Sistemska administracija UNIX / LINUX operacijskih sustava

Ćosić, Josip

Sveučilište Jurja Dobrile u Puli
Fakultet ekonomije i turizma
«Dr. Mijo Mirković»

JOSIP ĆOSIĆ

SISTEMSKA ADMINISTRACIJA UNIX/LINUX OPERACIJSKIH SUSTAVA

Diplomski rad

Pula, 2019.

Sveučilište Jurja Dobrile u Puli
Fakultet ekonomije i turizma
«Dr. Mijo Mirković»

JOSIP ĆOSIĆ

SISTEMSKA ADMINISTRACIJA UNIX/LINUX OPERACIJSKIH SUSTAVA

Diplomski rad

JMBAG:0303052025, redoviti student

Studijski smjer: Poslovna informatika

Predmet: Informacijski management

Znanstveno područje: Društvene znanosti
Znanstveno polje: Poslovna ekonomija
Znanstvena grana: Poslovna informatika

Mentor: doc. dr. sc. Darko Etinger

Pula, travanj 2019.

IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani _____, kandidat za magistra ekonomije/poslovne ekonomije ovime izjavljujem da je ovaj Diplomski rad rezultat isključivo mojega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Diplomskog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

_____

U Puli, _____, _____ godine

IZJAVA

o korištenju autorskog djela

Ja, _____ dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj diplomski rad pod nazivom _____ _____ koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, _____ (datum)

Potpis

_____

Sadržaj

1.Uvod

Sistem administrator kao jedna od uloga u IT odjelu često se zanemaruje te mu se ne pridodaje velika pažnja. Kao osoba zadužena za instaliranje, konfiguraciju te održavanje IT infrastrukture važna je fokalna točka koja podržava rad poslovnih organizacija i njenih svakodnevnih operacija. Osim navedenih obveza, sistem administrator također osigurava pravilan rad, performanse i sigurnost računala te time zadovoljava potrebe korisnika, s time da ne premašuje zadani budžet. Tema diplomskog rada odnosi se na sistemsku administraciju Unix/Linux operacijskih sustava s naglaskom na utjecaj rada poslovnih organizacija. Cilj je bio istražiti općenito ulogu sistem administratora, razlučiti što radi u poslovnim organizacijama, koji su benefiti rada sistem administratora, na koji način, kojim metodama i alatima utječe na rad poslovnih organizacija te time osigurava pravilan rad poslovnih procesa. Rad sistem administratora u poslovnim organizacijama usko je vezan za IT management. Sve aktivnosti sistem administratora mogu se povezati na neki financijski način ili na neke bazične funkcije managementa.

Produkt rada sistem administratora utječe na budžetiranje, osoblje, promjene u vodstvu te organiziranje i kontroliranje. Poslovne organizacije počivaju na profitabilnosti i likvidnosti. Korištenje određenih metoda i alata utječe na ROI kao najvažniju metodu a na koju sistem administrator utječe indirektno, korištenjem metoda i određenih standarda postiže se financijski dobitak. Kroz sam početak rada uvodimo se problematiku IT infrastrukture (kreiranje, management, komponente, tipovi te budućnost IT infrastrukture) kao stalnog staništa sistem administratora. Potom prelazimo na neke osnove sistem administracije, temelje svakodnevnog rada administratora. Samim time da titula zaposlenja sistem administratora polagano gubi na značaju, bitno je spomenuti prijelaz u DevOps ulogu gdje sistem administrator poprima novu ulogu i značenje. U radu se istražuju pojedina područja gdje sistem administrator svojim radom utječe na poslovna okruženja kao što su administracija korisničkih računa, upravljanje podacima i pohranom, nadziranje rada IT infrastrukture, automatizacija, upotreba kontejnerske tehnologije kao i važnost sigurnosti u poslovnim organizacijama.

2. IT infrastruktura

Drugo poglavlje bavi se tematikom vezanom za infrastrukturu, njezinim vitalnim komponentama, kreiranjem infrastrukture, upravljanjem infrastrukturom, raznim tipovima infrastrukture te budućnošću same IT infrastrukture.

Infrastruktura (IT infrastruktura) ima veliku ulogu u radu sistem administratora. To je iznimno bitna stavka kod sistem administratora pošto je stavljen u samo središte nje i glavna zadaća mu je vitalne dijelove infrastrukture održavati. Infrastruktura je temelj ili okvir koji podržava sustav ili organizaciju. U računalstvu, računarstvu ili informacijskoj tehnologiji, infrastruktura se sastoji od fizičkih i virtualnih resursa koji podržavaju protok, pohranu, procesiranje i analizu podataka. Infrastruktura može biti centralizirana u samom sklopu podatkovnog centra ili može biti decentralizirana i biti rasprostranjena preko par različitih podatkovnih centara koji su kontrolirani od strane samo organizacije ili od treće strane kao što je pružatelj cloud usluga.

2.1. Komponente infrastrukture

Infrastruktura podatkovnog centra sastoji se od struje, hlađenja i građevnih elemenata koji pružaju potporu hardveru podatkovnog centra. Infrastruktura hardvera podatkovnog centra uglavnom sadrži servere, podsustave za pohranu, mrežne uređaje, switch – eve, router – e, fizičke kablove te namjenske mrežne aparate, kao što su mrežni vatrozidovi.

Infrastruktura podatkovnog centra također zahtjeva pažljivu usklađenost sa sigurnošću IT infrastrukture. To može uključivati fizičko osiguranje zgrade, kao što je ulaženje pomoću elektroničkog ključa, konstantni video i ljudski nadzor nad vitalnim dijelovima, pažljivo kontrolirani pristupi serverima i prostoru za pohranu. Sve to omogućuje pristup infrastrukturi podatkovnom centru samo autoriziranom osoblju i smanjuje rizik pojave opasnosti ili krađe podataka.

Van podatkovnog centra postoji i internet infrastruktura, koja uključuje medije za prijenos, kao što su optički kablovi, sateliti, antene, router – i, agregati, repetitori i ostale mrežne komponente koje kontroliraju prijenosne puteve. Internet infrastruktura je dizajnirana, građena i kontrolirana od strane pružatelja usluga. Kad poslovni zahtjevi zatraže ISP[1], ISP uglavnom se veže za infrastrukturu podatkovnog centra s dodijeljenim i osiguranim prostorom za gradnju.

Uloga računalstva u oblacima mijenja na koji način infrastruktura se dizajnira i implementira. Računalstvo u oblacima omogućuje organizacijama da pristupe podatkovnom centru i njegovim uslugama određenog poslužitelja za određenu svotu. Infrastruktura kao servis (IaaS[2]) dopušta fleksibilnije računalstvo bez potrebe da korisnik lokalno ima resurse.

Softver kao servis (Saas[3]) pruža slične usluge za slične radne potrebe. Treća strana pruža hardver, softver, servere, pohranu i ostale infrastrukturne komponente te dopušta korisnicima da pristupe resursima pružatelja usluga umjesto da postavljaju i održavaju sve resurse lokalno.

[1] ISP – Organizacija koja pruža usluge pristupa, korištenja ili sudjelovanja u Internetu.

[2] Infrastrucure as a service – Forma računalstva u oblaku koja pruža virtualizirane računalne resurse putem Interneta.

[3] Software as a service – Model distribuiranja softvera u kojem treća strana pruža usluge korištenja aplikacija putem Interneta.

2.2. Kreiranje infrastrukture

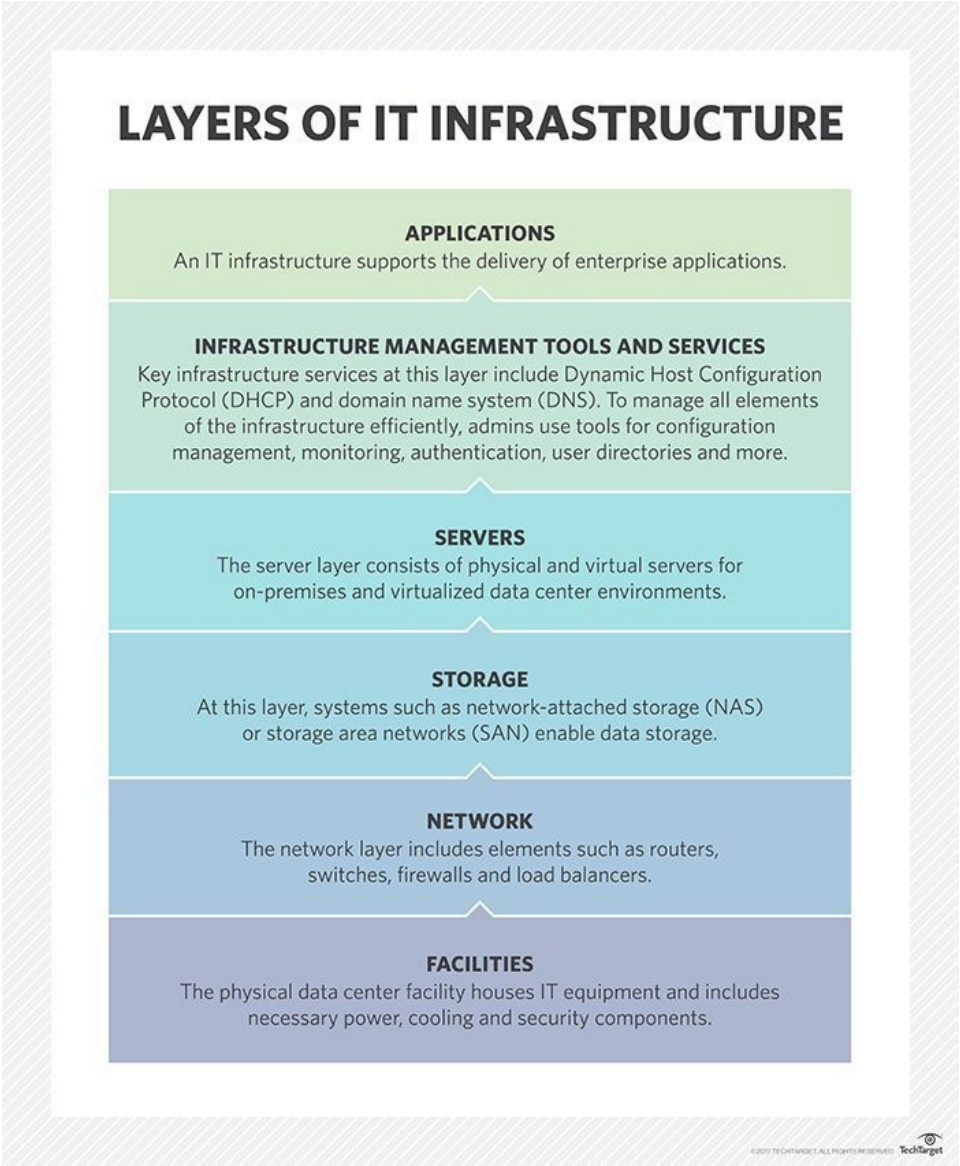
Tijekom kreiranja tradicionalne infrastrukture podatkovnog centra, organizacije najčešće slijede formalizirani proces koji počinje analiziranjem i pristupanjem poslovnim ciljevima, izrada arhitekture i dizajn odluka, građa i implementacija dizajna te potom optimizacija i održavanje infrastrukture. Proces uglavnom sadrži detaljnu ekspertizu, uključujući dizajn gradnje podatkovnog centra, podsustav i selekciju komponenti i kvalitetne tehnike izgradnje.

Unatoč tome, način na koji se IT infrastruktura kreira se kontinuirano mijenja. Tradicionalna heterogena infrastruktura i razvoj visoko su manualni procesi koji zahtijevaju enormnu integraciju, optimizaciju i upravljanje sustavom, pogotovo kad je u pitanju povezivanje servera, pohrane, mreže i ostalih komponenti od različitih dobavljača.

Današnji dobavljači nude već unaprijed integrirane i optimizirane kolekcije računala, pohrane i mrežne opreme koja optimizira IT hardver i platformu za virtualizaciju u jedan sustav koji može biti dodijeljen, proširen i lako upravljiv. Ovakav modularan pristup zovemo pretvorena infrastruktura (CI[4]). Ovakva notacija napredovala je do sustava s jednim dobavljačem, koja nudi užu integraciju i upravljanje svim resursima. Ovakav pristup zovemo HCI[5].

[4] Converged infrastrucure – Način na koji se strukturira informacijska tehnologija koja grupira više komponenti u jedan optimiziran računalni paket. Neke od komponenti jesu: serveri, uređaji za pohranu, mrežna oprema i softver za upravljanje IT infrastrukturom (automatizacija i orkestracija).

[5] Hyper – converged infrastructure – Kombiniranje pohrane, računalstva i mreže u jedan radni sustav. Smanjuje se kompleksnost podatkovnih centara i povećava se skalabilnost.

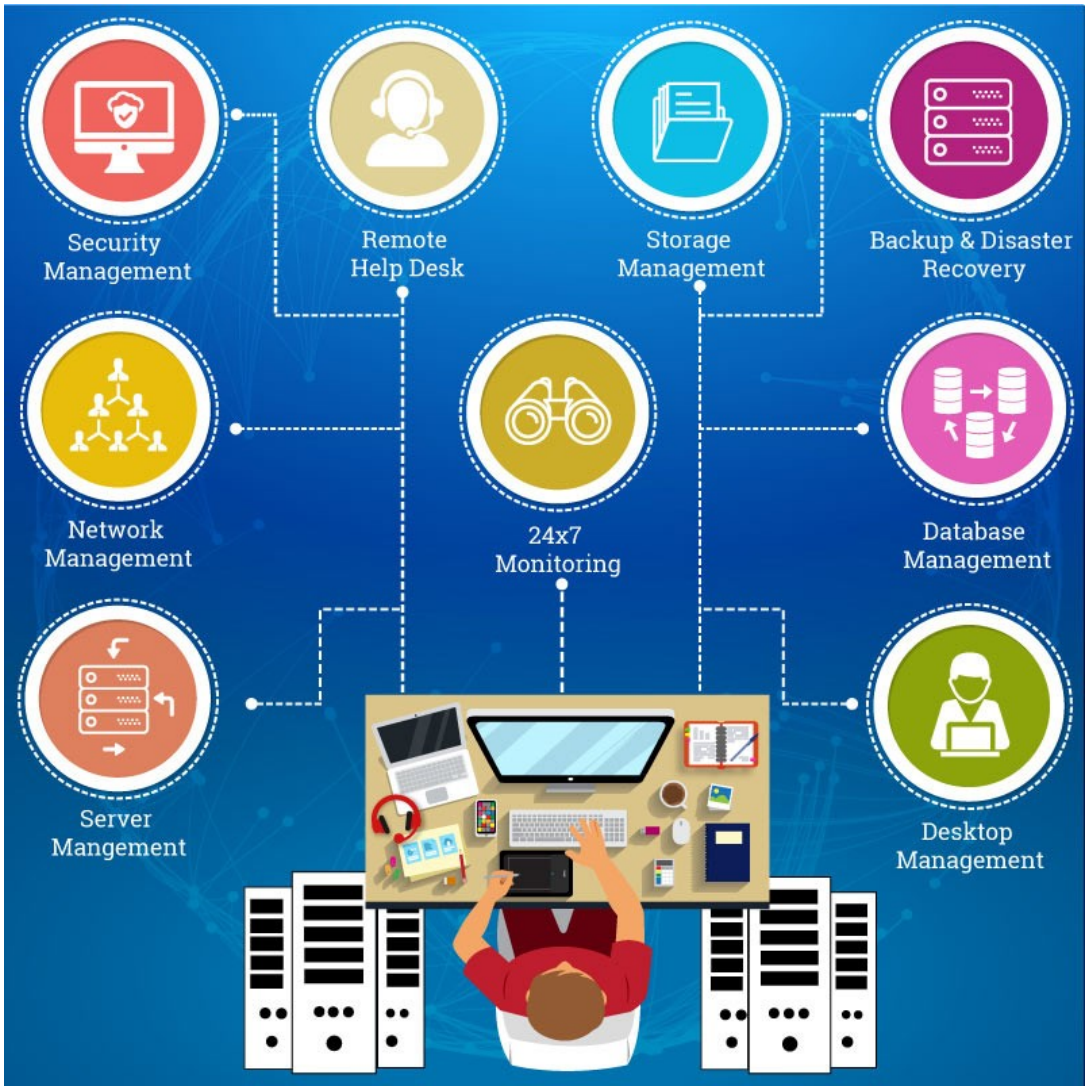


Slika 1. Prikazuje slojeve IT infrastrukture.

Izvor: Search Dana Centar, TechTarget, infrastructure (IT infrastructure), autor: Margaret Rouse, na web stranici: https://searchdatacenter.techtarget.com/definition/infrastructure (pristupljeno: 20. travnja, 2019. godine.)

2.3. Management infrastrukture

Bez obzira kako je kreirana, IT infrastruktura mora pružiti platformu za sve aplikacije i funkcije koje organizacija ili pojedinac treba. To znači da dizajn i implementacija bilo koje IT infrastrukture mora podržavati efikasan management infrastrukture. Softver alati moraju pružiti IT administratorima pogled na infrastrukturu kao pojedinačan entitet, kao i pristup i konfiguraciju bilo kojeg uređaja unutar infrastrukture. Sve to dovodi do efikasnijeg i efektivnijeg managementa infrastrukture. Solidan management također dopušta sistem administratorima da sami optimiziraju resurse za određene poslove.



Slika 2. Prikazuje strukturu managementa IT infrastrukture.

Izvor: Netweb.biz, IT Infrastructure Management, na web stranici: https://www.netweb.biz/work/it-infrastructure-management/ (pristupljeno: 24. travnja, 2019. godine.)

Management infrastrukture najčešće je podijeljen na više kategorija. Sustav managementa uključuje širok spektar alata za IT tim koji koriste za konfiguraciju i upravljanje serverima, pohranom i mrežnim uređajima. Alati sustava managementa proširuju se na podršku lokalno udaljenim podatkovnim centrima, skupa sa privatnim i javnim oblacima i njihovim resursima. Alati za upravljanje također koriste i automatizaciju kako bi poboljšali efikasnost te smanjili pogreške u radu.

2.4. Tipovi infrastruktura

Kako poslovne potrebe i dostupna tehnologija napreduju, organizacije koriste raznoliki asortiman infrastruktura podatkovnog centra koji zadovoljavaju poslovne potrebe i ciljeve.

Nepromjenjiva infrastruktura je pristup za upravljanje uslugama i implementaciju softvera za IT resurse gdje komponente mogu biti radije mijenjane nego u potpunosti zamijenjene. Aplikacija ili servis efektivno se mijenjaju svaki put ako dođe to ikakve promjene. Npr. zakrpa može ažurirati određenu aplikaciju, ali nepromjenjiva infrastruktura ne podržava nešto takvo, umjesto toga IT implementira noviju aplikaciju.

Skladna infrastruktura okvir je koji tretira fizički izračun, pohranu i mrežne resurse kao servise. Resursi su logički udruženi, tako da administratori ne moraju fizički konfigurirati hardver kako bi imali podršku specifične softver aplikacije. Administrator može sam organizirati i upravljati resursima putem softver alata koji koriste veliku razinu automatizacije.

Dinamična infrastruktura okvir je koji automatski određuje i prilagođava sebi promjene u opterećenju. Time se smanjuje vrijeme i potreba za upravljanjem infrastrukturom i znatno smanjuje pogreške, pritom osigurava da se resursi iskorištavaju što efikasnije. IT administratori mogu također birati da upravljaju resursima manualno.

Kritična infrastruktura okvir je gdje su resursi iznimno esencijalni za daljnje operacije te time se osigurava sigurnost određene nacije, njezina gospodarstva, zdravlja i sigurnost javnosti. Koncept koji okružuje visoku dostupnost (HA[6]) i otpornost ključni su ovdje, često uključuje udaljene podatkovne centre i resurse računalstva u oblacima za podupiranje viška radnog opterećenja.

Infrastruktura kontaktnog centra okvir je sastavljen od fizičkih i virtualnih resursa koje ustanova za pozivni centar mora učinkovito iskorištavati. Infrastrukturne komponente uključuju automatske distributere, integrirane jedinice za glasovnu reakciju, integraciju računala i upravljanje redom reda čekanja.

Infrastruktura oblaka sadrži sloj apstrakcije koji virtualizira resurse i logično ih prikazuje korisnicima preko interneta kroz aplikacijske programe sučelja i naredbenog retka omogućenog za API ili grafička sučelja. Dodatne mogućnosti uključuju korisničku samoposlugu, automatsku naplatu ili storniranje i izvješćivanje na strani korisnika, tako da korisnici mogu vidjeti resurse i servise koje implementiraju, kao i odgovarajuće troškove. Slično tome, infrastruktura za skladištenje oblaka je okvir sastavljen od hardvera i softvera koji podržava računalne zahtjeve privatnog ili javnog servisa za pohranu u oblaku.

Tamna infrastruktura dio je okvira koji se sastoji od nedokumentiranog, ali aktivnog, softvera ili usluga čije postojanje i funkcija nije poznata administratorima sustava unatoč činjenici da bi to moglo biti integralno za nastavak rada dokumentirane infrastrukture. To se često naziva Shadow IT[7], i to može postati ozbiljna sigurnost ili ranjivosti za organizaciju.

[6] High availability (HA) – Karakteristika sustava čime se osigurava dogovorena razina operacijskih performansi (pouzdanost sustava).

[7] Informacijski sustavi građeni i korišteni unutar organizacije bez službenog odobrenja.

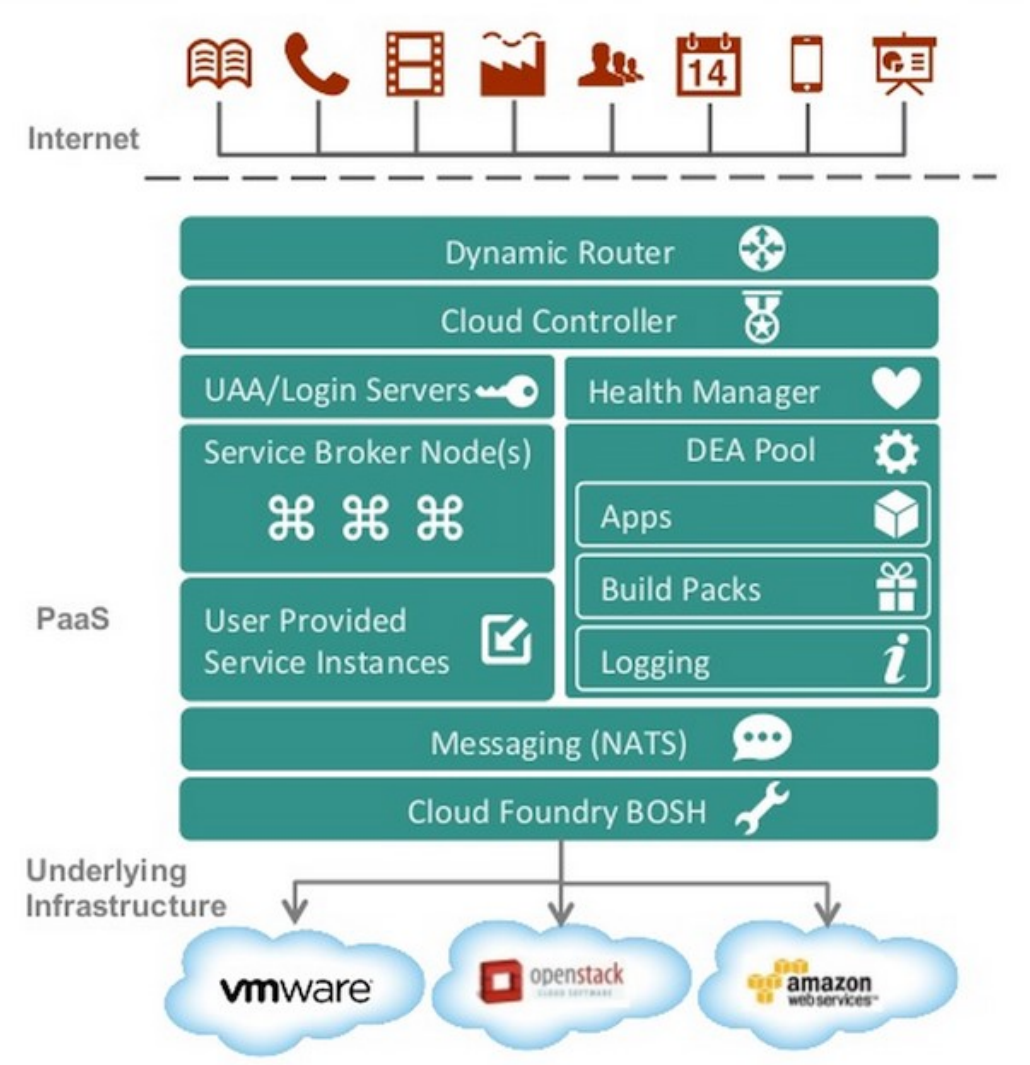
2.5. Budućnost IT infrastrukture

Hipervizori (virtualni operacijski sustavi, VirtualBox…) će i dalje biti dio podatkovnih centara dokle god postoje podatkovni centri, što zbog financijskih razloga ali i tehničkih. Organizacije koje dugoročno ulažu u transformaciju tehnologije imaju poslovni profit. Teško je imati podršku aplikacije od jedne verzije operacijskog sustava pa do druge verzije bez refaktoringa. Velika većina IT trgovina i dalje koristi stare verzije Solaris OS – a, zbog nemogućnosti refaktoringa ili kupnje nove verzije aplikacije koja se pokreće na novijim operacijskim sustavima. Predviđa se sve veća upotreba infrastrukture bazirane na hipervizor bazi zbog podrške za virtualne operacijske sustave, dok će fizički sloj biti malo drukčiji. Razvijat će se jednostavnije metode za implementaciju infrastrukture bazirane na virtualnim operacijskim sustavima. Kako se sve više širi utjecaj virtualizacije operacijskih sustava nad podatkovnim centrima, korisnici će prihvaćati hiper promjenjivu infrastrukturu i računalstvo u oblacima kako bi se upravljanje i podrška učinili jednostavnijim.

Zalet apstrakcije IT infrastrukture privremeno je zaustavljen. Kontejneri nude aplikacije bazirane na mikrouslugama (aplikacije građene kao modularna komponenta ili servis) kako bi lakše prenijeli trenutnu konstrukciju infrastrukture. Velika je mogućnost da će kontejneri u potpunosti zamijeniti virtualizaciju kao primarnu jedinicu unutar podatkovnog centra. Kontejner će dalje napredovati u pogledu performansi i sigurnosti. Kontejneri nude agilnost razvojnom toku aplikacije. Aplikacije se brzo razvijaju i pakiraju u takvom okruženju. Samo pakiranje koje se koristi za razvoj aplikacija koristi se i za implementaciju aplikacija. Kontejneri se tu vide kao ovlast DevOps – a. Kontejneri su formirali bazu za mikrousluge. Korištenjem softvera kao što je Google Kubernetes i Metosphere portfelj proizvoda, visoko skalabilni i prijenosni sustavi koriste kontejnere za implementaciju kompleksnih servisa. Napredni podatkovni centri koriste alate za prijenos mikrousluga od privatnih do javnih računalnih oblaka.

Glavni cilj je postizanje fokusa na servise bez poslužitelja (serverless). Termin koji je skovan od strane AWS Lambde i IBM OpenWhiska – a. Lambda je kreirala mikrousluge bazirane na događajima. Primjer toga je mogućnost lociranja vozila pomoću GPS – a ako je obavijest primljena preko aplikacije za dijeljenje prijevoza.

Još jedna usluga bez poslužitelja je i Cloud Foundry, usluga bazirana na PaaS[8] (platforma kao usluga) konceptu. Developeri kreiraju kod i šalju ga Cloud Foundry, kod zaobilazi koncepte infrastrukture, kao što su kontejneri ili virtualni operacijski sustavi. Cloud Foundry zove usluge kao što su MongoDB za usluge baze podataka i CloudBees za kod aplikacije. Nema potrebe za specificiranjem poslužitelja unutar koda.



Slika 3. Prikazuje arhitekturu Cloud Foundry (PaaS) platforme

Izvor: Medium, Scalable, How to Deploy WSO2 Middleware on Cloud Foundry, autor: Imesh Gunaratne, objavljeno: 1. svibnja, 2016. godine na web stranici: https://medium.com/scalable/how-to-deploy-wso2-middleware-on-cloud-foundry-3b50291734e2 (pristupljeno: 24. travnja, 2019. godine.)

[8] Platform as a Service (PaaS) ili Application Platform as a Service (aPaaS) – kategorija usluga u oblaku koja pruža platformu za razvijanje, pokretanje i upravljanje aplikacijama bez potrebe za gradnjom i održavanjem infrastrukture.

Budućnost infrastrukture je kreiranje i podrška resursima virtualizacije, kontejnerima i uslugama bez poslužitelja. Moguće su kombinacije ta tri modela. Nepoznato je kako će sva tri modela, tj. njihovi određeni slojevi biti usklađeni. Infrastruktura ima mogućnost implementacije kontejnera kao PaaS opcije na već postojećem hipervizoru. IT arhitekti unutar svojih mogućnosti imaju i kombinaciju cijele infrastrukture baziranoj na kontejnerima gdje se KVM[9] (Kernel – based – Virtual Machine) može se pokrenuti unutar samih kontejnera kako bi pružali hipervizorsku[10] podršku aplikacijama.

Kako bi pravilno alocirali resurse, IT manageri moraju imati na umu da kod dobavljača traže bazičnu fizičku infrastrukturu. Buduća IT infrastruktura može doći u obliku hiper promjenjive i promjenjive platforme. Takva infrastruktura može sadržavati i apstraktne resurse, putem usluga u cloudu (Amazon Web Services ili Azure). Krajnji cilj je apstrakcija fizičkog sloja i mogućnost platforme, suprotno poslužiteljima. Važno je IT timovima da zanemare infrastrukturu baziranu na hardveru kad uzimaju u obzir mogućnosti buduće IT infrastrukture.

[9] Modul zadužen za virtualizaciju Linux kernela.

[10] Računalni softver, firmver ili hardver koji stvara i pokreće virtualna okruženja.

3. Osnove sistem administracije

Treće poglavlje bavi osnovama sistem administracije, upoznajemo se dinamikom rada i bitnim stavkama u svakodnevnom radu.

