Postupak mrežnog sigurnosnog testiranja i oporavak nakon incidenta

Žagar, Filip

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

SVEUČILIŠNI STUDIJ

POSTUPAK MREŽNOG SIGURNOSNOG TESTIRANJA I OPORAVAK NAKON INCIDENTA

Diplomski rad

Filip Žagar

Osijek, 2019.

SADRŽAJ

1. UVOD

Zbog sve većeg oslanjanja na različite oblike suvremenih tehnologija u svakidašnjem životu sve više se postavlja pitanje koliko su svi ti uređaji i aplikacije zapravo sigurni, pri čemu računalna sigurnost dobiva sve veći značaj u današnjem svijetu. Razlog za sve veću zabrinutost prouzročili su različiti oblici računalnog kriminala kao što su krađa identiteta, krađa informacija, ometanje pružanja usluga, hakerski aktivizam pa čak i dio spektra terorizma. Privatne i javne organizacije su prisiljene djelovati u smjeru prevencije takvih događaja i razrade plana akcija nakon računalnog upada. U svrhu prevencije organizacije mogu izvoditi penetracijske testove, mrežno očvršćivanje (engl. *Network hardening*), očvršćivanje računala (engl. *Host hardening*), koristiti uređaje za detekciju i/ili prevenciju računalnih upada, angažirati operacijski tim za računalnu sigurnost (engl. *Security Operation Center* – SOC). S obzirom da niti jedan sustav nije i ne može biti apsolutno siguran organizacije moraju biti spremne da uz sve moguće mjere prevencije može doći do upada u slučaju čega im može pomoći jasno razrađen plan odziva na incidente. Jedna od vremenski najzahtjevnijih zadaća SOC-a je procesiranje i analiza velikih količina podataka koji su relevantni za sigurnost sustava. Od mnogo izvora podataka koji su relevantni za sigurnost sustava najistaknutiji su sustavi za detekciju upada (engl. *Intrusion Detection System – IDS*). Zajedno s ostalim izvorima podataka tipični SOC će na dnevnoj bazi prikupiti, analizirati i spremiti na desetke tisuća sigurnosnih događaja.

Prvo poglavlje ovoga rada sadrži uvodna razmatranja. U drugom poglavlju ovoga rada razmotreni su i kategorizirani događaji u računalnoj mreži koji utječu na sigurnost računalnih sustava. Treće poglavlje opisuje kvalitetan nadzor računalne mreže i metodologije koje je potrebno slijediti za uspješan odgovor na računalne prijetnje i incidente. Četvrto poglavlje razrađuje neke metode istraživačke analize podataka s naglaskom na analizu teksta i vizualizaciju podataka u svrhu informacijske sigurnosti. Opisuje generalno poznato ponašanje napadača i anomalije koje se manifestiraju probojima u računalne mreže. U petom poglavlju obrađeni su konkretni primjeri napada kroz detekciju, analizu i preporuke prilikom odziva na napad. Posljednje poglavlje rezimira dobivene rezultate i daje zaključke na temelju istih.

2. SIGURNOSNI DOGAĐAJI

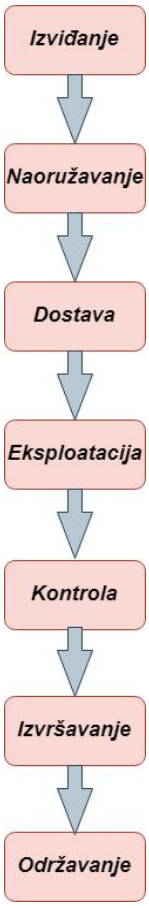
Događaj je svaka vidljiva pojava u sustavu i/ili računalnoj mreži. Događaji su ponekad indikatori da se sigurnosni incident događa npr. Alarm generiran od strane IDS-a. Događaj nije ništa više od sirovog podatka koji zahtjeva ljudsku analizu. Analiza je u ovom slučaju proces evaluacije značenja kolekcije sigurnosno relevantnih podataka, tipično pomoću posebnih alata. S obzirom da na analizu konstantno pristižu velike količine podataka potrebno je izvesti trijažu. Trijaža je proces sortiranja, kategorizacije i prioritizacije nadolazećih događaja. Tipično unutar jednog SOC-a postoje dvije grupe pri čemu događaj prvo dolazi u grupu razine 1 koja u rasponu od najviše 15 minuta procjenjuje događaj i po potrebi ga prosljeđuje grupi 2 koja zatim izvodi dublju analizu. Grupa razine 2 ili više ima odgovornost određivanja je li došlo do sigurnosnog incidenta. Sigurnosni incident je procijenjena pojava koja potencijalno ili stvarno ugrožava povjerljivost, integritet ili dostupnost informacijskog sustava ili informacija koje sustav obrađuje, sprema/prenosi, predstavlja povredu ili neposrednu prijetnju narušavanju sigurnosne politike, sigurnosnih procedura i prihvatljive politike korištenja. Jedan događaj može prouzrokovati incident ali za svaki će incident postojati tisuće događaja koji su benigni. SOC neće poduzimati protumjere na prve pokazatelje upada zbog sljedeća tri razloga:

1. SOC želi biti siguran da ne blokira benignu aktivnost
2. Akcije odziva na incident mogu utjecati više na usluge koje pruža institucija nego na sami incident
3. Razumijevanje ozbiljnosti i širine upada promatrajući napadača je ponekad učinkovitije od izvođenja statične forenzičke analize na kompromitiranim sustavima kada napadač više nije prisutan

Kada je moguće dovoljno dobro razumjeti znakove nekog napada moguće ga je kodirati u IDS potpis razumljiv računalima koji zatim mogu napade prevenirati koristeći HIPS (engl. *Host Intrusion Prevention System – HIPS*) ili NIPS (engl. *Network Intrusion Prevention System - NIPS*). Navedeni sustavi se koriste za blokiranje osnovnih napada zbog ograničenosti računala da provodi dublju analizu, stoga je svakodnevna ljudska interakcija neophodna [1].

2.1. Životni lanac kibernetičkog napada

Prema slici 2.1. vidimo tijek događaja prilikom izvršavanja kibernetičkog napada. Tijekom prve faze izviđanja napadač identificira i istražuje metu. U skladu s dobivenim informacijama napadač zatim izabire ili kreira set alata koji će mu služiti za dostavu i izvođenje malicioznog koda na žrtvinom sustavu. U trećem koraku izvodi se dostava alata za napad na metu. Nakon dostave napad se izvršava i prelazi se u iduću fazu u kojoj napadač uspostavlja kontrolu nad kompromitiranim računalima. Faza izvršavanja označava događaje u kojima napadač počinje ispunjavati ciljeve svoga napada. Kada se ispune ciljevi napada napadač ostavlja skrivena stražnja vrata (engl. *backdoors*) koja mu omogućavaju ponovni povratak prema potrebi što označava fazu održavanja [1].



Slika 2.1. Životni lanac kibernetičkog napada

Ovaj model napada nije savršen ali je dobar generalni opis kako se napadači ponašaju bez puno detalja. Postoje slučajevi koji odstupaju od ovog modela kao što su npr.:

- *Peer-to-peer* propagacija crva i *phishing* napadi oslanjaju se na pasivne maliciozne postupke i socijalni inženjering tj. oslanjaju se na metino otvaranje poveznice ili pristup datoteci što zahtjeva da mamac (npr. ime datoteke, scenarij) bude dovoljno atraktivan da privuče interakciju
- Crvi često spajaju faze izviđanja, naoružavanja, dostave i eksploatacije u jedan korak
- Napadi iznutra često preskaču prve faze životnog lanca npr. krađom lozinki drugih zaposlenika [2]

2.1.1. Životni lanac napada i detekcija

Nekoć su se SOC timovi fokusirali na idealan slučaj u kojem detektiraju incident još dok je napad u fazi istraživanja ili tijekom napada u fazi eksploatacije. Povećanjem sofisticiranosti i prikrivenosti napada SOC timovi su prisiljeni djelovati tijekom cijelog ciklusa napada. Koristeći znanje o cijelom ciklusu možemo koristiti više holistički pristup detekciji i analitici. Kao primjer toga možemo navesti *zero-day* napade koji nisu od prije poznati i stoga se tipični cilj kao što je eksfiltracija osjetljivih podataka može detektirati velikom količinom izlaznih podataka. Veliki broj napada događa se između dviju stranaka koje si međusobno vjeruju i na taj način se preskaču neki od koraka (npr. Industrijska špijunaža). Izvorno se napadača zamišlja kao tehnički potkovanog individualca koji pokušava otkriti određene slabosti sustava kako bi ukrao informacije ili datoteke. Ovo je klasičan primjer zainteresiranog napadača koji želi srušiti i kontrolirati određeni sustav s ciljem da pridobije novac, podatke, poštovanje drugih ili nešto sasvim drugo. Zainteresirani napadači su velika opasnost ali čine mali udio napadačke populacije. Najveći broj napada danas provodi se od strane nezainteresiranih napadača koji žele zauzeti što je moguće više računala. Nezainteresirani napadi su visoko automatizirani što i moraju biti jer su u velikom broju slučajeva bezuspješni. Zbog toga je i ovakve napade teško detektirati u početnih fazama životnog lanca jer npr. automatizirani crv će jednostavno lansirati napade na svako računalo u dometu bez obzira je li ono ranjivo. Nezainteresirani napadači se oslanjaju na alate i uvjerenje da će netko, negdje biti ranjiv i u većini

slučajeva neće ni biti svjesni da računalo postoji dok ne postane zaraženo. Analitičari trebaju biti svjesni ove ravnoteže između uobičajenih, „glupih“ (engl. *dumb*) i automatiziranih napada i rjeđih, inteligentnih i usmjerenih napada. Pametni napadači će se oslanjati na šum generiran od strane „glupih“ napadača a to utječe na pažnju analitičara koji može procesuirati ograničeni broj upozorenja tijekom radnog vremena [2].

