Softverski alat za analizu izvornih podataka o komunikacionim aktivnostima u računarskim mrežama

Lazar Smiljković, Marko Mišić, *Member, IEEE*, Pavle Vuletić, Slavko Gajin

***Apstrakt*—U ovom radu opisan je razvoj alata za detaljnu analizu i vizuelizaciju velikih skupova podataka koji opisuju komunikaciju u računarskoj mreži. Implementirani su tabelarni i grafički prikaz mrežnih tokova. Prikazi su upotpunjeni mogućnostima za izbor složenih kombinacija parametara kojima se podaci filtriraju, sortiraju i grupišu. Alat omogućava korisniku detaljan uvid u strukturu saobraćaja u proizvoljnim vremenskim intervalima kroz različite metrike radi uočavanja određenih pravilnosti, a čiji nedostatak može da ukaže na potencijalne rizike, kao i napade na mrežu. Zatim, omogućava da se pomoću posebno dizajniranog korisničkog interfejsa korisnik fokusira na određene delove i tipove saobraćaja kako bi te nepravilnosti dodatno i lakše ispitao. Takve detaljne analize pojedinačnih i grupisanih mrežnih tokova su od velikog značaja u forenzici upada u računarske sisteme i mreže.**

***Ključne reči*—mrežni saobraćaj; softverski alat; netflow; vizuelizacija.**

I. Uvod

Stalnim napretkom IT tehnologija, računarske mreže postaju sve ranjivije ako se ne tretiraju na odgovarajući način. Pošto mreža utiče na mnoge aspekte digitalnog poslovanja preduzeće ili funkcionisanja institucija, ovo bi trebalo da bude jedan od glavnih prioriteta. Da bi se postiglo ispravno funkcionisanje mreže nije dovoljno da se ona samo dobro dizajnira i konfiguriše, već da se sprovodi i tekuće održavanje mreže, uključujući praćenje, optimizaciju i nadogradnju.

Jasno je da je nadgledanje mreže veoma važan aspekt održavanja [1][2]. To podrazumeva proces u kome se nadgledaju mrežne komponente kao što su serveri, *firewall*, ruteri, svičevi, itd. Softverski alat koji se predstavlja u ovom radu predstavlja deo aplikacije za monitoring mreže zasnovan na *NetFlow* protokolu [3]. *NetfFow* je protokol koji je razvio *Cisco* i služi za evidenciju informacija o pojedinačnim komunikacijama u mreži. Podaci se u realnom vremenu prikupljaju na ruterima i eksportuju na odvojen server, gde se korišćenjem *NetFlow Analyzer* softvera podaci naknadno obrađuju, analiziraju, skladište i u obliku sumarnih statistika po potrebi prikazuju u izabranim vremenskim intervalima [4].

Ideja za realizaciju modula proistekla je iz potrebe da se osim sumarnih statističkih podataka, korisnicima omogući prikaz i analiza originalnih „sirovih“ neobrađenih podatka, radi detaljnijeg uvida u način korišćenja mreže. Pored jednostavnog iščitavanja podataka, modul treba da pruži svojim korisnicima slobodu u prikazu, grupisanju i analizi podataka po sopstvenom nahođenju. Ovo podrazumeva mnogobrojne opcije za sofisticirana filtriranja, sortiranje podataka i raznovrsne načine agregacije podataka. Modul treba da, pored ispisivanja podataka, pruži i raznovrsne načine vizuelizacije na graficima čime će se omogućiti ispitivanje trendova i neregularnosti u saobraćaju, lokalizovanjem anomalije i rasparčavanjem komunikacionih aktivnosti [5]. U ovom radu predstavljen je originalni pristup u forenzici upada u računarske mreže i identifikacija uzroka napada. Iako postojeći radovi uglavnom proučavaju opšte karakteristike mrežnih tokova, ova analiza omogućava dublje razumevanje ponašanja komunikacionih aktivnosti u mreži.

U ovom radu je predstavljeno efikasno rešenje za analizu izvornih komunikacionih podataka u računarskim mrežama, koje je obliku dodatnog modula integrisano u okružujuću aplikaciju. U drugom poglavlju rada su opisani korisnički zahtevi i sve funkcionalnosti koje je bilo neophodno implementirati zarad zadovoljavajućeg korisničkog iskustva. Način na koji je sistem realizovan, odabir i dizajn korišćene baze podataka, arhitektura sistema, implementacioni izazovi, način na koji su oni prevaziđeni i nedostaci aktuelnog rešenja su predstavljeni u trećem poglavlju ovog rada. Četvrto poglavlje predstavlja glavne funkcionalnosti realizovanog alata, prikaznanokroz odgovarajuće primere. Peto poglavlje bavi se analizom performansi alata tokom rada sa velikim količinama podataka. Finalno poglavlje daje zaključak i smernice za dalja unapređenja sistema i buduća istraživanja.