3.1. Uzbune

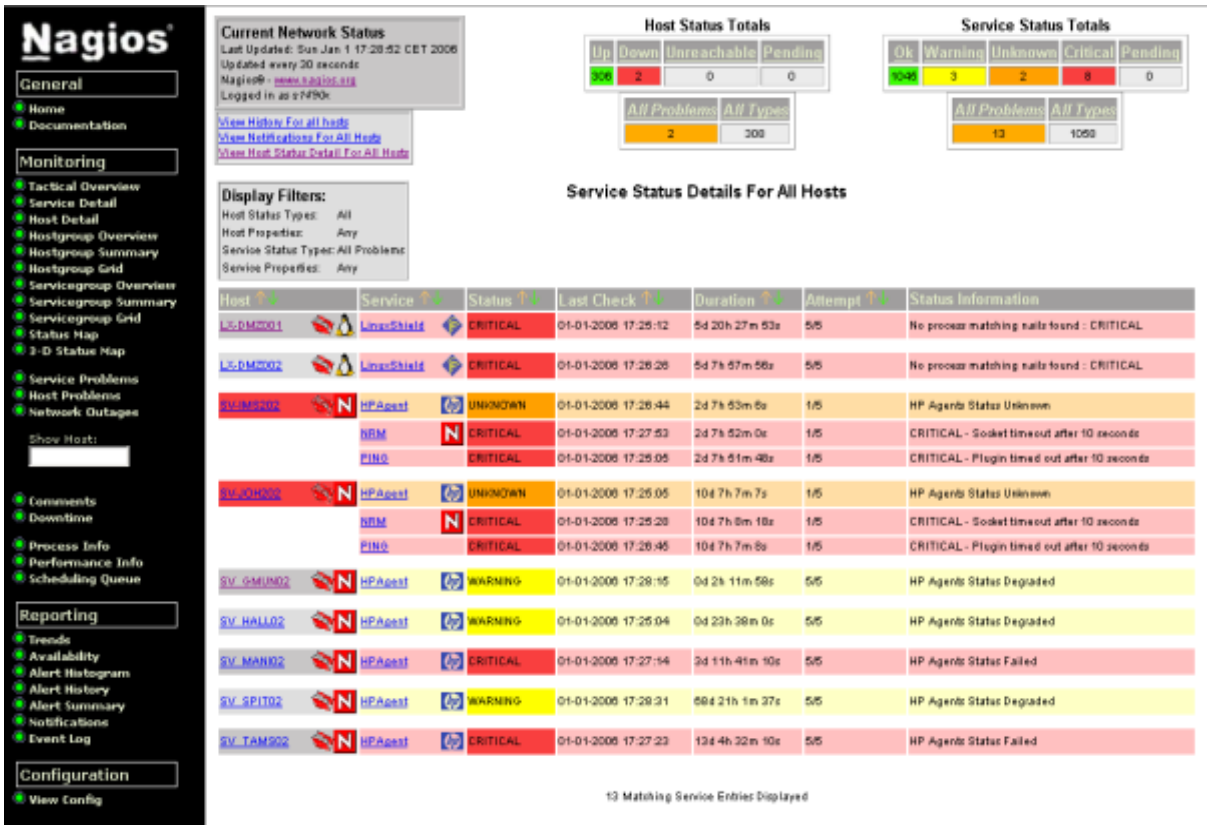
Kao i s bilo kojim poslom, odgovornosti vezane uz sistem administratore, DevOps – e i inženjere mogu biti različite, dok u nekim slučajevima uključuju kompletnu dežurnost. Ovih dana, naziv DevOps učinio je titulu posla sistem administrator da zvuči staromodno, kao i sistem analitičar koji ju je zamijenio. DevOps[11] pozicije po mnogočemu su drukčije od tipične pozicije sistem administratora u prošlosti te sada imaju više utjecaja i povezanost na razvoj softvera dok je prije bilo sve bazirano na čistom pisanju shell skripti. Rezultat tome je da te pozicije se daju ljudima koji imaju nekakvu programersku pozadinu bez ikakvog prijašnjeg iskustva u sistem administraciji. U prošlosti, sistem administrator ušao bi u rad kao junior sysadmin gdje bi tijekom tog razdoblja mentorirao senior sysadmin. DevOps ulaze u junior fazu bez ikakvog mentorstva.

Najveća greška sysadmina je nepravilno postavljanje sustava za promatranje, točnije takvo postavljanje koje reagira na apsolutno sve. Lako je promatrati sve aspekte servera, time dolazi do pretjerivanja u postavljanju svakojakih provjera sustava. Glavni zadatak za bilo koji sustav koji promatra je postavljanje pripadajućih znakova obavijesti kako bi se smanjile obavijesti koje ne označavaju veliku prijetnju sustavu, odnosno da smo samo obaviješteni ako je sustav u opasnosti od pada. Ako imamo previše provjera sustava, time se dovodimo da nam je održavanje sustava teže. Kritična upozorenja koja se postave samo na sustave koji rade 24/7, pomažu smanjivanju lažnih upozorenja i daju više prostora pravim problemima koje treba hitno riješiti. Svaka kritična obavijest mora uz sebe imati plan akcije kojom se sysadmin vodi.

[11] Set softverskih rješenja koja kombiniraju razvoj softvera i operacije informacijskih tehnologija kako bi se skratio vremenski period stvaranja funkcionalnog sustava koji mora biti u istoj ravnini sa poslovnim zahtjevima.

Slanje obavijesti sysadminu, koji bi uz sebe imali uređaj bilo je slično kao i kod doktora. Sustav za motrenje bio je dizajniran da šalje brojeve koji su označavali tip obavijesti, dok se danas također prakticira takva praksa ali pomoću mobilnih uređaja. Lažne (nevažne) obavijesti mogu predstavljati veliki problem, samim time sysadmin ima naviku ih zanemarivati. Danas pomoću određenih aplikacija moguće je postaviti razne filtere koji premještaju sve obavijesti istog značaja u određenu mapu, te emitira poseban zvuk, vibraciju za svaki novu e-mail, odnosno za više računa, jedan privatni i drugi za poslovne obveze.

Kad imamo postavljene obavijesti, sljedeći korak je konfiguracija opcija za eskalaciju, odnosno mogućnost dobivanja određene osobe ako je uređaj koji upotrebljava van funkcije. U slučaju toga, bitno je imati druge na umu kako bi oni bili obaviješteni o tome neki od zaposlenika ima komunikacijsku liniju van funkcije. Eskalacije su jedan aspekt gdje su sustavi za motrenje puno bolji. Jedan od takvih primjera je Nagios koji nudi bogati set rasporeda eskalacija.



Slika 4. Prikazuje sučelje Nagios alata za raspored i upravljanje eskalacijama.

Izvor: Microfocus, Nagios: Host and Service Monitoring Tool, autor: Rainer Brunold, objavljeno: 19. siječnja, 2006. godine, na web stranici: https://www.novell.com/coolsolutions/feature/16723.html (pristupljeno: 25. travnja, 2019. godine.)

Jednostavan sustav za eskalaciju ima sljedeće aspekte:

- Početna obavijest šalje se sysadminu i ponavlja se svakih 5 minuta.
- Ako sysadmin ne prepozna obavijest ili ne popravi kvar u sljedećih 15 minuta tada obavijest prelazi na sljedeću razinu i šalje se ostatku tima.
- Obavijesti se ponavljaju svakih 5 minuta sve dok ih netko ne uvidi obavijest ili riješi problem

Ideja je da obavijest daje vremena sysadminu da uvidi kvar. Sve ovisi o našem SLA – u[12], želimo skratiti ili produžiti vremenske periode između eskalacija ili ih učiniti sofisticiranijima pomoću dodavanja osobe kao backupa koja prima obavijest prije cijelog tima. Generalno, potrebno je organizirati eskalacije tako da postoji prava ravnoteža između osobe koja prima obavijest i prilike za odaziv na nju prije nego se obavijesti cijeli tim. Uz eskalacije tu postoji i rotiranje sysadmina, odnosno dežurstva (on – call rotation). Praksa je da svakih tjedan, pa do četiri tjedna se vrše rotacije gdje rotacije svakih 2 tjedna su najpogodnije. Što je veći tim, time su i rotacije lakše podnošljive za organizaciju i zaposlenike.

3.2. Automatizacija

Postoji velik broj razloga zašto je potrebno (dobro) automatizirati veliki dio svojih obaveza. Postoji i jedna izreka koja se spominje u sysadmin krugovima a ta je: „Budi oprezan ili ću te zamijeniti malom shell skriptom“. Dobar sysadmin mrzi raditi ponavljajuće zadatke, samim time postoji i veliki asortiman pristupa automatizacije.

Sa svom mogućom automatizacijom koja je već ugrađena u servere ovih dana, lako je zanemariti sve zadatke koje su u prošlosti radio sysadmin. Logovi se nisu rotirali automatski; backup se radio manualno. Iako dan danas postoje sysadmin koji sve rade ručno, logiranje u računalo i instalacija ili ažuriranje softvera, kao i konfiguracija servera. Instalacija modernog servera i OS – a može uzeti od 15 minuta pa do sat vremena. To su zadaci koji ne zahtijevaju nekakvu veliku ekspertizu sysadmina. Automatizacijom takvih zadataka, sysadmin može se posvetiti poslu koji zahtijeva određenu razinu ekspertize.

[12] Service-level agreement (SLA) – obveza između pružatelja usluga i korisnika. Glavni aspekti su kvaliteta, dostupnost i odgovornost.

Stvar oko ponavljanja istih zadataka više vremena je da postoji velika šansa da se pojavi greška, stvari koje radimo svaki dan više puta, eventualno postoji šansa da se zanemari ako je zadatak uspješno izvršen. Način na koji se zadatak izvršava može varirati od sysadmina do sysadmina. Automatizacijom određenih zadataka, tim se može složiti koji je idealan način da se odradi i zna se da ako se na taj način odradi zadatak da nema mogućnosti pojavljivanja greške.

Automatizacijom je moguće kompleksne procese svesti na samo jednu naredbu. Ta naredba postaje nešto što svi u timu mogu koristiti, bilo to da takav proces zahtijeva više senior članova u timu. Razvoj softvera može biti primjer toga, može postojati složeni raspored aktiviranja raspoređivača opterećenja i praćenja načina održavanja, softverske verzije za provjeru, ogledala za sinkronizaciju i usluge za ponovno pokretanje i testiranje. Iako ovi svi individualni koraci mogu biti rutinski, ako ih se kombinira, mogu postati vrlo kompliciranі za sve, pogotovo ako je u timu junior sysadmin. Automatizacijom takvog procesa, senior sysadmini mogu svu svoju ekspertizu i znanje uputiti na kreiranje pravog procesa koji odrađuje sve provjere. Automatizacijom dolazi i dokumentacija, tako da često tim provodi vrijeme i u tome. Dokumentacija ima još uvijek važno mjesto u svemu tome. Nije sve moguće automatizirati, čak i stvari koje izgledaju da su dobri kandidati za to, za neki period vremena neće biti. Bitno je znati što automatizirati.

Generalno, zadatke koji se izvode periodično (barem svaki mjesec) su dobri kandidati za automatizaciju. Što je zadatak više izvodiv, u teoriji možemo dobiti više vremena ako se automatizira. Zadaci koji sve izvode jednom godišnje nisu vrijedni automatizacije, samim time za njih postoji dobra dokumentacija. Ako bi mogli dokumentirati proces serije određenih naredbi, onda ih kopirali i zalijepili jednu po jednu u terminal zadatak bi bio izvršen, to je zadatak koji se više puta ponavlja te bi bio dobar kandidat za automatizaciju. S druge strane, zadaci koji su iznimno kratki te koji imaju različite inpute te koje nećemo morati više raditi su nešto što nije potrebno automatizirati.

Što je zadatak kompleksniji, to je veća mogućnost da se napravi greška ako se radi manualno, korak po korak. Ako zadatak ima više koraka te neki od koraka ima zahtjev da se uzme output od nekih koraka i koristi kao input za sljedeći korak, ili korak koji koristi naredbe s kompleksnim stringovima (argumentima) je odličan kandidat za automatizaciju. Što duže zadatak je potrebno obavljati (unos naredbe, čekanje da naredba odradi svoje te korištenje outputa naredbe) to je bolji kandidat za automatizaciju. Instalacija OS – a i konfiguracija su dobar primjer toga, kad se OS instalira, postoje periodi kad se unose postavke i periodi kad se čeka instalacija. Sve je to izgubljeno vrijeme. Automatizacijom zadataka koji traju duže vrijeme, možemo se posvetiti ostalom poslu.

3.3. Upravljanje i organizacija zadataka

Upravljanje i organizacija zadataka (etiketiranje) odnosi se na sustave koji dopuštaju sysadminu da prati zadatke koji dolaze interno i od kupaca ali i od radnih kolega. Postoji puno načina na koji se tiketiranje protumačiva pogrešno kao nešto nepotrebno organizaciji, tako velika većina sysadmina odbija ih koristiti ili ih koristi na silu. Takav pristup dobar je za developere dok za sysadmine može biti užasno, no kako sysadmin naziv polako izumire te polako ulaze u developer vode pod nazivom DevOps potrebno je da prihvate takav način. Kao dokumentacija, tiketiranje (karte) su jedna od važnijih stvari unutar zrele organizacije. Sustav za karte je važan bez obzira na veličinu organizacije. U velikim organizacijama, volumen karata je ogroman te ga je potrebno pratiti.

Sysadmini rade mnoštvo zadataka konstantno. Potrebno je balansirati između zadataka. Spremanjem svakog upita u kartu, bez obzira kako je zaprimljen, upit je spremljen, čak i ako se zaboravi na njega, upit je unutar sustava. Karte u sebi sadrže ono što točno žele ljudi, napisano njihovim riječima. Svaki upit je važan osobi koja ga postavlja. Svaki upit ne mora biti hitan prema nama ili timu. Kad su svi upiti spremljeni unutar karte, voditelj tima ili manager može proći kroz prikupljene karte te ih postaviti po pravom redoslijedu kako bi bili odrađeni kako treba. Sustavom za karte, voditelj tima ili manager ima kompletnu listu svih važnih zadataka.

Ako imamo samo jednog sysadmina, distribucija karata i projekata je lagana. Kad tim naraste, važna je podjela zadataka na timove kako ostali članovi ne bi bili previše opterećeni. Kad je upit zaprimljen, voditelj tima može dodati zadatak ili oduzeti zadatak pojedinom članu tima. Svaki put kad dođe do promjene unutar sustav, postoji šansa da se nešto pogorša. Pomoću sustava za karte, možemo izvući upit koji je bio u tijeku kad se dogodio kvar, pao sustav i slično. Karte na neki način predstavljaju i audita za upite koji zahtijevaju dokaz o rješavanju, kao što je kreiranje ili uklanjanje korisničkih računa, davanje ovlasti ili ažuriranje softvera. Ako trebamo dokazati da smo upotrijebili sigurnosno ažuriranje, možemo dati uvid na output naredbe koji smo uhvatili te ga spremili u određenu kartu.

Postoji puno sustava za upravljanje kartama te kad se ljudi žale na njih, žale se na loše sustave. Što je sustav jednostavniji, to bolje, što je proces brži time je i jednostavniji. Postoji par stanja u kojima se karte nalaze: otvorene, dodijeljene, u toku, riješene i zatvorene. Što manje polja ima karta, to bolje. Idealno, sustav za etiketiranje kreira karte iz same skripte, bilo to s e – maila poslanog na posebnu adresu ili preko API – ja. Ako postoji API koji dopušta mijenjanje stanja karte ili dodavanje komentara tim bolje, jer postoji mogućnost automatizacije pomoću skripte. Korištenje sustava koji dopuštaju kreiranje veza između karata je najbolje. Važno je znati da upit A ovisi o upitu B i da trebamo riješiti upit B prvo.

Svaki sustav za karte mora imati notaciju za stanje karata, a to su sljedeće:

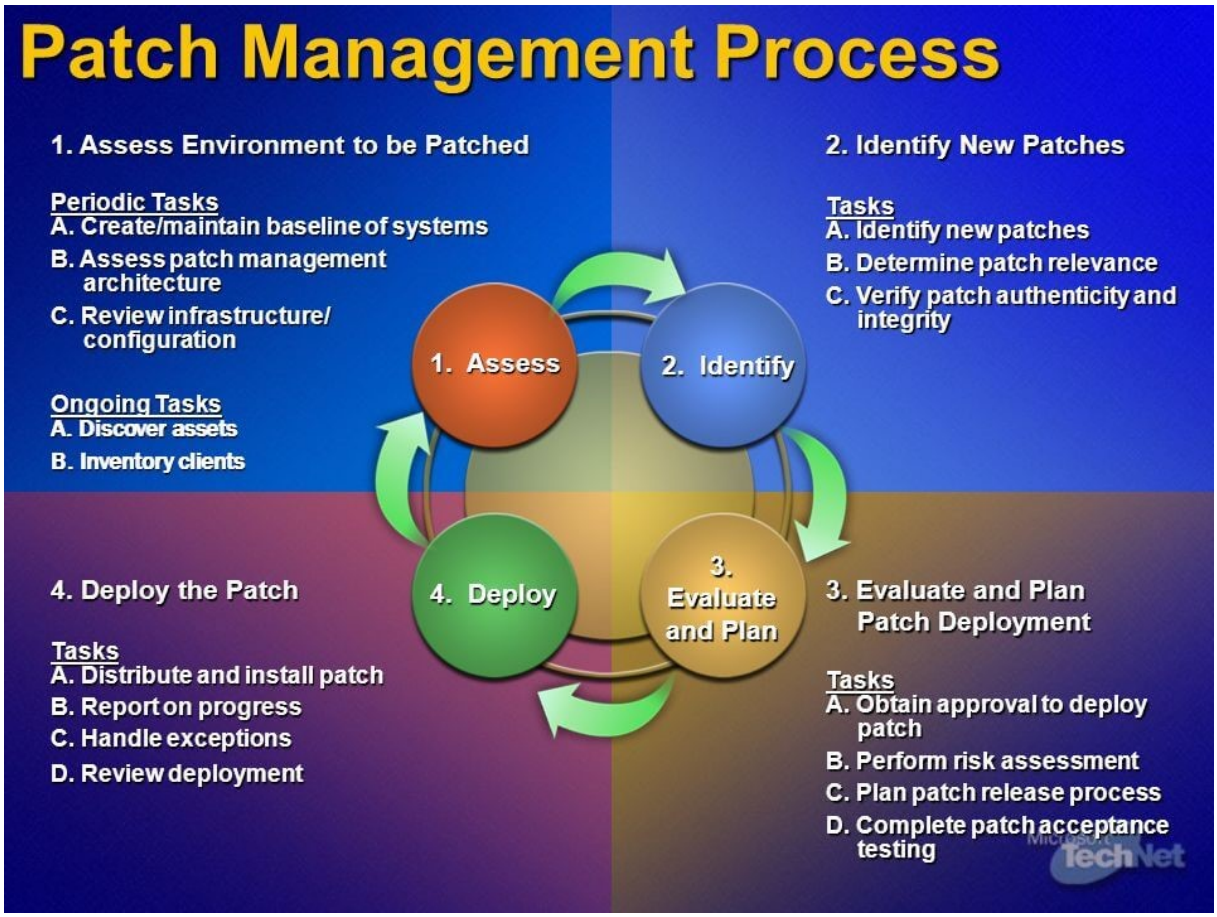
- **Otvoren:** upit koji treba izvršiti te nije dodijeljen nikome.
- **Dodijeljen:** upit koji se nalazi kod određene osobe, ali nije ga još odradila. Upit u ovakvom stanju može sigurno biti dodijeljen nekom drugome.
- **U radu:** Upit je dodijeljen nekome te netko na njemu aktivno radi. Prije dodjeljivanja karte nekom drugome, potrebno je prvo konzultirati s osobom koja na upitu trenutno radi.
- **Riješen:** Sysadmin je riješio upit te čeka na potvrdu osobe koja je uputila upit.
- **Zatvoren:** Upit je odrađen uspješno.

Dobar voditelj tima trebao bi imati mogućnost vaganja koji je upit bitniji nasuprot drugom upitu te mijenjati njihovu važnost na temelju važnosti na prema drugih upita. Upit može imati malu važnost prije dva tjedan no s vremenom može primiti bitniju važnost zbog toga što je dugo bio u stanju pripravnosti. Sljedeća važna stavka je da su svi upiti unutar sustava. Upit može biti da radni kolega dođe direktno vama te vam postavi upit i time preskače red unutar sustava. Kao manager bitno je raspodijeliti posao pošteno između članova. Ako imamo člana koji je ekspert u tom polju, nije potrebno da mu se uvijek dodjeljuje takav tip upita uvijek, bitno je da svatko unutar time iskusi i prođe sve moguće upite.

3.4. Upravljanje zakrpama

Upravljanje zakrpama odnosi se na sustave za koje imamo potrebu ažurirati softver te su na poslužitelju. Čak i konzervativnije distribucije kao što je Debian koje se strogo drže rasporeda izdavanja novih verzija (stable version) redovito puštaju u opticaj ažuriranja koje se tiču zakrpa ili sigurnosnih propusta. Organizacije koje koriste vlastite verzije, prerađene njihovim potrebama softver, iz razloga što developeri tako zahtijevaju, potrebno je napraviti određene izmjene unutar softvera. Idealno je imati sustav koji automatski pakira prerađene verzije softvera. Ako naša verzija softvera ima sigurnosni propust, potrebno je upotrijebiti zakrpu koja se nadovezuje na prerađenu verziju.

Upravljanje zakrpama započinje time da imamo saznanja koji softver treba ažurirati. Za naš temeljni softver, koji je glavna okosnica trebali bi biti pretplaćeni na mailing listu naše Linux distribucije kako bi izravno i istog trena bili obaviješteni o novom ažuriranju. Ako se ipak koristi softver koji nije originalno unutar distribucije, mora se pronaći način kako da se budu u toku za zakrpe vezane točno za taj softver. Kad nam pristigne nova sigurnosna obavijest, potrebno je pregledati detalje koji se tiču koliko je velik sigurnosni propust, jesmo li ugroženi te koliko je vremenski bitno implementirati sigurnosnu zakrpu. Neke organizacije koriste manualni način upravljanja zakrpama. Takvim sustavom, kad se pojavi sigurnosna zakrpa, sysadmin mora uočiti na kojim poslužiteljima se takav tip softvera koristi pomoću memorije i logiranja izravno na poslužitelje.

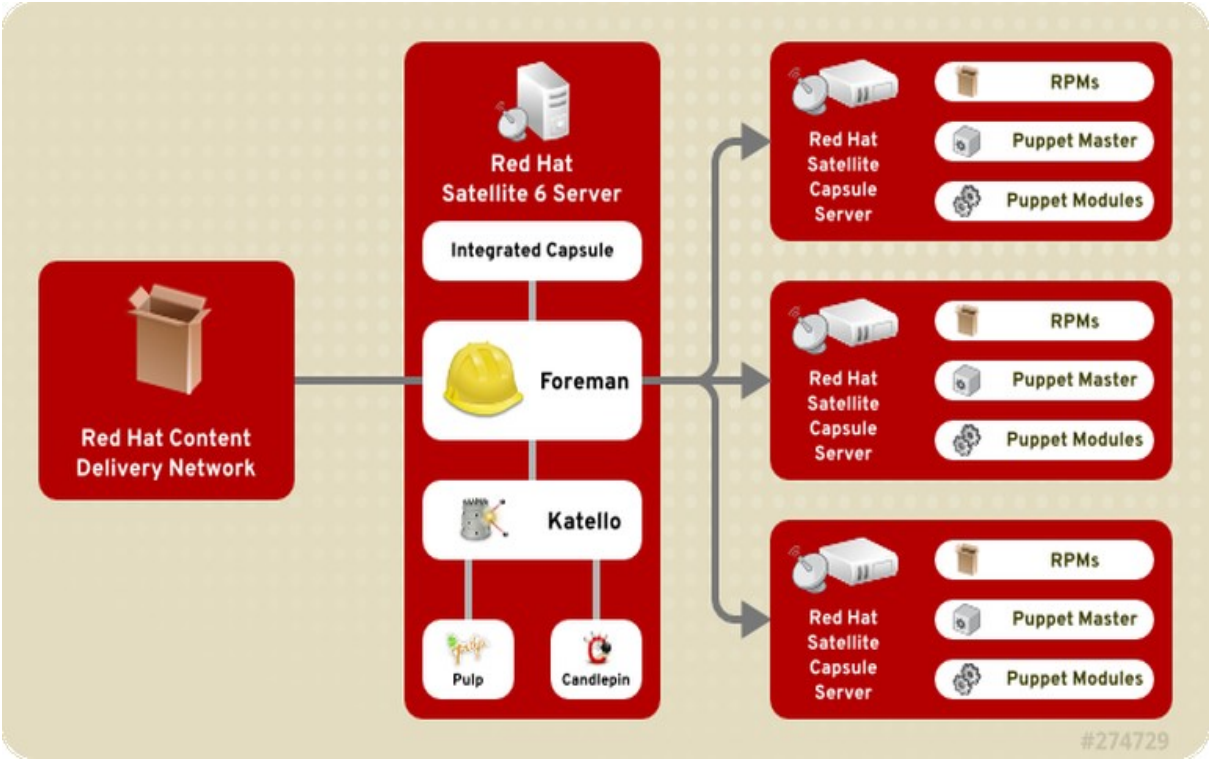


Slika 5. Prikazuje proces upravljanja zakrpama.

Izvor: Icctel Uganda, Patch management, objavljeno: 5. studenog, 2017. godine, na web stranici: http://icttel-uganda.com/patch-management/ (pristupljeno: 29. travnja, 2019. godine.)

Nakon toga sysadmin koristi integrirani sustav za upravljanje zakrpama koji se nalazi unutar poslužitelja te ažurira softver najnovijim paketima te distribucije. Potom se prelazi na sljedeći poslužitelj i tako dok svih poslužitelji nisu zakrpani. Postoji mnogo problema s manualnim upravljanjem zakrpama. Prvo je da zahtjeva puno radnog vremena te sysadmin može Drugi problem je da takvim načinom sysadmin vrlo lako može zaboraviti neki od poslužitelja ili ga u potpunosti zanemariti. Drugi problem je da se takvim načinom previše oslanjamo na mogućnosti sysadmina i njegovo pamćenje koji poslužitelj je zakrpan a koji nije, sve to dovodi do toga da poslužitelj može biti zaboravljen ili biti u radu bez sigurnosnih zakrpa. Što je sustav za upravljanje zakrpama brži i lakši time ćemo ga i više koristiti. Imati sustav koji u svakom trenutku može pokazati koji od poslužitelja u radnom stanju koristi koji i softver te koju verziju.

Postoje i razni alati koji služe za orkestracija takvih zadataka kao što je MCollective, Red Hat (Satellite) te Canonical i njegov Landscape kojima je moguće vidjeti verzije softvera koje se nalaze na poslužiteljima te implementirati zakrpe. Zakrpe bi trebale biti tolerantne na pogreške, imati mogućnost implementacije zakrpe te restartati poslužitelj vrlo brzo, bez većih zastoja. Ista ideja je i vezana uz ažuriranje kernela koji zahtijevaju ponovno podizanje sustava.



Slika 6. Prikazuje Satellite koji integrira Puppet (open source alat za upravljanje konfiguracijom) te ga koristi kao podatkovni centar i kao administracijski alat u računalstvu u oblaku.

Izvor: Zdnet, Red Hat Satellite 6 comes with improved server and cloud management, autor: Steven J.Vaughn-Nichols, objavljeno: 10. rujna, 2014. godine, na web stranici: https://www.zdnet.com/article/red-hat-satellite-6-comes-with-improved-server-and-cloud-management/ (pristupljeno: 29. travnja, 2019. godine.)

Sustav bi trebao biti u mogućnosti da bude u funkciji odmah nakon implementacije zakrpe bez ikakvih problema i zastajanja u radu (ShellShock ranjivost UNIX/Linux operacijskih sustava, odnosi na propust unutar Bash-a). Za nešto poput OpenSSL-a to zahtijeva restartiranje usluga, oprezan proces implementacije i ponovnog pokretanja uređaja pomoću orkestracije (upotreba MCollective, Satellite ili Landscape).

Kada zakrpa zahtijeva ponovno podizanje sustava, kad je u pitanju kernel, može potrajati neko duže vrijeme, ali uz orkestraciju i automatizaciju sve može ići puno brže. Mnoge organizacije i dalje imaju poslužitelje koji nikada ne pasti i kao rezultat toga nemaju potrebu biti osigurani novih sigurnosnim zakrpama. Brz i jednostavan sustav za upravljanje zakrpama je znak zrelosti i profesionalnosti sysadmin tima. Ažuriranje softvera je nešto što dio posla svakog sysadmina, ulažući vremena u sustave koji brzo i sigurno izvršavaju sve procese i zadatke puno su isplativiji od sigurnosti u dugom roku. Pomaže prepoznati loše odluke vezane za arhitekturu koje mogu dovesti do pada sustava. Pomaže u identifikaciji sustava koji stagnira te koji je zastario i kojeg je potrebno zamijeniti.

4. DevOps

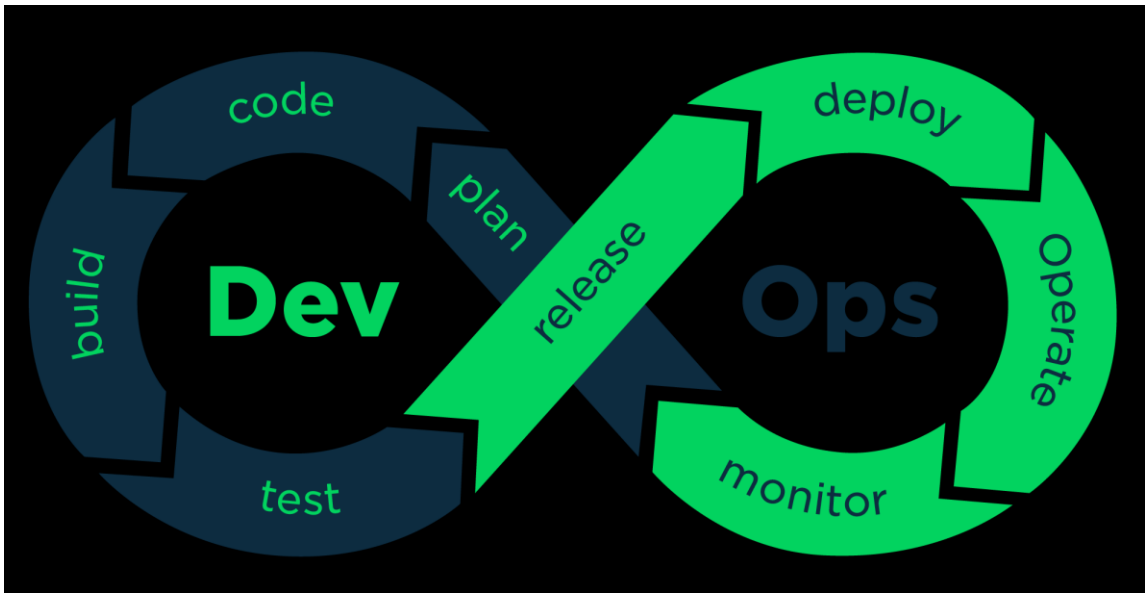
Četvrto poglavlje bavi se metodologijom DevOps (Development + Operations) te detaljno opisuje njezine prakse, različite inačice (BizDevOps), razlike u metodama razvoja softvera te povezanost s scrumom.