Faza izviđanja može se detektirati nadzorom netipičnih konekcija na vanjske web aplikacije, „*phishing“* kampanje ili drugih usluga orijentiranih prema vanjskom dijelu mreže pomoću sigurnosnih senzora. Naoružani „exploit“ (softverski alat dizajniran kako bi iskoristio grešku u računalnom sustavu s ciljem malicioznih radnji kao što je instaliranje malicioznog programa) može ući u mrežu na bezbroj načina ali zbog velikog uspjeha najčešće putem email *phishing*-a, kompromitacijom web stranice / vjerodajnica (engl. *credentials*) tvrtke ili putem *drive-by*/*watreing hole* napada na vanjske usluge, za koje je poznato da napadnuta organizacija koristi. Direktni napadi na vanjske aplikacije mogu se nakon ispitivanja otkriti u IDS zapisima ili web zapisima vatrozida. Sustavi za detekciju upada ili antivirusni sustavi mogu alarmirati analitičare tijekom svih faza osim faze naoružavanja. Faze kontrole, izvršavanja i održavanja omogućavaju najbolje šanse za uspješnu detekciju s obzirom da podaci moraju napustiti mrežu žrtve [3].

2.2. Detekcija sigurnosnih događaja

Alati koje koristimo za detekciju sigurnosnih događaja ne reagiraju uvijek kada se nešto loše dogodi i, u pravilu, ne znači svaka reakcija nekog od alata (npr. IDS) da stvarno nešto loše dogodilo. Svaki događaj koji neki alat generira pripada u jednu od četiri kategorije:

1. Istinito pozitivan (engl. *true positive*) – događaj je opasan i sustav ga je uspješno zabilježio
2. Lažno negativan (engl. *false negative*) – događaj je opasan ali nije ni generirano nikakvo upozorenje
3. Lažno pozitivan (engl. *false positive*) – Generirano upozorenje za događaj ali događaj zapravo nije opasan
4. Istinito negativan (engl. *true negative*) – Događaj nije opasan i nije zabilježen

Najveći izazov za svaki nadzorni sustav (posebice IDS) je postići što veći broj istinito pozitivnih događaja. Kreatori ovih sustava trude se da njihovi proizvodi nikada ne propuste uspješan proboj što

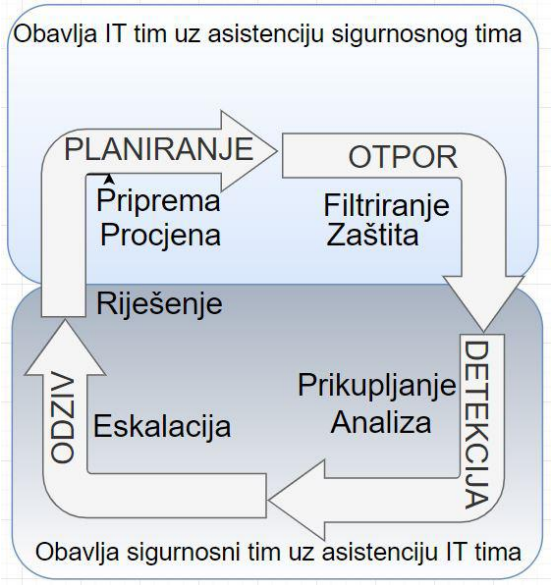
često dovodi do pojave lažno pozitivnih događaja [4]. Prema [4] čak i mali broj lažno pozitivnih događaja može sustav učiniti beskorisnim. Zbog previše lažno pozitivnih događaja analitičar gubi puno vremena proučavajući podatke ili čak ignorira događaje u potpunosti, a istinito pozitivan događaj se skriva u masi lažno pozitivnih događaja. Uzimajući u obzir velike količine podataka, koje IDS senzori mogu generirati potrebno je prema potrebi konfigurirati IDS senzore, kako bi skupljali podatke koje su relevantni za određeni sustav i na taj način smanjiti sveukupnu količinu podataka. U tome slučaju moglo bi se reći da ne postoje lažno pozitivni rezultati, već samo analitičari koji ne znaju ispravno interpretirati podatke, pri čemu je bitno razumjeti lažno pozitivan rezultat kao događaj generiran zbog potpisa čija je namjera bila da upozori na nešto opasno. Lažno pozitivan događaj nastaje zbog pogreške u preciznosti ili točnosti tj. dolazi do okidanja alarma u krivim okolnostima. Preciznost potpisa je uvijek izazov i dio rada s IDS-ovima jer većina sustava se trudi integrirati prioritet pobude događaja ali ne i njegovu povjerljivost.

3. NADZOR RAČUNALNE MREŽE

Sigurnosni nadzor mreže predstavlja prikupljanje, analizu i eskalaciju indikatora i upozorenja s ciljem odziva na upade. Analitičari trebaju alate kako bi pronašli upade ali metodologija je puno bitnija od programske podrške. Alati skupljaju i interpretiraju podatke ali metodologija omogućava konceptualni model. Bitno je znati kako koristiti alate kako bi se postigao određeni cilj, ali bitno je započeti s dobrim operacijskim modelom i onda odabrati alate koji ga podupiru [5].

3.1. Sigurnosni ciklus organizacije

Ciklus sigurnosti računalne mreže sastoji se od faza planiranja, otpora, detekcije i odziva pri čemu su sve četiri faze nužne pri zaštiti mreže od prijetnji. Prvi korak pri kreiranju modela je opis odnosa između planiranja, otpora, detekcije i odziva.



Slika 3.1. Sigurnosni ciklus

Slika 3.1. prikazuje odnose između četiri temeljne sigurnosne aktivnosti koje se u stvarnosti pojavljuju istovremeno jer su organizacije često izložene različitim upadima koji se nalaze u različitim fazama istovremeno tj. IT (engl. *Information Tehnology* – IT) i sigurnosni timovi planiraju nove načine obrane dok postojeće protumjere odbijaju neke od napada [5].

3.1.1. Faza planiranja

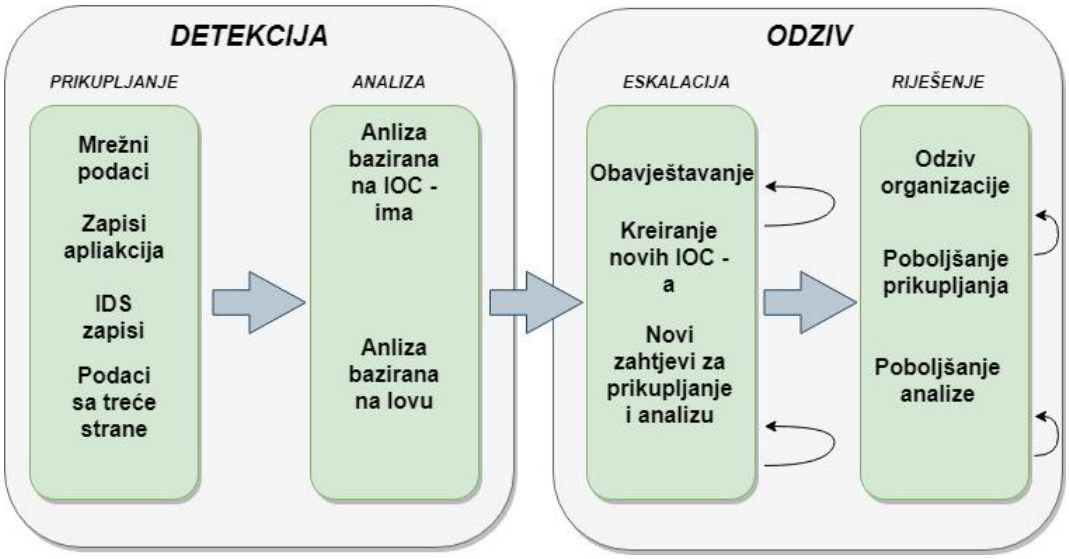
Cilj faze planiranja je da organizaciju pozicionira tako da se što je efikasnije moguće može oduprijeti upadima ili odgovoriti na ranjivosti koje napadač izrabljuje tijekom napada. U ovoj fazi IT i sigurnosni timovi pripremaju i procjenjuju situaciju, uspostavljaju obranu i procjenjuju njezinu efektivnost. Ovoj fazi pripadaju aktivnosti kao što su: budžetiranje, revizija, provjere usklađenosti, trening, sigurni razvoj programske podrške i slično. Simulacija napadača i penetracijsko testiranje su primjeri procjene mrežne sigurnosti [5].

3.1.2. Faza otpora

Tijekom faze otpora IT i sigurnosni timovi filtriraju i štite oraganizaciju. Automatizirane protumjere kao što su vatrozidi, antivirusni programi, zaštite od curenja podataka (engl. *data-leakage protection*), izrada popisa dopuštenih radnji/adresa (engl. *whitelisting*) i slične tehnologije koje za cilj imaju da zaustave napadača prije nego što dobije neovlašteni pristup dijelovima mreže. Trening osviještenosti o računalnoj sigurnosti, upravljanje ranjivostima i konfiguracija infrastrukture su protumjere dizajnirane da očvrsnu ljudsko i tehničko okruženje organizacije te također pripadaju ovoj fazi sigurnosti. Uz sve navedene protumjere predani napadači svejedno pronađu način za kompromitaciju sigurnosti organizacije, stoga su faze detekcije i odziva obvezne [5].

3.1.3. Faze detekcije i odziva

Ove dvije faze sastoje se od četiri elementa: prikupljanje, analiza, eskalacija i rješenje (engl. *resolve*). Prema Slici 3.1. element rješenja postavljen je bliže fazi planiranja s obzirom da je s njom usko povezan. Slika 3.2. grafički prikazuje odnose između ove četiri faze.



Slika 3.2. Odnosi između faza detekcije i odziva te njihovih podfaza prikupljanja, analize, eskalacije i rješenja

Prva faza se odnosi na prikupljanje podataka koji su potrebni za određivanje je li neka aktivnost normalna, sumnjiva ili maliciozna. U prikupljanje su uključeni tehnički i ne-tehnički procesi. Tehnički procesi su prikupljanje podataka sa rubnih dijelova mreže, mrežnih uređaja, IDS-ova i sl. Ne-tehnički procesi uključuju prikupljanje podataka treće strane (engl. *third parties*) kao što su poslovni partneri, policija, obavještajne agencije i sl. te od samih žrtava.