II. Opis funkcionalnosti primenjenog rešenja

U modernim računarskim mrežama, praćenje i analiza podataka se uobičajeno vrše na sloju 2, sloju 3 i sloju 4 OSI modela. U nastavku je opisano kakvi su podaci na raspolaganju realizovanom softverskom alatu.

Lazar Smiljković – Elektrotehnički fakultet, Univerzitet u Beogradu, Bulevar kralja Aleksandra 73, 11120 Beograd, Srbija (e-mail: lazarsmiljkovic@etf.rs), ORCID ID (https://orcid.org/0009-0004-8168-1924)

Marko Mišić – Elektrotehnički fakultet, Univerzitet u Beogradu, Bulevar kralja Aleksandra 73, 11120 Beograd, Srbija (e-mail: marko.misic@etf.bg.ac.rs), ORCID ID (https://orcid.org/0000-0002-7369-4010)

Pavle Vuletić – Elektrotehnički fakultet, Univerzitet u Beogradu, Bulevar kralja Aleksandra 73, 11120 Beograd, Srbija (e-mail: pavle.vuletic@etf.bg.ac.rs), ORCID ID (https://orcid.org/0000-0001-5600-2652)

Slavko Gajin – Elektrotehnički fakultet, Univerzitet u Beogradu, Bulevar kralja Aleksandra 73, 11120 Beograd, Srbija (e-mail: slavko.gajin@etf.bg.ac.rs), ORCID ID (https://orcid.org/0000-0002-8939-3589)

 ISBN 978-86-7466-969-3

A. Opis NetFlow izvornih podataka

NetFlow baza sa podacima se uobičajeno popunjava dnevno, dodajući po jedan indeks, što znači da je sačinjena od dnevnih indeksa u kojima se čuvaju podaci dobijeni o komunikacionim aktivnostima u mreži toga dana. Indeks predstavlja jedinicu organizacije podataka – sličan je bazi podataka u relacionim sistemima i sastoji se od skupa dokumenata koji se mogu smatrati jedinicama informacija. Na Sl. 1 je prikazano mapiranje za jedan dokument dnevnog indeksa za datum 29.08.2022.

Originalni *NetFlow* podaci se odnose u komunikaciji u jednom smeru (unidirekciono), identifikovani preko izvorišne i odredišne IP adrese, primenjenog protokola i izvorišnog i odredišnog porta, a za koje se evidentira ukupan broja prenetih paketa i bajtova. Kroz posebnu obradu podaci iz oba smera se uparuju i kao bidirekcioni zapisi se upisuju u bazu, što znači da nose informacije o oba smera komunikacije. Zato većina polja u nazivu sadrži jedan od sledeća dva prefiksa: *init,* što označava smer od inicijatora u komunikaciji između dva uređaja, ili *resp,* što označava smer od uređaja koji odgovara inicijalni upit (responder).

TABELA I
POLJA INDEKSA U BAZI

startTime	Početno vreme komunikacije
endTime	Završno vreme
duration	Trajanje komunikacije
initIP	IP adresa inicijatora
respIP	IP adresa respondera
initPort	Port inicijatora
respPort	Port respondera
initNextHop *respNextHop*	Sledeći ruter u nizu za prolaz paketa
initAS, respAS	Autonomni sistem
initFlags, respFlags	Indikatori za opis
initDSCP *respDSCP*	8-bitne vrednosti za kvalifikaciju saobraćaja
exporter	IP adresa uređaja preko kog je tekla komunikacija
initInterfaceIn *respInterfaceIn*	SNPM indentifikator ulaznog interfejsa
initInterfaceOut *respInterfaceOut*	SNPM indentifikator izlaznog interfejsa
initBytes, respBytes	Broj razmenjenih bajtova
initPackets *respPackets*	Broj razmenjenih paketa
initFlows, respFlows	Broj ramenjenih zapisa spojenih u jedan
bidirectional	Označava da li su podaci bidirekcioni


```
{
  "initIP" : "82.117.196.170",  "respInterfaceIn" : "0",
  "initPort" : "6052",          "respInterfaceOut" : "9",
  "respIP" : "8.8.8.8",         "initAS" : "0",
  "respPort" : "53",            "respAS" : "0",
  "protocol" : "17",            "duration" : "20.0",
  "initPackets" : "1",          "startTime" : "2022-08-29T16:24:42.777 CEST",
  "respPackets" : "1",          "endTime" : "2022-08-29T16:24:42.797 CEST",
  "initBytes" : "86",           "initDSCP" : "192",
  "respBytes" : "143",          "respDSCP" : "0",
  "initFlows" : "1",            "initAS" : "0",
  "respFlows" : "1"             "respAS" : "0",
  "exporter" : "172.16.0.3",    "initFlags" : "0",
  "initInterfaceIn" : "9",      "respFlags" : "0",
  "initInterfaceOut" : "0",     "bidirectional" : true  }
```