DevOps predstavlja relativno novi termin koji nastaje iz kolizije dva velika trenda. Prvi trend je agilna infrastruktura ili agilne operacije; kreće se od korištenja agilnih i lean metodologija u radu. Drugi trend predstavlja znatno prošireno razumijevanje vrijednosti same suradnje između razvojnog i operativnog odjela i osoblja kroz sve faze razvojnog ciklusa od stvaranja i rada usluge te same spoznaje kako su takve operacije postale važne iz razloga što se sam svijet okreće i bazira na uslugama. DevOps je skup programa za razvoj softvera koji kombinira razvoj softvera (dev) i informatičku tehnologiju (ops) kako bi se skratio životni ciklus razvoja sustava usporedno s time pružajući nove značajke, popravke i česta ažuriranja u bliskom usklađivanju s poslovnim ciljevima organizacije.

“DevOps“ ne razlikuje između različitih ishodišnih disciplina – “Ops“ je okvirni pojam za inženjere sustava, sistem administratore, operativno osoblje, mrežne inženjere, stručnjake za sigurnost i razne ostale poddiscipline kao i nazive radnih mjesta. “Dev“ se primarno koristi za programere posebice, dok je u praksi to jako širok termin te označava “sve ljude uključene u razvoj proizvoda“.

DevOps ima snažnu povezanost s agilnim i lean metodologijama kao što je i prije spomenuto. Prijašnji pogled na operacije odnosio se na “Dev“ koja je označavala tvorce proizvoda i “Ops“ kao stranu koja se bavi stvaranjem proizvoda nakon prvotnog rođenja kao same ideje. DevOps se može tretirati kao rezultat rasta agilnog razvoja softvera koji propisuje blisku suradnju kupaca, odjela za upravljanje proizvodom i developera kako bi se popunile praznine i poboljšao već stvoreni proizvod. Dostava usluge i način na koji aplikacija i sustav imaju interakciju jesu temeljni dio vrijednosti za svakog kupca. DevOps ima mnogo značenja mnogim ljudima zato jer sama diskusija oko toga pokriva mnoga područja.

DevOps nije tehnologija, iako postoje određeni alati koji se koriste u razvojnom okruženju. To uključuje kontinuiranu integraciju i kontinuirano dostavljanje ili alate za kontinuirano dostavljanje, s naglaskom na automatizaciju zadataka. Ostali alati koji podupiru rad DevOps – a jesu promatranje u realnom vremenu (real-time monitoring) i sustav koji reagira incidente, odnosno pogreške u radu sustava kao i ostale platforme namijenjene suradnji.



Slika 7. Prikazuje metodologije i infrastrukturu koja opisuje DevOps filozofiju kroz odjele Dev i Ops.

Izvor: suse, na web stranici: https://www.suse.com/solutions/devops/ (pristupljeno 7. kolovoza 2019. godine).

DevOps pristup može zajedno postojati s metodama agilnog razvoja softvera; okviri za upravljanje uslugama, kao što je ITIL [13] (IT Infrastructure Library); direktive project managementa, kao što su lean i six sigma te ostale strategije kojima se izvršavaju IT projekti koji se ravnaju s poslovnim potrebama organizacije. Računarstvo u oblacima kao i definirana softverska infrastruktura, mikroservisi i automatizacija implementirani su zajedno s DevOps metodologijama. Industriji nije dosta samo imati kombinaciju Dev – a i Ops – a, te bi termin DevOps trebao uključivati i poslovnu stranu, čime dobivamo termin BizDevOps, tako i sigurnosna strana ima svoj termin DevSecOps kao i testiranje. Uspješno DevOps okruženje treba uključivati sve kritične grupe koje su neophodne za IT uspjeh.

[13] Set detaljno opisanih praksi za upravljanje IT uslugama koji se fokusiraju na ravnanje usluga s poslovnim potrebama.

4.1. DevOps prakse

Postoji malen broj ključnih praksi koje pomaže organizacijama da što brže dolaze do inovacije te da brzo teku kroz automatizaciju i protok razvoja softvera i infrastrukturnog upravljanja procesima. Većina praksi je uspješno kompletirana uz prave alate. Bitna praksa je izvođenje malih ali čestih ažuriranja. Time organizacije brže dolaze do inovacija koje kasnije dolaze do korisnika kao krajnji proizvod. Ažuriranja se provode u inkrementima nasuprot običnih ažuriranja koji dolaze u tradicionalnim ciklusima puštanja u opticaj. Česta ali malena ažuriranja smanjuju rizik vjerojatnosti pojavljivanja grešaka. Time timovi brže pronalaze greške jer mogu identificirati zadnje ažuriranje koje sadrži grešku. Frekvencija puštanja ažuriranja i njihova veličina varira, organizacije koristeći DevOps model puštaju u opticaj ažuriranja češće nego tradicionalne tehnike razvijanja softvera.

Organizacije također koriste arhitekturu mikrousluga time čineći da aplikacije budu što fleksibilnije time dopuštajući brze izmjene. Arhitektura mikrousluga odvaja velike i kompleksne sustave u jednostavne, neovisne projekte. Aplikacija se dijeli u više komponenti (usluge) s svakom uslugom koja ima opseg jedne funkcije te funkcionira odvojeno od usluga viših razina i same aplikacije. Takva arhitektura smanjuje preklapanje koordinacije kad je u pitanju ažuriranje aplikacija. Kad je usluga sparena s manjim, agilnim timom koji preuzima vlasništvo svake usluge, organizacije se kreću brže.

Kombinacija mikrousluga i povećane frekvencije puštanja ažuriranja dovodi do operativnih izazova. Iako kontinuirana integracija i dostavljanje nekim dijelom rješavaju taj problem i puštaju da organizacije daljnja ažuriranja vrše u sigurnome okruženju. Određene prakse dopuštaju takav tempo bez straha od stvaranja operativnih zastoja.

4.1.1. Kontinuirana integracija

Praksa razvoja softvera gdje developeri svakodnevno spajaju promjene koda u centralni repozitorij, nakon čega se vrše testiranja. Referira se na izgradnju ili fazu integracije softvera i njegovog procesa puštanja u opticaj gdje spaja komponentu automatizacije. Cilj je brži pronalazak i prijava grešaka, poboljšavanje kvalitete te smanjenje vremena validacije i puštanja ažuriranja u opticaj. Prijašnja praksa je bila odvojeni rad u nekom određenom vremenskom periodu te spajanje koda u glavnu granu (*master branch*). Takva praksa otežavala je promjene koda i bila vremenski previše ovisna čime bi se greške samo skupljale bez ispravaka. Kontinuiranom integracijom, developeri promjene mogu unositi putem dijeljenog repozitorija koristeći sustav kontrole verzija kao što je Git. Prije svake promjene, moguće je vršiti testove koda kao dodatna provjera prije spajanja s ostalim kodom. Promjene koda automatski se grade, testiraju i pripremaju za puštanje u opticaj. Na to se nadovezuje trajna isporuka puštajući cijeli kod u testno okruženje.

4.1.2. Trajna isporuka

Trajna isporuka (continuous delivery) praksa gdje se kod automatski priprema za puštanje u opticaj. Pravilnom implementacijom, developeri će uvijek imati kod koji je prošao kroz testni proces. Omogućava se automatizacija testiranja kako bi se verificirala aplikacija prije puštanja u opticaj kupcima. Testovi uključuju UI test, učitavanje, test integracije i API pouzdanost. Pomoću računarstva u oblacima olakšava se i smanjuju se troškovi automatizacije kreiranja i repliciranja raznih okruženja za testiranje. Trajnom isporukom, svaka promjena koda se gradi, testira te se prosljeđuje na testiranje u namješteno okruženje. Može biti više, paralelnih testiranja prije puštanja u proizvodnju. Razlika između trajne isporuke i kontinuiranog uvođenja je prisutnost odobravanja fizičke osobe za ažuriranje te potom puštanje u proizvodnju. S trajnom isporukom, proizvodnja se događa automatski bez eksplicitnog odobravanja.

4.1.3. Mikrousluge

Arhitektura mikroservisa je dizajnerski pristup gradnje jedne aplikacije kao seta manjih usluga. Svaka usluga izvršava svoj proces i komunicira s ostalim servisima kroz detaljno definirano sučelje koristeći lakši mehanizam, najčešće sučelje (API) bazirano na HTTP – e programiranju. Mikrousluge su građene oko samih mogućnosti poslovanja, svaka usluga ima jednu jedinstvenu ulogu. Koristeći različite radne okvire ili programske jezike moguće je graditi mikrousluge te ih puštati u opticaj neovisno, kao jedna usluga ili kao grupu servisa.

4.1.4. Infrastruktura kao kod

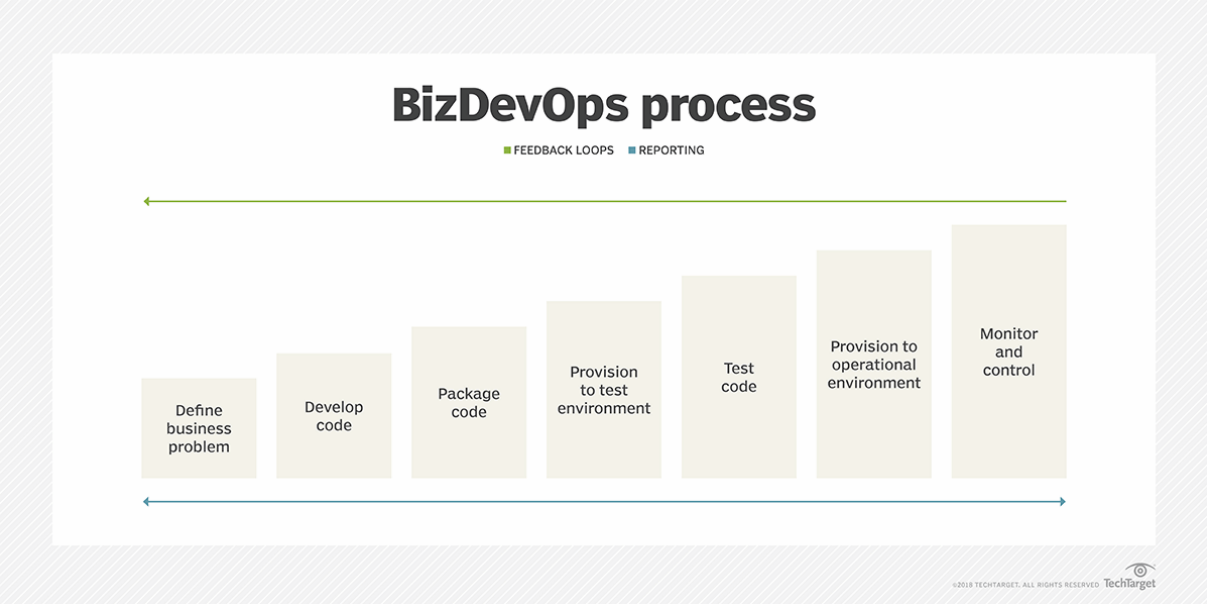
Praksa kojom se infrastruktura dodjeljuje i upravlja koristeći kod i razvojne tehnike, kao što je kontrola verzija i kontinuirana integracija. Model koji se temelji na API – ju oblaka dopušta developerima i sistem administratorima programsku interakciju s infrastrukturom nasuprot potrebe za ručnim postavljanjem i konfiguriranjem resursa. Inženjeri mogu ostvariti interakciju s infrastrukturom koristeći alate bazirane na kodu te tretirati infrastrukturu kao što bi tretirali kod aplikacije. Iz razloga što se baziraju na kodu, infrastruktura i serveri mogu biti brzo pušteni u rad koristeći standardne radnje, ažurirane zadnjim zakrpama i verzijama.

4.1.5. Monitoring i bilježenje

Organizacije prate metriku i logove kako bi promatrali kako aplikacije i infrastruktura te ostvarene performanse utječu na iskustvo krajnjih korisnika proizvoda. Dohvaćanjem, kategorizacijom te analizom podataka i logova generiranih kroz aplikacije i infrastrukturu, organizacije shvaćaju kako promjene ili ažuriranje utječu na korisnike. Aktivno promatranje postaje sve važnije jer usluga mora biti dostupna 24/7 jer aplikacija i infrastruktura primaju sve više ažuriranja. Kreiranjem upozorenja ili provodeći analizu u realnom vremenu tih podataka pomaže organizacijama da proaktivno nadziru svoje usluge.

4.3. BizDevOps

BizDevOps, također nosi i naziv DevOps 2.0 označuje pristup razvoju softvera koji potiče developere, operacije kao i poslovne timove na zajedničku suradnju kako bi organizacija razvila softver što brže, bila što spremnija na zahtjeve od strane klijenta te time maksimizirala prihod. Uobičajeno da svi odjeli, pogotovo developeri, operacije i poslovni odjeli rade odvojeno, što je tradicionalna praksa. Developeri kreiraju kod, operacije održavaju kod te management pregledava podatke koji čine glavni čimbenik mjerenja uspjeha (KPI [14]) i postavlja uvjete za daljnje projekte.



Slika 8. Prikazuje cjelokupni BizDevOps proces (izvještavanje + povratne informacije).

Izvor: Techtarget, No-code/low-code app development evolves from loathed to loved, BizDevOps (Business, Development and Operations), autor: Margaret Rouse, na web stranici: https://searchsoftwarequality.techtarget.com/definition/BizDevOps-Business-Development-and-Operations

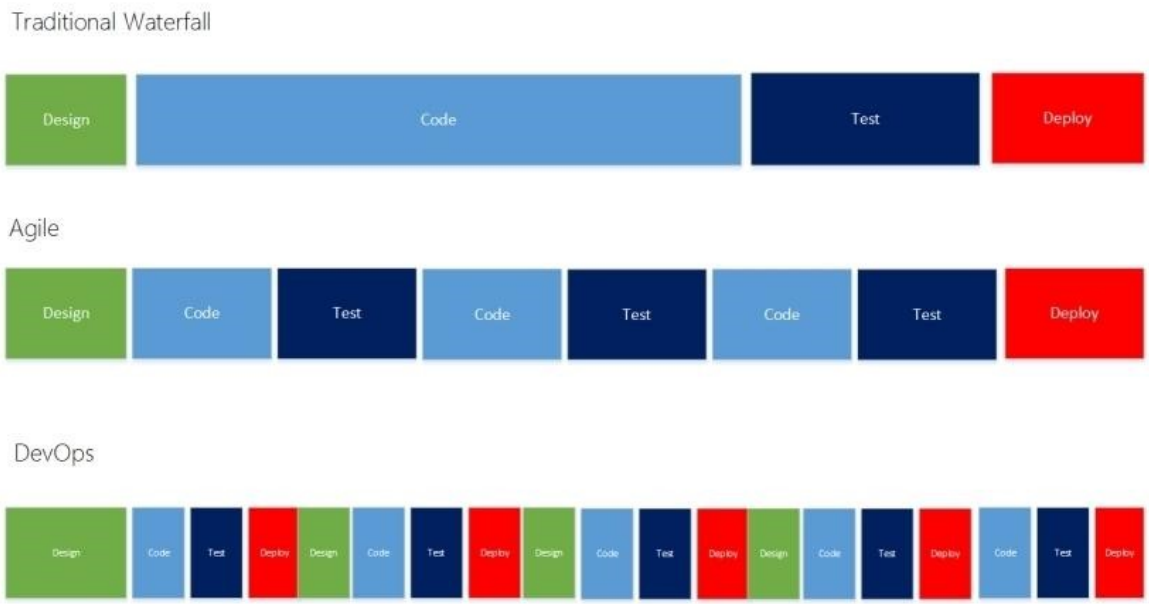
[14] Key Performance Indicator – vrsta mjerenja uspjeha, mjeri se pojedina aktivnost organizacije (projekti, programi, proizvodi…)

Jedna od tehnologija koja se koristi je i *real-time analytics*. Što označuje upotrebu, kapacitet upotrebe, podatke i povezane resurse čim podaci uđu u sustav. Izraz *real-time* odnosi se razinu kojom sustav odgovara na korisnikov podražaj. Takva analitika sastoji se od tri komponente, agregat (skupljanje podataka i njihovo slanje), broker (dostupnost podataka) i modul (analiza podataka). Koristeći alate za monitoring performansi aplikacije, kao i analitičke alate, organizacije imaju mogućnost prikupljati podatke vezane performanse aplikacije i podatke vezane za ponašanje krajnjeg korisnika te ih kvantificirati kako uspješno podržavaju mjerenje uspjeha organizacije.

Dostupnost podataka omogućava izmjene u aplikacijama čime izdanja dolaze do krajnjih korisnika frekventnije. U nekim slučajevima, ovakav pristup zahtjeva povećanu automatizaciju, pogotovo kod testiranja i osiguranja kvalitete. Sve veća potreba za brzinom, omogućava automatizaciju nekih rutina kod razvoja. Kod nekih organizacije, BizDevOps tim uključuje i sigurnosni tim kao službene predstavnike iz odjela za testiranja i osiguravanja kvalitete.

4.4. DevOps vs. Waterfall

Waterfall razvoj uključuje testiranje novog koda u izoliranim uvjetima za što bolju kvalitetu (*quality assurance*) te ako kod zadovolji uvjete, kod se pušta u opticaj, zajedno u komadu s ostalim izdanjima kako bi IT odjeli kontrolirali procese. Tim zadužen za operacije pušta u opticaj program te ga od tog trenutka održava. Takav razvoj traje dulje vrijeme između puštanja softvera u opticaj te zbog toga odjel developera i operacija rade odvojeno, odjel developera nije uvijek svjestan rada operacija unatoč tome što od njih se može pojaviti da kod ne funkcionira kako je planirano.



Slika 9. Prikazuje razliku između metoda u trajanju određenih stavki i broju ponavljanja istih.

Izvor: Itweb, Business, Is DevOps for you? autor: Alison Job, objavljeno: 25. lipnja 2018. godine na web stranici: https://www.itweb.co.za/content/VgZeyqJAO2AMdjX9/G3mYZRXM9yMOgA8L (pristupljeno 7. kolovoza 2019. godine.)

DevOps model spaja razvoj softvera i puštanje u opticaj u pojednostavljeniji proces koji se ravna s razvojem, kvalitetom i IT operacijama time pružajući kontinuiraniji radni protok. Ovakav pristup zvan *shift left* prebacuje određene odgovornosti operativnog tima na tim zadužen za razvoj kako bi se olakšao kontinuirani razvoj, kontinuirana integracija, kontinuirane isporuke i praćenje kontinuiranih procesa. DevOps praksa ruši standardni poredak koji izolira korake u procesima za isporuku softvera za što brže i češće puštanje koda u opticaj.

4.5. DevOps vs. Agile Development

Agilni pristup je tip razvoja softvera koji se striktno fokusira na inkrementalne i rapidne cikluse kreiranja koda kao i dostavu koda, poznatije pod nazivom *sprint.* Svaki sprint se nadograđuje na onaj posljednji, čime se stvara veća razina fleksibilnosti, IT administratori lakše evaluiraju promjene, raspon i smjer između svakog ciklusa. DevOps nastaje iz uspjeha agilnih metoda te brzine razvoja no pojavljuje se problem komunikacije, odnosno nepostojanje komunikacije između odjela za razvoj i odjela za operacije, kao i između IT i poslovne strane organizacije što postaje smetnja pri brzini i fleksibilnosti koda i dostavljanje koda korisnicima.

Učinkovitost agilnog razvoja dovela je toga da IT organizacije razdvoje odjele razvoja i operacija. Kod agilnog radnog toka, odjeli imaju različite ciljeve i vodstva. Organizacija kada koristi DevOps i agilne metode, odjeli mogu upravljati kodom bez smetnji u komunikaciji. DevOps nema službeni radnik okvir, brzina nije glavni fokus. Agilni razvoj se često povezuje i nadovezuje na radni okvir kao što je Scrum.

4.6. Scrum

Scrum predstavlja radni okvir za *project management* koji stavlja fokus ma timski rad, odgovornost i iterativno napredovanje prema krajnjem cilju. Okvir ima jednostavnu pretpostavku, započinjanje projekta s onim što je vidljivo ili znano. Potom se napredak prati. Tri glavna stupa su transparentnost, pregled i usvajanje. Okvir je često dio agilnih metoda razvoja softvera, ime je dobilo formaciji u ragbiju. Svatko ima neku ulogu u svemu. Kad je u pitanju razvoj proizvoda, *Scrum* uloge uključuju *product owner – a*, *Scrum mastera* i *Scrum* razvojni tim.

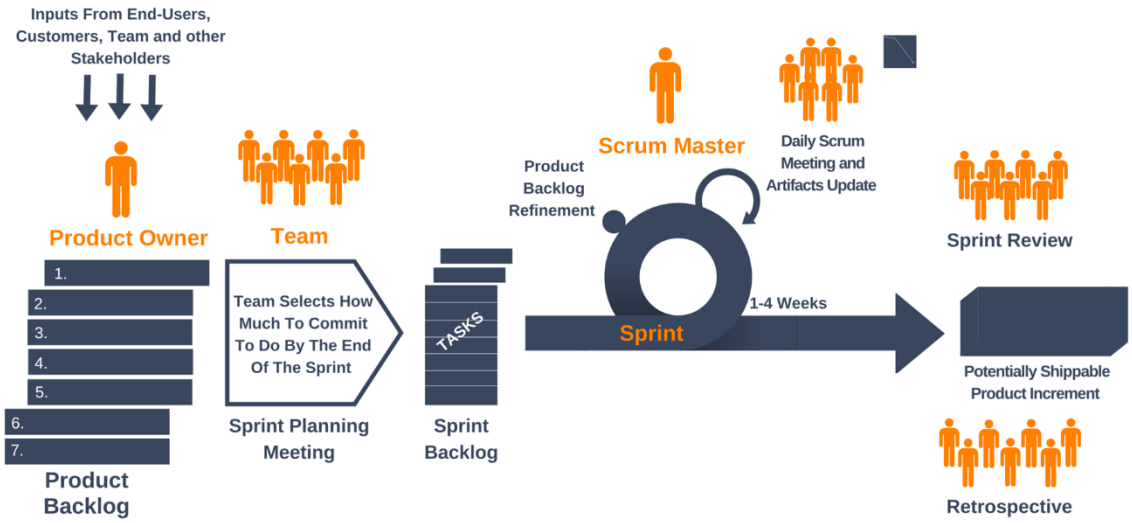
Product owner je uloga u razvojnom timu koja se odnosi na zastupanje poslovanja ili zajednice korisnika te je odgovorna za rad s korisnicima kako bi se utvrdilo koje značajke će biti uključene u krajnjem proizvodu. Poveznica između odjela za razvoj i odjela za rad s korisnicima leži u tome da *product owner* mora blisko surađivati s oba odjela kako bi se došlo do zajedničkog sporazuma vezanog za značajke koje će se nalaziti u proizvodu ili aplikaciji.

Scrum master je pokretač agilnog razvojnog tima. Upravlja procesima u odnosu na to kako se informacije izmjenjuju. *Scrum* analogija korištena je u proizvodnji papira od strane Hirotake Takeuchi – ija i Ikujiro Nonake. Tokom razvoja proizvoda, održavaju se dnevni sastanci, koji se nazivaju “scrums“, gdje *scrum master* ispituje članove time određena pitanja. Glavne zadaće mastera su pomaganje timu da dođu do zajedničkog jezika kad je u pitanju što se može postići za određeni vremenski period, pomaže timu u dnevnim sastancima, pomaže timu da ostane fokusirano na već dogovorena pravila rada, micanje prepreka koje štete napretku tima te pruža zaštitu timu od vanjskih utjecaja.

Bitno je napomenuti termin znan kao sprint. *Sprint* označuje vremenski period u kojem je potrebno da se određeni posao obavi te je spreman za pregled od strane *scrum mastera*. Svaki sprint započinje sastankom timova, gdje *product owner* i razvojni tim dolaze do zajedničkih uvjeta i poslova koji će se obaviti u nadolazećem sprintu. Trajanje sprinta određuje *scrum master*, tradicionalno on traje 30 dana, iako glavnu brojku određuju timovi unutra sprinta. Tokom trajanja sprinta, održavaju se dnevni sastanci gdje se raspravlja o progresu i raznim idejama.

4.6.1. Scrum proces

Scrum proces potiče one koji ga prakticiraju da rade s onim što imaju te da vrše evaluaciju što funkcionira a što ne funkcionira. Komunikacija je važan proces, koje se proteže kroz svakodnevne sastanke, zvane Events (događaji).



Slika 10. Prikazuje Scrum proces.

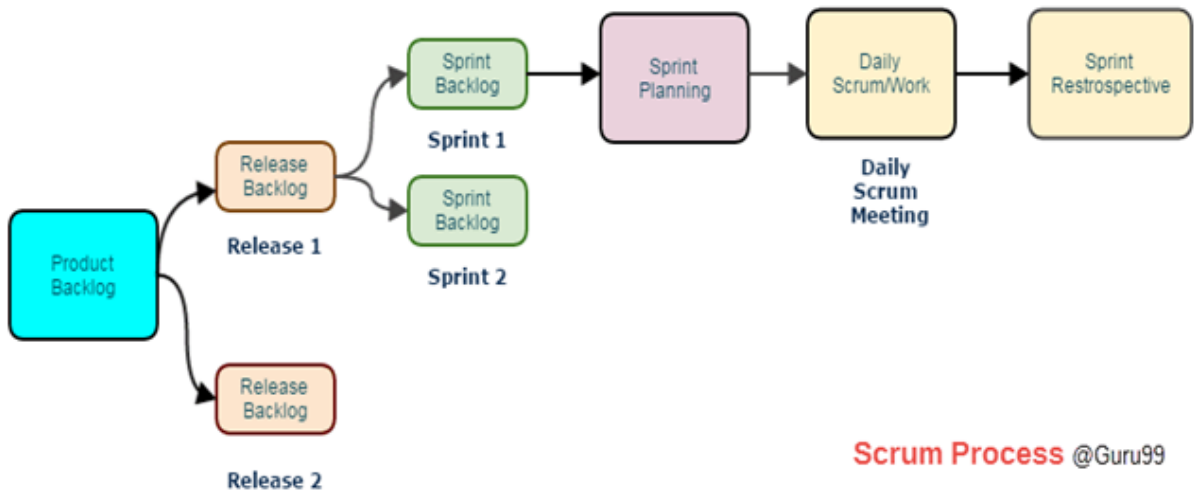
Izvor: Yanado, Ultimate Guide to Scrum Project Management Framework, autor: Lana Pavkovic, objavljeno: 25. veljače 2016. godine, na web stranici: https://yanado.com/blog/ultimate-guide-to-scrum-project-management-framework/ (pristupljeno 9. kolovoza 2019. godine).

Daily Scrum je naziv za kratke sastanke koji se održavaju svakog dana na istom mjestu i u isto vrijeme. Na svakom sastanku, tim prolazi kroz odrađeni rad urađen prijašnji dan te planira što će raditi u narednom danu. Ovo služi za članove time da se osobno izjasne u vezi problema koji bi mogli spriječiti završetak projekta. *Sprint* koji je prije spominjan, odnosi se na vremenski okvir u kojem planirani rad/zadatci trebaju biti kompletirani, njegovo trajanje je najčešće 30 dana. Svi sudjeluju u postavljanju ciljeva, te na kraju sprinta barem jedan inkrement mora proizaći iz toga, a to je korisni komad softvera. Proizvod (softver) se službeno prikazuje na sprint review – u. Nakon svakog završenog Sprinta, održava se *Sprint Retrospective* gdje svi članovi daju svoja mišljenja u vezi *Sprint* procesa u kojem su sudjelovali.

4.6.2. Scrum artefakti

Artefakti po svojoj definiciji označuju nešto od povijesnog značaja. U Scrum razvoju, artefakti služe kako bi provjerilo što je urađeno kao i što je u redu za čekanje. Scrum artefakti jesu *product backlog, Sprint backlog, product increment te burn – down.*

Product backlog referira se na listu koja sadrži popis onoga što treba uraditi. Tokom pregleda tih lista, razvojni tim blisko surađuje s vlasnikom tvrtke, dioničarima kako bi se posložili prioriteti koji su stavljeni na listu čekanja. Lista se može "podešavati" tokom procesa zvanog *backlog refinement. Sprint backlog* predstavlja listu zadataka koji se moraju izvršiti prije nego odabrani sadržaji *product backlog* liste mogu biti pušteni u opticaj. Dijele se u korisničke priče bazirane na vremenu okviru. *Product increment* odnosi se na završni proizvod prethodnog *Sprinta*, prikazuje uspješnost koliko je napretka postignuto. *Burn – down* je vizualna prezentacija količine posla koja je preostala za obaviti. Grafikon sadrži Y os (rad) i X os (vrijeme). Idealni prikaz bio bi padajući grafikon, što označava manji posao koji je preostao za obaviti.



Slika 11. Prikazuje Scrum proces s pripadajućim artefaktima.

Izvor: Guru99, Scrum Testing Methodology Tutorial: What is, Process, Artifacts, Sprint, na web stranici: https://www.guru99.com/scrum-testing-beginner-guide.html (pristupljeno 11. kolovoza 2019. godine).

5. Uloga sistem administratora u poslovnim organizacijama

Peto poglavlje bavi se detaljnijim uvidom u ulogu sistem administratora u poslovnim organizacijama, njegov utjecaj, hijerarhijski položaj te evoluciju kroz promjene poslovanja.