Druga faza predstavlja proces validacije onoga što mislimo o nekom događaju. Postoji dvije vrste analize, ona koja se temelji na indikatorima kompromitacije (engl. *Indicators Of Compromise* – IOC) i ona koja se ne temelji na IOC. IOC-i ubrzavaju ovaj proces s obzirom da su oni formalno manifestacije uočenih napadačevih radnji. Neformalno IOC-i su načini za kodiranje napadačevih aktivnosti kako bi tehnički sustavi mogli detektirati upade u digitalnom svijetu. Analiza koja se oslanja na IOC-e zove se i usporedbom (engl. *matching*) jer analitičari uspoređuju IOC-e s dokazima kako bi otkrili sumnjivu ili malicioznu aktivnost i tako potvrdili svoja otkrića. Osim usporedne analize moguće je provesti i napredniju analizu koja se ne temelji na IOC-ima i naziva se lovom (engl. *hunting*).

Treća faza je čin obavještavanja organizacije o statusu kompromitirane imovine i izrade dokumentacije o događajima. Poželjno je incidente dokumentirati prema nekim od standarda zajednice kao što je VERIS (engl. *Vocabulary for Event Recording and Incident Sharing*). Pri

obavještavanju potrebno je identificirati kompromitiranu imovinu, pronaći osobu ili grupu odgovornu za napad i dostaviti izvještaj o incidentu stranci zahvaćenoj napadom.

Četvrta faza uključuje akcije koje se provode od strane organizacije ili tima za sigurnost s ciljem smanjenja rizika od gubitka. Tijekom ove faze događa se prijelaz između stanja rizika u sigurno stanje. Kakav će prijelaz biti ovisi o incidentu, mogućnostima i toleranciji na rizik tima za sigurnosne incidente CSIRT (engl. *Cybersecurity Incident Response Team*). Svaka organizacija mora balansirati između rizika od gubitka podataka, izmjene, uskraćenja usluge i poslovnih zahtjeva kompromitirane imovine [5].

3.2. Izvori podataka i alati za nadzor mreže

Sigurnosni analitičar u pravilu treba tri stvari kako bi mogao obavljati kompetentni mrežni nadzor:

1. Inicijalni uvid u stanje putem IDS-a baziranog na potpisima ili ponašanju, uključujući mogućnost izrade vlastitih potpisa i uvida u detalje ponašanja koje je izazvalo okidanje
2. *NetFlow* zapise koji sadrže sažetke komunikacije u oba smjera sa računala koji se nalaze u zapisima IDS-a nekoliko dana i tjedana prije i poslije okidanja događaja
3. Snimku prometa (engl. *packet capture*) za pakete koji su okinuli događaj, po mogućnost cijele sjednice u formi *libpcap* formatiranih podataka (engl. *packet capture* - PCAP)

IDS alarmi i *NetFlow* događaj trebali bi biti indeksirani u PCAP podacima za lakšu analitiku.

3.2.1. Sustavi za detekciju upada (IDS)

Sustav za detekciju upada - IDS je hardverski ili softverski proizvod koji prikuplja i analizira informacije iz različitih područja računalne mreže, kako bi identificirao moguće sigurnosne propuste koji uključuju napada iznutra i izvana. Mrežni sustav za detekciju upada - NIDS (engl. *Network Intrusion Detection System*) prikuplja i analizira mrežni promet tražeći potencijalne upade, maliciozne aktivnosti i anomalije [6]. Mrežni promet se u stvarnom vremenu provjerava prema zadanim potpisima, definiciji prihvatljivog/normalnog ponašanja ili prema nekom drugom skupu heuristika. Tipovi detekcije mogu se generalno podijeliti u četiri kategorije: zloupotreba, anomalije, specifikacija i hibridna detekcija [7].

Detekcija prema zlouporabi uspoređuje aktivnosti kao što su mrežni promet ili akcije na razini sustava i uspoređuje ih s potpisima iz baze podataka već poznatih napada. Kada dođe do određene razine sličnosti aktivnost se označava kao sumnjiva (npr. repetitivno skeniranje otvorenih portova). Ovaj tip detekcije postiže dobre rezultate za već poznate tipove napada i zahtijeva stalno ažuriranje potpisa.

Detekcija anomalija izgrađuje model tipičnih aktivnosti koji onda omogućuje usporedbu trenutne aktivnosti s modelom u kojem postoje alarmi. Primjer aktivnosti koja može biti označena kao sumnjiva je iznenadno spajanje poslužitelja na netipičnu adresu ili uslugu. Ova metoda detekcije postiže dobre rezultate u detekciji novih napada iako često uzrokuje velik broj lažno pozitivnih događaja ako se model ne ažurira redovno.

Specifikacijska detekcija kombinira atribute prethodne dvije kategorije. Sadrži detekciju anomalija prema prethodno definiranom modelu i bazu podataka s potpisima ali sada se svaka aktivnost mora potvrditi kao maliciozna od strane čovjeka. Ova kategorija postiže rezultate visoke preciznosti ali unosi kašnjenje u kreiranju potpisa zbog ljudske interakcije [9].

Hibridna detekcija sadrži kombinaciju svega navedenog gdje se slabosti jedne metode nadopunjavaju prednostima druge [7].

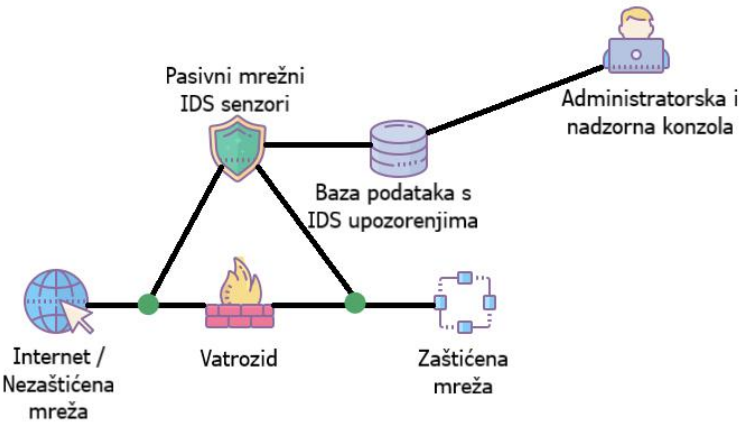
U posljednje vrijeme razvijaju se IDS rješenja bazirana na strojnom učenju s ciljem dobivanja boljih rezultata i oslobađanja čovjeka dijela posla u procesu detekcije i procesuiranja događaja. Velik broj tih rješenja usmjeren je prema sustavima visoke preciznosti, s niskom razinom lažno pozitivnih događaja što ovi sustavi zaista i postižu [9][10].

Kada se za neki događaj generira upozorenje ono treba sadržavati detalje koji su nužni analitičaru da razumije što to upozorenje znači i što treba činiti. Tipično upozorenje sa IDS-a koji se temelji na potpisima sastoji se od sljedećih značajki:

- identifikacijski broj događaja;
- vrijeme i datum okidanja;
- izvorišna i odredišna IP adresa;
- izvorišni i odredišni UDP/TCP port;
- ime i/ili ID potpisa;
- ozbiljnost događaja;
- tekstualni opis potpisa ili link na vanjski repozitorij/bazu podataka s detaljima o potpisima;

- primljeni i poslani bajtovi podataka tijekom cijele sjednice za koju je upozorenje okinuto;
- dodatne informacije koje sadrže posebna polja u nekima od protokola kao što su SNMP,SMTP, POP3, HTTP,FTP,SSL.

Upozorenja nekad sadrže i poveznicu na PCAP datoteku koja sadrži pakete koji su okinuli potpis ili cijelu sjednicu. U velikoj organizaciji tipično postoji više NIDS senzora postavljenih na ključnim točkama u mreži, kao što su periferije mreže, spojevi na Internet ili na glavnim preklopnicima i usmjerivačima. Prikaz tipične IDS arhitekture nalazi se na Slici 2.2.



Slika 2.2. Prikaz tipične mrežne arhitekture pri korištenju sigurnosnih uređaja vatrozida i IDS-a

Tipičnih IDS - ovi analiziraju promet svih protokola u mreži te često zbog ravnoteže moraju zanemariti dublju analizu specifičnog prometa. Stoga su razvijeni sustavi koji se baziraju na nekom od protokola i zapravo predstavljaju spoj IDS-a, IPS-a i vatrozida. Neki od specifičnih protokola/područja za koja se specijaliziraju su: XML (engl. *Extensible Markup Language*), SQL (engl. *Structured Query Language*) ili HTTP protokol. Njihova sposobnost otkrivanja i blokiranja napada u tom specifičnom području je bolja u usporedbi s IDS – ovima generalne namjene [1].

Postoje neki generalni nedostaci NIDS – a. Podložan je podvalama (engl. *spoof*), napadima, može propustiti dio prometa i može ga se zaobići. Često ne može odrediti uspješnost ili neuspješnost

napada i ne može analizirati kriptirani promet. Bilo kako bilo, bitno je razumjeti da IDS logovi nisu ništa više nego mnoštvo podataka relevantnih za sigurnost sustava koji svoje značenje dobivaju analizom od strane stručne osobe. Znanstveni radovi u području detekcije upada nastavljaju se fokusirati na poboljšanje brzine i efektivnosti, posebice zbog porasta brzine prijenosa podataka. Razvoj je usmjeren i prema primjeni detekcije i odziva na nove tehnologije kao što su SCADA (engl. *Supervisory Control And Data Acquisition*) i IoT (engl. *Internet of Things* – IOT) sustavi. IDS sustavi su u posljednjih nekoliko godina prošli razvoj od samostalnih i usmjerenih sustava do distribuiranih sustava i koordinirane detekcije s „Big data“ analitikom [11].