Sl. 1. Primer dokumenta u dnevnom indeksu

B. Vizuelizacija podataka

Podatke je potrebno prikazati tabelarno i grafički. U tabeli se prikazuju podaci koji ispunjavaju naznačene uslove tako da jedan red tabele prikazuje jedan zapis (bidirekcioni komunikacioni tok), a kolone predstavljaju polja iz dokumenta. Zaglavlje tabele pruža mogućnosti za odabir filtriranja, sortiranja i grupisanja podataka.

Grafički prikaz vizuelizuje količinu podataka (vertikalna osa) u vremenu (horizontalna osa) za odabrane parametre. Radi boljeg tumačenja podataka na graficima je omogućeno zumiranje na manji vremenski interval. U zavisnosti od vrsta saobraćaja na grafiku se ocrtavaju i pozitivna i negativna osa (vertikalno) tako da su podaci od izvornog uređaja u komunikaciji na pozitivnoj osi, a od uređaja koji odgovara na negativnoj. Iskorišćene su dve vrste grafika, linijski za prikaz jednog parametra u vremenu (najčešće kod ukupnog saobraćaja) i grafik naslaganih površina kada se prikazuju grupisani i sumirani podaci na osnovu nekog od polja za deset najvećih dobijenih vrednosti. U drugom slučaju uz grafik se prikazuje i legenda za lakše razumevanje prikazanih podataka sa dodatnim podatkom *Others* koji predstavlja ostale izostavljene vrednosti.

C. Operacije nad podacima

Nad podacima je omogućeno vršenje operacija sortiranja, filtriranja i grupisanja. Sortiranje se može vršiti po jednom polju (podrazumevano *endTime*) u oba smera i ima uticaja samo na tabelarni prikaz.

Filtriranje se podrazumevano izvršava na osnovu odabranog vremenskog intervala, a moguće je dodatno filtrirati podatke i po proizvoljnom broju ostalih polja na osnovu jednakosti, različitosti i odnosu manje/veće, a kod IP polja i pripadnosti zadatoj mreži.

D. Filtriranje na osnovu spoljnih elemenata

Podaci se prikazuju u kontekstu elementa izabranog u okružujućoj aplikaciji, tako što se automatski primenjuje implicitni filter. Tako je moguće filtrirati na osnovu eksportera ili određenog interfejsa na nekom od eksportera, korisnika čije se komunikacije analiziraju, definisane složene šablone saobraćaja, kao i na podmreže i skupove podmreža u ovim šablonima.

E. Korisnički interfejs

Alat pruža mogućnost za jednostavan izbor vrsta i mera saobraćaja koji se želi prikazati. Tako je moguće odabrati meru za prikaz – protočnost i volumen, jedinicu za prikaz nad kojom će se vršiti sumiranje pri grupisanju – bajtovi, paketi ili zapisi i

 ISBN 978-86-7466-969-3

u slučajevima podataka za interfejse moguće je izabrati izvorni ili destinacioni saobraćaj.

F. Agregacija podataka

Podaci mogu da se grupišu po odabranom polju (agregiraju), a zatim se nad tim grupama izvrši određena agregaciona funkcija (najčešće *sum*). Rezultat agregacije u tabeli se prikazuje tako da svaki red odgovara jednoj grupi, dok se nad ostalim poljima primenjuje odgovarajuća agregaciona funkcija: kvantitativni podaci - bajtovi, paketi i trajenje se sumiraju, a za početno i krajnje vreme se računa minimalno i maksimalno vreme, respektivno.

