoliko entuzijastičnih fanatika, Linux danas broji milijune, kako običnih, tako i profesionalnih korisnika, diljem svijeta, što je svakako nevjerojatno dostignuće.

Velike korporacije i tvrtke nisu više na oprezu kad je u pitanju Linux. Koristi ga BBC i njemačka državna uprava, a japanska vlada ozbiljno razmišlja o uvođenju. Mnogobrojne aplikacije koje podržavaju taj operacijski sustav i pouzdanost uvele su Linux u mnogobrojne urede i poslovne prostore. Što se tiče upravo pouzdanosti, Linux se sasvim sigurno dokazao tijekom mnogobrojnih napada virusa. 1999. računalnim prostorom harao je CIH virus, a godinu dana kasnije Love bug, no Linux sistemi ostali su imuni na štete koje mogu prouzročiti ovi

naizgled veoma jednostavni programčići. Jedna od vodećih svjetskih računalnih kompanija IBM²⁷ ulaže kroz posljednjih nekoliko godine milijardu dolara godišnje u razvoj i opremanje Linuxa. Američka vlada uložila je tajne iznose kako bi njena NSA²⁸ razvila SELinux, ekstenziju za kernel koja eksponencijalno povećava sigurnost Linux sustava i dovodi ga u stanje da je spreman za korištenje u najtežim uvjetima kao što je primjer vojske. Nije loše za projekt dokonog studenta koji “neće biti velik i profesionalan kao GNU”.

GNU/Linux

The Soft Revolution



Grafika 7: GNU/Linux logotipovi, simbolika suradnje

puta koristite Linux. Poznata je anegdota da se Microsoft za vrijeme jedne od najvećih napada na njihovu mrežu računala pogodnjenih Windows NT sustavima zaštitio mrežom Linux računala.

Linux problemi

Iako odličan i nezamjenjiv na poslužiteljima, Linux još nije pronašao put do srca kućnih korisnika računala. Problem je u manjku njima poželjnog softvera za Linux (očito, igara) kao i u podršci za novi hardver.

Softver se za široku upotrebu proizvodi s namjerom da se proda. Kako je Linux sustav koji se malo koristi, za njega se ne proizvodi softver jer je fond potencijalnih kupaca manji (da ne kažemo da je teško prodati nekome komad

²⁷ International Business Machines (eng.)

²⁸ National Security Agency (eng.), Agencija za nacionalnu sigurnost

softvera ako ga je on naviknut dobivati ne samo besplatnog, već slobodnog). Kako se za Linux proizvodi malo softvera, ima malo korisnika. Krug je zatvoren. Rješenje ovog problema je da kompanija koja može podnijeti manju dobit kroz neko vrijeme ponese baklju i napravi dobar i kvalitetan Linux softver iako ga neće prodati u onoj količini kao za druge OSove. Dok se to ne dogodi, postoje i naponi da se Windows alati mogu pokretati na Linuxu. Projekt Wine²⁹ i njegova “djeca” postigli su da se razmjerno složena aplikacija može pokrenuti uz zanemarive probleme.

Drugi problem je podrška za nove uređaje. Kao i sa softverom, proizvođačima hardvera se ne isplati proizvoditi upravljačke programe i za Linux. Druga je stvar da Linuxaši vole imati izvorni kôd tih upravljačkih programa, a kompanije koje proizvode hardver boje se za svoje patente koji su strogo čuvana poslovna tajna. Jedino rješenje je da se neki zainteresirani član zajednice potruži i metodom reverznog inženjeringa³⁰ dobije kakav – takav upravljački program. Jasno je da bez točnih specifikacija uređaja ne može dobiti da sve opcije rade kako treba i najčešće uređaj ne radi punom brzinom ili sa svim opcijama koje bi mu bile dostupne uz ispravan upravljački program. Rješenje ovog problema je kupovina samo onih uređaja čiji proizvođači priznaju Linux kao operativni sustav na kojem žele da njihovi uređaji rade. Ovo za svaku pohvalu postaje trend među jačim proizvođačima hardvera. Uređaji koji koriste neke javno propisane specifikacije kao npr. USB³¹ diskovi rade bez ikakvih problema sa upravljačkim programima napisanim od zajednice. Još je gori primjer uređaja s prefiksom “win” u imenu, npr. winmodem ili winpisač. Ovi uređaji nemaju neke osnovne dijelove koji su im potrebni za funkcioniranje, a za njihovo emuliranje koriste vaše računalo. Budući da su jeftiniji od njihove prave braće, vjerojatno imate neki od njih u računalu. Upravljački programi za te uređaje se proizvode **isključivo** za Windows OS pa se ne morate čuditi ako vam neće raditi. Onaj “win” prefiks nije tu bezveze.