Postoji velik broj radnih mjesta koja se ne cijene dovoljno, jedno od tih mjesta je sistem administrator. Predstavljaju prvu liniju koja brani IT infrastrukturu od mogućih zastoja, pada servera, problema oko nadogradnje. Organizacije mogu nastaviti s radom bez straha da će njihov rad biti prekinut te ostali zaposlenici ne moraju brinuti o IT infrastrukturi. Način na koji sistem administrator podupire rad organizacija je nevidljiv, no njegovo prisustvo je krucijalno. Upravljanje i održavanje IT infrastrukture poslovne organizacije nije nimalo lagan posao. Sve dok IT infrastruktura funkcionira normalno te zadovoljava potrebe poslovanja ne postoji kontakt između. Kada korisnici se susretnu s nekim problem vezanim za rad IT infrastrukture, sistem administrator je prva i jedina točka doticaja. Rješavanjem problema, sistem administrator omogućava zaposlenicima da se usredotoče na vlastiti posao bez potrebe za brigom. Sistem administrator je izravno odgovoran za neprekidan rad, performanse i sigurnost sustava kojim upravlja a na koji se oslanja poslovanje organizacije. Organizacije koje podupiru rad sistem administratora trebaju osigurati dovoljna ulaganja u IT infrastrukturu, uključujući sigurnosno kopiranje i mobilnost u oblaku, time omogućavajući izgradnju i upravljanje sustavima. Održavanje koraka s stalnim promjenama u tehnologiji je iznimno teško, no to je nešto što se očekuje od sistem administratora. Imaju ključnu ulogu u potpori i održavanju kritičnih IT poslovnih operacija i performansi. Rade na projektima svih veličina, od IT nadogradnje pa do zahtjevnijih projekata, kao što su ubrzanje performansi, konsolidacija podataka te poboljšanje analitike. Održavanje sustava i mreže je jedna od važnijih uloga u IT odjelima. Performanse mogu imati duboki utjecaj na organizaciju i njeno poslovanje stoga je rad sistem administratora i njegova uloga krucijalna u organizacijama. Osiguravanjem rada IT podatkovne infrastrukture jedna je od boljih stvari koje sistem administrator može uraditi za poslovanje organizacije. Poslovanje se temelji na sve češćoj promjeni podatkovne infrastrukture. Korištenjem automatizacije smanjuje se vrijeme, trošak i rizik uvođenja promjena.

Uloga tradicionalnog sistem administratora se mijenja. Sam naziv “sistem administrator“ gubi svoj značaj. To je neizbježni proces evolucije posla. Promjene koje je moguće uočiti jesu rezultat dolaska PC – a i Windowsa, gdje sama komercijalizacija pokreće transformaciju, krećući s osnovnim alatima. Valuta kojom se napreduje u svijetu Windowsa vezana je poslovnim ciljevima, PC je stvoren za potrebe poslovanja. Nasuprot tome, oni koji dolaze u svijet administracije sustava iz polja UNIX – a[15] odavno su ovladali sustavom takvim. PC kao i ostale inačice računala uvedene su u standardne programe i poslovne strategije tako jednostavnost koju je moguće ponoviti može pojednostaviti uspjeh organizacije. Današnji poslovi sistem administratora uvelike su olakšani napretkom tehnologije, kao što je upravljanje automatiziranom konfiguracijom, web serverima, sustavima za upravljanje sadržajem i sustavima za upravljanje paketima. Današnji IT izazovi uključuju orijentaciju uslugama, poslovnu agilnost te upravljanje znanjem, no sistem administrator u budućnosti morat će ovladati novim vještinama upravljajući poslovnim vrijednostima te tražeći liniju između agilnost i stabilnost. Tradicionalna organizacija ima slojeve, tj. odjele. Neki su zaduženi za management, prodaju, razvoj poslovanja dok neki su zaduženi za tehnički aspekt poslovanja organizacije. Odvajanje odjela ima svoju svrhu, vještine i tipovi osobnosti potrebni za obavljanje prodaje i managementa drukčiji su od potrebe tehničke prirode. Poslovni slojevi, koji uključuju management, striktno se bave pitanjem prihoda i pitanjem što je svrha same organizacije. Tehnička razina je upravljana predviđenim procesima inženjerstva i upravljanja resursima. Uloga sistem administratora u poslovnim organizacijama okružena je raznim nedaćama i nezahvalnim situacijama. Rješavanje tih problema postiže se boljom komunikacijom između IT i poslovne strane. Sistem admin je zadnji u hijerarhiji do kojeg dođu vijesti vezane za potrebe i promjene u sklopu poslovanja organizacije. Imajući strategiju, višegodišnju pomaže svima da lakše se prebrode svakodnevni izazovi. Potrebna je dokumentacija utjecaja rada sistem administratora u odnosu na poslovanje, jedan od načina je naplaćivanje usluga u modelu nabave.

[15] UNIX ili (Unix) je višezadaćni i višekorisnički operacijski sustav koji se pojavio u ranim 70-im godinama 20. stoljeća, kao plod rada zaposlenika AT&T-a u Bell Labsu, uključujući Kena Thompsona, Denisa Ritchieja, i Douglasa McIlroyja.

Postizanje bolje komunikacije između IT i managementa je krucijalno za dobrobit poslovanja organizacije. IT odjel treba preuzeti inicijativu te analizirati i savjetovati management. Time se povećava percepcija samopouzdanja i osigurava brz nastanak ideja. Ako management zna da It odjel ima vrijedni input, obratiti će se za savjet. Jaka komunikacija i vještina dolaze do izražaja. Sistem administratori trebaju naučiti poslovni žargon te izbjegavati IT žargon, govoreći više o utjecaju njihovog posla, misiji te troškovima. Točka uspjeha je kad management vrednuje toliko mišljenje IT odjela da ga potom stavi u svoj management tim. Jedan od načina formaliziranja takve veze je stvaranje petogodišnjeg plana za IT odjel. Time se IT odjel uključuje u razinu definiranja problema ako se on pojavi. Sistem administratori trebaju ostvariti vezu povjerenja s managementom. Manjak povjerenja dovodi do manjka poslovne vjerodostojnosti i niskog statusa sistem administratora u hijerarhiji. Manjak povjerenja dovodi do financijskog gubitka.

Management obično nema vremena da u potpunosti shvati složene tehničke probleme, čime se jednostavnije okruženje čini mnogo poželjnijim. Postoji percepcija da jednostavnije okruženje košta manje što se tiče održavanja: manje hardvera i softvera, manje IT osoblja te manje vještina potrebne za osoblje. Za sistem administratore, nova tehnologija može biti neodoljiva ali instalacija i konfiguracija označava gubitak vremena i troškove. Postavlja se par pitanja, kodiranje vlastitog softvera ili kupnja novoga, ali da li ta odluka košta ako se gleda dugoročno? Da li nova stavka zadovoljava poslovne potrebe? Koji je utjecaj na poslovanje organizacije? Da li se trošak infrastrukture povećava ako je ona složenija ili je samo potpora vanjskim zahtjevima? Mnoge organizacije nemaju povjerenja u promjene i razne varijacije. Pojavljuje se problem na koji način IT objašnjava managementu što će sustav točno raditi ako se to čini njima jako komplicirano? Kompleksni sustavi koštaju puno?

IT mora puno bolje razumjeti poslovne potrebe s ciljem bolje komunikacije. Time rečenim, pokušaji pojednostavljivanja nisu strani koncept rada sistem administratora. Homogenost sustava je opasnost predviđanju u upravljanju serverima, čime se podrazumijeva da se previše oslanja na pristup koji upravlja očekivanjima. Argument protiv toga da dosljednost ima svoje prednosti ako ne postoji potreba za varijacijom, čime se smanjuje potrebna razina IT znanja. Sve se svodi na razliku između željene ili kontrolirane varijacije i neželjene varijacije. Sistem administratori mogu pretvoriti temu kompleksnosti protiv jednostavnosti u razgovor s managementom na najbolji način a da pritom podržava osnovnu misiju poslovne organizacije. Management može ponuditi određene poticaje IT odjelima kako bi pojednostavili dokumentiranje infrastrukture. Želja za pojednostavljenim okruženjem uzrokovat će probleme kad se pojavi problem nedostatka razumijevanja. IT mora na neki način "prodati" to managementu, može puno više koštati u dugoročnom roku, ali i služiti na duge staze. Jedan od načina na koji management je pokušao razumjeti potrebe IT – ja je kroz SLA. Termin originalno skovan kao legalni dokument između pružatelja usluga i kupaca usluga, time dokumentirajući obećane radnje i posljedice ako usluga nije isporučena. Biti u mogućnosti dati takva obećanja je veliki pokazatelj samopouzdanja. Sistem administratori samim time moraju dati dozu racionalnosti u takvu raspravu u ime organizacije. Agilnost u IT i managementu zahtjeva susret s iznenadnim izazovima. Jedan od načina poboljšanja agilnosti je modularizacija. Modularnost sustava je primjer univerzalne nauke danas. Izgradnja sustava "of the shelf" ima svoje prednosti, uključujući ekonomiju razmjera te mogućnost outsourcinga. Kompleksnost i fragmentacija IT operacija donijela je pregršt novih standarda u zadnjih 25 – 30 godina. Dolaskom COBIT – a[16] i ITIL – a[17] koji sebe predstavljaju kao najbolji radni okviri. Takvi standardi uključuju razne razine preporučenih radnji za upravljanje IT – om i isporukom usluga.

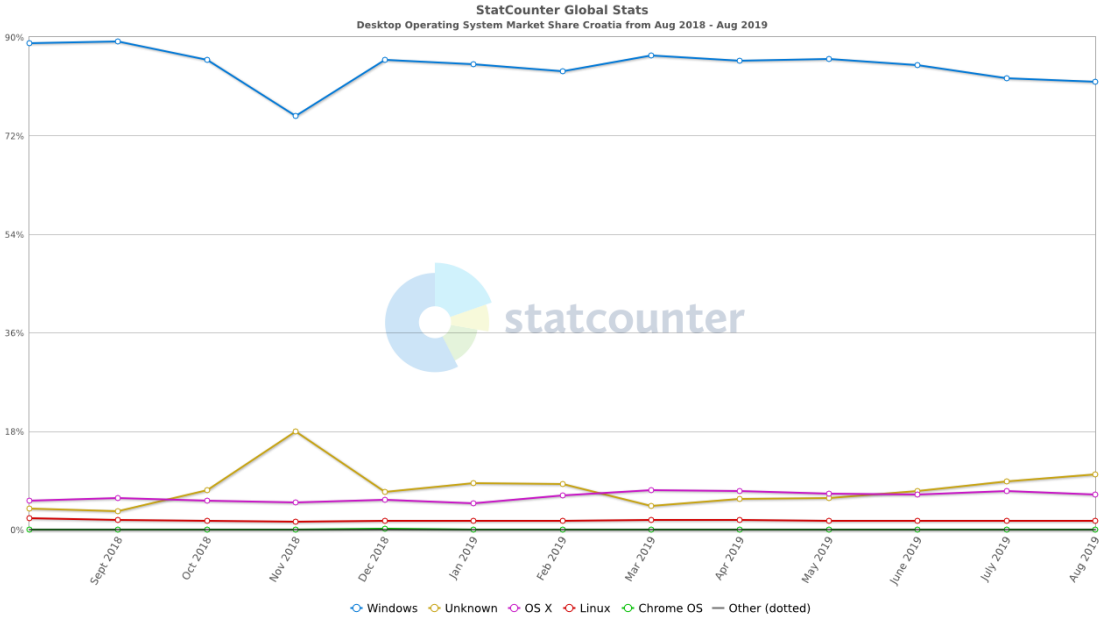
[16] Control Objectives for Information and Related Technologies – radni okvir namijenjen informacijskom managementu. Pruža implementirani set kontrola nad informacijskom tehnologijom i organizira ih u logičke radne okvire i IT procese.

[17] Information Technology Infrastructure Library – Set detaljno opisanih radnji za usluge IT managementa koji se fokusiraju na ravnanje IT usluga s potrebama poslovanja.

6. Linux u poslovnim organizacijama

Šesto poglavlje bavi se financijskim prednostima implementiranja Linuxa u rad poslovnih organizacija.

Linux se zadnje desetljeće dokazao kao pouzdan, siguran operacijski sustav koji može smanjiti totalni trošak vlasništva (TCO[18]) te pruža fleksibilnost i kompetitivnu prednost nad ostalima te time dovodi poslovanje organizacije na višu razinu uspjeha. Linux definitivno nije najpopularniji operacijski sustav kad je u pitanju odabir, ali je omiljen od strane industrije. Većina poslovnih organizacija kao i njihovi IT manageri preferiraju Linux naspram npr. Windowsa jesu manji troškovi, sigurnost, pouzdanost, otvorenost i sloboda odabira. S Linuxom u poslovnom okruženju, IT manageri mogu se koncentrirati na poboljšanje poslovanje radije nego provoditi vrijeme na ažuriranju softvera i dodavanju mnogobrojnih sigurnosnih zakrpa. Ovakav robustan operacijski sustav naširoko je prihvaćen od IT profesionalaca u rastućem poslovanju zbog visoke kvalitete, pouzdanosti i cijene.



Grafikon 1. Prikazuje udio različitih operacijskih sustava u razdoblju od 8.mj 2018. godine pa do 8.mj 2019. godine za Republiku Hrvatsku. Izvor: GlobalStats statcounter, na web stranici: https://gs.statcounter.com/os-market-share/desktop/croatia#monthly-201808-201908 (pristupljeno: 8. rujna 2019.).

[18] Total cost of ownership – Kupljena cijena proizvoda uvećana za troškove rada tog proizvoda.

Linux štedi novac, jednostavno rečeno. Bazični Linux operacijski sustavi su otvorenog koda i besplatno dostupni. Čak i naprednije verzije jeftinije su od Windows servera. Kad se kombinira poslovanje i financijski paketi s Linuxom može se zaraditi ogromna ušteda od ne uzimanja raznih licenci Windowsa, manja je administracija i troškovi održavanja. Linux dolazi s nevjerojatno količinom snage i performansi za jako malu cijenu ukupnog troška vlasništva nego bilo koji drugi operacijski sustav. Jače verzije Linux distribucija manje koštaju od vlasničkih operacijskih sustava, troškovi hardvera i softvera su manji. Ostali bitni faktori troškova jesu administracija i vremenski pad sustava. Mrežne i sistemske upravljačke sposobnosti Linuxa su jako robusne, što znači da IT manageri i vlasnici provode manje vremena na administriranju i upravljanju Linux platformama, samim time investicije u Linux postaju isplativije kroz fiskalne godine. Ako je poslovno okruženje kombinacija više vrsta operacijskih sustava ili samo Linuxa, daje nam se fleksibilnost migriranja na sustave koji najbolje odgovaraju poslovnoj organizaciji. Također uporabom Linuxa, produžuje se životni vijek hardvera iz razloga što je snaga procesiranja puno učinkovitija. S time poslovanje organizacije može ovisiti o starijoj opremi te time nema potrebe za daljnjim ulaganjima u noviji hardver.

7. Radnje sistem administratora u poslovnim organizacijama

Sedmo poglavlje bavi se svakodnevnim radnjama i njihovim opisima administratora u poslovnim organizacijama.

Uloga sistem administratora u poslovnim organizacijama ovisi o specifičnim zahtjevima. U nekim organizacijama sistem administrator poslovnih organizacija ima ulogu sličnu onoj administratora baza podataka u smislu da ta osoba ima široku odgovornost koja obuhvaća razvoj i upravljanje bazom podataka poslovne organizacije. Uz te obaveze tu je i postavljanje hardvera kao i instalacija potrebnog softvera. Ostale organizacije imaju ograničeni raspon rada sistem administratora, koji je pod direktnom kontrolom nekog od senior članova IT odjela. U takvim situacijama, sistem administrator može poslužiti kao predstavnik korisnika kad je u pitanju situacija odgovaranja pitanja korisnika i potrošača.

Jedna od uloga sistem administratora u organizacijama je postavljanje hardvera, uključujući odabir prijenosnih računala i drugog hardvera potrebnog za funkcioniranje organizacije. Tom svrhom administrator mora stalno nadzirati računala, time sprječavajući bilo kakve probleme vezane za softver ili vezane za rad hardvera. Takvi zadaci uključuju periodične nadogradnje softvera i hardvera. Još jedna bitna uloga administratora, iako nije primarna pošto za nju postoji zasebna uloga u organizaciji je razvoj baze podataka koja služi kao repozitorij, glavno spremište za podatke koje organizacija generira svojim radom. U skladu s tim treba postojati odgovarajući režim zaštite pohrane podataka u slučaju pojavljivanja negativnih radnji. Potrebno je održavati rad servera, provoditi periodične provjere rada servera kako bi se osiguralo da radi optimalno. Neke organizacije, ulogu sistem administratora više usmjeravaju prema repetitivnim zadacima i očuvanju poveznice između organizacije i vanjskog svijeta, u pogledu informacijske tehnologije. Administrator također može imati ulogu između organizacije i zaposlenika koji imaju probleme vezane za IT.

Točne radnje koje sistem administrator obavlja su raznovrsne. Pošto je veliko polje sama administracija, zadaci koji se izvršavaju odudaraju od jednog do drugog opisa. Ako bi skupili sve moguća saznanja što točno radi sistem administrator u poslovnim organizacijama, zadatke koji se dalje dijele u manje zadatke, započeli bi s instalacijom i konfiguracijom Linux sustava. Glavna odgovornost Linux sistem administratora je **instalacija i postavljanje Linux sustava i servera**, najčešće za implementaciju na cijeloj organizaciji. Održavanje servera i rad individualno s korisnicima te time se osigurava da sustav radi pouzdano i brzo. Moguće je i postavljanje arhitekture, uključujući baze podataka i skripte za specifične aplikacije i situacije. Administrator također vrši **rutinska održavanja sustava** i rješava probleme vezane za rad servera ako se kojim slučajem pojavi. To uključuje pregled logova i prijavljenih kvarova. Potrebne su radnje s izvornim kodom sustava kako bi se primijenile promjene te osigurale stabilne performanse.

Kreiranje sigurnosnih kopija podataka je iznimno važno za rad i postojanje organizacija. Takve radnje obavljaju se tjedno, dnevno ili frekventno više oviseći prostoru sustava i potrebama organizacije. Ovime se poboljšava povratak sustava radu nakon pada sustava te podržava integritet podataka osiguravajući da ne postoji veći gubitak vitalnih podataka. Uz rutinska održavanja sustava postoji **promatranje performansi sustava** čime se smanjuju moguće usporavanje rada sustava i pad. Pregledavanje dnevnih logova i izvještaja daje na uvid stanje trenutno. **Pružanje tehničke podrške i smjernica** korisnicima i ostalim administratorima. U većini slučajeva Linux administrator surađuje blisko s ostalim developerima kako bi dao odgovore na tehnička pitanja i riješio probleme vezane za rad servera. Asistiraju pri instalaciji i održavanju programa baziranih na Linuxu. **Održavanje sigurnosti sustava** aktivnim identificiranjem sumnjivih radnji i ranjivih područja unutar arhitekture sustava. Uključuje znanje o virusima i ostalim izvorima ranjivosti sustava. **Evaluacija hardverskih i softverskih tehnologija** čime se postiže najbolji dizajn i konfiguracija za radno okruženje poslovne organizacije. Sistem administrator također se bavi **pisanjem dokumentacije** koja pomaže korisnicima sustava i samim administratorima da iskoriste efikasnost i efektivnost sustava.

7.1. Administracija korisničkih računa

Dvije najbitnije obaveze sistem administratora u poslovnim organizacijama je administracija korisnika i monitoring svih korisničkih računa na. Dobar uvid u ovo polje smanjuje rizik pada korisničkih računa te otkriva potencijalne sigurnosne propuste prije nego postanu ozbiljniji problem. Administracija korisnika i računa iznimno je bitna stavka zbog same cirkulacije zaposlenika koji dolaze i odlaze iz organizacije. Moguća pojava odlaska zaposlenika iz određenog razloga (raskid ugovora, otkaz i slično) ostavlja korisnički račun i datoteke različitog sadržaja i razine značaja organizaciji (poslovni planovi, strategije rasta, bilanca…) bez upotrebe i kontrole. Tu nastupa sistem administrator te po potrebi briše korisnika i njegov korisnički račun no datoteke koje su bitne za poslovanje organizacije dijele se vodećim ljudima odjela u kojima je zaposlenik radio te kojima datoteke po potrebi odgovaraju i potrebne su za daljnji rad ili ih se sprema na repozitorij koji ima takvu namjenu.

Dolaskom novih zaposlenika na radno mjesto stvara se potreba za kreiranje korisničkih računa kako bi se ostvario pristup poslovnom sustavu organizacije, njezinim podacima i intranetu. Uz stvaranje korisničkog računa paralelno se stvaraju i grupni korisnički računi. Moguće je napraviti u sustavu organizacije grupne račune koji se dijele na određene odjele u organizaciji. Od IT odjela, prodaje, financija, računovodstva (knjigovodstva), marketinga, upravno – pravnog odjela i slično. Time se određeni zaposlenik može smjestiti u grupu u koju spada oviseći o svojoj poziciji na poslu, tj. poziciji koja je striktno vezana za određeni odjel (zaposlenik: Ivan, radno mjesto: tehnička podrška, grupa: IT odjel). Moguća je pojava da osoba radi blisko s više odjela te mu se time odobrava pristup u više od jedne grupe čime ima uvid i u dokumente. Pripadanje grupi daje i pravo pristupa određenim dokumentima i sadržajima (zaposlenik: Marko, radno mjesto: CEO, grupa: management, dokumenti: bilanca poslovanja 2018./2019: godine). U daljnjem tekstu pobliže će biti praktično objašnjeni primjeri radnji koje sistem administrator obavlja svakodnevno u radu te predstavlja jedan od najvažnijih zadataka.

Bitna vrsta računa kojom sistem administrator upravlja te ima pristup i kojim obavlja spomenute radnje je tzv. **root account**[19], još poznatiji kao **superuser.** Tim se računom ostvaruje kompletna i neometana kontrola kompletnog sustava. Superuser može izvršiti bilo koju naredbu bez ograničenja. Samim time se pretpostavlja da ovo račun sistem administratora. Računi sustava potrebni su za funkcioniranje specifičnih dijelova sustava kao što e – mail računi i **sshd**[20] računi. Ovakvi računi služe samo u specifične svrhe rada sustava, ne smije se modificirati na ikoji način jer time se utječe na rad sustava. Korisnički računi koji se stvaraju zaposlenicima i koji ih koriste svakodnevno u svom radu, pružaju interaktivni pristup sustavu, korisniku i grupi korisnika. Obični korisnici većinom su povezani ovim tipom računa te imaju ograničeni pristup kritičnim datotekama sustava i direktorijima. Unix podržava koncept *Group Account* koji logički grupira određeni broj računa (grupa=pojedini odjel organizacije). Svaki račun pripada nekoj grupi. Ovakav koncept igra veliku ulogu u upravljanju dopuštenjima prema određenim datotekama i upravljanju procesima.

Upravljanje grupama i računima odrađuje se putem četiri glavne datoteke. Prva je **/etc/passwd** koja čuva korisnički račun i informacije vezane za šifru te sadrži većinu informacija o računu Unix sustava. Druga datoteka je **/etc/shadow** koja sadrži enkriptirane datoteke s šiframa točnog računa. Treća vrsta je **/etc/group** te sadrži informacije vezane za grupu pojedinog računa. Zadnja datoteka je **/etc/gshadow** koja u sebi pohranjuje sigurnosne informacije grupnih računa. Sve navedene datoteke moguće je otvoriti putem **cat** naredbe (**cat /etc/passwd /etc/shadow /etc/group /etc/gshadow**). Sljedeći popis sadrži naredbe koju su dostupne na većini Unix sustava a koje se vezane za kreiranje i upravljanje računima i grupama. (**useradd, usermod, userdel, groupadd, groupmod, gropudel**). Na sljedećim primjerima pobliže ćemo se upoznati s korištenjem navedenih naredbi. Hipotetska situacija bi bila dolazak novog zaposlenika u organizaciju te stvaranje grupnog korisničkog računa, modificiranje grupe, kreiranje računa za zaposlenika, modifikacija računa zaposlenika, brisanje računa kao i brisanje grupe.

[19] Root, admin, supervisor ili administrator. Vrsta korisničkog računa koja se koristi za administratorske zadatke.

[20] Proces servera koji prima pristigle veze koristeći SSH protokol i time preuzima ulogu servera za taj protokol. Upravlja autentifikacijom korisnika, enkripcijom, vezama unutar terminala i prijenosom datoteka.

Prije samog kreiranja korisničkog računa, potrebno je kreirati grupni korisnički račun. Moguće je koristiti i grupe već stvorene instalacijom određene Linux distribucije. Lista svih grupa nalazi se u jednoj od prije spomenutih datoteka sistem administratora (**/etc/groups**). Sve unaprijed postojeće grupe jesu u biti specifične grupe računa sustava te nije ih preporučljivo koristiti za običen račune. Sljedeća sintaksa prikazuje stvaranje novog grupnog računa: **groupadd [-g gid [-o]] [-r] [-f] groupname.** Bitno je napomenuti i parametre koji se nalaze te koji se mogu mijenjati po potrebi. **-g GID** predstavlja numeričku vrijednost ID – ja grupe, **-o** označava dopuštenje dodavanja grupe koja ima *non-unique* GID, **-r** prikazuje smjernicu naredbi **groupadd** da se doda kao račun sustava. **-f** je opcija koja se izvršava ako već postoji grupa (isto ime) te **groupname** daje opciju imenovanja grupe. Ako ne specificiramo niti jedan parametar, sustav postavlja parametre sustava. Sljedeći primjer kreira grupu imena newemps (new employes). **groupadd newemps** kreira se grupa u koju ćemo kasnije smjestiti korisnički račun zaposlenika. Nakon kreiranja možemo modificirati grupu pomoću **groupmod** sintakse, **groupmod -n new_modified_group_name old_group_name** ovom sintaksom mijenja se naziv grupe **groupmod - n newemps oldemps.**

Ako postoji potreba za brisanjem grupe određenog odijela zbog gašenja rada tog odjela, prebacivanja svih računa u drugu grupu, moguće je napraviti s jednostavnom naredbom **groupdel.** Brišemo našu stvorenu grupu s **groupdel newemps** (ili **groupdel oldemps**). Ovime brišemo samo grupu, datotekama se može još uvijek pristupiti koje su bile vlasništvo brisane grupe. Datotekama mogu pristupiti njihovi vlasnici. Prije spomenuto stvaranje računa započeli smo stvaranjem grupe, koja je preduvjet tome. Stvaranje računa vrši se s **useradd -d homedir -g groupname -m -s shell -u userid accountname.** Kao i prijašnja naredba za stvaranje grupe, ova također sadrži opcije parametara. Parametri su **-d homedir** koji specificira *home* direktorij za račun, **g- groupname** tim se račun stavlja u grupu po želji (radno mjesto=odjel=grupa), **-m** kreira *home* direktorij ako ne postoji, **-s shell** specificira shell za račun (**/bin/sh**), **-u userid** mogućnost specificiranja korisničko id – ja te **accountname** davanje konkretnog imena računu. Bez specificiranja parametara, stvara se račun s tvorničkim vrijednostima. Sljedeći primjer kreira račun imena ivan, postavlja se putanja prema direktoriju **home/ivan** i grupa **newemps,** ljuska koja bi se koristila bila bi **sh.** Sintaksa glasi: **useradd -d /home/ivan -g newemps -s /bin/sh ivan.**

Nakon kreiranja računa **ivan** možemo postaviti i šifru kojom će korisnik pristupati računu. Šifra se kreira naredbom **passwd ivan.** Stvoreni račun moguće je i modificirati pomoću **usermod.** Koristi iste argumente kao i **useradd,** uključujući i opciju **-l** kojom se mijenja ime računa. Ako želimo mijenjati ime iz **ivan** u **ivan2** sljedeća sintaksa to čini: **usermod -d /home/ivan2 -m -l ivan ivan2.** Zadnja radnja koja preostaje je brisanje korisničkog računa. Brisanje se čini u pogledu poslovanja organizacije, ako zaposlenik napusti organizaciju uslijed raskida ugovora, otkaza ili nekog određenog razloga. Naredba **userdel** briše korisnika. Naredba dolazi s parametrom **.r,** koji miče *home* direktorij računa i datoteku e – maila. Ako želimo sadržati direktorij za potrebe povrata podataka, povrat datoteka bitnih za rad odjela u kojem je zaposleni radio te datoteka bitnih za poslovanje organizacije. Brisanje računa radi se s **userdel - r ivan2** s time da tu postavljamo parametar **– r** koji zadržava direktorij.

7.2. Secure Shell (ssh) pristup

S konstantnim rastom kompleksnosti i veličine sustava uzrokovanih digitalnom transformacijom, poslovne organizacije mogu sadržavati veliki broj **SSH**[21] ključeva. Koja je svrha tih "ključeva" u organizacijama? Njima se omogućava da zaposlenici ostvare pristup udaljenom serveru koji ima pohranjene podatke i kreira veze između računala za automatizirane transakcije. Tipična organizacija ne vodi računa o pravilnom upravljanju tim ključevima, time čekajući da se stvori ogromna sigurnosna rupa koja samo čeka da bude otkrivena od strane hakera ili nekog iz unutrašnjosti organizacije u najgorem slučaju. Razna istraživanja koja su provedena na tu temu dovele su da organizacije s više od jednog milijuna ključeva, 10% njih ima svojstvo "root" označavajući neograničenu upotrebu i pristup serverima. Tom činjenicom krši se polica internog osiguranja organizacije kao i točka kvara kod revizije PCI – DSS – a[22]. To je samo jedan od primjera kojima se organizacije suočavaju usprkos sigurnosti udaljenog pristupa podacima u reguliranom okruženju. Kako organizacija dalje ide putem digitalne transformacije, broj zaposlenika i uređaja povezuje se s mrežom organizacije raste eksponencijalno. Potrebno je osigurati da identitet i upravljanje pristupom ne krši upravljanje organizacijom i kontrolom. Organizacije po pravilu okruženje SSH ključeva ima pod kontrolom. Odjel zadužen za osiguranje rada organizacije usredotočeno je izgradnju perimetra. Iako sistem administratori i oni koji odgovaraju za interni pristup svjesni da postoji na desetke tisuća ključeva koji su izgubljeni, oslobođeni ili zanemareni na računalnoj mreži tvrtke. Slučajevi zanemarivanja ključeva postoje, najpoznatiji je slučaj Edwarda Snowdena, čija su saznanja dovela do toga da NSA [23] ima sposobnost dekripcije SSH ključeva, čime se omogućava čitanje sadržaja određenih SSH sesija. Odmetnuti SSH ključ koristi se kao druga opcija napada hakera. Ulaskom u mrežu putem phishinga, malware ili botneta, pronalazi se neoštećen SSH ključ čime se ostvaruje kretanje mrežom, pristup serverima bez detekcije.