3.2.2. Zapisi toka podataka

IDS pregledava cijeli sadržaj mrežnog prometa, ali potrebna nam je mogućnost sumiranja cjelokupnog mrežnog prometa uz ulaganje malo računalne snage. Jedan od važnih komplemenata IDS upozorenjima su Netflow zapisi kod Cisco proizvoda ili Traffic flow zapisi kod Mikrotik proizvoda, često se nazivaju i tokovima (engl. *flows*). Umjesto snimanja i analiziranja cijelog sadržaja, svaki zapis toka sadrži sažetak svake mrežne konekcije. Zapis toka ne sadrži sadržaj komunikacije nego stvara metapodatke o komunikaciji kao što su početak, kraj, izvorišna i odredišna IP adresa, izvorišni i odredišni port, protokole četvrtog sloja OSI modela, količina poslanih i primljenih podataka, postavljene TCP zastavice ako se radi o TCP toku. Iako sadržaj neke mrežne konekcije može biti reda veličine gigabajta podataka, zapis toka takve konekcije je reda veličine kilobajta. Općenito zapisi toka ne sadrže komunikaciju iznad četvrtog sloja OSI modela. Takav način rada ima svoje prednosti i nedostatke. Neke od prednosti su ušteda procesorske snage, zauzimanje malo prostora za pohranu ili prijenos, nema dodira s enkripcijom dok je mana što ne zna ništa od sadržaju komunikacije. Analiza toka je primjenjiva tijekom svih faza kibernetičkog napada dok su IDS – ovi tradicionalno orijentirani prema fazi izviđanja i eksploatacije. Primjer tragova koji se mogu uvidjeti samo iz zapisa toka su promet elektroničke pošte sa web servera, jedna radna stanica prenosi znatno više podataka iz organizacije od drugih ili prijenos velike količine podataka izvan radnog vremena.

Zapisi toka mogu se generirati na usmjerivačima, preklopnicima, nekim NIDS – ovima ili programskim paketima izrađeni za tu namjenu kao što su SiLK, Argus, S/GUIL [1].

3.2.3. Snimka prometa

Kada dođe do ozbiljnog incidenta koji zahtjeva aktivni odziv ili pravne akcije trebamo konkretan dokaz što se dogodilo. To možemo saznati izvlačenjem podataka iz zaraženog računala, pri čemu je potpuno snimljeni promet često presudan. Potpuno snimljeni promet može riješiti probleme kao što su sadržaj sumnjivog prometa, potpuni sadržaj (engl. *payload*) malicioznog programa ili koji podaci su razmijenjeni nakon infekcije. Vrlo je bitno efektivno skaliranje snimke prometa pa se promet neprekidno snima na glavnim perifernim točkama u mreži gdje su i IDS – ovi ili specifično prema potrebi na potencijalno zaraženom uređaju. Promet je moguće snimiti alatima kao što su tcpdump, netsniff-ng ili Wireshark. Najveći izazov prilikom snimanja prometa je količina podataka koja može uzrokovati gubitak paketa i velike zahtjeve za pohranom.

3.2.4. Upravljanje sigurnosnim informacijama i događajima

SIEM (engl. *Security Information and Event Management*) alati prikupljaju, agregiraju, filtriraju, spremaju, trijažiraju, koreliraju i prikazuju sigurnosne podatke, u stvarnom vremenu ili u svrhu kasnije analize. Svrha ovih alata je da velike količine podataka pretvore u značajnu informaciju i na taj način omogućavaju analitičaru da te informacije pretvori u znanje prema kojem se može djelovati. Kvalitetni alati štede vrijeme analitičaru ali su skupi i u nekim slučajevima nepotrebni za manje jedinice nadzora stoga se često koriste IDS u kombinaciji s sustavima za rukovanje zapisima LMS (engl. *Log Management System*). LMS - ovi prikupljaju, spremaju, obavještavaju i provode jednostavnu analitiku dok ne inkorporiraju bogata sučelja i znatnu korelaciju događaja. Usporedba ELK stog LMS-a sa komercijalnim SIEM sustavom Splunk iz [12] pokazala je prednosti LMS-a pri implementaciji na organizacije male i srednje veličine zbog cijene, performansi pretraživanja i mogućnosti vizualizacije. U ovome radu primijeniti će se upotreba LMS baziranog na ELK stogu koji se sastoji od aplikacija Elasticsearch, Logstash i Kibana. Elasticsearch omogućava gotovo stvarnovremensko pretraživanje i analizu svih vrsta podataka. Logstash omogućava prikupljanje, agregaciju i obogaćivanje podataka te njihovo spremanje u Elasticsearch. Kibana omogućava interaktivno pretraživanje, vizualizaciju i dijeljenje uvida u podatke kao i upravljanje i nadzor nad stogom.

4. ANALIZA I VIZUALIZACIJA PODATAKA

Efektivni sigurnosni alarmi su korisni samo kada su predstavljeni pomoću efikasne, precizne i po mogućnosti automatizirane analize podataka. Ispravan razvoj metoda i praksi odziva na incident zahtjeva dobar plan i temeljni okvir (engl. *framework*) za svaki CIRT. Pronalazak sigurnosnih incidenata i korisnih tragova je vrlo težak, stoga bez plana i okvira CIRT se može lako izgubiti u velikoj količini podataka ili ostati bez korisnih podataka za analizu. Jednostavna kupnja i postavljanje SIEM – a ili LMS - a s ciljem promatranja podatka koju oni sami pružaju će u početku dobro funkcionirati. Tijekom vremena CIRT će naići na sofisticiranije prijetnje i bez kontekstualne interpretacije podataka i samostalnog pretraživanja neće moći izvršiti kvalitetan odziv na incident. Vizualizacija podataka je bitan element analize koji daje jednostavan i brz uvid u anomalije koje se događaju.

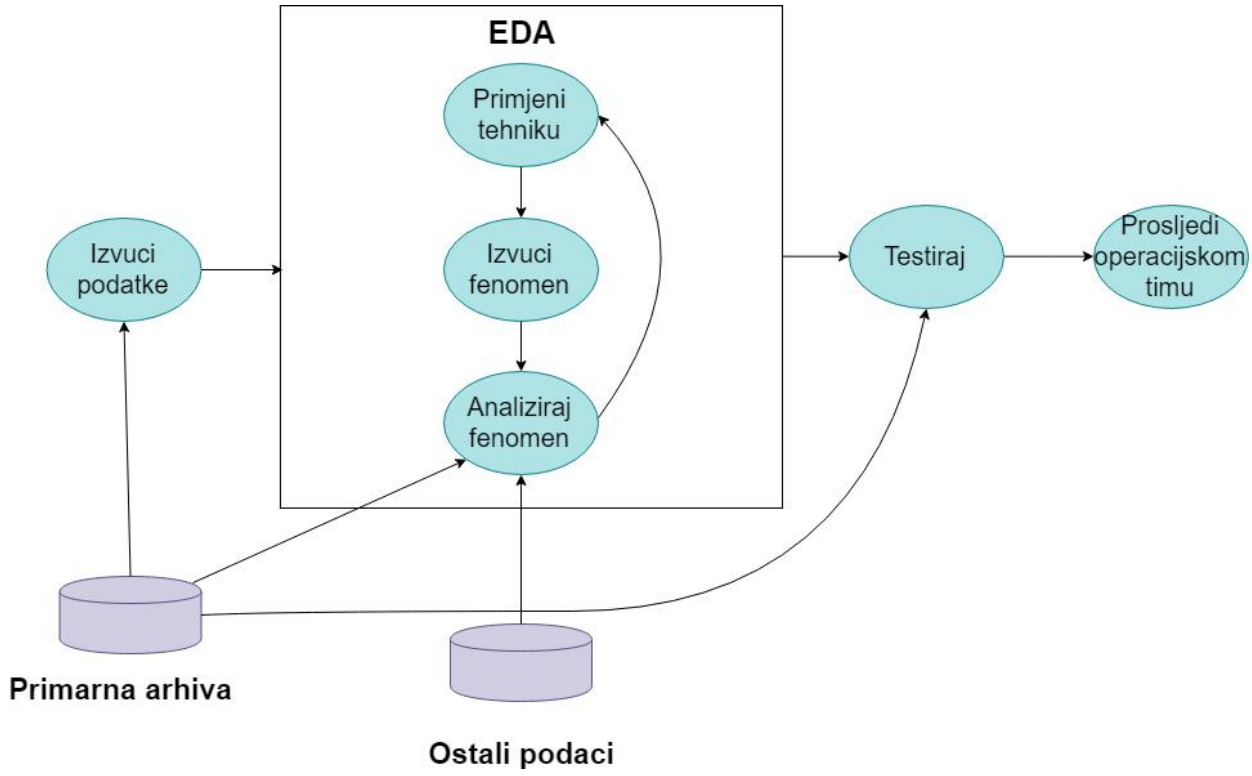
4.1. Istraživačka analiza podataka

Istraživačka analiza podataka EDA (engl. *Exploratory Data Analysis*) je proces ispitivanja skupa podataka bez prethodnih pretpostavki o podacima i njihovom ponašanju. Podaci iz stvarnog svijeta su zbrkani i kompleksni te zahtijevaju progresivno filtriranje i stratifikaciju kako bi se identificirali fenomeni korisni za upozorenja, detekciju anomalija i forenziku. Napadači i Internet sam po sebi su pokretne mete i analitičari se suočavaju s konstantnim priljevom čudnih informacija što čini analizu konstantnim procesom. Cilj analize je dobiti uvid u podatke prije nego se primjeni klasična statistička analiza jer u tom slučaju često dolazi do grešaka. Na primjer, analitičari često izračunaju srednju vrijednost i standardnu devijaciju skupa podataka s ciljem izračuna nekakvog praga (obično 3.5 standardnih devijacija od srednje vrijednosti). Ovaj prag je određen uz pretpostavku da je skup podataka normalno distribuiran, ali ako to nije slučaj jednostavno prebrojavanje će dati puno kvalitetnije rezultate. Poanta procesa analize je stvaranje modela koji može biti formalna reprezentacija podataka ili može biti jednostavno paljenje alarma kada se vidi „previše stvari“ pri čemu je „previše stvari“ jasno kvantificirano. EDA se u informacijskoj sigurnosti provodi zbog četiri cilja: konstrukcija alarma, forenzika, konstrukcija obrane i situacijska svijest. Kada se koristi kao alarm, analitički proces uključuje generiranje nekakvog broja i usporedbe s normalnom