Na grafičkom prikazu se izvršava posebna vrsta agregacije i kada nije odabrano polje za grupisanje. Koristi se agregaciona funkcija koja podatke deli po vremenskim intervalima i sumira ih unutar intervala kako bi se te vrednosti prikazale kao tačke na grafiku. Dodatno, kada je naznačeno grupisanje, podaci se dele u grupe na osnovu istog ključa, a zatim dodatno u svakoj grupi dele na vremenske intervale i sumiraju, kako bi se na kraju na grafiku prikazali podaci za ključeve sa 10 najvećih vrednosti suma. Jednačina (1) prikazuje način računanja vrednosti u slučaju kada se agregira po polju *IP* adrese inicijatora

$$sum(initFlow[initIP]) \tag{1}$$

gde *Flow* predstavlja odabranu jedinicu nad kojom se vrši sumiranje, a *IP* odabrano polje po kome se podaci grupišu i prikaz se vrši na pozitivnoj osi grafika, dok se na negativnoj osi prikazuje suma *respField* polja. U formuli se sumira po odabranoj jedinici, ali uz to da su podaci grupisani tako da jednu grupu predstavlja jedna ista vrednost polja koje je izabrano kao ključ. Formula se koristi na svakom podintervalu grafika koji će biti predstavljen jednom tačkom. Varijacije u formuli postoje u odnosu na vrstu polja i informacije da li polje predstavlja jednu stranu u komunikaciji ili celokupnu komunikaciju i na osnovu spoljnih filtera koji su implicitno primenjeni.

G. Bidirekciono grupisanje

Za bolju analizu događaja u mreži, korisno alat nudi i mogućnost grupisanja vrednosti i u *init* i u *resp* polju istovremeno. To pruža mogućnost da se bolje sagleda sav saobraćaj u kome je jedan uređaj učestvovao, na kojoj god strani komunikacije da je bio. Jednačina (2) predstavlja formulu za sarčunavanje bidirekcionalnog saobraćaja kada je eksterno primenjen filter za interfejs:

$$\begin{aligned} &sum(initFlows[initIP]\ gde\ važi\ int = \\ &initInterfaceOut) \ + \\ &sum(respFlows[respIP]\ gde\ važi\ int = \\ &respInterfaceOut) \end{aligned} \tag{2}$$

za izlazni izvorni saobraćaj na pozitivnoj osi gde *Flows* predstavlja odabranu jedinicu čija suma se predstavlja na grafiku. *IP* predstavlja polje nad kojim se vrši grupisanje, dok je *int* identifikator interfejsa za koji se podaci prikazuju. Sumiraju se vrednosti odabrane jedinice, a grupisanje se vrši po polju, kako je prethodno opisano. Kako je implicitno primenjen filter za interfejse potrebni su i dodatni uslovi pri pretrazi, a pre samog grupisanja. Interfejsi sadrže sufiks *Out* jer se sumira izlazni saobraćaj. Bidirekcionalno grupisanje podrazumeva da se podaci grupišu po ključu koji sada predstavlja polje u obe strane komunikacije (prefiksi *init* i *resp*), a da bi se sračunao izvorni saobraćaj prefiksi se poklapaju kod ključa i jedinice. U slučaju destinacionog saobraćaja kod izvornih (*init*) ključeva, jedinice koje se sumiraju su destinacione (*resp*), a kod destinacionih ključeva jedinice su izvorne jer iz perspektive *resp* uređaja u komunikaciji destinacija je uređaj na suprotnoj strani tj. *init* uređaj.

III. DETALJI IMPLEMENTACIJE

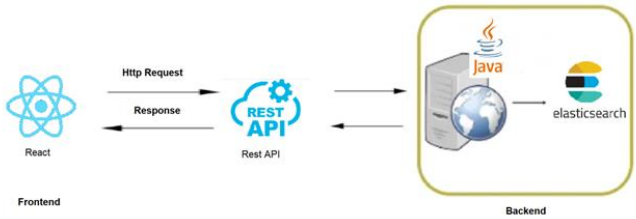
U ovom poglavlju su dati najvažniji detalji implementacije realizovanog alata. Za izradu softverskog modula su korišćene sledeće tehnologije: *HTML, CSS, Javascript, React, chart.js, react-table, Java, GWT, Kibana, PostgreSQL, Java High Level REST Client, JsInterop*, kao i nerelaciona baza podataka *ElasticSearch*.