[21] SSH je sigurni protokol za udaljenu administraciju računala

[22] Payment Card Industry Data Security Standard – informacijski sigurnosni standard (kreditne kartice).

[23] National Security Agency – Američka vladina obavještajna agencija koja nadzire i analizira komunikaciju u državi ali i u inozemstvu. Zadužena je i za zaštitu državnih komunikacija i vladine infrastrukture od sličnih organizacija u svijetu.

Realnost je da sistem administratori, tehnička podrška te developeri stvaraju ključeve, čuvajući par njima bitnijih ključeva, zanemarujući ključeve koje su prosljeđivali. Neuspjeh rješavanja problema zadržava neprihvatljivo veliku izloženost riziku, skeniranjem okruženja SSH ključeva. Međunarodna banka, srednje veličine s značajnim digitalnim podacima može sadržavati 10 000 servera pod kontrolom. S 10 000 Unix/Linux računala, banka ima u cirkulaciji 1.5 milijuna ključeva koji omogućavaju razne razine pristupa i pristup preko 75 000 ključeva koji su u vlasništvu svakog sistem administratora. Tim se dolazi do milijarde autentifikacija na godišnjoj razini prema sustavima i podacima. Skeniranje SHH okruženja pronalazi oko 90% ključeva koji su u cirkulaciji i zastarjeli te ih je potrebno ukloniti iz razloga jer su stvoreni za bivše zaposlenike, bivše suradnike na projektu ili za nepotrebne procese.

Poslovne organizacije izložene su velikim rizicima ako određene faktore ne postave pravilno u skladu s radom organizacije.

- Vidljivost SSH ključeva, njihovih veza i mogućnost praćenja
- Procesi za dodjeljivanje vlasništva, korekciju, opoziv i rotaciju ključeva
- Pravila za SSH pristup
- Upravljanje SSH ljuskom i izvještavanje.

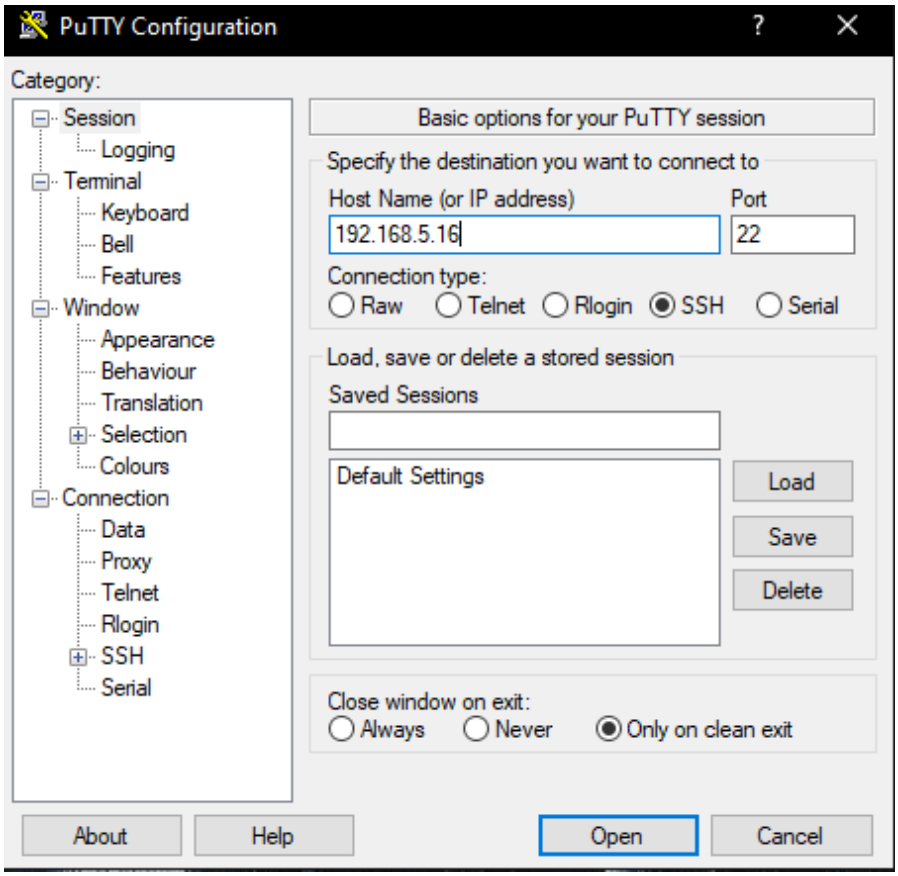
Najbolje radnje za upravljanje SSH ključevima započinje sveobuhvatnim upravljanjem ključevima i njihovim životnim ciklusom. Pomoću pravih alata organizacije mogu:

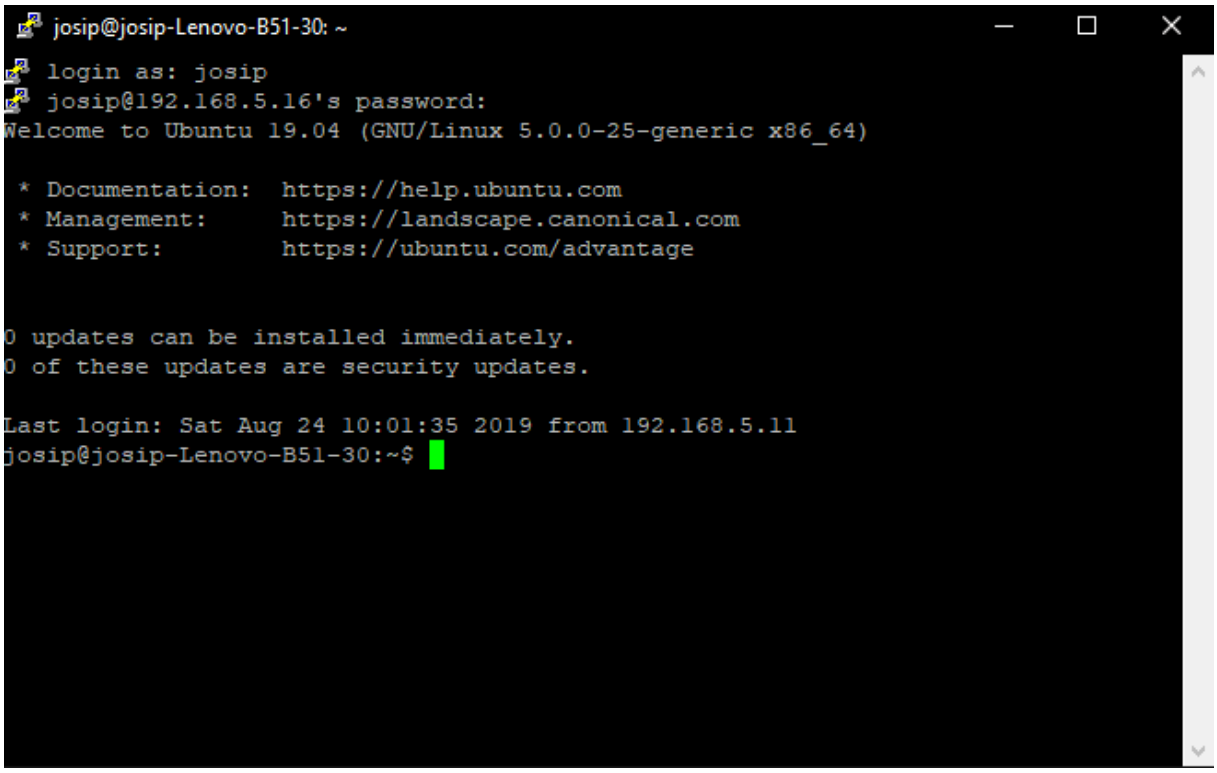
- Okruženja širokog obuhvata staviti pod kontrolu
- Smanjenje administrativnog rada povezanog s ručnim dodjeljivanjem i otklanjanjem poteškoća s pristupom temeljenim na ključu
- Smanjenje potencijalne površine napada
- Izbjegavanje financijskih sankcija
- Smanjenje rizika od pojave poremećaja u poslovanju

7.2.1. PuTTY

Prije samog postavljanja SSH klijenta čime se osigurava sigurna konekcija na SSH server na udaljeno računalo, po potrebi možemo instalirati PuTTY, Linux verziju koja podržava SSH, telnet i rlogin protokole. Omogućava spajanje na udaljene poslužitelje, koristi se za potrebe otklanjanja grešaka. Moguće je koristiti unaprijed instaliranu inačicu OpenSSH za bazičnu upotrebu. Tvorničkim postavkama nije instaliran, gotovo na svim distribucijama instalira se na način: **sudo aptitude install putty,** ostali povezani alati dolaze u individualnim paketima: **sudo aptitude install putty-tools.**

PuTTY je originalno bio napisan za Windowse, no prebačen je i na druge operacijske sustave. Podržava više varijacija sigurnosnog pristupa i nudi kontrolu pristupa preko SSH enkripcijskih ključeva i verzija protokola. Mrežni komunikacijski sloj podržava IPv6. Dolje navedene slike prikazuju ostvarivanje konekcije putem GUI – ja. Potrebne radnje su upisivanje imena računala na koje se spajamo ili IP adresu te broj porta, koji je u svim Linux distribucijama postavljen na 22 no moguće ga je mijenjati po potrebi.





Slika 12 i 13. Prikazuju GUI i postupak ostvarivanja veze između Windowsa 10 i Ubuntu – a 19.04. preko PuTTY – ja.

Izvor: Vlastito računalo

7.3. Važnost upravljanja podacima i pohranom

Organizacije koje nisu svjesne važnosti upravljanja podacima i pohranom imaju male šanse za preživjeti u modernom doba ekonomije. Podaci su najvažnija imovina svake organizacije. Kako bi se kreirala organizacija s fokusom na upravljanje podacima treba razumjeti interne i eksterne utjecaje. Podatak predstavlja temelj poslovnih informacija, znanja i na kraju mudrost donošenja pravih odluka i akcija. Ako je podatak relevantan, kompletan, točan, značajan i ima mogućnost korištenja definitivno pomaže rastu organizacije. Organizacije su primorane upravljati životnim ciklusom podataka jer podaci se stvaraju, pohranjuju, održavaju, koriste i naposljetku uništavaju. Kad se upravljanje odvija učinkovito, životni ciklus započinje i prije nego je podatak pribavljen.

Upravljanje podacima je funkcija planiranja, kontroliranja i dostavljanja podataka učinkovito organizaciji. Učinkovito upravljanje podacima pomaže u minimizaciji potencijalnih grešaka i smanjuje štetu nastalu od "loših" podataka. Učinkovita strategija upravljanja podacima i kvaliteta podataka mora biti implementirana za bolju kontrolu imovine organizacije. Ako se podacima pravilno upravlja, ažurira i poboljšava učinkovitost organizacije raste eksponencijalno, iako su podaci netočni, skloni greškama i loše upravljani, može potrošiti ogromnu količinu vremena i resursa. Sigurnost podataka od iznimne je važnosti i pravilno upravljanje pomaže u osiguranju da vitalni podaci nisu nikada zagubljeni i da su zaštićeni unutar organizacije. Sigurnost predstavlja esencijalnu točku, štiti zaposlenik i organizaciju od mogućih gubitaka podataka, krađe i proboja sigurnosti. Bolje upravljanje podacima pomaže u osiguravanju kvalitete podataka i njenom pristupu. Bolji rezultati pretraživanja u kombinaciji s boljim i bržim pristup podacima organizacije pomaže u donošenju bitnih odluka.

Ako uzmemo primjer SME (Small and Medium enterprise), mala i srednjih poduzeća koji su najčešći tip poslovne strukture. Prema podacima Europske komisije (2018. godine) čak 99% poslovnih entiteta u Europi jesu mala i srednja poduzeća. Tom činjenicom se otkriva da konkurencija i rast idu u ekstremne veličine. Upravo zbog toga dobro upravljanje podacima i pohranom je ključ uspjeha poslovnog kontinuiteta. 93% organizacija koje izgube svoj podatkovni centar na vremenski period od 10 ili više dana imaju veće šanse otići u bankrot u roku od godinu dana. 50% organizacija koje ne posjeduju niti jedan oblik upravljanja podacima u istom periodu prijeti bankrot. Snažan plan upravljanja podacima bitan je uspjeh organizacije. Dobro upravljanje podacima čini organizaciju produktivnijom, s druge strane slabo upravljanje podacima dovodi ne neefikasnosti. Olakšava zaposlenicima potragu za podacima i daje bolje razumijevanje informacije koja im je potrebna za posao. Omogućava se lakša validacija rezultata ili zaključaka. Benefit upravljanja podacima je izbjegavanje nepotrebnih dupliciranja. Brzina kojom organizacija donosi odluke ključ je uspjeha organizacije.

Ako organizacija treba neko vrijeme da reagira na promjene na tržištu ili na konkurenciju to može značiti propast. Dobrim planom upravljanja podacima dopušta se zaposlenicima pristup informacijama i daje im se obavijest o promjeni na tržištu ili radu konkurencije. Dodatno postoji i više sigurnosnih rizika ako podacima se ne upravlja pravilno te upadnu u ruke krivih ljudi. Pravi sustav značajno smanjuje rizik vjerojatnosti takvog tipa događaja. Smanjuje se rizik gubitka vitalnih podataka. Pravom strategijom važni podaci su sigurnosno kopirani i mogući za pristup ako prvotni izvor postane nedostupan. Organizacije koriste različite izvore informacija za planiranje, analizu trendova te upravljanje performansama. Unutar organizacije zaposlenici koriste različite izvore da bi odradili isti zadatak ako ne postoji proces upravljanja podacima. Vrijednost informacije onoliko je dobar koliko i sam izvor informacije. Nepravilno upravljanje podacima može biti od velike štete i iznositi ogromne iznose. Organizacije dolaze do saznanja da imaju problem tek kad se problem i stvori, umjesto da se bude proaktivan, organizacije rade suprotno što ih u dugom roku košta financijski.

7.3.1. Upravljanje podacima

Upravljanje podacima, datotekama je važno kao što smo ustanovili prije. Potrebno je imati sustav koji upravlja podacima te time osigurava prednost naspram konkurenata u pogledu promjena na tržištu i radnji konkurenata. Ako gledamo s stajališta sistem administratora u Unix/Linux području, također ima važnu ulogu u tim radnjama. Samim time što unutar poslovne organizacije administrator ima važnu ulogu administracije korisnika i njihovih računa, također ima i pristup podacima preko pojedinih načina pristupa, najčešće ssh putem. U ovom dijelu striktno se bavimo upravljanjem podacima. Za početak treba raščlaniti pojedine stavke. Svi podaci unutar Unix – a organizirani su u podatke. Podaci su podijeljeni u direktorije. Direktoriji su organizirani u strukturu poput stabla zvanu podatkovni sustav. Sistem administrator održavajući korisničke račune zaposlenika time radi i s datotekama. Kao što je kreiranje, brisanje, kopiranje, preimenovanje i stvaranja linkova međusobno. Unix posjeduje tri vrste tipova podataka:

- **Obične datoteke** – Datoteka unutar sustava koja sadrži podatke, tekst ili programske instrukcije.
- **Direktorij** – Direktorij pohranjuje obične datoteke i specijalne datoteke (hard disk, modem…). Direktorij je ekvivalent mapama u Windowsima i Mac OS – u.
- **Specijalne datoteke** – Pruža se pristup hardveru kao što je hard disk, modemi i Ethernet adapteri. Ostale specijalne datoteke slične su aliasima ili prečacima kako bi se pristupilo datoteci koristeći drugi naziv.

Sistemska administracija u pogledu upravljanja datoteka započinje s naredbom **pwd** (present working directory). Otkriva se trenutni direktorij u kojemu smo prisutni. Ako jesmo u nekome direktorij tipa ***Business Files*** napuštanje direktorija vrši se s **cd** (change directory) bez sufiksa (ime direktorija). Popis svih direktorija unutar podatkovnog sustava ostvaruje se s **ls** naredbom. Ako želimo proširiti opis samih podataka uz naredbu se stavlja **- l.** Takva opcija da listu s sedam kategorija. Lista se sastoji od tipa datoteke i dopuštenja koja ima, broj memorijskih blokova, vlasnika datoteke, grupe kojoj pripada, veličinu (u bajtima), datum i vrijeme stvaranja te ime direktorija gdje se nalazi.

Kod pronalaska datoteka, koje ne možemo pronaći regularnim putem, koristimo se metakarakterima (*****, **?**), imaju posebno značenje u Unixu. Koristeći ***** stvaramo parametar podudaranja 0 ili više karaktera, dok **?** podudara se s jednim karakterom, **ls ch*.doc** pretraga direktorija i svih datoteka s prefiksom **ch*** i završava s **.doc,** * funkcionira kao metakarakter koji stvara parametra pretraživanja.

No, ako želimo ispis svih datoteka koje završavaju s **.doc** koristimo **ls *.doc.** Pretraga i pronalazak datoteka bitna je stavka, brzina kojom se nalazi datoteka koja odlučuje uspjeh poslovne organizacije je krucijalna. U slučaju direktorija koji u sebi sadrži na stotine ili tisuće datoteka s istom ekstenzijom, ali znamo par karaktera početnih datoteke koje nam je potrebna, gornje naredbe u kombinaciji s metakarakerima olakšava poslovanje organizacije u određenim sustavima. U samom podatkovnom sustavu Unix – a uz datoteke i direktorije koje možemo ispisati s naredbama, postoje i datoteke koje su skrivene, odnosno datoteke koje sam korisnik nema potrebe koristiti, ali sistem administratoru mogu pomoći. Nevidljive datoteke karakteriziraju se s (**.**) karakterom, većinom sadrže konfiguracijske informacije. Neki od primjera takvih datoteka su **.profile** (skripta Bourne ljuske), **.kshrc** (skripta Korn ljuske), **.cshrc** (skripta C ljuske) i **.rhosts** (konfiguracijska datoteka za udaljeni pristup). Prve tri datoteke vezane su za tipove ljuski koje se koriste za automatizaciju sustava i zadataka, dok zadnja se odnosi na konfiguracija pristupa na udaljena računala (ssh, telnet, rlogin). Ispis nevidljivih datoteka vrši se parametrom **-a.** Ispisom se saznaje koji direktorij je trenutan (**.**) i direktorij roditelja (**..**).

Većina zaposlenika u organizaciji koriste Windowse i njihov Office paket. Za stvaranje pisanih datoteka (Word), proračunskih tablica (Excel), prezentacija (PowerPoint). Fokusiramo se na stvaranje pisanih datoteka. Unix posjeduje puno vrsta editora (vi, vim, emacs…). Zašto je bitno korištenje raznih editora u poslovanju organizacije? Ako gledamo sa stajališta sistem administratora, zaposlenika koji je zadužen za održavanje infrastrukture i nadogradnju, korištenje editora je od iznimne važnosti što se tiče automatizacije Unix/Linux operacijskih sustava. Editori se koriste za pisanje skripti. Jedna od važnijih stavki svake Linux distribucije.. Skripte automatiziraju pojedine manualne radove. Monitoring performansi sustava, stvaranje sigurnosnih kopija itd. Datoteke (skripte) stvaraju se s **vim ime datoteke.** Stvara se datoteka s danim imenom, pritiskom na **i** ulazi se u edit mode. Izmjena napisanog sadržaja vrši se s **vi ime datoteke** (stvaranje i izmjena ista sintaksa).

Prikaz napisanog sadržaja obavlja se s naredbom **cat ime datoteke** (**-b** za ispis broja linija). Podatak o broju linija, riječi i karaktera postiže se s **wc (word counting) ime datoteke,** gdje ispis daje četiri kolone (broj linija, riječi, bajtova i ime datoteke.). Kopiranje datoteka iz direktorija u direktorij česta je radnja sistem administratora. Kopiranje datoteka za rad zaposlenika, **cp izvorna datoteka odredišna datoteka** (cp bilanca bilanca 2020).

Hipotetski postavljeno poslovno okruženje u potpunosti je prešlo na otvoreni softver, točnije na različite distribucije Unix/Linux – a, koje po potrebi su prilagođene radu zaposlenika različitih odjela. Svaki zaposlenik ima vlastite datoteke, koje su raspoređene u direktorije koji su potom strukturirani u podatkovni sustav. Fokus stavljamo na upravljanje direktorijima. Kako je prije spomenuto, upravljanje podacima je važno jer podatak predstavlja najvažniju imovinu poslovne organizacije. Stvoren ili pribavljen podatak mora biti sigurno pohranjen i nadziran. Direktorij gdje takva datoteka boravi ima zadatak pohrane datoteka i povezanih informacija. Sve datoteke, bile one obične, specijalne ili čak i direktoriji, pohranjeni su u direktorij. Unix koristi hijerarhijsku strukturu za organizaciju datoteka i direktorija. Takva struktura naziva se ***directory tree.*** Stablo ima jedan jedini ***root node (/)*** i sve direktorije sadržane. Direktorij u kojem se zaposlenik prvi put nađe kad se ulogira zove ***home directory.*** Veliku većinu posla obavlja se u tom direktoriju i poddirektorijima koji se stvaraju tokom svakodnevnog rada. Ulazak u ***home directory*** radi se s **cd ~** (lamda=*home directory*).

U situaciji kada sistem administrator ima potrebu za ulazak u neki od direktorija zaposlenika za potrebe kopiranja, stvaranja i povezivanja datoteka to radi pomoću **cd ~ ime korisnika.** Direktoriji su raspoređeni u hijerarhiji s ***root (/)*** na samom vrhu. Pozicija svake datoteke u hijerarhiji je opisana putanjom. Elementi putanje odvojeni su s ***/.*** Apsolutna putanja postoji ako je povezana s ***root,*** iako su takve putanje započinju s ***/.*** (/**etc/passwd**). Putanja može biti relativna, tj. povezana s radnim direktorijem. Relativne putanje nikad ne započinju s **/.** (**personal/res**). Tokom rada stvara se potreba za stvaranjem direktorija za bolju organizaciju rada. Direktorij se stvara s **mkdir ime direktorija,** direktorij ima apsolutnu ili relativnu putanju prema direktoriju koji se stvara, npr. **mkdir mojdirektorij** stvara direktorij unutar trenutnog direktorija, dok **mkdir /tmp/test-dir** kreira direktorij **test-dir** unutar **/tmp** direktorija. Naredba ne stvara izlazni podatak ako je uspješno kreira direktorij.

Upravljanje pravom pristupa datotekama važna je komponenta Unix – a čime se osigurava sigurna metoda pohranjivanja datoteka. Neovlašteni pristup pojedinim datoteka od strane pojedinaca može uništiti organizaciju. Krađa povjerljivih datoteka, uvid u dokumente od značaja organizaciji, kao što su poslovni planovi, investicije, dugoročna ulaganja i slično. Svodi se na poantu da pojedini odjeli unutar organizacije ne smiju (ne bi smjeli) imati pristup datotekama, točnije određenima. Ako odjeli usko surađuju na projektima, tipa marketing i odjel za istraživanje i razvoj, međusobno će razmjenjivati podatke i datoteke te time imati uvid u određene datoteke koje su od značaja za projekt. No odjeli koji ne surađuju na projektima, ali simbiozno borave u organizaciji nemaju pravo na pristup datotekama. Voditelj odjela, npr. marketinga, ima pravo pristupa u direktorij **Marketing 2020,** pravo uvida u datoteke, njihovu izmjenu, kreiranje, brisanje, dijeljenje, čitanje i pisanje . Bliski suradnici kao što je zamjenik imati će određene privilegije uvida, no ne sve, imati će samo pravo čitanja. Asistent će imati pravo čitanja datoteke koje spadaju u opseg njegovog posla. Sve se skalira s obzirom na hijerarhiju u organizaciji i odjelu. Sistem administrator ima svemoguće ovlasti u radu s direktorijima i datotekama, sve radnje može raditi. Svaka datoteka unutar Unix – a ima sljedeće atribute:

- **Dopuštenja vlasnika** – Dopuštenja vlasnika određuju koje radnje vlasnik datoteke smije izvršiti na datoteci.
- **Dopuštenja grupe** – Dopuštenja grupe određuje koje radnje korisnik, tko je član grupe kojemu ta datoteka pripada, može izvršavati određene radnje.
- **Ostala dopuštenja** – Dopuštenja za ostale indicira koje radnje ostali korisnici mogu izvršavati nad datotekom.

Koristeći naredbu **ls – l** ispisuju se razne informacije vezane za dopuštenja datoteke. Dopuštenja se dijele na tri grupe i svaka pozicija u grupi daje specifično dopuštenje ovim redoslijedom: čitaj (read), piši (write) i izvrši (execute). Prva tri karaktera (2 – 4) odnosi se na dopuštenja vlasnika datoteke. Druga grupa karaktera (5 – 7) sastoji od dopuštenja koje grupa smije izvršavati nad datotekom koja pripada njihovoj grupi. Posljednja grupa (8 – 10) prikazuje dopuštenja za sve ostale. Dopuštenja nad datotekama prva su linija obrane unutar sigurnosti Unix sustava. Mijenjanje dopuštenja vrši se korištenjem naredbe **chmod** (change mode). Postoje dva načina korištenja naredbe, simbolični i apsolutni.

Najlakši način je korištenje simboličkog načina. Moguće je dodavati, brisati ili specifirati dopuštenje koristeći operatore. Drugi način korištenja (apsolutno dopuštenje) je uporaba broja čime se postavlja određeno dopuštenje nad datotekom. Svakom dopuštenju pripisuje se vrijednost.

Sr.No.	Chmod operator & Description
1	**+** Adds the designated permission(s) to a file or directory.
2	**-** Removes the designated permission(s) from a file or directory.
3	**=** Sets the designated permission(s).

Slika 14. Prikazuje popis operatora i opisa.

Izvor: Tutorialspoint, Unix / Linux - File Permission / Access Modes, na web stranici: https://www.tutorialspoint.com/unix/unix-file-permission.htm (pristupljeno: 25. kolovoza, 2019. godine.).

Number	Octal Permission Representation	Ref
0	No permission	---
1	Execute permission	--x
2	Write permission	-w-
3	Execute and write permission: 1 (execute) + 2 (write) = 3	-wx
4	Read permission	r--
5	Read and execute permission: 4 (read) + 1 (execute) = 5	r-x
6	Read and write permission: 4 (read) + 2 (write) = 6	rw-
7	All permissions: 4 (read) + 2 (write) + 1 (execute) = 7	rwx

Slika 15. Prikazuje popis vrijednosti i svojstava.

Izvor: Tutorialspoint, Unix / Linux - File Permission / Access Modes, na web stranici: https://www.tutorialspoint.com/unix/unix-file-permission.html (pristupljeno: 25. kolovoza, 2019. godine.).

Tokom kreiranja korisničkog računa na Unix – u, računu se dodjeljuje ID vlasnika i ID grupe. Sva prijašnja dopuštenja također se dodjeljuju na temelju vlasnika i grupe. Dvije bitne naredbe su za promjenu vlasnika i vlasništva datoteka te grupe. **chown** (change owner) mijenja vlasnika datoteke. Primjer bi bio: **chown korisnik ima datoteke.** Vrijednost korisnika može biti ili ime korisnika i ID korisnika (**uid**) na sustavu. Sljedeći primjer pojašnjava: **chown josip datoteka,** mijenja se vlasništvo prema **josip.**

chgrp (change group) mijenja grupu kojoj datoteka pripada. Sintaksa glasi: **chgrp group datoteka.** Vrijednost grupe može biti ime grupe ili ID grupe (**gid**) na sustavu. Sljedeći primjer pojašnjava koncept: **chgrp Marketing 2020 datoteka.** Mijenja se grupa kojoj pripada **datoteka,** te spada pod vlasništvo grupe **Marketing 2020.**

Često kad je naredba izvršena, da bude izvršena zahtjeva da bude specijalne privilegije da bi se zadatak izvršio. Primjer toga je promjena šifre s naredbom **passwd,** gdje se potom šifra pohranjuje unutar datoteke **/etc/shadow.** Zaposlenik, kao običan korisnik nema potrebe za pristupom toj datoteki, no ako zaposlenik promijeni šifru on mora imati dopuštenje za pisanje (write). To označava da naredba **passwd** mora dati zaposleniku dodatna dopuštenja kako bi se omogućilo pisanje unutar **/etc/shadow** datoteke. Dodatna dopuštenja koja se dodjeljuju jesu ***Set User ID (SUID)*** i ***Set Group ID (SGID).***

7.3.2. Upravljanje pohranom

Sam termin *storage management* povezuje tehnologije i organizacijske procese kako bi maksimizirali ili poboljšali performanse njihovih resursa za pohranu podataka. To je ogromna kategorija koja uključuje virtualizaciju, replikaciju, zrcaljenje, sigurnost, kompresiju, analizu prometa i automatizaciju procesa. Količina digitalnih informacija pohranjenih u sustave svake godine se duplicira. Kao rezultat toga, organizacije imaju konstantan pritisak proširenja kapaciteta pohrane. Povećanje kapaciteta pohrane organizacije svake godine financijski je skup poduhvat. S ciljem smanjenja troškova i poboljšanja sposobnosti i sigurnosti pohrane, organizacije se okreću raznim rješenjima upravljanju pohranom.