aktivnošću te određivanja treba li promatrana aktivnost privući pozornost analitičara. Anomalija nije nužno napad, a napad ne zahtjeva nužno reakciju. Dobar alarm će se temeljiti na fenomenima koji se mogu predvidjeti u normalnim uvjetima a napadač mora utjecati na njih da bi došao do svog cilja. Problem u operativnoj informacijskoj sigurnosti nije kreiranje alarma nego rukovanje istima. Prvu stvar koju analitičar treba učiniti, kada dobije alarm je dati objašnjenje npr. je li opasnost stvarna, je li relevantna, odrediti razine štete i preporučiti akcije koje slijede. Većina sigurnosne analize je forenzička analiza koja se odvija nakon događaja i može započeti kao odziv na informacije od bilo gdje: IDS signala, alarma, korisničkih prijava ili novinskih članaka. Forenzička analiza polazi od nekih početnih podataka, kao što su IP adrese inficiranih računala ili zaražena web stranica. Počevši od toga istražitelj mora otkriti što je više moguće o napadu – razinu štete, druge aktivnosti napadača, vremensku crtu događaja tijekom napada itd. Prilikom takve analize potrebno je korelirati velike količine podataka iz različitih izvora, kao što su IDS zapisi, zapisi o tokovima prometa itd. Osim reaktivnih mjera kao što su alarmi i forenzička analiza moguće je podatke iskoristiti i proaktivno te konstruirati obranu pomoću preporučenih smjernica, pravila u vatrozidu, IPS-u ili autentifikacije [2].

4.1.1. Tijek rada istraživačke analize podataka

Tijek rada prilikom istraživačke analize podataka temelji se na petlji primjene EDA tehnika, ekstrakcije fenomena i dubinske analize. Cijeli proces prikazan je na Slici 4.1. EDA tehnika je proces odabira skupa podataka i njegova sažimanja na način na koji će omogućiti osobi da identificira fenomen vrijedan istraživanja. Većina EDA tehnika su vizualizacije, dok ostale obuhvaćaju neke od pristupa rudarenja podataka kao što je klasteriranje ili klasične statističke tehnike kao što je regresijska analiza. EDA tehnike pružaju tragove na temelju ponašanja koji se mogu iskoristiti za dublju analizu originalnih podataka, stoga se ova petlja tehnika – ekstrakcija – analiza može ponavljati do beskonačnosti. EDA proces jednom mora stati, a po završetku analitičar će uobičajeno imati više potencijalnih mehanizama za odgovor na inicijalno pitanje. Na primjer, za periodične fenomene moguće je koristiti autokorelaciju, fourierovu analizu ili histograme. Koju od opcija koristiti određuje se procesom testiranja i operativnim zahtjevima. Proces testiranja treba upotrijebiti tehnike razvijene tijekom EDA i odrediti koje su najprikladnije za operativno korištenje, konstruirati alarme i izvještaje.



Slika 4.1. Proces istraživačke analize podataka

4.2. Vizualizacija podataka

Vizualizacija podataka je najkorisnija EDA tehnika. Razlikuju se ovisno o tipu podataka koji se istražuje i cilju analize a za pravi odabir bitno je razumjeti varijable. Varijabla je karakteristika entiteta koji se može mjeriti ili izbrojiti. Varijable se mogu mijenjati tijekom vremena ili između entiteta. S obzirom na to varijable se dijele u četiri kategorije:

- Intervalne – varijabla kod koje interval između dvije vrijednosti ima značenje ali omjer između tih vrijednosti nije bitan
- Omjerne (engl. *ratio*) – omjerna varijabla je isto što i intervalna varijabla samo što posjeduje značenje nultog događaja
- Redne (engl. *ordinal*) – podaci se nalaze poredani na brojevnom pravcu ali nemaju fiksne intervale
- Nominalne – ovakvi podaci su određeni nekim nazivom tj. Nemaju numeričku vrijednost ni poredak (npr. broj porta)

Intervalne, omjerne i redne varijable se nazivaju i kvantitativnima a nominalne kvalitativnim ili kategoričkim varijablama. Daljnja podjela je moguća na diskretne i kontinuirane. U mrežnom prometu velika većina podataka je diskretna [2].

4.2.1. Jedno-varijantna vizualizacija

Najosnovnije vizualizacije primjenjuju se na jedno-varijantne podatke koji se sastoje od jedne promatrane varijable po mjerenoj jedinici. Primjer jedno-varijantnog mjerenja je veličina paketa u bajtima ili broj IP adresa u nekom vremenskom periodu.

Histogram je temeljni graf za intervalne i omjerne podatke a predstavlja razdiobu vrijednosti neke varijable tj. koliko često varijabla poprima koje vrijednosti. Sastoji se od skupova podataka, koji su diskretni rasponi vrijednosti, i frekvencija. Koristan je u analizi podataka jer omogućava pronalazak strukture u distribuciji varijable, a struktura omogućava daljnju istragu. U slučaju histograma ta struktura je generalno mod – vrijednost koja se najčešće pojavljuje. Modovi se manifestiraju vrhovima na grafu. Analiza histograma se u velikom broju slučajeva sastoji od određivanja razdiobe podataka i modova. Modovi postavljaju nova pitanja i usmjeravaju ka izvornim podacima za dublju analizu.

Stupčasti graf je analogan histogramu kada se radi s jedno-varijantnim kvalitativnim podacima. Razlika između histograma i stupčastog grafa je u grupiranju podataka. Kvalitativni podaci mogu se grupirati u raspone koji predstavljaju grupe u histogramu. Grupe su aproksimacije i raspon varijabli koje sadrže može se po potrebi mijenjati. U slučaju stupčastih grafova različite potencijalne vrijednosti podataka su diskretne, prebrojive, i često bez poretka.

Statistički sažetak podataka se često daje pomoću pet brojeva: minimalna vrijednost, prvi, drugi i treći kvartil te maksimalna vrijednost. Ovo se može jednostavno vizualizirati pomoću kutija-grafa (engl. *boxplot*) koji se sastoji od pet linija, po jedna za svaku od pet prije navedenih vrijednosti. Centralne tri su spojene tako da tvore „kutiju“ [2].

4.2.2. Dvo-varijantna vizualizacija

Dvo-varijantni podaci sastoje se od dvije promatrane varijable po mjerenoj jedinici. Primjer dvo-varijantnih podataka su broj bajtova i paketa u toku podataka (dvije kvantitativne varijable) ili broj paketa po protokolu (kvalitativna i kvantitativna varijabla). Najčešće vizualizacije u ovome slučaju su raspršeni grafovi (engl. *scatterplots*) za usporedbu dviju kvantitativnih varijabli i višestruki kutija-grafovi za usporedbu kvalitativnih i kvantitativnih varijabli te kontingencijske Tablice za usporedbu dvaju kvalitativnih varijabli.

Grafovi raspršenja pokazuju povezanost dviju intervalnih, omjernih ili rednih varijabli. Primarni cilj analize ovih grafova je određivanje strukture podataka. Česta svojstva u ovim grafovima su klasteri, praznine i linearne ovisnosti. Klasteri i praznine su promjene u gustoći grafa raspršenja. Ako su dvije varijable povezane postojati će promjena u gustoći podataka negdje u grafu. Linearne ovisnosti se pojavljuju na grafu u obliku linije a jakost linearne povezanosti može se procijeniti iz gustoće točaka oko linije.

4.3. Analiza teksta

Analiza teksta je sveprisutna u informacijskoj sigurnosti. Proteže se od analize strukturiranog ili polustruktuiranog teksta kao što su IDS i DNS zapisi sve do nestrukturiranog teksta kao što je sadržaj paketa.

4.3.1. Kodiranje teksta

Kodiranje se odnosi na pravila koja govore računalu da vrijednost 65 treba predstaviti sa slovom A ili da se vrijednost 0x43E predstavlja slovom o. Trenutni sustavi se oslanjaju na Unicode standard kodiranja, najčešće UTF-8, ali zbog održavanja kompatibilnosti sa starijim sustavima postoji velika nasljedna infrastruktura iznad toga. Kodiranje teksta zahtjeva upravljanje velikim brojem rubnih slučajeva i nasljednog procesuiranja. Veliki broj standarda vodi do dvosmislenosti što odgovara napadačima. Informacija koja prolazi između entiteta u mreži biti će kodirana nekoliko puta, često implicitno. Na primjer, HTTP protokol podržava kompresiju podataka, stoga ako mrežni senzor između čvorišta ne obraća pozornost na kompresiju doći će do propusta. Unicode je set znakova tj. indeks koji povezuje svaki znak sa jedinstvenom numeričkom vrijednosti (engl. *code point*).

Kodiranje je mehanizam za predstavljanje tih brojeva u standardiziranom binarnom obliku. Znak nije slika nego ideja koja stoji iz grupe slika. Glif (engl. *glyph*) je reprezentacija znaka koji je indeksiran svojim „code point-om“. Razlika između glifa i znaka je vrlo bitna jer različite sheme kodiranja interpretiraju iste reprezentacije znatno drugačije. Stoga događa se da se isti glif reprezentira znatno drugačijim kodiranjem kao na primjer slovo ä može ovisno o kodiranju biti predstavljeno kao „a s prijeglasom “, „a“ iz kojeg slijedi prijeglas ili „a“ iza kojeg slijedi razmak i prijeglas. Izvan unicode standarda situacija je puno lošija gdje razlike između „code point-a“ i njegovog kodiranja nije baš jasna. Unicode ne radi pretpostavke o tome kako treba reprezentirati „code point“ dok Windows kodiranje, macOS kodiranje i ISO standardi rade takve pretpostavke. Unicode je istisnuo ali ne i potpuno zamijenio ostale sheme kodiranja kao što su JIS za japanske sustave ili KOI8 za ruske sustave. Kodiranje unutar unicode standarda se odvija primarno putem UTF-8 (engl. *Unicode Text Format - UTF*) kodne scheme specificiran s RFC 3629. UTF-8 je kodna schema promjenjive dužine specifično dizajnirana za rješavanje brojne probleme koje uključuje prijenos podataka i prethodni standard UTF-16. UTF-8 je kompatibilan i s ASCII (engl. *American Standard Code for Information Interchange* - ASCII) 7-bitnim standardom kodiranja.