A. Baza podataka

Odabrana nerelaciona baza podataka *ElasticSearch* ispunila je zahteve za brzinom i lakoćom pisanja složenih upita potrebnih za pretragu velike količine izvornih podataka [6] . To je distribuirana baza prilagođena za brzu i efikasnu pretragu i analizu. Osim toga, pruža mogućnost grupisanja informacija kako bi se otkrili trendovi i obrasci u pretraživanim podacima. *ElasticSearch* je veoma brz i omogućava pretragu u skoro realnom vremenu [7]. Distribuiran je i može da se proširi na stotine servera i da rukuje petabajtima podataka. Ovim karakteristikama su ispunjeni svi potrebni zahtevi za izradu softverskog alata.

B. Arhitektura sistema

Alat je razvijen uz pomoć *GWT* frejmvorka i *Java*-e na serverskoj strani i *React* frejmvorka na klijentskoj. Na klijentskoj strani nalazi se korisnički interfejs na osnovu kog se formiraju i šalju složeni *HTTP* zahtevi na serversku stranu gde se u jeziku *Java* formiraju upiti ka *ElasticSearch* bazi, dohvataju se podaci, zatim formatiraju i na kraju šalju nazad klijentskoj strani u vidu *HTTP* odgovara kako bi se prikazali korisniku.



Sl. 2. Arhitektura implementiranog sistema

Radi očuvanja uniformnog stanja u alatu neophodno je uvesti dodatnu komunikaciju između klijenta i servera i za to je upotrebljena *JsInterop* odlika *GWT*-a. Na ovaj način se postiže laka komunikacija između *Java* i *Javascript*-a, jednostavnim otkrivanjem metoda i polja anotacijama.

U sistemu se primenjuje objektni uzorak ponašanja Stanje, koji omogućava dinamičko menjanje grafičkog interfejsa i tabela i grafik se prilagođavaju stanjima koja se razmenjuju između serverske i klijentske strane. Objekat stanja sadrži različite informacije o korisniku, odabranom elementu,

 ISBN 978-86-7466-969-3

trenutnom kontekstu i odabranom vremenskom prozoru, a ove informacije se čuvaju u biblioteci *React Redux*, koja pruža optimizaciju i definiše *API*-je za interakciju komponenti sa *Redux Store*-om kako bi se automatski ažurirale kada se podaci promene.

Tokom implementacije korišćeni su i *React Hooks* koje omogućavaju dodavanje stanja, životnog ciklusa i drugih funkcionalnosti funkcionalnim komponentama. Najčešće korišćen hook je *useState*, koji omogućava čuvanje i promenu stanja komponente. Drugi često korišćen hook je *useEffect*, koji se koristi za manipulaciju životnog ciklusa komponente i reagovanje na promene u njoj. Upotreba *useEffect*-a je često vezana za definisanje liste zavisnosti, tj. Stanja koja utiču na promenu u komponenti.

Ostali primeri korišćenih *hook*-ova u modulu uključuju *useMemo*, koji se koristi za čuvanje skupih podataka iz baze, *useRef* i *useTable*.

C. Komunikacija sa bazom podataka

Serverska strana se bavi formiranjem upita za pristup bazi podataka i dohvatanje onih zapisa koji su potrebni za prikaz na klijentskoj strani. *Java* biblioteka – *Java High Level REST Client*, se koristi za formiranje *ElasticSearch* upita, pružajući jednostavne *API*-je za kreiranje *QueryBuilder*-a putem poziva metoda. Na serveru se nalaze metode za dohvatanje podataka za tabelu, neagregirani grafik, agregirani grafik i bidirekcionalno grupisanje.

Filteri su deo *bool* upita, koji se sastoji od nekoliko delova: *must, should, must not* i drugih. Ovi delovi menjaju logičke operatore pri kombinovanju uslova, poput “i”, “ili” i “ne”. Za jednakost se koristi *term* upit, a za opsege vrednosti se koriste *range* upiti.