Mnoge tehnike upravljanja pohranom, kao što je virtualizacija, deduplikacija i kompresija, dopušta organizaciji da bolje koriste postojeću pohranu. Neke od prednosti korištenja takvih tehnika jesu smanjenje troškova, uključujući trenutne kapitalne troškove i operativne troškove za dulji period za održavanje takvih uređaja. Većina tehnika upravljanja pohranom pojednostavljuju upravljanje mrežnom pohranom i uređajima. Time se organizacijama omogućava ušteda vremena, čak i smanjenje IT radne snage koja održava sustave za pohranu čime organizacija zauzvrat smanjuje operativne troškove pohrane. Poboljšavaju se performanse podatkovnog centra. Kompresija omogućava brži I/O i automatizaciju pohrane čime se podiže brzina dodjeljivanja resursa pojedinim aplikacijama. Dodatno, virtualizacija i automatizacija poboljšavaju agilnost organizacije. Takve tehnike omogućuju brže dodjeljivanje prostora za pohranu u skladu s poslovnim potrebama, smanjuju nepotreban prostor i poboljšava mogućnost bržeg odgovora na promjene na tržištu. Tehnike kao replikacija, zrcaljenje i sigurnost, pomažu podatkovnim centrima u pogledu pouzdanosti i dostupnosti. Takve tehnike bitne su stvaranje kopije podataka i arhiviranih pohrana. IT odjeli time pomažu u postignućima povezanima s SLA – ima i postizanju zadanih ciljeva.

Storage management termin je usko povezan s terminom *Storage Resource Management* (SRM[24]). SRM se referira kao jedinstveni softver korišten za upravljanje mrežnom pohranom i uređajima. U kontrastu, *"storage management"* referira se na uređaje i procese, i na sam softver. SRM označuje specijalni softver za alociranje kapaciteta pohrane baziranom na politici i trenutnim događajima u organizaciji. To uključuje upravljanje imovinom, upravljanje kapacitetom, upravljanje konfiguracijom, prijenos podataka i medijske migracije, upravljanje događajima, upravljanje performansama i dostupnost, upravljanje politikom, upravljanje kvotama i mogućnosti upravljanja medijima. Ukratko, SRM je podskup upravljanja pohranom, iako se termini izmjenjuju međusobno. Upravljanje pohranom blisko je povezano s mrežnim rješenjima pohrane kao što je *storage area networks* (SAN [25]) i *network – attached storage* (NAS [26]). Korištenje SAN – a i NAS – a je kompliciranije nego koristiti DAS [27], većina organizacija implementira SRM softver kad implementiraju okruženje za mrežnu pohranu.

Iz razloga što je upravljanje pohranom široka kategorija, teško je pružiti neke instrukcije kako instalirati ili kako koristiti tehnologije upravljanje pohranom. Generalno, tehnologije se mogu implementirati kao softver ili može biti implementirano unutar hardvera samog uređaja. Tehnike se koriste primarno za stvaranje kopija podataka ili arhiviranih pohrana. Procedure implementacije ovise o tipu upravljanja pohranom i odabranom dobavljaču.

[24] Proaktivni pristup optimiziranju efikasnosti i brzine s kojom dostupni prostor se koristi. Pomaže u radu sistem administratora kod upravljanja konfiguracijama i nadziranjem performansi.

[25] Mreža (podmreža) velike brzine koja spaja prisutne dijeljene uređaje za pohranu više servera.

[26] Pohrana računala (servera) spojena na mrežu pružajući podatke heterogenoj grupi korisnika.

[27] Direct – attached storage – Digitalna pohrana koja je spojena direktno na računalo te nije dostupna ostalim računalima.

Primarna organizacija povezana s uspostavljanjem upravljanja pohranom je *Storage Networking Industry Association* (SNIA [28]). Nekoliko važnih specifikacija vezanih za pohranu je stavljeno u javnost od strane organizacije. *Storage Managment Initiative Specification* (SMI – S [29]) i *Cloud Data Management Interface* (CDMI [30]). SMI – S definira pojedine atribute kao što su hardver pohrane, optički kanali, NAS uređaji. Također se odnosi na probleme s softverom za upravljanje pohranom, otkrivanje podataka, dodjela resursa, upravljanje troškovima, upravljanje događajima i zaštitu podataka. CDMI specifikacija pruža standarde za pohranu u oblaku, omogućujući interoperatibilnost među različitim rješenjima za upravljanje pohranom.

Upravljanje pohranjenim podacima jedan je od većih IT izazova s kojim se organizacije susreću. Količina podataka koja se skuplja eksponencijalno raste, tako da veličina podataka raste sve češće. Rezultat svega toga je veća potreba za sigurnošću tih podataka, ali i potreba pretraživanjem, arhiviranjem te povratom podataka. Opremanje organizacije s dovoljno prostora za danas, traži više prostora za sutra, što predstavlja skupu opciju. Prema podacima iz 2010. godine, troškovi pohrane porasli su za 40% do 50% budžeta IT odjela, dok organizacije koriste samo 20% prostora gdje 80% ostaje prazno i čeka na podatke da budu pohranjeni.

[28] Krovna organizacija koja se bavi savjetovanjem oko pohrane, standarda te raznih tehnologija čime promiču učinkovitiji management, kretnju te sigurnost informacija.

[29] Standard stvoren i održavan od strane SNIA – e. Smatra se i ISO standardom. Omogućava široko upravljanje interoperabilnošću heterogenih dobavljača sustava za pohranu.

[30] Standard (SNIA) koji definira protokole za samostalno dodjeljivanje, administriranje i pristupanje pohrani u oblaku.

Veliki broj tehnologija je dostupno danas koje pomažu u upravljanju pohranjenih podataka čime istodobno smanjuje infrastrukturne troškove i troškove na osoblje. Neke od tehnologija jesu:

- **Pohrana u oblaku:** Nudi sredstva organizacijama za smanjenje količine skupih uređaja koje je potrebno imati na licu mjesta, a nudi plaćanje samo prostora koji nam je potreban za skladištenje količine podataka koji odredimo. Preporuča se skladištenje nebitnih podataka kako bi se testirala pouzdanost odabranog pružatelja usluga i postigla razina sigurnosti.
- **Virtualizacija pohrane:** Omogućuje organizacijama da ponovno koriste postojeće platforme za pohranu čime administratori upravljaju tehnologijama raznih pružatelja usluga na homogeni način time također optimizirati prostor za pohranu.
- **Tehnologije učinkovitosti pohrane:** Proizvodi koji pružaju dedupliciranje, kloniranje, replikaciju nude smanjenje troškova smanjivanjem količine podataka koji se trebaju pohraniti.

Uz ključne tehnologije upravljanja pohranom, dizajnirane da smanji administraciju i troškove, organizacije koje provode takve prakse upravljanja podacima. To uključuje prakse kao što je upravljanje životnim ciklusom informacija (ILM[31]), strategije za upravljanje pohranom kojima diktiraju poslovni ciljevi organizacije. Takva pravila pomažu organizaciji u razlikovanju ključnih podataka od nebitnih podataka te time mapirati strategiju pohrane na one razine koje čuvaju najpovjerljivije podatke ili kojima se često pristupa. Pošto se fokusiramo na rad sistem administratora koji svoj svakodnevni rad bazira na Unix/Linux administraciji sustava, sljedeći dio se odnosi na svakodnevne radnje u organizaciji. U samom Linuxu postoji puno alata za upravljanje pohranom, dok se samo neki koriste za svakodnevno održavanje i administraciju. Proći ćemo kroz najčešće korištene prakse za upravljanje pohranom.

[31] Information Lifecycle Management – Strategija za upravljanje sustavima za pohranu te za učinkovitiji IT management.

Najčešće, najvažnija informacija koju želimo saznati o pohrani na sustavu je sam kapacitet i trenutno korištenje spojenih uređaja za pohranu. Mogućnost uvida u pojedina računala zaposlenika daje nam na pregled sve podatke o stanju pohrane. Najčešće se uvid ostvaruje koristeći prije u radu spomenuto **ssh** logiranje. Ulaskom u sustav zaposlenika možemo provjeriti koliko je prostora pohrane dostupno u cjelini kao i trenutno stanje iskoristivosti diskova. Koristeći naredbu **df** tvornički, po sustavu daje nam rezultate mjerene u 1K blokovima, što nije baš korisno. Dodavajući opciju **-h** dobiva se detaljniji opis. Ispisom se dobije uvid u koja particija je gdje spojena na uređaj kao i postotak popunjenosti. Bitno je pratiti stanje pohrane i njezine iskorištenosti.

Generički termin *block device* je oznaka uređaja za pohranu koji čita i piše u blokovima specifične veličine. Ovaj termin se primjenjuje na skoro svaki tip uređaja, uključujući tvrde diskove (HDD), SSD – eve i slično. *Block device* označuje fizički uređaj gdje podatkovni sustav je pisan. Podatkovni sustav zauzvrat, diktira kako su podaci i datoteke pohranjene. Naredba **lsblk** koristi se za prikaz informacija o uređaju. Specifične mogućnosti naredbe ovisi o instaliranoj verziji, ali generalno gledajući **lsblk** može se koristiti za prikaz informacija o samom uređaju, kao i informacije vezane za particioniranje i podatkovnog sustava na koji se piše. Bez ikakvih argumenata, **lsblk** prikazuje imena svih uređaja, male i velike brojeve (Linux kernel koristi za evidenciju uređaja), ako je uređaj moguće maknuti, veličina, ako je spojen na način da samo čita (read – only), tip (diska ili particije) kao i točku spajanja (mounting point). Neki sustavi zahtijevaju **sudo** za prikaz informacija. Kao bi dobili relevantnije informacije vezane za disk i particiju, koristimo **–fs** na nekim verzijama. Ako je to nedostupno na verziji sustava, moguće je ručno replicirati izlazne podatke koristeći **-o** kako bi se zatražio specifični izlazni podatak. Prije korištenja novog diska, moramo ga podijeliti na particije, formatirati ga s podatkovnim sustavom te ga potom spojiti na uređaj ili particiju.

Spajanje je nešto se radi frekventnije nego particioniranje i formatiranje. Spajanje podatkovnog sustava čini ga dostupnim serveru kao točka spajanja. Točka spajanja je direktorij pod kojim podatkovnom sustavu se može pristupiti. Dvije bitne naredbe koje se koriste za ovakav tip upravljanja spajanja: **mount** i **umount. mount** se koristi za spajanje podatkovnog sustava trenutnom stablu podataka. Unutar Linux sustava, jedna jedinstvena hijerarhija, tj. datoteka koristi se za cijeli sustav bez obzira od koliko fizičkih uređaja se koristi. **umount** se koristi se za odspajanje podatkovnog sustava, dodatno naredba **findmnt** korisna je za skupljanje informacija o trenutnom stanju spojenih podatkovnih sustava.

Najjednostavniji način korištenja **mount** je slanje u formatu formatiranog uređaja ili particije i točke na koju se spaja. Npr. **sudo mount /dev/sda /mnt.** Točka spajanja je glavni parametar koji specificira gdje u hijerarhiji novi podatkovni sustav treba biti spojen. Potrebno je odabrati specifične opcije kod spajanja. Iako **mount** može pogoditi tip podatkovnog sustava, uvijek je bolje staviti tip sustava s **-t** opcijom. Primjer za Ext4 podatkovni sustav je: **sudo mount -t ext4 /dev/sda1 /mnt**. Postoji puno više opcija koje utječu na kako je podatkovni sustav spojen. Postoje generičke opcije spajanja, koje je moguće pronaći u **FILESYSTEM INDEPENDENT MOUNT OPTIONS** sekciji pod **man mount.** Podatkovni sustavi također imaju sekciju pod **FILESYSTEM SPECIFIC MOUNT OPTIONS.** Za prikaz opcija spajanja korištenih za specifičnu vrstu spajanja, prosljeđuje se prema **findmnt** naredbi. Ako gledamo samo točku spajanja (read – only) napisali bi **findmnt /mnt. umount** naredba koristi se za odspajanje podatkovnog sustava. Generalna forma je imenovanje točke spajanja ili uređaja trenutnog spojenog podatkovnog sustava.

7.4. Upotreba Linux servera u poslovnim organizacijama

Linux se nakon dugog niza godina postavio kao preferirana opcija kod odabira pokretanja servera. Postao je preferirana platforma za infrastrukturu računalstva u oblaku. Kredibilitet se nije dogodio preko noći, počeo je od skromnih početaka i strpljive gradnje ekosustava pa sve do bogatog portfelja aplikacija. Najvažnije je da se kredibilitet izgradio preko zadovoljnih kupaca koji su uvidjeli da Linux zadovoljava njihove poslovne potrebe za ugodno radno okruženje, nudi sigurnost i robusnu podršku za aplikacije. Unutar široke kategorije, poduzeća teže distribucijama namijenjenima za podršku rada infrastrukture. Uporaba Linux servera dovodi do efektivnijih operativnih troškova, bolje pouzdanosti i dostupnosti.

Takve beneficije dovode boljeg povrata investicija (ROI[32]) za kupce (poduzeća). IT odjel ima cilj pružanja kvalitetne IT usluge koja pomaže organizacijama da budu uspješnije i kompetentnije i idealno da nadmaše konkurenciju a time drže liniju IT troškova i operativnih troškova. Realnost je da većina organizacija, tj. njihovi operativni troškovi generiraju veći trošak na budžet. Kao rezultat toga, cilj je smanjiti operativne troškove i povećati ROI. Softver otvorenog koda, Linux dugo je bio viđen kao rješenje te time nudeći različite vrste podrške kroz tehnologije otvorenog koda dostupne u paketima, od onih koji ne koštaju ništa, onih podržanih od same zajednice pa do komercijalno podržanih proizvoda. Sljedeći benefiti uporabe Linux servera jesu:

- Optimizaciju troškova IT infrastrukture, to uključuje manji broj potrebnih servera koji rade na istim radnim protocima te ostvaruju manje troškove softvera.
- Bolje korištenja osoblja IT odjela kroz korištenje virtualizacije.
- Smanjuje utjecaj neplaniranog pada sustava i učinak na krajnje korisnike i poslovne procese.
- Podržava poslovni rast kroz skalabilnost i visoke performanse.

[32] Return on investment – Odnos između net profita (tokom perioda) i troška investicije (investiranje u određene resurse). Visoki ROI označuje da investitor dobiva zauzvrat više nego što gubi. Koristi se za evaluaciju učinkovitosti investicije ili da se usporede učinkovitosti više različitih investicija.

Upotreba Linux servera pruža troškovno, efikasan i pouzdan temelj za pokretanje važnih poslovnih aplikacija. Smanjuju se troškovi IT infrastrukture jer organizacija treba manje servera koji podržavaju identičan obujam posla. Linux operacijski sustavi imaju manje troškove održavanja kao i troškove samog softvera. Povećava se produktivnost osoblja IT odjela na način da za rad je potrebno manje radne snage, manje vremena za implementaciju, održavanje i upravljanje istim radnim tokovima. Smanjuje rizike u radu što dovodi do veće produktivnosti korisnika, serveri su pouzdaniji, minimiziraju izlazne troškove sustava i aplikacija koji se tiču korisnika. Linux serveri pružaju agilnost i performanse u pogledu zahtjeva organizacije kod radnih opterećenja po pitanju performansi i kapaciteta te potpore virtualizacije. Uz prije spomenute uštede troškova kod fizičke infrastrukture, ušteda se pojavljuje u područjima kao što je struja, mrežna oprema i održavanje operacijskih sustava i servera, sve uštede proizlaze iz samog područja otvorenog koda. Potrebno je manje osoblja za implementaciju, upravljanje i administraciju. Ima još faktora kao što je mogućnost održavanja serverskog okruženja, lakoća virtualizacije, manje problema kompatibilizacije s hardverom i softverom te mogućnost efikasnijeg ažuriranja sustava.

Zbog same stabilnosti sustava, potrebno je manje vremena utrošiti na korisničku podršku i održavanje u usporedbi s drugim alternativnim serverima. Pruža se otpornost i neprekidan rad koji zahtjeva organizacija. To rezultira s manje učestalih i neplaniranih padova sustava, smanjeni utjecaj na korisnike i poslovne operacije u odnosu na standardne industrijske servere. Uz minimiziranje štete koja utječe na korisnike, Linux pomaže održati poslovni kontinuitet. Smanjuje se mogućnost pada sustava koji dovode do utjecaja na poslovne operacije koje stvaraju prihode organizaciji. Uz smanjene troškove rada i osoblja, Linux podržava rad poslovnih operacija kroz skalabilnost i visoke performanse. Može služiti kao platforma za razvoj aplikacija, čime se povećava efikasnost razvoja aplikacija, vrijeme potrebno za implementaciju virtualnih servera je značajne manje te brže nadogradnje ključ su uspjeha poslovnih organizacija, dovodi do poboljšanih performansi aplikacije i veće produktivnosti za korisnike aplikacije. Sve navedene prednosti prebacuju se na poboljšane poslovne rezultate.

Prije smo spomenuli povećanje ROI – ja kao glavnog faktora koji dolazi do izričaja kod korištenja Linux servera. Smanjenje troškova IT infrastrukture, smanjeno osoblje IT odjela, povećanje produktivnosti korisnika te povećani prihodi organizacije. Ulaganja nadilaze početne i godišnje troškove uvođenja Linux servera, uključujući migracije, planiranje, savjetovanje, konfiguracije ili održavanje te osposobljavanje osoblja i korisnika. ROI je omjer neto sadašnje vrijednosti (NPV[33]) i diskontiranog ulaganja te razdoblje povrata je točka na kojoj su kumulativni benefiti jednaki inicijalnom ulaganju. Manji troškovi te benefiti ROI – ja povezani su s korištenjem Linux servera, ali s ROI prednostima, korisnici se susreću s određenim izazovima koji utječu na postizanje dugoročne efikasnosti. Postizanje ROI – ja zahtjeva standardizaciju servera s istim softverskim izdanjima i iste metodologije upravljanja. Rješenje toga je postizanje standardizacije kao jednog od ključnih faktora boljih operativnih i privlačnijih ROI učinaka. Kupci koji imaju veliku raznolikost u IT infrastrukturi najviše će osjetiti benefite standardizacije, standardizacija se odnosi na sve slojeve softverskog stoga, a ne samo na razinu operacijskog sustava. Sljedeći izazov je ometanje rada IT operacija. Troškovi optimizacije započinju konsolidacijom servera na manjem broju korištenih strojeva. Povrat na ulaganje povezan je s efikasnim upravljanjem i niskim troškovima infrastrukture te može povećati uštedu troškova povezanih s smanjenjem broja fizičkih servera u podatkovnom sustavu organizacije.

Organizacije sve više i više zahtijevaju od svojih IT odjela da prihvate digitalnu transformaciju i da budu efikasno konkurentni s širokom transformacijom koja se odvija oko njih. To označava da IT organizacije moraju koristiti IT infrastrukturu koja nudi skalabilnost, pouzdanost, performanse i fleksibilnost za poslovne aplikacije. No, to zahtjeva odgovarajuće financijske resurse. Organizacije su pod stalnim pritiskom optimizacije operativnih troškova u skladu s infrastrukturom i osobljem. Organizacije pogonjene Linux serverima ostvaruju određene benefite, smanjenje servera, osoblja te veća dostupnost.

[33] Net present value – Razlika između trenutne vrijednosti prihoda i trenutne vrijednosti troškova u određenom periodu vremena. Koristi se u planiranju investicija kod analiziranja profitabilnosti planiranih investicija ili projekata.

8. Nadgledanje IT infrastrukture u poslovnim organizacijama

Osmo poglavlje bavi nadziranjem (monitoringom) IT infrastrukture i njegovom važnošću za poslovanje organizacije, načinima nadziranja te nadziranjem od strane sistem administratora.

S današnjom dinamičnom infrastrukturom, serveri, pohrana i mrežni uređaji nisu u opasnosti od utjecaja na poslovanje organizacije svojim potencijalnim padom tokom rada. Zahvaljujući računalstvu u oblaku, virtualnom računalstvu, kontejnerima i softverski definiranim resursima, nove funkcionalnosti mogu biti implementirane bez znanja krajnjeg korisnika. Način na koji biznis konzumira IT resurse putem usluga ili aplikacija, bez izlaganja infrastrukturi. Kao rezultat toga, IT nadziranje infrastrukture i rada ne bi trebalo započeti s prikupljanjem podataka, nego s poslovnim ciljem na umu. Prema procjenama, do 2021. godine, 60% investicija u IT nadziranje uključivat će fokus na metriku baziranu na poslovnim ciljevima, manje od 20% nego u 2017. godine. Nadziranje s perspektive poslovanja označava važnost prikupljanja informacija koje pomažu zdravlju poslovanja. Performanse poslovanja je jedina bitna stvar u pogledu ostvarivanja ciljeva IT nadziranja. Cilj takvog nadziranja nije u kontroli digitalnih procesa poslovanja direktno, umjesto toga daje poslovanju moć vlastito nadziranje koje je efektivno i fleksibilno no istovremeno i troškovno efikasno.

Računala su strojevi te da bi funkcionirala svojom punom radnom snagom, potrebno ih je rutinski održavati. To se odnosi na mnogobrojne virtualne inačice koje se koriste. Računalni sustavi uvijek će biti u potrebi za korekciju. Monitoring, nadziranje IT sustava dopušta različite preventivne akcije prije samog nastanka problema koji može utjecati na poslovanje poduzeća. Ako i kada problem se pojavi, većina nadziranja pruža informacije tehničaru za minimizaciju vremena potrebnog za obnovu sustava. Vrijeme odaziva je ključ uspjeha u ITSM [34] (IT service management). Idealni alat za nadziranje prikuplja sve podatke koji su potrebni za donošenje strateški odluka baziranih na prikupljenim informacijama bez efekta na usluge nadziranja.

[34] IT service management – odnosi se na aktivnosti određene politikom organizirane i strukturirane u procesima i procedurama, provode se u organizaciji za dizajniranje, planiranje, dostavu i kontrolu usluga informacijske tehnologije kupcima.

Postoje dva tipa nadziranja, čiji se jedan dijeli u dvije potkategorije:

- **Nadziranje bazirano na agentu:** Agent radi unutar aplikacije i presretanjem određenog ponašanja aplikacije prikuplja i analizira relevantne podatke o trenutnoj izvedbi.
- **Nadziranje bez agenta**
    - **Lokalni način rada:** Izvođenje na istom fizičkom uređaju (ili virtualnom) može prikupljati podatke koristeći mogućnost posrednika za izvoz; log datoteke, API ili koristiti infrastrukturne mogućnosti bazirane na mjerenim podacima.
    - **Način rada uređaja:** Pokretanje softverske ili hardverske komponente na udaljenoj lokaciji koja je odvojena od ciljanog okruženja, ali i dalje prikuplja relevantne podatke putem mreže ili sličnih metoda lokalnog načina rada.

Nadziranje bazirano na agentu pruža najbolji uvid u detalje. Bez obzira koliko je dobro nadziranje bez agenta, tj. alat baziran na tome, činjenica da postoji agent unutra nadzirane komponente ili aplikacije omogućava detaljniji uvid u rad. Agent zahtjeva resurse kao što je memorija i CPU, što diže troškove održavanja.

Alati bazirani na nadziranju bez agenta ne troše resurse, iako trebaju ovisiti o aplikaciji ili komponenti, npr. korištenje log datoteka ili alata za dohvat takvih podataka. Takvo nadziranje oslanja se prikupljanje podataka indirektno kroz radnik okvir OS – a ili mreže te zahtjeva detaljniju analizu i evaluaciju. Četiri elementa treba uzeti u obzir kod nadziranja IT infrastrukture. Njihova važnost ovisi o relaciji prema određenoj industriji ili poslovnom sektoru. Velike organizacije imaju implementirane ITSM strategije za sva četiri faktora:

- Nadziran na razini hardvera
- Nadziranje rada OS – a
- Nadziranje komponenti aplikacije
- Nadziranje komponenti aplikacija u oblaku

Evidentno je da sve organizacije ovise o sigurnosti, pouzdanosti servera, mrežnim uređajima i poslovnim aplikacijama. S ciljem da se maksimizira efikasnost IT infrastrukture, proaktivni pristup bolji je nego reaktivni pristup. Identifikacija i rješavanje problema prije nego utječu na korisničko iskustvo ili na sigurnost podataka. Benefiti nadziranja povezani su s rezanjem troškova i dugoročnim poslovnim strategijama.

- **Maksimizacija ROI – ja:** Velika IT infrastruktura je velika i dugoročna investicija. Nadziranje efektivnosti aplikacije omogućuje trenutan pristup analizi podataka i trendova i pruža skladišni kapacitet sposoban za izvještavanje diljem organizacije. Budući da IT sustavi počinju postajati složeniji i međusobno povezaniji za učinkovito upravljanje kapacitetima postaje potrebno za praćenje ROI – ja i strategija za planiranje resursa.
- **Predviđanje i rješavanje problema hardvera i softvera prije utjecaja na proizvodnju:** Osnovna svrha nadziranja sustava. Sprječavanje neplaniranih i nepredvidljivih problema u pogledu performansi i kapaciteta koji ometaju ili zaustavljaju proizvodnju usluga i proizvoda. Nadzor IT infrastrukture osigurava vidljivost, pravodobnu intervenciju, omogućuje alarme u okviru procesa za niz vitalnih komponenti sustava. Mreža, memorija, pohrana, struja i/ili okolišni čimbenici kao što su hlađenje, praćenje IT infrastrukture osigurava učinkovito ublažavanje pada rada hardvera i softvera.
- **Maksimiziranje efikasnosti osoblja:** Osoblje je najveća imovina svake organizacije kao i najveći trošak. Koristeći prave alate za nadziranje, osoblje se može fokusirati na bitnije aspekte poslovanja. Pristupi temeljeni na projektima pokazali su se učinkovitijima i omogućavaju učinkovitije korištenje osoblja.

- **Održavanje razine kvalitete i usluge:** Nedostatci u izvedbi mogu ostati nevidljivi dok nenamjerno ne ošteti povjerenje kupaca i poslovnih partnera. IT nadziranje pruža veliku kolekciju mjernih podataka koji se mogu analizirati i koristiti za detaljnije izvještaje o usluzi i usklađenosti, time organizacija ispunjava preduvjete standardizirane usluge.

- **Održavanje kontrole sustava i rasta:** Praćenje rada IT sustava je temelj upravljanja IT – om. Stavlja temelj za buduće širenje automatizacije, skalabilnosti i direktno doprinosi pouzdanosti i otpornosti.

8.1. Načini na koji nadziranje poboljšava poslovanje organizacija

Nadziranje IT infrastrukture pomaže u poboljšanju poslovanja, radu s kupcima te budućem rastu u mnogobrojnom načinima. Za početak možemo spomenuti da upravljanje kapacitetom mrežne propusnosti dovodi do manjih troškova. Od aplikacija do korisnika mora se znati kakvi se podaci razmjenjuju, kolika je propusnost koju mreža koristi te kada i za koju upotrebu. Prepoznavanjem koji tokovi uzrokuju gušenje mreže nema potrebe za implementacijom tehnologije za pronalazak greške, minimizira operativne i kapitalne rashode te omogućuje timovima bolje planiranje i donošenje odluka kroz razumijevanje kompozicije mrežnog prometa. Suvremena rješenja za praćenje performansi otkrivaju promjene na mreži koje utječu na performanse te omogućuju promjene prije bilo kakvog poremećaja usluge. Brzina kojom virtualni uređaji dolaze i odlaze na mrežu označava da upravljanje performansama mora biti na detaljnoj razini kako bi se otkrile promjene koje mogu biti prolazne no mogu ostaviti značajne posljedice.

Aplikacije koje loše odrađuju posao guše propusnost Interneta, usporava mrežni promet, smanjuje korisničko iskustvo te negativno utječe klijenta. Brza rješenja poboljšanja performansi može značajno poboljšati korisničku produktivnost i uvelike poboljšati iskustvo korisnika. Proaktivna analitika i obavijesti od značajnih alata za nadziranje omogućuje planiranje održavanja poslovnih učinaka. Nadziranje omogućuje organizacijama da planiraju troškove infrastrukture te ih dodjeljuju unutarnjim troškovnim centrima ili krajnjim kupcima na temelju potrošnje. Za potrebe pružatelja usluga, ovo je korisno za naplatu na temelju korištenja usluge. Ista mogućnost može se koristiti za optimizaciju konfiguracija kako bi se smanjili troškovi infrastrukture. Korištenje uređaja u vlasništvu zaposlenika (BYOD) dugi niz godina je rastući fenomen koji organizacije prihvaćaju. Istodobno je potrebna IT podrška kako bi se omogućio širok raspon uređaja i proizvođača što je problem za mrežu organizacije. Snažna rješenja za praćenje mreže i infrastrukture pruža neposrednu vidljivost ključnih mjernih podataka o uspješnosti podržanih tehnologija omogućujući različite vrste uređaja.

8.2. Sistemska administracija (nadziranje) IT infrastrukture poslovnih organizacija

Nadziranje i održavanje rada IT infrastrukture poslovnih organizacija svakodnevni je posao sistem administratora. Nadziranje rada IT infrastrukture moguće kroz razne alate koji sami po sebi imaju ukomponirane razne naredbe za pregled, održavanje i rutinske preglede. U ovom dijelu rada posvetiti ćemo se pregledu i popisu naredbi koje imaju svrhu cjelokupnoga nadzora rada IT infrastrukture koja funkcionira pod Linux distribucijama. Naredbe koje će biti obrađene dostupne su u svim verzijama Linuxa i mogu biti korisne za nadziranje i pronalazak aktualnih krivaca za probleme vezane za performanse.