Za napadače je kodiranje teksta alat za izbjegavanje i provlačenje informacija kroz filtere koji su namijenjeni da ih zaustave te odgađanje uspješne detekcije. Napadači se mogu koristiti brojnim tehnikama potaknuti implicitnim ovisnostima unutar strategija kodiranja teksta. Base64 kodiranje je tehnika kodiranja originalno razvijena za prijenos elektroničke pošte a trenutno se koristi za prijenos podataka, između ostalog i malicioznog koda. Cilj base64 kodiranja je omogućavanje mehanizma za kodiranje proizvoljnih binarnih vrijednosti koristeći prepoznatljive znakove što čini mapiranjem binarnih vrijednosti u raspon vrijednosti od 0 do 63 i zatim dodjeljuje svaki „code point“ poznatom lako prepoznatljivom znaku čime mu kodira vrijednost koristeći taj znak. Standard uključuje 64 „code point-a“ koji predstavljaju znakove od A-Z, a-z i 0-9 i dva znaka koja ovise o standardu - + i /. Base64 kodiranje je „jeftino“ i sveprisutno sa već razvijenim funkcijama u jezicima kao što su Python ili Javascript. Pored base64 kodiranja, HMTL podržava u potpunosti drugačije kodiranje NCR (engl. *numeric character references*). NCR kodira vrijednosti koristeći format &#DDDD, gdje je DDDD indeks pripadajućeg „code point-a“ u Unicode standardu. Najbrži način za otkrivanje base64 kodiranja je traženje znakova koje base64 ne podržava a očekuju se u normalnom tekstu (zarezi, razmaci, apostrofi) dok se NCR prepoznaje po & znaku.

Sveprisutno je i neformalno kodiranje teksta koje se pojavljuje prilikom pokušaja napadača da zaobiđe detekciju. Kod neformalnog kodiranja ne postoje pravila ali se mogu raspoznati česti načini kao što je jednostavno obrtanje slijeda znakova, pridodavanje ili „1337sp3ak“ [2].

Kompresija je jednostavan i robustan (otporan na greške) način za obilazak detekcije. Ovo nije nov problem i mnogi alati za dublju analizu paketa danas adresiraju ovaj problem ali problem je što se novi algoritmi za kompresiju stalno razvijaju što dodatno otežava posao pri analizi. Kompresiju je kao i enkripciju jednostavno i brzo moguće provjeriti izračunom entropije a međusobno ih je moguće razlučiti analizom zaglavlja u kojem je označen algoritam kompresije [2].

4.3.2. Tehnike za analizu teksta

Prilikom analize teksta moguće je tekst prvo obraditi n-gram-skom analizom. N-gram je sekvenca od n ili više simbola. Na primjer, riječ „diplomski“ moguće je podijeliti u trigrame na sljedeći način „dip“,“ipl“,“plo“,“lom“,“oms“,“msk“,“ski“. N-gram analiza proizvede puno manjih nizova iz jednog izvora a proces analize n-grama je računalno zahtjevan, iako koristan kada nismo sigurni što je tekst kojeg analiziramo. Na pojedinim riječima ili n-gramima moguće je provesti izračun Hammingove udaljenosti s ciljem određivanja sličnosti među nizovima. Hammingova udaljenost između dva niza jednake duljine je ukupan broj individualnih znakova koji se međusobno razlikuju. Jednostavna je za izračunati ali ograničena veličinom niza i manjkom normalizacije što znači da su vrijednosti malo nejasne. Prilikom izračuna bitno je jasno definirati što je definicija različitosti jer ako na primjer koristimo individualne bitove kao mjerilo udaljenosti isti niz kodiran UTF-16 i UTF-8 kodnim shemama daje različite vrijednosti udaljenosti. Udaljenost je korisna pri detekciji eksploatacije „fat fingering“ tehnike prijevare, gdje napadači kreiraju domene koje izgledaju vrlo slično legitimnim i često korištenim domenama, pri čemu će udaljenost među tim nizovima biti manja nego između legitimnih domena, kojima je i cilj da se razlikuju jedne od drugih.

Već je spomenuta primjerna entropije pri detekciji kompresije i enkripcije. Entropija signala je vjerojatnost da će se neki znak pojaviti u uzorku tog signala. Entropija se matematički opisuje prema izrazu (4-1).

$$H(X) = -\sum_{x \in X} P(x) \log_2 P(x) \qquad (4\text{-}1)$$

Prema izrazu (4-1) entropija je suma umnožaka vjerojatnosti simbola i njihovog vlastitog sadržaja informacije. Konačni rezultat opisuje broj bita potrebnih da bi se opisali svi promatrani znakovi u nizu. Signali visoke entropije imaju puno šuma tj. Podaci izgledaju kao da su nasumično generirani, jer je vjerojatnost pojave bilo kojeg znaka podjednaka. Signali male entropije usmjereni su ka uzorcima koji se ponavljaju i opće je pravilo da su strukturirani i moguće ih je komprimirati. Govorni jezik ima malu entropiju kao i većina mrežnih datoteka zapisa dok je enkriptirani i komprimirati sadržaj ima veliku entropiju. Selektivno uzorkovanje i procjena entropije je dobar i brz način provjere kompresije i enkripcije podataka [2].

4.4. Bihevioristička analiza napadača

Analiza ponašanja napadača potpomaže detekciju identifikacijom generalnih uzoraka ponašanja. Generalna i česti postupci iza sebe ostavljaju karakteristične anomalije čijom detekcijom možemo dobiti trag za daljnju analizu događaja u mreži.

4.4.1. Prtljanje

Prtljanje (engl. *fumbling*) se odnosi na proces sustavno neuspješnog povezivanja na metu pomoću reference. Referenca može biti IP adresa, URL ili e-mail adresa. Ono što čini mumljanje sumnjivim je to da bi legitimni korisnik trebao imati reference koju su mu potrebne. Kako bi saznali potrebne informacije (npr. adresu e-mail servera) napadači moraju pogađati, krasti, izviđati takve podatke iz sustava pri čemu čine greške koje su često velike i sistematične. Prvi korak u analizi je identifikacija tih grešaka i njihovo razlikovanje od slučajnih pogrešaka.

Prtljanje kod TCP protokola znači da jedan entitet u mreži nije uspio doći do drugog entiteta tj. njegove IP adrese/porta, dok kod HTTP protokola prtljanje označava nemogućnost pristupa URL-u. Individualno napipavanje je normalno ali zabrinutost treba stvoriti prtljanje koje se ponavlja. Ponavljajuće mumljanje nastaje iz nekog od tri razloga: greške u konfiguraciji, automatiziranog

softvera ili skeniranja. Najčešće, prtljanje nastaje jer adresirano odredište ne postoji što može biti prijelazan fenomen zbog krivog adresiranja, kretanja ili zato što je netko adresirao resurs koji nikad nije ni postojao. U slučaju HTTP protokola možemo zaključiti da ljudi rijetko samostalno unose adrese a posebice IP adrese. URL-ovi se češće otvaraju preko poveznica ili se kopiraju, više nego što se samostalno upisuju tako da kada netko krivo adresira URL velika je vjerojatnost da su grešci prethodile krive konfiguracije. Prilikom kretanja napadača kroz mrežu krivo adresiranje je vrlo česta pojava pri kojoj meta postoji, ali napadač nije informiran o adresi pa upisuje kriva imena ili IP adrese ili koristi stare adrese računala. Svaka privatna mreža uobičajeno ima neiskorišteni IP prostor i portove. Neiskorištene IP adrese i portovi se nazivaju „tamni prostor“ (engl. *dark space*) i legitimni korisnici im rijetko pristupaju ali napadači gotovo uvijek. Kada dođe do greške u adresiranju karakteristično je vidjeti veliki broj istih besmislenih nizova koji se pojavljuju s velikog broja mjesta. Skeniranje je najčešći oblik napadačkog prometa koji se može uočiti na mreži. Skeniranje mreže je toliko česta pojava da se ne može uvijek klasificirati kao napad. Skeniranje se može specificirati kao vertikalno – skeniranje svih portova na jednom računalu ili horizontalno – skeniranje svih računala na određenom portu. Nepisano pravilo je da napadači izvršavaju horizontalna skeniranja dok administracija i obrana izvršava vertikalna skeniranja. Razlog je primarno oportunistički jer napadači traže mete koje posjeduju one ranjivosti koje znaju iskoristiti, iako će u napadač skenirati i vertikalno ako je zainteresiran specifično za određenu metu.

Detekcija prtljanja odvija se u dvije faze: određivanje što prema korištenom protokolu znači da je netko neuspješno pristupio resursu i određivanje je li neuspješnost prolazna/stalna i globalna/lokalna. Lažno pozitivni rezultati mogu se dogoditi zbog krive konfiguracije, promjena u mreži ili korisničkih grešaka. Stoga je bitno razlučiti uzorak namjernog i slučajnog fenomena. prtljanje se identificira pomoću nekoliko tehnika kao što su već spomenuta komunikacija s „tamnim prostorom“, raspršenje adresa, neuspješne sjednice, povećan broj ICMP upozorenja, povećan broj upozorenja za specifičnu uslugu. Raspršenost adresa odnosi se na činjenicu da većina računala interno komunicira s malim brojem različitih internih adresa pa će stoga komunikacija s neproporcionalno velikim brojem adresa u kratkom vremenu biti znak upozorenja. Za detekciju neuspješne TCP sjednice potrebno je razumjeti kako radi TCP protokol. Ukratko, normalna TCP sjednica zahtjeva barem tri paketa samo za uspostavu veze a uz standardni MTU od 1500 bajta legitimna sjednica će se sastojati od nekoliko desetaka paketa. Prilikom detekcije ove anomalije analiza zapisa toka je efektivnija nasuprot analize paketa, jer neuspješne sjednice svakako ne

posjeduju sadržaj. Osnovne tehnike detekcije uključuju proučavanje zastavica, prebrojavanje paketa i veličina sadržaja paketa. Zastavice su dobar indikator prtljanja, ali su komplicirane za korištenje zbog rubnih slučajeva koji se uspješno iskorištavaju pri skeniranju. Na primjer, kada klijent šalje ACK zastavicu iako prije toga nije primio SYN i ACK zastavice od poslužitelja. Dakle, bez prethodnog odgovora od strane poslužitelja, klijent ne bi trebao slati ACK zastavice i kao posljedica toga tokovi sa SYN zastavicom, a bez ACK zastavice su dobri indikatori mumljanja. Napadači izrađuju pakete s čudnim kombinacijama zastavica s ciljem određivanja stoga i konfiguracije vatrozida. Primjer ovoga je „božićno drvce“ paket koji ima postavljene sve moguće zastavice, SYN, ACK, FIN, PUSH,URG i RST. Paketi koji se sastoje samo od ACK zastavice su normalna pojava kod korištenja protokola koji duže vrijeme održavaju vezu npr. SSH. Nije svaki mali broj paketa s iste adrese pokušaj skeniranja i nije svako skeniranje prijetnja. Mali broj organizacija redovito skenira cijeli Internet s ciljem otkrivanja ranjivosti npr. Censys, Shodan, Project Sonar.