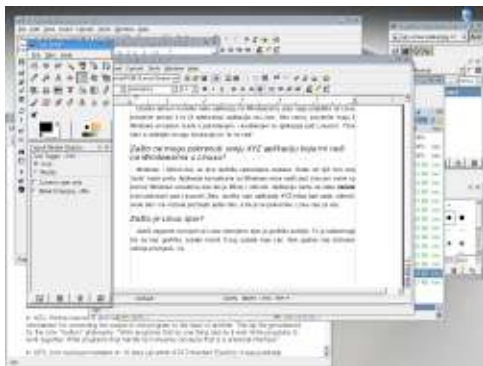
²⁹ WINdows Emulator

³⁰ Mukotrpan postupak otkrivanja kako nešto radi na temelju metode pokušaj/pogreška.

³¹ USB (eng. Universal Serial Bus – univerzalna serijska sabirnica) je sučelje koje nam omogućava da na svoje računalo priključimo najrazličitije uređaje, od prijenosnih diskova do digitalnih kamera.

Pitanja i odgovori

Je li Linux za mene?



Grafika 8: Linux desktop

Ukoliko koristite Windows OS bez problema i zadovoljni ste njime, nema potrebe da ga mijenjate. Suprotno mišljenju, Linux korisnici nisu fanatici i neće vam u bilo kojem slučaju preporučiti Linux kao glavno i jedino rješenje svih vaših problema.

Ukoliko aktivno koristite neku aplikaciju na Windows OSu, prije nego prijeđete na Linux provjerite postoji li ta (ili adekvatna)

aplikacija za Linux. Ako nema, provjerite imaju li Windows emulatori sreće s pokretanjem i korištenjem te aplikacije pod Linuxom. Time ćete si uštedjeti mnogo frustracija jer “smeće ne radi”.

Zašto ne mogu pokrenuti svoju XYZ aplikaciju koja mi radi na Windowsima u Linuxu?

Windows i GNU/Linux su dva različita operacijska sustava. Svaki od njih ima svoj “jezik” kojim priča. Aplikacije kompilirane za Windows neće raditi pod Linuxom (osim uz pomoć Windows emulatora kao što je Wine) i obrnuto. Aplikaciju samu od sebe **nećete** moći pokrenuti pod Linuxom! Zato, ukoliko vam aplikacija XYZ treba baš sada, odmah, svaki dan i ne možete proživjeti jedan dan, a da je ne pokrenete, Linux nije za vas.

Zašto je Linux spor?

Jedini segment na kojem je Linux razmjerno spor je grafičko sučelje. To je uslijed toga što se kao grafičko sučelje koristi X.org sučelje koje već 15ak godina nije doživjelo važnije promjene. Osim toga, sporost dolazi i od činjenice da je grafičko sučelje izdvojeno iz kernela (što kod npr. Windows Osa nije slučaj) pa je

po cijenu robusnosti izgubljena brzina (ako se grafičko sučelje sruši, Linux radi i dalje). U skoroj budućnosti nadamo se poboljšanju (ili čak i promjeni) kompletnog grafičkog sučelja kako bi se osigurale dobre performanse uz dobar izgled i funkcionalnost. Primjer za to je Mac OSX koji je izgrađen na Unix temelju, ali uz vrhunsko grafičko sučelje.

Zašto je tu sve komplicirano? U Windows ja to s nekoliko klikova...

Uz ovaj stav preporuka je da ostanete na postojećem OSu. Filozofija rada Unixa (kojeg je Linux klon) i Windowsa se jako razlikuje pa je normalno da se korisnik izrazito naviknut na jedan teško prilagođava drugome (da, vrijedi i za Linux korisnika prisiljenog raditi na Windows OSu, osjećaj skučenosti i iritacije je stalno prisutan).

Windows je prilagođen radu korisnika – početnika te ga pokušava spriječiti da uradi glupost ili nešto što “ne želi”. Linux je po prirodi takav da pretpostavlja da je korisnik pametniji od njega samoga te da korisnik zna što želi uraditi, a tko je OS da mu sudi, on je tu da slijedi naredbe. Upravo taj faktor (Linux zajednica to naziva “usrećivanje”) izaziva najveću odbojnost Linux korisnika prema Windows OSu.

Ono što je sigurno da je krivulja učenja Linuxa strma na početku, ali nakon što prođete kroz početničke muke, postajete ne samo stručnjak za Linux već i za računala jer OS ne pokušava od vas sakriti da su računala upravo ono na čemu se izvršava.

Korisni resursi

<http://www.kernel.org>

Web sjedište Linux kernela sa svim dosad izdanim verzijama.

<http://www.gnu.org>

Web sjedište GNU projekta.

<http://www.tldp.org>

The Linux Documentation Project, sva moguća Linux dokumentacija.

<http://www.distrowatch.com>

Web čija je glavna uloga recenziranje gomile Linux distribucija.

<http://www.freshmeat.net>

Najveći repozitorij OSSa na Mreži.

<http://www.linuxcompatible.org>

Hoće li vaš uređaj raditi pod Linuxom?

<http://www.linuxquestions.org>

<http://www.linuxbasis.com>

Web sjedišta namijenjena Linux – početnicima.

<http://www.linuxjournal.com>

<http://www.linuxgazette.com>

Linux online časopisi.

<http://www.linuxlinks.org>

<http://www.linux-directory.com>

Repozitoriji Linux linkova, nešto bogatiji od ovoga.


point

25