Naredba **top** jedna je od najčešćih naredbi namijenjena za nadziranje performansi i rada te korištena od strane praktički svih sistem administratora te je dostupna pod svim verzijama Unix/Linux operacijskog sustava. Naredba se koristi za ispis svih procesa koji se izvršavaju te za ispis svih aktivnih *real – time* procesa ispisanih u listi koji se regularno ažuriraju. Prikazuje upotrebu CPU – a, upotrebu memorije, veličinu *cache* memorije, *swap* memoriju, *buffer size*, PID procesa te korisnika. Također prikazuje visoku upotrebu memorije i procesorske snage za pokretanje svih procesa. Naredba **vmstat** upotrebljava se za prikaz statistike virtualne memorije, procesa kernela, diskova, sistemskih procesa, I/O blokova i aktivnost CPU – a. Tvornički nakon instalacije naredba nije dostupna te je potrebno instalirati paket **sysstat** koji uključuje **vmstat** program. Naredba **lsof** koristi se u većini Unix/Linux operacijskih sustava za prikaz otvorenih datoteka i procesa. Otvorene datoteke su datoteke diska, mrežni priključci, uređaji te procesi. Glavna upotreba naredbe je ako imamo situaciji da određeni disk ne može biti odspojen od podatkovnog sustava te prikazuje greške kod korištenja datoteka. Naredbom je moguće identificirati koje su datoteke točno u upotrebi. **tcpdump** je jedna od najkorištenijih naredbi namijenjena za praćenje razmjene podataka na mreži, korišten za hvatanje ili filtriranje TCP/IP paketa koji se primaju ili razmjenjuju na određenom sučelju mreže. Nudi opciju spremanja uhvaćenih paketa u datoteku za kasniju analizu.

netstat naredba je korištena za nadziranje pristiglih i odlaznih paketa koji sadrže statistiku te sadrži sučelje za pregled statistike. Korištena je za nadzor rada mreže i rješavanje problema povezanih s mrežom. Naredba **htop** je naprednija verzija prijašnje naredbe **top.** Interaktivnija je te nudi određene značajke kao što je korisničko sučelje za upravljanje procesima, prečace (tipkovnica), vertikalni i horizontalni pregled procesa. Naredba nije uključena u Linux operacijske sustave te ju je potrebno instalirati zasebno (YUM [35]). **Iotop** slična je **top** i **htop** naredbi, ali ima funkcije pregleda te ispisa u realnom vremenu za I/O diska i procesa. Korištena je za pronalazak točnih procesa i diskova koji čitaju/pišu procese. **Iostat** je naredba koja prikuplja i prikazuje ulaznu i izlaznu statistiku uređaja za pohranu. Korištena je za praćenje performansi uređaja za pohranu kao što su uređaji, lokalni diskovi, udaljeni diskovi kao što je NFS[36]. **iptraf** je alat za nadziranje rada mreže u realnom vremenu (IP LAN). Skuplja razne informacije kao što je praćenje prometa IP – a na mreži, uključujući informacije TCP – a, detalje ICMP – a, detaljne preglede mreže TCP/UDP i pakete TCP konekcije. Također prikuplja generalne informacije i detaljnu statistiku TCP, UDP, IP, ICMP i aktivnosti sučelja.

psacct ili **acct** alati korisni su za nadzor pojedinog korisnika i njegove aktivnosti. Oba procesa pokreću se u pozadini te pomno prate aktivnost svakog korisnika unutar sustava i resurse koji se troše od strane tih korisnika. Korisno je za praćenje aktivnosti pojedinog korisnika, što rade, koje naredbe koriste., koje resurse troše te koliko dugo su aktivni u sustavu. **nethogs** je program (sličan **top** naredbi) koji prati rad svakog mrežnog procesa sustava. Također prati aktivnost u realnom vremenu, mrežne propusnosti prometa korištenog od svakog programa ili aplikacije. **iftop** još jedan od alata za nadzor koji prikazuje frekventno ažuriranu listu mrežnih resursa (izvor i destinacija) koji prolaze kroz mrežno sučelje sustava.

Sve navedene naredbe moguće je koristiti pojedinačno, jednu po jednu upisivati u terminal. Druga mogućnosti je automatizacija, odnosno pisanje skripti kojima je moguće ukomponirati sve naredbe koje se izvršavaju sekvencijalno te time se olakšava posao nadziranja i održavanja IT infrastruktura organizacija.

[35] Yellowdog Updater, Modified - Alat za upravljanje instalacijama paketa.

[36] Network File System - Dozvoljava korisniku klijentskog računala pristup podacima preko mreže kao što bi se pristupalo pohrani.

9. Automatizacija

Deveto poglavlje bavi se pojmom automatizacije (poslovna automatizacija i IT infrastrukture) te prednostima automatizacije u pogledu sistemske administracije operacijskih sustava.

Automatizacija podrazumijeva kreiranje tehnologije i njezine primjene s ciljem da kontrolira i nadzire proizvodnju i dostavu različitih proizvoda i usluga. Obavlja zadatke koje su prethodno izvodili ljudi. Cilj automatizacije je povećanje efikasnosti i pouzdanosti. Sastoji se od ključnih elemenata , sustava i poslovnih funkcija u gotovo svim industrijama. Automatizacija se koristi u velikom broju područja kao što je proizvodnja, transport, obrana, radnje te informacijska tehnologija. Automatizacija se izvodi na puno načina u različitim industrijama. Primjer toga je informacijska tehnologija, skripta može testirati softver te skripta stvara izvještaj o radu aplikacije. Postoje razni softverski alati dostupni na tržištu koji mogu generirati kod za aplikaciju. Korisnik mora jedino konfigurirati alat i definirati proces. U ostalim industrijama, automatizacija poboljšava produktivnost, time stvarajući uštedu na vremenu i financijama. Automatizacija se razvija brzo te poprima svoj oblik i u poslovnoj inteligenciji i njenim aplikacijama je nova forma visoke automatizacije. U domeni tehnologije, utjecaj automatizacije raste rapidno, u softveru i hardveru. Unatoč prednostima, neke manualne intervencije su potrebne i preporučljive i kad alat može sve obaviti. Automatizacija je krucijalna za upravljanje, mijenjanje i prilagodbu ne samo na IT infrastrukturu, ali i način na koji organizacija posluje kroz te procese. Pojednostavljenjem promjena kroz automatizaciju, dobiva se potrebno vrijeme i energija za fokusiranje na inovacije. Automatizacija nije zamjena za ljudsku radnu snagu. To je mogućnost radi potrebe za ljudskom interakcijom, fokus je na prednostima nađenima u produktivnosti, konzistentnosti i efikasnosti. To predstavlja paradoks, postajemo efikasniji koristeći automatizaciju, ljudski doticaj postaje bitniji no manje učestaliji. Prednosti su bolja produktivnost, osoblje može više vremena provoditi na bitnijim projektima. Bolja pouzdanost, smanjenjem potrebe za ljudskom interakcijom smanjuje se pojava eventualnih problema. Time se ostvaruje znanje gdje koji proces, testiranja, ažuriranja će se dogoditi. Loše strane automatizacije je trošak, uz trošak tu je i vrijeme potrebno za postizanje željenog učinka.

9.1. Automatizacija poslovnih procesa

Automatizacija u poslovnim organizacijama vodi se terminom BPA[37] (Business Process Automatization). Automatizacija poslovanja je ravnanje BPM – a[38] (Business Process Management) i BRM – a[39] (Business Rules Management) pomoću razvoja modernih aplikacija kojima se zadovoljavaju potrebe tržišta. Koristi se za automatizaciju raznih procesa čime se povećava efikasnost i kontrola troškova unutar cijele organizacije i njenih odjela. To je postignuto kroz BPM i BRM. BPM i BRM izvrsne su tehnologije, ako djeluju same nisu dovoljne. Digitalna transformacija je ključ uspjeha na tržištu a za to je potrebno partnerstvo IT – ja i poslovanja. Fokus samog IT – ja prebačen je s usluživanja unutarnjih potreba, kao što je efikasnost i kontrola troškova prema stvaranju odnosa s vanjskim kupcima i time kreirajući poslovne prilike. Tradicionalno poslovanje treba automatizacijski model za napredak. Aplikacije mogu biti smještene u kontejnere, time dopuštajući da ih se lakše održava, ažurira i distribuira. Kontejnerizacija poslovnih rješenja dopušta aplikaciji da njoj bude pristupano preko većeg broja platformi te ujedno dopuštajući developerima i poslovnim korisnicima da upravljaju životnim ciklusom aplikacija.

Fokus je na puštanju aplikacija u rad kao mikrousluga preko skalabilne infrastrukture. Kombiniranjem modela stvorenih od poslovne strane i IT – ja stvara aplikacije bazirane na procesima. Takve aplikacije su agilnije i mogu se prilagoditi potrebama korisnika. Automatizacija poslovanja ubrzava procese razvoja, skraćujući vrijeme između puštanja u opticaj i povratne informacije korisnika. BPA i BPM ostaju samostalne strategije koje se mogu koristiti na poboljšanje efikasnosti i smanjenje troškova rada organizacije.

[37] Tehnološka automatizacija kompleksnih poslovnih procesa.

[38] Disciplina operacijskog managementa koja koristi metode za otkrivanje, analizu, modeliranje, mjerenje, poboljšanje, optimiziranje i automatiziranje poslovnih procesa.

[39] Softver koji se koristi za definiranje, implementaciju, izvršavanje, nadziranje i održavanje kompleksnih logičkih odluka a koristi se od strane operacijskih sustava unutar organizacije.

Automatizacija poslovnih procesa uključuje puno pokretnih dijelova. Implementiranje ponovno korištene automatizacije dopušta kontrolu takvih procesa, oslobađajući vrijeme i resurse. Automatizacija poslovnih procesa pojavljuje se kad želimo automatizirati poslovnu rutinu. BPA nam može poslužiti samostalna strategija s ciljem da organizacija bude što efikasnija, ili može biti efikasno kombinirana s inicijativama poslovnih procesa. Dok BPM dopušta da se razumije *end – to – end* poslovni procesi, BPA se koristi za kontinuiranu provjeru i napredovanje efikasnih procesa. Automatizacija poslovnih procesa produžuje se na sustav IT – ja te time automatizira procese kreirane specifične za potrebe organizacije. Kreiranje specifičnih automatizacija može koštati i biti vremenski konzumirajuće. Rješenja BPA sve više koriste umjetnu inteligenciju za bolje razumijevanje i prilagodbu nestrukturiranim bazama podataka. Time se stvara bolje iskustvo korisnika. Automatizacija poslovnih procesa daje bolju transparentnost i kontrolu nad procesima upravljajući raznim pregledima, dopuštenjima i pristupima. Ubrzava procese i smanjuje greške manualnog rada, jer čak i malene greške u radu mogu biti veliki trošak, tjerajući organizacije da troše vrijeme i resurse na ponavljajuće radnje. Smanjuje ljudsku grešku i štedi vrijeme centralizirajući proces. Automatizacija je sve u svemu za jednu stvar, davanje vremena i energije za fokusiranje na rješavanje važnijih problema. Automatizacija radi usko s algoritmima kako bi se bolje organizirali poslovni procesi. Puštajući automatizaciji da ubrza i ravna procese, organizacija time dobiva više vremena i energije za potrebe klijenata.

9.2. Automatizacija IT infrastrukture

IT automatizacija, nekada se referira kao automatizacija infrastrukture je uporaba softvera za kreiranje repetitivnih instrukcija i procesa koji zamjenjuju ili smanjuju ljudsku interakciju s IT sustavima. Softver za automatizaciju radi u okvirima instrukcija, alati i radni okviri obrađuju zadatke bez potrebe za ljudskom intervencijom. Automatizacija je ključ IT optimizacije i digitalne transformacije. Moderno dinamično IT okruženje ima potrebu za bržim skaliranjem nego ikada i IT automatizacija je krucijalna za postignuće toga. U teoriji, ako je to zadatak unutar IT – ja, neka razina automatizacije se može primijeniti na takav zadatak. Automatizacija se može integrirati i primijeniti na bilo što, bilo to automatizacija mreže pa do infrastrukture i računalstva u oblaku i upravljanja konfiguracijom. Automatizacija se širi i na područja kao što je kontejneri, metodologije tipa DevOps te u polja kao računalstvo u oblaku, sigurnost, testiranje te nadziranje infrastrukture.

Holistički pristup IT automatizaciji pomaže ukloniti repetitivne procese. Time se timovima daje više prostora na produktivnosti, smanjenju grešaka, poboljšanju suradnje među timovima te oslobađa vrijeme koje se kasnije koristi na bitnije stvari. Kako bi pokrenuli poslovne sustave infrastruktura mora biti postavljena. Većina onoga što radimo definirano je u softveru, a premještanje na softver povećalo je opseg i kapacitet onoga što je moguće obaviti. Time se ostvaruje pravo kodifikacije procesa, što pomaže zadovoljiti zahtjeve poslovanja pod povećanim savjesti o troškovima i vremenskim ograničenjima. Zahvaljujući kodifikaciji ima se predložak po kojem se radi. Nisu sve aplikacije stvorene na isti način. Zahtijevaju različite postavke, podatkovne sustave, priključke i korisnike. Nakon automatske dodjele resursa, potrebno je odrediti što će ti resursi činiti. Potreban je robustan sustav za upravljanje konfiguracijom koja omogućuje developerima jednostavno definiranje infrastrukture na način da svi razumiju unutar IT odjela. Što je jednostavnije automatizirati zadatke i rutine upravljanja sustavom, lakše je obaviti zadani posao. Što je IT sustav kompleksniji, sve je teže upravljati pomoćnim dijelovima. Potrebno je kombinirati više automatiziranih zadataka i njihovih konfiguracija preko različitih grupa sustava i računala. To je orkestracija, ukratko. Povrh toga, moguće je kontrolirati orkestracije s robusnim automatizacijama. To nam omogućuje praćenje svih i povezivanje svih naprednih sustava.

Bilo da koristimo tradicionalniji pristup implementaciji aplikacije ili da koristimo kontinuiranu integraciju i kontinuiranu implementaciju (CI/CD). Uspješna implementacija aplikacije ovisi o potpuno učinkovitom skupu automatiziranih zadataka i mogućnosti. Time se smanjuje mogućnost pojave ljudske pogreške te istovremeno se poboljšava efikasnost i protok. IT automatizacija nam omogućuje da svoje aplikacije implementiramo s sigurnošću, konfiguriramo potrebne usluge iz samog starta te pokrenemo sve aplikacije. IT automatizacija i automatizacija poslovnih procesa nije jedno te isto. Poslovna automatizacija se mijenja konstantno. Potrebno je usvojiti strategije automatiziranja kako bi se omogućila digitalna transformacija. U budućnosti biti će sve veća potreba za autonomijom kao inteligencija ugrađena u takve sustave. Proširit će vidike na više područja te time pokriva veći raspon rada.

10. Kontejnerska tehnologija

Deseto poglavlje bavi se uporabom tehnologije kontejnera i njezinom implementacijom u poslovne organizacije kao i prednosti korištenja takve tehnologije.

Kontejnerska tehnologija, znana kao i *OS-level virtualization* referira se na paradigmu operacijskog sustava u kojoj kernel dopušta postojanje više izoliranih procesa. Takvi procesi se nazivaju kontejneri, zone, virtualni privatni serveri, particije, virtualna okruženja ili virtualni kernel. Program pokretan na običnom operacijskog sustavu može promatrati sve resurse (spojene uređaje, datoteke i mape), mrežu, snagu CPU – a tog računala. Programi pokretani unutar kontejnera mogu vidjeti samo njegov sadržaj i uređaje pripisane kontejneru. Na UNIX/Linux operacijskim sustavima takva opcija se vidi kao naprednija implementacija **chroot** [40] mehanizma, koji mijenja **root** mapu za trenutne procese i podprocese. Uz izolacijske mehanizme, kernel nudi i upravljanje resursima kako ne bi došlo do međusobnog utjecaja kontejnera.

Na običnim operacijskim sustavima za osobna računala, program može pristupiti resursima sustava. Kao što su hardverske mogućnosti, (CPU i mrežna konekcija), podaci koji se mogu čitati ili pisati (datoteke i mape) te spojena periferija (web kamera, printer, skener ili faks). Operacijski sustav može dopustiti ili zabraniti pristup takvim resursima baziranima na tome što program traži. Program ima vezu s tim resursima i operacijski sustav ima kontrolu nad interakcijom. S virtualizacijom operacijskih sustava, moguće je pokretati programe unutar njih, gdje su određeni resursi su alocirani. Program očekuje cijelo računalo, no kad je unutar kontejnera vidi samo alocirane resurse dodijeljene njemu. Nekoliko kontejnera je moguće stvoriti na sustavu, gdje svaki koristi dio resursa. Svaki kontejner može sadržati bilo koji broj računalnih programa. Programi rade istovremeno ili odvojeno i čak mogu imati interakciju jedan s drugim. Kontejnerizacija je česta kod virtualnih okruženja gdje se koristi za sigurno alociranje hardverskih resursa između više korisnika. Sistem administratori koriste kontejnere također za konsolidiranje serversko hardvera premještajući usluge na odvojena računala u kontejnere na jednom serveru.

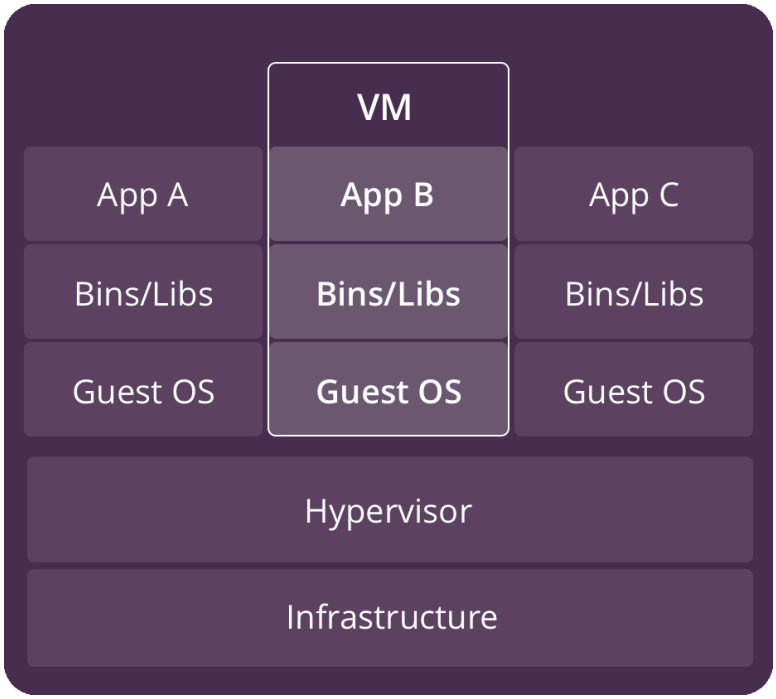
[40] Operacija koja mijenja root direktorij za trenutni proces koji se izvodi.

Tipičan scenarij je odvajanje više programa u više odvojenih kontejnera za poboljšanje sigurnosti, neovisnosti hardvera i dodanih značajki upravljanja resursima. Poboljšana sigurnost osigurana je korištenjem **chroot – a.** Kontejnerizacija manje opterećuje nego korištenje same virtualizacije zato je programi u virtualnim particijama koriste normalne pozive sustave i ne trebaju biti emulirani ili biti pokretani u virtualnim računalima kao što je slučaj s punom virtualizacijom ili paravirtualizacijom. Takav tip virtualizacije ne zahtjeva podršku hardvera za efikasne performanse. Kontejnerizacija nije fleksibilna kao ostali pristupi iz razloga što ne može biti pokretano na drukčijem operacijskom sustavu osim onoga na kojem je trenutno. Linux i različite distribucije imaju tu mogućnost, dok Windowsi ne mogu biti pokretani. Neke implementacije pružaju mehanizam ***copy – on – write*** [41] (CoW) čime se olakšava stvaranje kopije podataka i oslobađa se više prostora.

[41] Tehnika za upravljanje resursima za efikasno kopiranje operacije na modificiranom resursu.

10.1. Kontejneri vs. Virtualna računala (VMs)

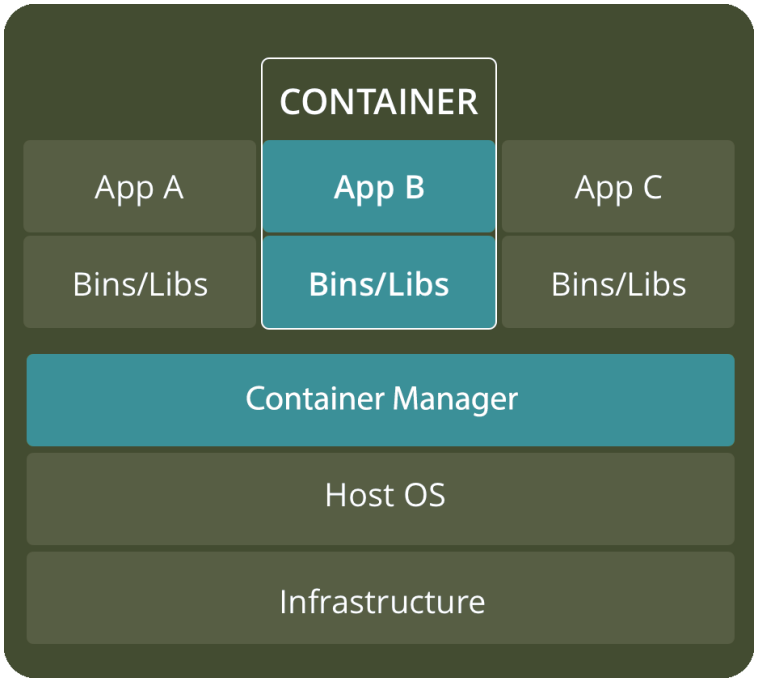
Virtualno računalo je emulacija računalnog sustava. Operativni sustav i njene aplikacije dijele resurse hardvera iz jednog računala, ili iz više njih. Svako virtualno računalo zahtjeva vlastiti operacijski sustav gdje je hardver postaje virtualiziran. Hipervizor, predstavlja softver, hardver i firmver koji kreira i pokreće virtualna računala. Nalazi se negdje između hardvera i virtualnog računala te je potreban za virtualizaciju servera. Od nastanka prihvatljive tehnologije virtualizacije i usluga računalstva u oblaku, IT odjeli objeručke su prihvatili virtualna računala za dobitak benefita kao što je smanjenje troškova i povećanje efikasnosti. Vms zauzima puno resursa sustava, iz razloga što ne pokreće samo kopiju operacijskog sustava ali i virtualnu kopiju hardvera koji pokreće sustav. Time se dodaje puno ciklusa trošenja CPU – a i RAM – a. I dalje je ekonomično ako se usporedi s pokretanjem zasebnih računala, ali za pojedine aplikacije ova opcija može biti kobna, što je dovelo do nastanka kontejnera. Neke prednosti Vms – a jesu da su svi resursi operacijskog sustava dostupni aplikacijama, uspostavljen su alati za upravljanje, uspostavljeni sigurnosni alati i bolja kontrola sigurnosti



Slika 16. Prikazuje arhitekturu Vms – a.

Izvor: Backblaze, What's the Diff: VMs vs Containers, autor: Roderick Bauer, objavljeno: 28. lipnja 2018. godine, na web stranici: https://www.backblaze.com/blog/vm-vs-containers/

Kontejneri su druga priča, umjesto virtualizacije kompletnog sustava, jedino što se virtualizira je operacijski sustav. Kontejneri se nalaze na samom vrhu fizičkog servera i njihov računalo (host) tipično je Linux ili Windows. Svaki kontejner dijeli kernel sustava. Dijeljene komponente su samo za funkciju čitanja (read – only). Dijeljenje resursa znatno smanjuje potrebu za stvaranjem koda sustava, što označava da server može pokretati više radnih tokova s jednom instalacijom operacijskog sustava. Kontejneri su iznimno, lagani, njihova veličina iznosi megabajta i treba samo par sekundi da se pokrenu. U usporedbi s kontejnerima, virtualnim računalima potrebno je par minuta da se pokrenu. U kontrastu s virtualnim računalima, sve što kontejner treba je dovoljno operacijskog sustava, podržanih programa i resursa sustava, to znači da može se staviti dva do tri puta više aplikacija na jedan server nego što je moguće s virtualnim računalima. Dodatno treba spomenuti, kontejneri su prijenosni, konzistentno okruženje za rad, razvoj, testiranje i implementaciju. Prednosti kontejnera su smanjenje potrebe za resursima, smanjena veličina kopija (snapshot), brže pokretanje aplikacija, smanjeno i jednostavnije ažuriranje sigurnosti te manje koda za transfer i migraciju.



Slika 17. Prikazuje arhitekturu kontejnera.

Izvor: Backblaze, What's the Diff: VMs vs Containers, autor: Roderick Bauer, objavljeno: 28. lipnja 2018. godine, na web stranici: https://www.backblaze.com/blog/vm-vs-containers/

Kontejneri i virtualna računala imaju prednosti i mane, konačna odluka ovisi o specifičnim potrebama. Virtualna računala bolja su programe koji zahtijevaju sve resurse operacijskog sustava i funkcionalnosti kad je potrebno pokretanje više aplikacija na serveru, ili kad želimo imati više sustava na raspolaganju. Kontejneri su bolja opcija kad želimo maksimizirati broj aplikacija na minimalnom broju servera. Za većinu, idealna postavka bila bi obje opcije. S trenutnim stanjem tehnologije virtualizacije, fleksibilnosti virtualnih računala i minimalnih zahtjeva kontejnera moguće ih je zajedno ukomponirati te im pružiti okruženje za rad. Ako organizacija pokreće veliki broj instanci na istom operacijskom sustavu, onda bi za to odgovarali kontejneri, samim time uštedjeli bi značajno vrijeme i novac nasuprot virtualnih računala.

What's the Diff: VMs vs Containers

VMs	Containers
Heavyweight	Lightweight
Limited performance	Native performance
Each VM runs in its own OS	All containers share the host OS
Hardware-level virtualization	OS virtualization
Startup time in minutes	Startup time in milliseconds
Allocates required memory	Requires less memory space
Fully isolated and hence more secure	Process-level isolation, possibly less secure

Slika 18. Prikazuje najznačajnije razlike između kontejnera i virtualnih računala.

Izvor: Backblaze, What's the Diff: VMs vs Containers, autor: Roderick Bauer, objavljeno: 28. lipnja 2018. godine, na web stranici: https://www.backblaze.com/blog/vm-vs-containers/

10.2. Poslovna vrijednost implementacije kontejnera u poslovanje

Postalo je iznimno bitno za poslovanje shvaćanje tehničke strane apliciranja tehnologije kontejnera. Iste značajke koje čine kontejnere bitnim alatom developera također stvara poslovnu vrijednost organizacijama. Kontejnerizacija pruža ključne tehničke prednosti s potencijalom da od toga beneficira poslovna strana.

Predvidivost i pouzdanost kontejnera razvija put kojim oni funkcioniraju. Svaki kontejner je za sebe izolirano okruženje koje sadrži aplikaciju i sve što treba aplikaciji za pokretanje, funkcionira zasebno, tako ništa izmijenjeno u drugom kontejneru ne utječe na aplikaciju u drugom kontejneru. Ako aplikacija radi u kontejneru, kontejner može biti premješten bilo gdje bez da utječe na funkcioniranje aplikacije, ako radi jednom, radi svugdje. Developeri mogu slobodno mijenjati aplikaciju unutar kontejnera, nakon što su uvjeti postavljeni, ne mogu se mijenjati. Time se izbjegavaju situacije u kojim nestandardni operacijski sustavi se čuvaju kako bi pokrenuli određenu aplikaciju. Ako se i pojavi neki problem, ili je s platforme ili kontejnera, lako je identificirati izvor. Time se smanjuje vrijeme potrebno za rješavanje problema, smanjuje vrijeme zastoja te poboljšava ukupnu predvidljivost usluge.

Kontejnerizacija pomaže u uklanjanju tehničkih poteškoća koje stoje između razvoja i implementacije. Tehnička ovisnosti su većinom zanemarene ili nisu u potpunosti razumljive u pogledu stvaranja nove aplikacije, te predstavlja faktor koji dovodi do velikih usporavanja. Time se dovodi sukoba između misije razvoja, čiji operativni timovi i usporavanje procesa premještanja aplikacije od razvoja do proizvodnje. Razvojni timovi motivirani su kreiranjem najboljeg proizvoda čak i kada u potpunosti ne razumiju novi proizvod. Operativni timovi, gledaju kako maksimizirati stabilnost čak i ako to uključuje ignoriranje inovacijskih prilika. To znači da negdje u radu operacija zaustavit će se implementacija novog koda razvojnog tima ili čak od strane razvojnog tima njega samoga će ga staviti u proizvodnju čak i ako ignoriraju upute operativnog tima. Kontejnerizacija pruža jednostavno rješenje tom problemu. Ako razvoj gardi aplikaciju unutar kontejnera, operacije samo održavaju kontejner, bez brige za aplikaciju unutar kontejnera. Time se ravna proces razvoja i ubrzava se plasiranje aplikacije na tržište.

Puno je lakše i brže ažurirati i dodati nove značajke aplikaciji unutar kontejnera te testirati nove značajke. Iz razloga što su kontejneri na neki način diskretni, developeri samo ažuriraju jednu značajku bez obzira na utjecaj na druge procese. Time se ubrzava proces nadogradnje i nadogradnje novih značajki. Kontejneri nude opciju testiranja aplikacija na malom uzorku kupaca prije puštanja na tržište. Time organizacija je puno odgovornija prema kupcu i sprječava skupe implementacije novih značajki koje korisnici ne žele. Kontejnerizacija nije prikladna za sve tipove aplikacija. Dizajnirani su takvim životnim ciklusom da radije nego da opada kvaliteta njih, ako prestanu funkcionirati optimalno, brišu se i zamjenjuju drugim. Kontejnerizacija nije efektivna za aplikacije koje spremaju podatke na duže vrijeme. Aplikacije namijenjene analitici, streamingu, web servisima idealni su kandidati za kontejnere. Banka ne bi mogla staviti u kontejner zahtjev za upravljanje osjetljivim financijskim podacima korisnika, dok bi mobilna aplikacija bila savršena. Razumijevanjem primjene kontejnera, organizacije mogu osigurati sebi predvidivost i pouzdanost, brže reagiranje na tržište i brže ažuriranje aplikacija koje će imati najviše koristi od toga.

10.3. Poslovna vrijednost uporabe Docker – a

Docker (Enterprise Edition) smanjuje troškove IT infrastrukture i poveća učinkovitost u tri ključna područja: smanjenje troškova IT infrastrukture i povećana optimizacija, produktivnost developera te učinkovitost rada IT operacija.

Čak i podatkovni centri koji su većinom virtualizirani popunjeni su serverima s niskom stopom iskoristivosti, često ispod 50%. Dok je IT pokušao uklopiti virtualna računala na manje servera, svaka aplikacija je zahtijevala vlastito virtualno računalo i operacijski sustav. Samim "kontejneriziranjem" aplikacija i pokretanja kontejnera u jednom virtualnom računalu (nasuprot pokretanja svake aplikacije u zaseban Vm), organizacije smanjuju upotrebu resursa za 25% ili više za isti opseg rada. Docker nudi pokretanje na čistim serverima samo s operacijskim sustavom servera, bez hipervizora ili virtualnih računala. Time se smanjuju zahtjevi resursa za 45% ili više. Svaki od pristupa smanjuje upotrebu resursa na skali od 30 do 35% te zahtjeve RAM – a za 7%. Pokretanje aplikacija na manjem broju OS instanci generira uštede od 40% ili više na samom hardveru. Smanjuje utisak podatkovnom centra, smanjuje potrošnju energije i troškove fizičkog prostora te smanjuje troškove softvera smanjenjem broja procesora.