IV. DEMONSTRACIJA SOFTVERSKOG ALATA I DISKUSIJA

Na slici 3, prikazani su tabelarni podaci koji odgovaraju vremenskom intervalu 22:10 – 22:35 na dan 09.04.2023. Nad podacima nisu izvršene nikakve dodatne operacije osim implicitnog filtriranja i sortiranja po vremenu. Grafik na slici 4.a prikazuje količinu razmenjenih bajtova u mreži kroz vreme, a u tabeli se prikazuju zapisi koji odgovaraju tim komunikacijama sa svim dodatnim informacijama o njima. Može se primetiti nagli skok u količini podataka nakon 22:15 koji odudara od trenda približnih vrednosti u ostatku intervala – vrednost se učetvorostručila u veoma kratkom intervalu, a zatim naglo opala na ustaljeni nivo.

Tokom procesa nadgledanja mreže, detaljnije ispitivanje situacije kao što je pomenuta mogu biti od velikog značaja, kako za sprečavanje malicioznih napada, tako i za bolje upoznavanje same mreže. Implementirani alat omogućava korisnicima da lokalizuju delove komunikacije koji su im od interesa i da saznaju odakle ovakve anomalije u saobraćaju potiču, ocene rizik i reaguju u skladu sa tim [8].

Jedan od načina za pristup ovakvim neregularnostima u saobraćaju je smanjivanje vremenskog intervala tako da se obuhvati samo deo saobraćaja u kome se nalazi nagli skok, što je omogućeno zumiranjem grafika. Zatim biranjem polja za grupisanje i sortiranjem podataka po sumi razmenjenih paketa jer je primećen skok u broju razmenjenih paketa. Odabirom polja *initPort* za grupisanje, dobijamo spisak portova preko kojih je tekla komunikacija, sortiranih po broju razmenjenih paketa. Na slici 4.b prikazan je rezultat primene ovih operacija.

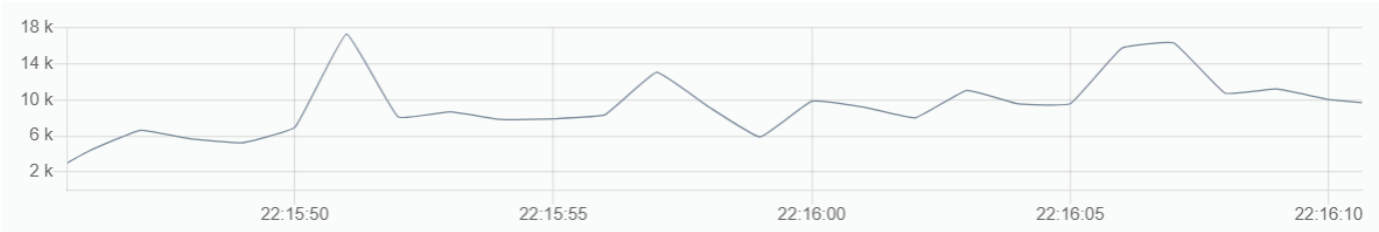
Na grafiku i u legendi prikazani su podaci za 10 najvećih vrednosti dobijenih sumiranjem paketa grupisanih po istom portu (*initPort*). U tabeli se prikazuju svi zapisi, sortirano po vrednostima sume, uz dodatne informacije o trajanju komunikacije, količini bajtova i paketa u oba smera. Iz tabele i legende se vidi da je najveći udeo komunikacije prolazio preko portova 443 i 80 koji označavaju *HTTPS i HTTP* protokole – dostupne su informacija o poljima prelaskom preko njih, dobijene razrešavanjem imena. Ovi protokoli često mogu biti meta napada kao što su skeniranja web aplikacija, DNS eksfiltracije i *DDoS* napadi, pa je potrebna dublja analiza radi provere bezbednosti. Sada kada se zna koji portovi su bili najčešće korišćeni, dodatno se može uvesti i filter sa polje porta sa vrednošću 443, kako bi se otkrile IP adrese koje su učestvovale u spornim komunikacijama, a zatim primenom filtera za nekoliko adresa u vrhu po broju paketa može se doći i do podataka ko je bio sa druge strane komunikacije. U našem slučaju skoro sva komunikacija je bila između adresa u mreži koja se posmatra i spoljnih mreža koje ne predstavljaju pretnju, što je odlučeno na osnovu podataka o njima, dobijenih putem *DNS* protokola. Ovime, ispitivanje može da se završi jer je utvrđeno da nije došlo do napada na mrežu.