Mumljanje na razini usluge je često rezultat skeniranja, automatiziranog malicioznog koda ili nekog od alata za izviđanje. Za razliku od mumljanja na razini mreže ovo mumljanje se može jasno identificirati jer postoje kodovi za kontrolu grešaka za većinu velikih usluga pomoću kojih je moguće razlikovati legitimne konekcije od malicioznih. Jedna od danas najraširenijih usluga, web posluživanje, radi putem HTTP protokola. Svaka transakcija u HTTP protokolu šalje povratnu informaciju u obliku statusnog koda od tri broja. Kodovi formata 4xx su rezervirani za klijentske greške. Najvažniji od 4xx kodova su 404 – „not found“ i 401 – „unauthorized“. 404 indicira da resurs nije dostupan na zahtijevanom URL-u. Ovaj kod će se pojaviti kod normalnih pojava kao što je krivi unos URL-a ili objava URL-a koji ne postoji. Međutim ova greška može indicirati i automatizirani softver koji skenira HTTP stranice i traži poznate ranjivosti. Na ovaj fenomen ukazuje analiza zahtjeva koji neće imati veze s pravom strukturom stranice jer napadači jednostavno pogađaju da nešto postoji i vode se dokumentacijama i dobrim praksama. 401 greške odnose se na osnovne mehanizme autentifikacije ukomponirane u HTTP protokol vrlo davno i ne bi se trebali koristiti jer su vrlo nesigurni zbog korištenja nekriptiranih base64 kodiranih lozinki [2].

4.4.2. Ovisnost volumena prometa i vremena

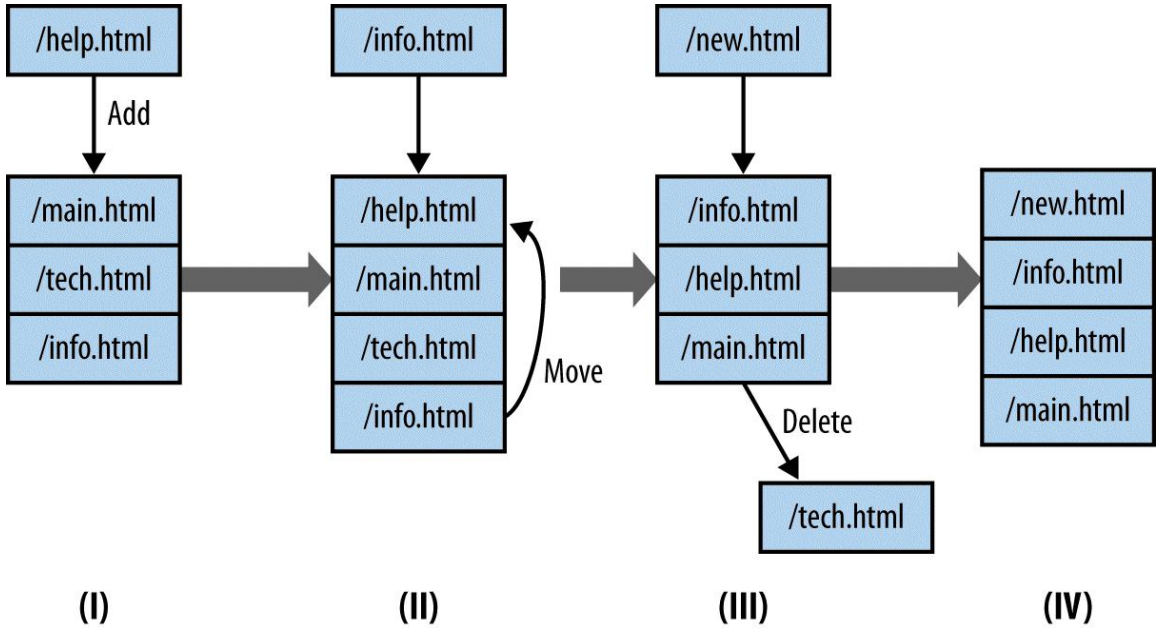
Analiza količine prometa kroz vrijeme može otkriti neke fenomene korisne za detekciju napadača. Neki od fenomena su „beaconing“, krađa datoteka, uskraćivanje usluge DoS (engl. *Denial of Service*). „Beaconing“ se odnosi na komunikaciju internog računala sa nepoznatom adresom u pravilnim intervalima što je mogući znak komunikacije s upravljačkim i kontrolnim poslužiteljem (engl. *Command and Control*) koji se koristi pri modeliranju APT-a (engl. *Advanced Persistent Threat* – APT), botneta i slično. Velike količine podataka usmjerene izvan mreže indiciraju krađu podataka iznutra, dok DoS napadi onemogućavaju normalno pružanje usluga. Volumen mrežnog prometa jako varira bez znatne povezanosti između volumena i značajnih događaja ali postoje neki od načina koji uz dobru ljudsku procjenu mogu indicirati opasnost.

Najveći dio prometa neke organizacije generiraju njezini zaposlenici pa će volumen mrežnih podataka pratiti radno vrijeme zaposlenika. Dakle, promet će početi rasti oko 08:00 h, doseći će vrhunac oko 13:00 h i padati sve do 18:00 h. Prema ovome socijalnom fenomenu, poznavajući gdje se nalazi mreža koju nadziremo, možemo predvidjeti događaje i količinu prometa. „Beaconing“ se manifestira kao tok informacija u pravilnim intervalima između zaraženog sustava i nepoznatog poslužitelja. Neki od mogućih lažno pozitivnih događaja su paketi održavanja na životu usluga kao što je SSH, provjere ažuriranja aplikacija ili vremena i vijesti u aplikacijama koji se također često odvijaju u pravilnim intervalima. Detekcija se odvija u dva koraka: određivanja stalnih signala i upravljanje inventarom. Algoritam za određivanje stalnih signala je sljedeći: podjeli sekvence zapisa toka u jednake grupe, provjeri median vremenske udaljenost između grupa i podjeli IP adrese prema medianu i toleranciji te ako postoji veliki broj tokova blizu mediana pronađen je redoviti događaj. Drugi korak uključuje provjeru mogućnosti da se radi o legitimnim aplikacijama a ne malicioznoj komunikaciji.

Krađa podataka je najosnovniji napad na bazu podataka ili web stranicu, posebno ako je web stranica namijenjena samo za internu upotrebu. Sitnice razlikuju kopiranje podataka od legitimnog pristupa s obzirom da je svrha poslužitelja da poslužuje podatke ali mora se manifestirati porastom volumena razmijenjenih podataka. Prvo je potrebno izraditi model normalnog volumena podataka kroz vrijeme sa poslužitelja i izračunati pragove. Ovdje je vrlo bitna upotreba vizualizacije za određivanje razdiobe podataka i određivanja praga a u tom slučaju najkorisniji je histogram. Velik broj usluga na Internetu redovno kopira sadržaj stranica, kao što su „web spiders“ i Internet arhiva a

ako je stranica striktno interna primjer su sigurnosna kopiranja. Vizualizacija je bitna i zbog određivanja lažno pozitivnih događaja jer omogućava konstrukciju popisa adresa koje je potrebno ignorirati. Idući fenomen na koji je potrebno obratiti pozornost je lokalitet.

Lokalitet je tendencija grupiranja referenci (IP adresa, memorijskih lokacija, URL – ova). Na primjer, praćenjem web stranica koje neki korisnik posjećuje pronaći ćemo uzorak prema kojem je većina web stranica koje korisnik posjećuje locirana na malom broju predvidljivih lokacija (prostorni lokalitet) i da korisnik stalno iznova posjećuje iste stranice (vremenski lokalitet). Lokalitet je koristan komplement volumetričkoj analizi jer su korsnici generalno predvidljivi, tipično posjećuju mali broj stranica i govore s malim brojem ljudi što se može lako modelirati radnim setom (engl. *working set*). Prema Slici 4.2. radni set je implementiran kao zadnji korišteni član LRU (engl. *Least Recently Used* – LRU) red fiksne veličine a namijenjen je praćenju web aktivnosti pribavljanjem URL – ova sa HTTP poslužitelja i dodavanjem na stog.



Slika 4.2. Radni set [2]

Kada radni set dobije referencu radi jednu od tri stvari: ako postoje prazne reference dodaje se nova referenca na kraj reda, ako je red napunjen i referenca već postoji pomiče ju na kraj reda; ako je red popunjen a referenca ne postoji onda dodaje referencu na kraj reda a referenca na početku reda se briše. Ovo možemo modelirati promatrajući vjerojatnost zamjene vrijednosti u radnom setu kao funkciju veličine radnog seta. Pareto model je prikladan za modeliranje normalne korisničke

aktivnosti a radni skupovi generalno imaju idealnu veličinu nakon koje je povećanje kontraproduktivno.