Developeri troše značajno vrijeme na samo postavljanje infrastrukture, što može potrajati danima ili čak tjednima a da projekt nije ni započeo. Nakon što je projekt službeno započet, određeni problemi u razvoju zagorčavaju napredak razvojnih timova. Teško je i vremenski konzumirajuće replicirati probleme bez točne infrastrukture. To točnije označava postavljanje par virtualnih računala samo da bi se stvorila potpora za testiranje i rješavanje nadolazećih problema. Troši se vrijeme na svakom novom projektu jer je potrebna nova infrastruktura. Docker kontejneri dopuštaju developerima da eliminiraju rutinske poslove te dobiju više na produktivnosti i kreativnosti. S manje nepotrebne infrastrukture, uvođenje developera je do 65% brže, tako da novi developeri čak i prvog dana posla mogu poslati svoj kod. Dopušta se izolacija koda i njegovih ovisnih dijelova zajedno bez potrebe za kreiranjem konflikata aplikacije s drugim kontejnerima. To na kraju označuje slobodu biranja alata za posao, nego biti ograničen infrastrukturom ili kompatibilnosti aplikacije.

Krajnji rezultat je puštanje koda do 13 puta frekventnije. Kad je Docker implementiran, naknadni projekti se implementiraju puno brže. Dobivanje boljeg softvera kao proizvod, kreira stvarnu poslovnu prednost na tržištu. Prednosti u stvaranju učinkovitosti od korištenja virtualizacije i ostale tehnologije, postojeća infrastruktura i aplikacije i dalje su financijski veliki trošak za održavanje. Razna ažuriranja, testiranja i rješavanje problema u radu i dalje uzimaju poveći dio radnog prostora IT – a. Docker omogućava do 300% brže puštanje IT projekata u rad, timovi nemaju potrebe se brinuti o infrastrukturi i manje se posvećuju testiranju te se time više posvećuju strateškim projektima. IT pojednostavljuje održavanje aplikacija i podršku. Docker pojednostavljuje IT okruženje, olakšava ažuriranje aplikacija i promjena puno brže iz razloga što se kontejneri pokreću zasebno unutar infrastrukture. Time se vraća pouzdanost aplikacija i dostupnost te dopušta IT – u da proaktivno upravlja sigurnosnim rizicima, osiguravanje aplikacije lakše je ako je uključeno manje varijacija nje.

Dopušta se digitalna transformacija time stvarajući most između tradicionalnog i modernog IT – a. S jednom platformom i lancem opskrbe koji upravlja svim aplikacijama. Nudi se sloj kroz horizont IT infrastrukture, alata i aplikacija. Uštedom financija i vremena na postojećoj infrastrukturi i aplikacijama, organizacija može reinvestirati i novac i vrijeme na transformaciju poslovanja. Docker pomaže IT – u i razvojnim timovima da rade učinkovitije. Moderniji rad daje organizacijama potrebnu agilnost za pomicanje cjelokupnog poslovanja brže te time čineći iskorak u digitalnoj transformaciji.

11. Uloga IT sigurnosti u poslovnim organizacijama

Jedanaesto poglavlje bavi se važnošću kibernetičke sigurnosti, zaštite podataka organizacije, načinima zaštite.

Društvo, poslovanje i organizacije sve više i više se isprepleću s tehnologijom. Svi podaci koji su spremani na običan komad papira, sada se pohranjuju na tvrdim diskovima za brži pristup, lakše prenošenje s i u udaljene lokacije. Podaci kupaca, e-mailovi, telefonski brojevi i financijski podaci također su pohranjeni elektronički. Uloga IT sigurnosti (*cyber security*) u organizaciji vitalna je za zaštitu podataka i osiguranje da usluge će i dalje biti dostupne bez neočekivanih zastoja. Moderne organizacije praktički ovise o računalnim sustavim za pohranu podataka, kontaktiranje kupaca i izvršavanje raznih zadataka kao što je pretraživanje, marketing i strateško planiranje. Financijski uspjeh organizacije kao i uspješna implementacija ciljeva ovisi o stanju računalnih sustava. Vitalno je da sustav ostane netaknut od najezde softvera (*third party software*) koji ima za cilj ući u sustav. Bezuspješno osiguranje sustava dovodi do gubitka podataka, gubitak kompetitivne prednosti (patenti), gubitak zaposlenika/kupaca i njihovih podataka pa sve do gubitka povjerenja od javnosti.

Trenutno informacijska sigurnost je krucijalna svim organizacijama, bez obzira na njihovu veličinu. Informacija (podatak) je najvažnija imovina svake organizacije. Za organizaciju, podatak je vrijedna varijabla u cijelom poslovanju te treba biti pravilno zaštićena. Sigurnost je kombiniranje raznih sustava, operacija i internih kontrola kako bi se osigurao integritet i povjerljivost podataka i operacijskih procedura u organizaciji. Informacijska sigurnost odrađuje četiri važne funkcije unutar organizacije, osigurava siguran način rada organizacije, štiti podatke koje organizacija prikuplja i koristi, štiti tehnološku imovinu koju organizacija koristi te štiti mogućnost organizacije da funkcionira besprijekorno.

Sigurnost se implementira kroz kontrolu pristupa i principe CIA. Kontrola pristupa se odnosi na proceduru kontroliranja tko ima pristup informacijama i do kojeg opsega mogu mijenjati ili koristiti tu informaciju. Kontrola pristupa također se odnosi i na fizičku kontrolu (zgrade, sobe…). CIA princip označuje povjerljivost (*Confidentiality*), integritet (*Integrity*) i raspoloživost (*Availability*) te se odnosi na tri stanja koja podaci kojem čuvamo trebaju posjedovati. Podaci trebaju biti povjerljivi (nema neozakonjenog pristupa), treba zadržati svoj integritet (nema izmjena, manipulacije ili uništavanja podataka) te treba biti dostupno u bilo kojem trenutku.

U organizacijama, podatak je važan poslovni atribut i esencijalan je za poslovanje te zbog toga treba biti propisno zaštićen. To je posebno važno u poslovnom okruženju u kojem podatak je izložen velikom broju prijetnji i ranjivosti. Maliciozni kodovi, hakiranje kao i odbijanje rada usluge postaju sve češći, sve ambiciozniji i sve napredniji. Implementiranjem polica sigurnosti u organizaciju, štiti se tehnološka imovina koja se koristi u organizaciji. U pogledu zaštite funkcionalnosti organizacije, management i IT management odgovorni su za implementaciju sigurnosne politike koja štiti mogućnost funkcioniranja. Informacija je najvažniji element organizacije a koji je potreban za poslovanje. Osim toga da organizacije čuva informacije korisnika, stoga je važno štititi informacije. Bez informacija, poslovanje se ne može odvijati. Osiguravanjem sigurnosti, omogućava se pokretanje poslovanja te njegovo održavanje.

11.1. Unix/Linux sigurnost u poslovnim organizacijama

Poslovne organizacije koje koriste sustave bazirane na Unix/Linuxu susreću se s izazovima kad je u pitanju upravljanje privilegiranim pristupom (PAM). Takva tehnologija čini temelje velike većine srednjih poduzeća te je prisutna duže od postojanja Windows – a ili nekog novijeg operacijskog sustava. Kao i s svakom tehnologijom, pravi izazovi usvajanja Unix/Linux – a u radno okruženje predstavljaju izazove tokom implementacije. U poslovnom svijetu gdje postoje unutarnje prijetnje, virusi te napredne ustrajne prijetnje samim time stvara se potreba za povećanom sigurnošću tehnologije. Današnja radna okruženja Unix – a moraju ispoštovati stroge zahtjeve usklađenosti, različite sofisticirane prijetnje i potreba za rad s potpunim s sustavima van Unix – a, kao što su Microsoft Windowsi. Kad je u pitanju kibernetička sigurnost u bilo kojem okruženju, bio to Unix ili Windowsi, postoje četiri stupa koje treba imati na umu za odgovarajući pristup:

- **Provjera autentičnosti** – Provjera identiteta osobe ili sustava koji zahtijeva pristup Unix sustavu.
- **Ovlaštenje** – Osiguravanje odgovarajuće razine pristupa autentičnoj osobi ili sustavu.
- **Administracija** – Sastoji od zadataka i aktivnosti povezanih s održavanjem životnog ciklusa korisničkog identiteta.
- **Revizija (audit)** – Aktivnosti koje se izvršavaju kako bi se zadovoljili zahtjevi u vezi s sukladnosti kojima je potrebna provjera autentičnosti, autorizacija i administracija u skladu s utvrđenim pravilima i najboljim praksama.

Kad je riječ o upravljanju Unix/Linux korisničkim računima, posebice privilegiranim računima, teško je išta postići s izvornim sposobnostima ugrađenima u Unix. Činjenica je da upravljanje identitetima (računima) i pristupom za više Unix sustava kompleksno i neučinkovito jer svaki sustav je za sebe tj. nepovezan. Unix koristi veliki broj tipova korisničkih računa, ali svaki Unix sustav mora imati root račun. Važno je da svaki korisnik ima samo jedan račun na sustavu. Unix koristi prije spomenute, privilegirane račune, oni imaju dvije karakteristike: to su najjači korisnici u bilo kojoj organizaciji te oni uključuju dijeljene korisničke račune i super korisnik račune.

Privilegirani korisnički računi jesu identiteti koji imaju sposobnost izvršavati administrativne funkcije, konfiguracijske promjene, instalacija i izvršavanje programa. Ako ne upravljamo tim računima na pravilan način, riskiramo rad organizacije kroz: gubitak liste šifre koja cirkulira organizacijom (i one koje nikada nisu mijenjanje), zaposlenici koji napuste organizaciju te znaju šifre, promjene poslova ili uloga gdje su dane dodatne ovlasti, zaposlenici kopiraju i međusobno dijele SSH ključeve, davanje pristupa krivim osobama te određene unutarnje prijetnje. Prije spomenuti root korisnički račun, tzv. super korisnik je tvornički postavljen račun na svakom Unix sustavu. Takav račun ima neograničene ovlasti (instalacija, modificiranje, brisanje te pokretanje programa). S takvim tipom moći, postoje ozbiljne prijetnje. Tome može pomoći naredba **sudo** ali ima određena ograničenja. Pogrešno upravljanje root računom i ostalim pravima korisnika predstavlja rizik organizaciji, uključujući financijske gubitke, gubitak reputacije, penaliziranje od strane regulatora te gubitak korisnika.

Za smanjenje rizika oko dijeljenih i privilegiranih korisničkih računa, treba uzeti i odgovornost. Korisnici trebaju biti svjesni kako koriste svoje privilegije. Za takve situacije koristi se koncept “zadnja privilegija“ koji smanjuje pristup takvim računima. Ograničavanje pristupa kritičnim serverima ne znači da sistem administrator neće biti u mogućnosti da obavlja svoje zadatke. Pristup će samo bolje biti kontroliran na način da se kontroliraju dopuštenja korisnika na određene datoteke. Svaka datoteka i direktorij na Unix sustavima označena je s tri seta dopuštenja koji označuju na način koji ime se pristupa i od koga: dopuštenje za vlasnika, grupu i za sve ostale korisnike te svaki set sadrži dopuštenja kao što su čitaj, piši i izvrši. Root račun u potpunosti ignorira dopuštenja, dopuštajući pristup svakoj datoteci na sustavu bez obzira na dopuštenja vezana za datoteku. Većina datoteka u vlasništvu je root računa te imaju dopuštenje za smanjenje dopuštenja ili potpunog blokiranja pristupa drugim korisničkim računima. Auditiranje (revizija) je jedan od važnijih aspekata Unix/Linux sigurnosti iz dva razloga. Revizija pomaže u dijagnosticiranju problema unutar radnog okruženja te sigurnost i sukladnost zahtijevaju povećanu razinu vidljivosti pristupa pravima individualaca i aktivnosti koje izvode s tim pravima.

11.2. Rad na sigurnosti sistem administratora u poslovnim organizacijama

Postoji mnogo aspekata sigurnosti Linux sustava, od postavljanja korisničkih računa pa od osiguravanja toga da pravi korisnici nemaju više ovlasti nego im je potrebno za obavljanje svojih svakodnevnih poslova. Ovime dijelom daje se uvid u najvažnije naredbe terminala koje sistem administrator koristi u svom svakodnevnom radu te njihov opis. Naravno, postoji mnoštvo alata s GUI – jem koji uvelike pomažu u tome, kombiniraju više naredbi skupa i njihove značajke te pritom daje vizualni (ljepši) prikaz cijelog stanja sustava. Svaki administrator mora imati snalaženje i vještine terminala u malom prstu, kao i određeni broj naredbi znati napamet oviseći o situaciji. Ove naredbe vezane su striktno za sigurnost sustava i njezinih korisnika. Moguće je napisati skripte koje sve same odrađuju jednim pritiskom Enter tipke, no za početak ćemo prolaziti jednu po jednu naredbu zasebno, kasnije sekvencijalno izvođenje možemo nadodati.

Pokretanje (izvršavanje) naredbi s **sudo**, umjesto prebacivanja korisnika na root je jedna od boljih praksi koja pomaže osigurati korištenje root privilegija samo kad je potrebno i time limitira utjecaj grešaka. Pristup naredbi **sudo** ovisi postavkama unutar datoteke **etc/sudoers** i **etc/group.** Za potrebe praćenja aktivnosti na sustavu koriste se naredbe **who** i **w** te prikazuju tko je ulogiran na sustav iako **w** prikazuje puno više informacija kao što je izvor logiranja, kad su se ulogirali te koliko im traje sesija. Praćenje prijašnjih logiranja korisnika korisno je kada želimo pronaći promjene ili neke aktivnosti, naredba za to je **last.** Za potrebe detaljnog pretraživanja, kad je u pitanju sigurnost sustava, kad trebamo pronaći npr. datoteke koje nemaju vlasnika ili imaju dopuštenja kao što je pisanje i izvršavanje. Naredba **find** laka je za shvatiti ali zahtjeva znanje njenih opcija. Za potrebe otkrivanja o kakvoj datoteci je riječ, koristi se naredba **file** čime se određuje tip datoteke po njenom sadržaju a ne imenu. Ako se dogodi situacija probijanja sigurnosti sustava, npr. ako je Trojanac ubačen u podatkovni sustav i u lokaciju koja nam se prikazuje u putanji pretraživanja, potrebno je koristiti naredbu **which** kojom se identificira izvršna datoteku koju planiramo pokrenuti kad upišemo njeno ime. To je dobar razlog za da putanja pretraživanja uključuje direktorije kao što je **/usr/bin** prije dodavanja novih lokacija i trenutnog direktorija (.).

Sigurnost se tiče i mrežnih konekcija kao i prometa koji se razmjenjuje obostrano putevima. Naredba **ss** je alat za pregled mrežnih priključaka te dopušta listanje priključaka i aktivnih konekcija. Bez dodavanja ograničenja, naredba ispisuje i više nego što nam je potrebno, jer većina dijelova operacijskog sustava komunicira putem priključaka. Ima mnogo kombinacija ove naredbe, za ispis liste s trenutnim konekcijama koristimo **ss -t** ili **ss -ltn.** Na samom računalu s Linuxom nema potrebe za firewall – om, ako je u pitanju organizacija te održavanje infrastrukture onda ćemo imati postavljen firewall. Važan korak u imanju takve vrste zaštite je kontrola pristupa sustavu, naredbe koje se koriste za pokretanje/stopiranje, uključi/isključi, izmjeni i prikaži status ili aktivna pravila jako su važna. Naredba koja se koristi je **ufw** (uncomplicated firewall) koji se nalazi striktno na većini Ubuntu sustava.

Zaustavljanje procesa koji je potencijalno opasan i ugrožava fluidan rad sustava također je bitna stavka održavanja sigurnosti. Postoji mnogo vrsta naredbe **kill** (pkill, xkill i killall). **kill** naredba funkcionira ako joj se pridoda ID procesa te broj 9 (opcija “sure kill“). Kernel zaustavlja proces bez vremena da se izvrši, iako ova naredba dovodi do gubitka podataka ako se ne koristi oprezno. Naredba **pkill** funkcionira na način da ako se želi izvršiti ne pridodaje se ID procesa nego ime procesa te šalje signal SIGTERM (isto kao i kill naredba). Zadnja naredba u **kill** opciji je **killall** koja ima sličnosti s **pkill** te također šalje SIGTERM. Što se tiče šifri, naredba **passwd** je jako korištena kad je u pitanju sigurnost. Imati pravu politiku osiguranja za promjenu šifra, pogotovo kod korisnika koji dolaze i odlaze ili mijenjaju svoje uloge u organizaciji. **passwd** se ne koristi samo promjenu šifri, moguće je postaviti šifre i ostalim korisnicima (**sudo**), zaključati/otključati račune, provjera statusa računa te promjena postavki koje određuju kada šifra istječe. Za potrebe provjere datoteka **/etc/passwd** i **etc/shadow** u smislu da se provjerava da li su sva polja prisutna, datoteke i direktoriji koristi se naredba **pwck.** Jako važna stavka sigurnosti je prije spomenute dozvole, odnosno dopuštenja nad pojedinim datotekama. Pomoću naredbi **setfacl** i **getfacl** moguće je dati dopuštenje nekome tko nije vlasnik datoteke i nije član grupe pristup datoteci. Naredbe **sestatus** i **apparmor** prikazuju status SELinux – a i **apparmor** alata koji pružaju izolaciju između aplikacija koristeći obveznu kontrolu pristupa.

12. Zaključak

Svi oblici i veličine poslovnih organizacija u svojim mnogobrojnim odjelima imaju IT odjel. Cijelo poslovanje s rastom i napretkom tehnologije potpuno se digitalno transformiralo te time se još više približilo simbiozi s informacijsko komunikacijskih tehnologijama. Uz samu vrstu poslovnih organizacija i njenih hijerarhija, postoji i IT infrastruktura kao kostur cijele poslovne organizacije. Kroz cijeli rad fokus je stavljen na sistem administratora i administraciju operacijskih sustava baziranih na Unix/Linux – u. Prikazane su određene prednosti korištenja takve vrste operacijskih sustava u radu, posebice Linux servera koji uz mnogobrojne prednosti najviše se vidi dobit na financijskom polju u pogledu optimizacije troškova IT infrastrukture i bolje korištenje IT osoblja kroz uporabu virtualizacije. Uz virtualizaciju, spomenuta je kontejnerska tehnologija kao pandem virtualnim računalima. Uporaba kontejnera utječe na poslovanje organizacija čime se stvara prednost na tržištu bržim plasiranjem aplikacija, smanjuje se vrijeme potrebno za rješavanje problema čime se poboljšava ukupna predvidljivost usluga koje se isporučuju.

Uporabom sustava baziranih na Unix/Linux unutar poslovnih organizacija i sistem administratorom kao glavnom osobom zaduženom za održavanje rada IT infrastrukture dokazani su benefiti. Poslovnim organizacijama bit je maksimizirati dobit uz najmanje moguće rashode. Povezanost financijskih metoda s ovakvom vrstom operacijskog sustava je iznenađujuća. Uz metode kao ROI, NPV i ostale metode u kombinaciji s pravim alatima, metodama i standardima prikazuje se simbioza IT, IT managementa i ekonomije te je najbolji prikaz kako tehnologija daje oslonac za dugoročno poslovanje i boljitak poslovne organizacije. Ovim radom dokazano je da svaka poslovna organizacija koja nastoji dugoročno poslovati te pritom ostvarivati dobit a imati što manje rashode treba implementirati ovakvu vrstu operacijskog sustava te time dati na važnosti ulogu sistem administratora u poslovnoj organizaciji.

13. Literatura

1. Clive Longbottom i Stephen J. Bigelow, SearchDataCenter, na web stranici: https://searchdatacenter.techtarget.com/definition/infrastructure (pristupljeno: 20. travnja 2019.).
2. Keith Townsend, SearchITOperations, Modern Infrastructure, 2016, na web stranici:https://searchitoperations.techtarget.com/opinion/The-real-future-of-IT-infrastructure-lies-in-abstraction (pristupljeno: 23. travnja 2019.).
3. Kyle Rankin, Linux Journal, Sysadmin 101: Alerting, 2017, na web stranici: https://www.linuxjournal.com/content/sysadmin-101-alerting (pristupljeno 25. travnja 2019.).
4. Kyle Rankin, Linux Journal, Sysadmin 101: Automation, 2017, na web stranici: https://www.linuxjournal.com/content/sysadmin-101-automation (pristupljeno 28. travnja 2019.).
5. Kyle Rankin, Linux Journal, Sysadmin 101: Ticketing, 2017, na web stranici: https://www.linuxjournal.com/content/sysadmin-101-ticketing (pristupljeno 28. travnja 2019.).
6. Kyle Rankin, Linux Journal, Sysadmin 101: Patch Management, 2017, na web stranici:https://www.linuxjournal.com/content/sysadmin-101-patch-management (pristupljeno 29. travnja 2019.).
7. Ernest Mueller, the agile admin, What is DevOps?, 2010, na web stranici: https://theagileadmin.com/what-is-devops/ (pristupljeno 5. kolovoza 2019.).
8. Emily Mell, Alex Gillis i Chris Riley, SearchITOperations, DevOps, na web stranici:https://searchitoperations.techtarget.com/definition/DevOps (pristupljeno 7. kolovoza 2019.).
9. Margaret Rouse, SearchITOperations, real-time analytics, na web stranici: https://searchcustomerexperience.techtarget.com/definition/real-time-analytics (pristupljeno 8. kolovoza 2019.).
10. Valentine Silverthone, SearchSoftwareQuality, BizDevOps (Business, Development and Operations), na web stranici: https://searchsoftwarequality.techtarget.com/definition/BizDevOps-Business-Development-and-Operations (pristupljeno 8. kolovoza 2019.).

11. Yvette Francino, SearchSoftwareQuality, product owner, na web stranici: https://searchsoftwarequality.techtarget.com/definition/product-owner (pristupljeno 9. kolovoza 2019.).
12. Margaret Rouse, WhatIs.com, scrum master, na web stranici: https://whatis.techtarget.com/definition/scrum-master (pristupljeno 9. kolovoza 2019.).
13. Margaret Rouse, SearchSoftwareQuality, sprint (software development), na web stranici: https://searchsoftwarequality.techtarget.com/definition/Scrum-sprint (pristupljeno 9. kolovoza 2019.).
14. Lana Pavkovic, Project management, Ultimate Guide To Scrum Project Management Framework, 2016, na web stranici: https://yanado.com/blog/ultimate-guide-to-scrum-project-management-framework/ (pristupljeno 9. kolovoza 2019.).
15. Guru99, Scrum Testing Methodology Tutorial: What is, Process, Artifacts, Sprint, na web stranici: https://www.guru99.com/scrum-testing-beginner-guide.html (pristupljeno 11. kolovoza 2019.).
16. Amazon Web Servis, What is DevOps?, na web stranici: https://aws.amazon.com/devops/what-is-devops/ (pristupljeno 12. kolovoza 2019.).
17. David Weldon, Information management, 6 reflections on the key role that system administrators play, na web stranici: https://www.information-management.com/list/6-reflections-the-key-role-that-systems-administrators-play (pristupljeno 17. kolovoza 2019.).
18. Carolyn Rowland, The Business Value of System Administration, na web stranici: http://markburgess.org/blog_busval_short.html (pristupljeno 17. kolovoza 2019.).
19. wiseGEEK, What Does a Business System Administrator Do?, na web stranici: https://www.wisegeek.com/what-does-a-business-system-administrator-do.htm# (pristupljeno 19. kolovoza 2019.).
20. JobHero, Linux Administrator Job Description, na web stranici: https://www.jobhero.com/linux-administrator-job-description/ (pristupljeno 19. kolovoza 2019.).
21. JobHero, Linux Systems Administrator Job Description, na web stranici: https://www.jobhero.com/linux-systems-administrator-job-description/ (pristupljeno 19. kolovoza 2019.).

22. Jim McIntyre, TechRepublic, User administration in Linux, 2000, na web stranici: https://www.techrepublic.com/article/user-administration-in-linux/ (pristupljeno 21. kolovoza 2019.).
23. tutorialspoint, Unix / Linux - User Administration, na web stranici: https://www.tutorialspoint.com/unix/unix-user-administration.htm (pristupljeno 21. kolovoza 2019.).
24. Markku Rossi, Global banking and finance, Financial Organizations Are in Denial About SSH Keys, 2019, na web stranici: https://www.globalbankingandfinance.com/financial-organizations-are-in-denial-about-ssh-keys/ (pristupljeno 22. kolovoza 2019.).
25. Ssh.com, PuTTY - Graphical Terminal & SSH Client for Linux, na web stranici: https://www.ssh.com/ssh/putty/linux/ (pristupljeno 22. kolovoza 2019.).
26. Steve Lehr, RingLead, The Importance of Data Management In Companies, 2019, na web stranici: https://www.ringlead.com/blog/the-importance-of-data-management-in-companies/ (pristupljeno 24. kolovoza 2019.).
27. Shehan Marasinghe, NerdyNaut, The importance of Data and Storage Management in SME, 2018, na web stranici: https://www.nerdynaut.com/the-importance-of-data-and-storage-management-in-sme (pristupljeno 24. kolovoza 2019.).
28. blue-pencil, What Is Data Management And Why It Is Important, 2015, na web stranici: https://www.blue-pencil.ca/what-is-data-management-and-why-it-is-important/ (pristupljeno 24. kolovoza 2019.).
29. Justin Ellingwood, DigitalOcean, How To Perform Basic Administration Tasks for Storage Devices in Linux, 2016, na web stranici: https://www.digitalocean.com/community/tutorials/how-to-perform-basic-administration-tasks-for-storage-devices-in-linux (pristupljeno 28. kolovoza 2019.).
30. Vangie Beal, webopedia, storage management, na web stranici: https://www.webopedia.com/TERM/S/storage_management.html (pristupljeno 28. kolovoza 2019.).
31. Storage Management, na web stranici: http://download.101com.com/GIG/Custom/2010PDFS/StorageMgt/Storage_Management2010.pdf (pristupljeno 28. kolovoza 2019.).

32. Mark Eveleens, opslogix, Why monitor your IT infrastructure?, 2014, na web stranici: https://www.opslogix.com/monitor-infrastructure/ (pristupljeno 31. kolovoza 2019.).
33. Sony Shetty, Gartner, How to Start an IT Monitoring Initiative, 2017, na web stranici: https://www.gartner.com/smarterwithgartner/how-to-start-an-it-monitoring-initiative/ (pristupljeno 31. kolovoza 2019.).
34. Ravi Saive, TecMint, 20 Command Line Tools to Monitor Linux Performance, 2014, na web stranici: https://www.tecmint.com/command-line-tools-to-monitor-linux-performance/ (pristupljeno 1. rujna 2019.).
35. techopedia, Automation, na web stranici: https://www.techopedia.com/definition/32099/automation (pristupljeno 1. rujna 2019.).
36. Red Hat, What's business automation?, na web stranici: https://www.redhat.com/en/topics/automation/whats-business-automation (pristupljeno 1. rujna 2019.).
37. Red Hat, Automation, na web stranici: https://www.redhat.com/en/topics/automation (pristupljeno 1. rujna 2019.).
38. Red Hat, What's IT automation?, na web stranici: https://www.redhat.com/en/topics/automation/whats-it-automation (pristupljeno 1. rujna 2019.).
39. Roderick Bauer, BackBlaze, What’s the Diff: VMs vs Containers, 2018, na web stranici: https://www.backblaze.com/blog/vm-vs-containers/ (pristupljeno 2. rujna 2019.).
40. Ross Beard, ShadowSoft, Why Application Containerization? The Business Value of Containers, 2018, na web stranici: https://shadow-soft.com/application-containerization/ (pristupljeno 2. rujna 2019.).
41. Docker, Understanding the Business Value of Docker Enterprise Edition, 2017, na web stranici: https://goto.docker.com/rs/929-FJL-178/images/WP_BusinessValueofDocker_06.26.2017.pdf (pristupljeno 4. rujna 2019.).
42. Ankontini, “What is the role of cyber security in an organization?” na web stranici: https://ankontini.com/what-is-the-role-of-cyber-security-in-an-organization/ (pristupljeno 4. rujna 2019.).

43. Uni Assignment, na web stranici: https://www.uniassignment.com/essay-samples/information-technology/importance-of-information-security-in-organizations-information-technology-essay.php (pristupljeno 4. rujna 2019.).
44. Derek A. Smith, BeyondTrust, A Modern Take on Best Practices for Unix and Linux Security, 2018, na web stranici: https://www.beyondtrust.com/blog/entry/modern-take-best-practices-unix-linux-security (pristupljeno 5. rujna 2019.).
45. Sandra Henry – Stocker, NetworkWorld, 22 essential Linux security commands, 2018, na web stranici: https://www.networkworld.com/article/3272286/22-essential-security-commands-for-linux.html?nsdr=true (pristupljeno 6. rujna 2019.).
46. Robert Diaz, Jr, Reduce Costs and Improve Business Performance with Linux, 2008, na web stranici: https://http-download.intuit.com/http.intuit/CMO/qbes/resources/pdfs/QuickBooks_Enterprise_Solutions_Linux_Advantage.pdf ((pristupljeno 8. rujna 2019.).

Sažetak

Diplomski rad iz kolegija “Informacijski management“ bavi se tematikom uporabe i sistemske administracije Unix/Linux operacijskih sustava s naglaskom na implementaciju u poslovne organizacije te financijskim benefitima. Rad započinje uvodom u samu definiciju IT infrastrukture kao žarišne točke poslovne organizacije. Potom prelazi u polje sistem administracije i njenih temelja, DevOps kao pozicije u koju sistem administratori sve više zalaze. Slijedi pregled uporabe u poslovnim organizacijama kroz određena polja (administracija, virtualizacija, uporaba kontejnera, kibernetička sigurnost) u kojima se pojedinačno mogu vidjeti financijski dobitci.

The thesis "Information Management" focuses on the use and system administration of Unix/Linux operating systems with an emphasis on implementation in business organizations and financial benefits. The work begins with an introduction to the definition of IT infrastructure as a focal point of the business organization. It then moves into the field of the administration system and its foundations, DevOps as a position in which the system administrators increasingly go. The following is an overview of the use in business organizations through certain fields (administration, virtualization, use of containers, cybersecurity) where financial gains can be seen individually.

Ključne riječi: Informacijski management, poslovna organizacija, sistem administrator, Linux, UNIX/Linux, operacijski sustav, sistemska administracija

Keywords: Information management, business organization, system administrator, Linux, UNIX/Linux, operating system, system administration