Init IP	Init Port	Resp IP	Resp Port	Protocol	Duration	Init Packets	Init Bytes	Init Flows	Resp Packets	Resp Bytes
	11187		41800	6	0	1	44	1	0	0
	52524		2375	6	0	1	40	1	0	0
	52524		2375	6	0	1	40	1	0	0
	60798		28015	6	0	1	40	1	0	0
	44795		53	17	10007	2	146	1	1	73
	56635		65024	6	0	1	46	1	0	0
	48035		3389	6	0	1	40	1	0	0
	52053		5432	6	0	1	46	1	0	0
	54269		8000	6	135	4	208	1	3	1024
	44168		655	6	0	1	44	1	0	0
	45444		23	6	0	49	2110	3	0	0
	49472		22	6	3909	12	2106	1	14	2428
	38452		16888	6	0	1	44	1	0	0
	40347		28015	6	0	1	40	1	0	0

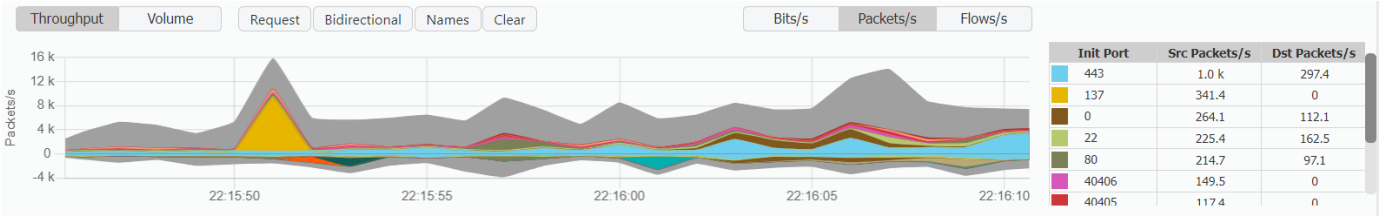
Page 1

Sl. 3. Prikaz tabele bez aktivnih dodatnih parametara

 ISBN 978-86-7466-969-3



Sl. 4.a Prikaz grafika bez aktivnih parametara



Sl. 4.b Prikaz grafika sa grupisanjem po *initPortu*

V. PERFORMANSE SISTEMA

Velike količine podataka koje opisuju komunikacione aktivnosti u mreži podrazumevaju određena ograničenja u efikanosti operacija nad podacima. Ograničenja se tiču kako prostora potrebnog za čuvanje podataka, tako i vremenske složenosti pri dohvatanju podataka. Testiranje performansi izvršeno je nad indeksom za datum 09.12.2022. koji u sebi sadrži 47.026.708 dokumenata sakupljenih u toku tog dana, a veličine je 6.6GB.

Vreme potrebno za dohvatanje podataka iz tako velikog indeksa povećava se u skladu sa povećanjem složenosti upita. *ElasticSearch* omogućava jako brzo filtriranje podataka u odnosu na ostale operacije. Zato se vreme odziva naglo povećava sa povećanjem vremenskog okvira u kome se podaci pretražuju. Upotreba agregacija dovodi do naglog skoka u trajanju izvršenja upita, a kako je neophodno da se agregacione funkcije izvršavaju nad velikim brojem polja, dohvatanje podataka i njihovo grupisanje na velikim vremenskim intervalima nije praktično. Ono što manje vremenske intervale čini poželjnijim je veoma brzo, prvobitno filtriranje velikog skupa podataka tako da se formira mnogo manji skup koji odgovara malom vremenskom intervalu. Nakon toga primenjuje se grupisanje i agregacione funkcije na manjem skupu, pa iako su skupe operacije, odziv je, naizgled, trenutan.

Dodatno, na odziv sistema utiče i doba dana u kom se pretražuje, tokom kasnih noćnih sati ima manje komunikacionih aktivnosti, pa je i vreme potrebno za dohvatanje podataka manje.

Rezultati dobijeni prilikom testiranja performansi su sledeći:

- kada nije odabrano grupisanje, vreme potrebno da se dohvate podaci za grafik i tabelu, ne prelazi preko 1s, za periode od 24h u toku istog dana. Kada taj period od 24h preklapa preko dva različita dana, vreme raste jer se tada pretraga vrši nad dva, umesto nad jednim indeksom.
- u slučaju kad je odabrano grupisanje, odziv je manji od 1s za periode od 2h u toku noći i 1h u toku dana. Za veće periode vreme raste 8s za periode od 12h i dolazi do 18s za pretragu u celom indeksu od 24h. U ovim situacijama je vreme dominatno određeno vremenom koje je potrebno za dohvatanje tabelarnih podataka u odnosu na podatke za grafik, jer je za tabelu neophodno izvršiti više agregacionih funkcija

Iako vremenska složenost raste sa veličinom vremenskog intervala, to nije ograničavajuće za rad aplikacije, zato što vrednost alata dolazi do izražaja pri analizi manjih vremenskih intervala, dok se veći vremenski intervali koriste za brz pregled i detekciju vidljivih odstupanja u trendu saobraćaja, a za to nije neophodno primenjivati grupisanja.

VI. ZAKLJUČAK

Cilj ovog rada je bio prikaz implementacije pouzdanog i upotrebljivog softverskog alata koji korisnicima omogućava širok spektar mogućnosti pri analizi izvornih podataka u računarskim mrežama. Veliki broj različitih operacija koje je moguće izvršiti nad podacima i njihovo kombinovanje je dovelo do željenog efekta, a vizuelni prikaz je dodatno obogatio mogućnost analize.

Odabrane tehnologije su se pokazale kao dobar izbor. Velike brzine i distribuiranost *ElasticSearch* mašine dovele su do prihvatljive brzine dohvatanja podataka, čak i pri najkomplikovanijim upitima sa ogromnom količinom podataka, reda veličine terabajta.

Postoji nekoliko smerova za buduća istraživanja i dalja unapređenja implementiranog softverskog alata. Ta unapređenja uključuju integraciju alata sa alatom specijalno dizajniranim za detekciju anomalija i jednostavnim prelaskom iz jednog u drugi. Dodatno, moguće je obezbediti da se pritiskom u legendi izdvoji samo jedan ključ na grafiku radi boljeg pregleda i mogućnost vraćanja na prethodni pogled.

ZAHVALNICA

Ovaj rad je delimično finansiran od strane Ministarstva nauke, tehnološkog razvoja i inovacija Republike Srbije, broj ugovora: 451-03-47/2023-01/200103.

LITERATURA

[1] Svoboda, Jakub, Ibrahim Ghafir, and Vaclav Prenosil. "Network monitoring approaches: An overview." Int J Adv Comput Netw Secur 5.2 (2015): 88-93.

[2] Lee, Sihyung, Kyriaki Levanti, and Hyong S. Kim. "Network monitoring: Present and future." Computer Networks 65 (2014): 84-98.

[3] Islam, Rafia, Vishnu Vardhan Patamsetti, Aparna Gadhi, Ragha Madhavi Gondu, Chinna Manikanta Bandaru, Sai Chaitanya Kesani and Olatunde Abiona. "Design and Analysis of a Network Traffic Analysis Tool: NetFlow Analyzer." International Journal of Communications, Network and System Sciences 16.2 (2023): 21-29.

[4] Ivanović, Ivan, and Slavko Gajin. "Recommendations for network traffic analysis using the NetFlow protocol." (2016).

[5] Chovanec, Martin, Martin Hasin, Pavol Tkač, and Eva Chovancova, "Two ways to analyze data from the netflow protocol with a view to non-relational databases" 2023 IEEE 21st World Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl'ani, Slovakia, 19-21 January 2023

[6] Zamfir, Vlad-Andrei, Mihai Carabas, Costin Carabas, and Nicolae Tapus, "Systems Monitoring and Big Data Analysis Using the Elasticsearch System" 22nd International Conference on Control Systems and Computer Science, Bucharest, Romania, 28-30 May 2019.

[7] Shaik, Subhani, and Nallamothu Naga Malleswara Rao "A Review of Elastic Search: Performance Metrics and challenges" International Journal on Recent and Innovation Trends in Computing and Communication, 5, 11, 222 – 229

[8] Patcha, Animesh, and Jung-Min Park "An overview of anomaly detection techniques: Existing solutions and latest technological trends", Computer Networks, vol. 51, issue 12, 22 August 2007, pages 3448-3470.

ABSTRACT

This paper describes the development of a tool for in-depth analysis and visualization of large datasets that describe communication in a computer network. The tool includes both tabular and graphical representations of the data. The reports are enhanced with the ability to select complex combinations of parameters that filter, sort, and group the data. The tool enables the user to view overall traffic in different time intervals through various metrics to identify certain irregularities. Then, using a specially designed user interface, the user can focus on specific parts and types of traffic to further and more easily investigate these irregularities.

Software tool for the raw data analysis of communication activities in computer networks

Lazar Smiljković, Marko Mišić, Pavle Vuletić, Slavko Gajin

 ISBN 978-86-7466-969-3