DoS napadi rezultiraju nemogućnošću pristupa resursu sa udaljenje lokacije. Većina ih je implementirana kao distribuirani DoS (DDoS) napadi gdje napadač koristi mrežu zaraženih računala za izvođenje napada. Neki od način za izvođenje DoS napada su sljedeći: iscrpljivanje na razini usluge (engl. *Service level exaustion*), SYN poplava, iscrpljenje propusnosti i jednostavni fizički napadi. Iscrpljivanje na razini usluge odvija se kada ciljani resurs sadrži javno dostupnu uslugu, koristeći botnet (mreža zaraženih računala) napadač pokreće set klijenata na meti, gdje svaki radi neku trivijalnu ali akciju specifičnu za tu uslugu. SYN poplava je klasičan DDoS napad, kod meta s otvorenim TCP portom napadač koristi klijente koji pomoću SYN paketa otvaraju nove veze prema resursu ali ih ne koriste. Iscrpljivanje propusnosti odnosi se na slanje velike količine podataka prema resursu s ciljem pretrpavanja veze. Koliko resursa napadač treba za uspješno izvođenje napada ovisi o njegovoj implementaciji. Nepisano pravilo je da što je napad više na OSI modelu, veće opterećenje stvara na metu i potrebno je manje zaraženih računala. Na primjer, pogađanje usmjerivača iscrpljivanjem propusnosti pomoću SYN poplava napada treba jednostavno iscrpiti metin TCP stog dok na višoj razini alati kao što je Slowloris efektivno kreiraju djelomične HTTP konekcije iscrpljujući resurse web poslužitelja. Kod detekcije DoS napada najčešći lažno pozitivni događaji su takozvani bljesak gomile (engl. *flash crowds*) i odrezani kabel (engl. *cable cuts*). Bljesak gomile je nagli priljev legitimnog prometa nakon neke objave ili obavijesti dok termin odrezani kabel označava bilo koju grešku u infrastrukturi. DDoS napadi su mehanički i obično se uključuje i isključuje trenutačno kako napadač izdaje naredbe zaraženim računalima. Kada napad započne odmah počinje iskorištavati maksimalni kapacitet mreže i volumen podataka ostaje konzistentan. Dio zaraženih računala će biti uklonjeno u pokušaju zaustavljanja napada, ali napadači često regrutiraju više računala nego što je potrebno. Kada se napad dogodi najbolje što se može napraviti je identificirati uzorke u prometu i blokirati ono što čini najviše štete. Zbog velike količine šuma koju generira promet, prilikom detekcije DoS napada, bolje je za detekciju koristiti zapise toka i na temelju njega djelovati u nekom od smjerova, kao što su: filtriranje prometa koji se odnosi samo na legitimna računala, a ne „mračni prostor“ adresa; odvajanje legitimnih TCP sjednica pomoću minimalnog broja paketa ili prisutnosti PSH zastavica; postavljanje pragova na broj bajta primljenih podataka [2].

4.4.3. Razlučivanje anomalija povezanih s brojem porta

Broj porta prvi je indikator provjere što neka usluga u mreži predstavlja. Broj porta i usluga na Internetu povezani su jedino socijalnim konvencijama bez striktnih tehničkih ograničenja. IANA (engl. *Internet Assigned Numbers Authority*) organizacija održava javni registar broja porta i usluga koje se s njima povezuju. S obzirom da je dodjela broja porta efektivno proizvoljna i napadači tu činjenicu aktivno iskorištavaju za izbjegavanje detekcije koristeći ne korištene portove ili dobro poznate (engl. *well known*) portove koji još uvijek nose dovoljno legitimnog prometa da se putem njih može znati korišteni protokol. Dodjela portova je kaotična jer sve što netko treba napraviti je odabrati broj porta i nadati se da ga nitko drugi ne koristi. Službeni registar koji održava IANA usmjeren je na protokole specificirane s RFC (engl. *Request For Comments*) koje izdaje IETF (engl. *Internet Engineering Task Force)* dok se ostali protokoli mogu pronaći na stranicama Wikipedia, SpeedGuide.net i SANS Internet Storm Center. Dakle, veliki broj portova je rezerviran za određene aplikacije dok se ostatak koristi konvencinalno od strane ostalih aplikacija od kojih je malo broj stvarno bitan. Neki bitni dobro poznati portovi i usluge koje se na njima pružaju nalaze se u Tablici 4.1.

Tablica 4.1. Neki dobro poznati portovi

Port/ protokol mrežnog sloja	Protokol aplikacijskog sloja
80 /TCP	HTTP
25 /TCP	SMTP
53 /UDP	DNS
179 /TCP	BGP
161-162 /UDP	SNMP
22 /TCP	SSH
23 / TCP	TELNET
123 /UDP	NTP
20-21 / TCP	FTP

Svaka simetrična UDP ili TCP transakcija koristi dva broja porta – poslužiteljski port koji se koristi od strane klijenta za slanje prometa na poslužitelj i klijentski port koji poslužitelj koristi za odgovore. Klijentski portovi se dodjeljuju na kratko vrijeme i recikliraju se iz grupe kratkotrajnih portova koja ovisi o TCP stogu i korisničkoj konfiguraciji sustava. Od svih konvencija vezanih za dodjelu portova najvažnija je ona vezana za portove broja 1024 i manje jer gotovo svaki operacijski sustav za otvaranje konekcija na jednom od ovih portova zahtjeva administratorske ovlasti. Sukladno tome pri legitimnom korištenju samo administrator može pokrenuti uslugu kao što je web poslužitelj ili poslužitelj elektroničke pošte ali ovi portovi postaju privlačni napadačima jer omogućavaju administratorske ovlasti. Dakle, generalno portovi od 0-1024 koriste se za poslužiteljske priključnice što ne znači da ih nije moguće koristiti na klijentu ali to ne bi imalo smisla jer se pokreće klijentska usluga s administratorskim ovlastima. IANA je dodjelila standardni raspon od 49152 do 65535 za kratkotrajne portove iako je taj raspon još u procesu prilagodbe i različiti operacijski sustavi imaju drugačije definirane raspone iz Tablice 4.2. [2].

Tablica 4.2. Raspon kratkotrajnih portova u različitim operacijskim sustavima

Operacijski sustav	Raspon
Windows XP	1025-5000
Windows, od verzije Vista pa na dalje	49152-65535
Linux	32768-65535

4.5. Upotreba grafova u detekciji, analizi i odzivu na incident

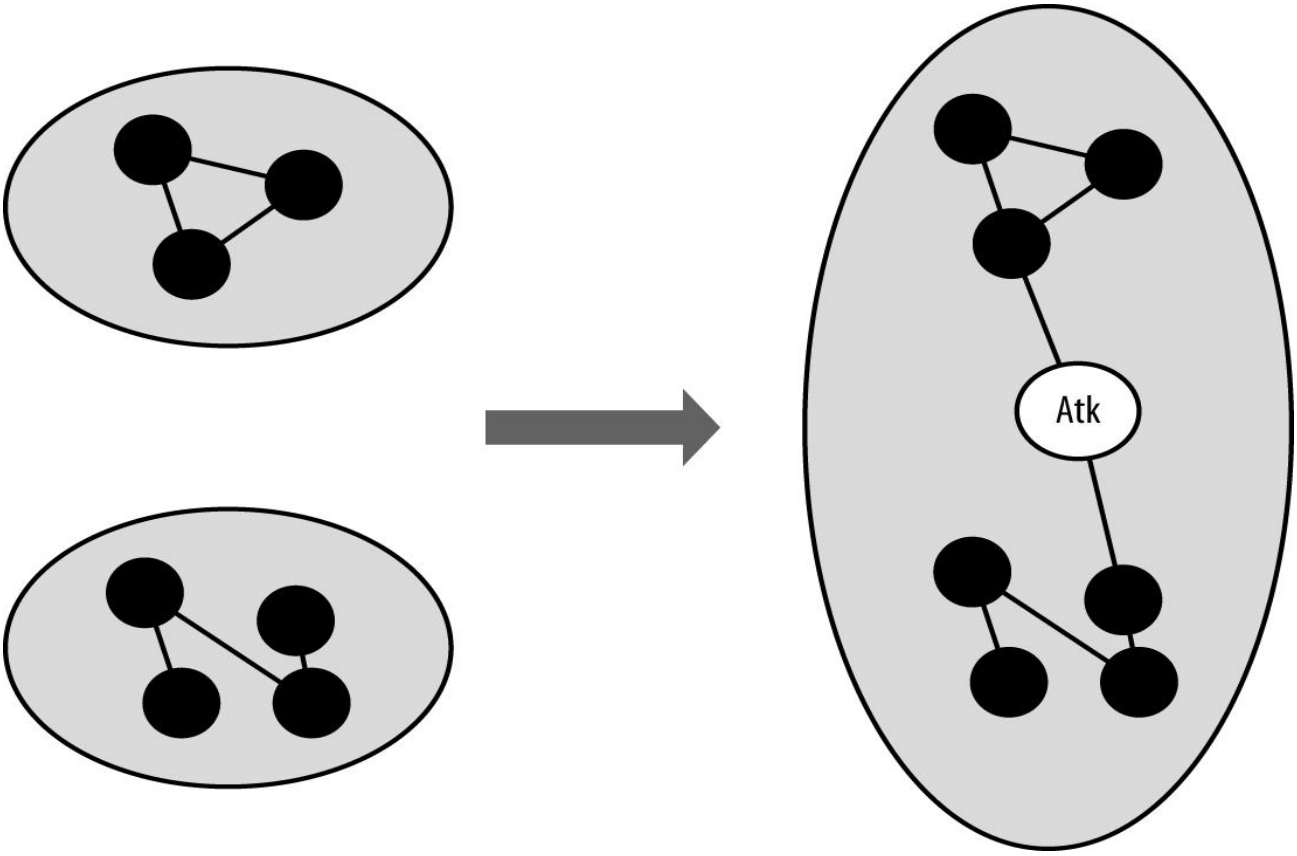
Graf je matematička konstrukcija koja se sastoji od jednog ili više čvorova međusobno povezanih vezama. Korisni su za modeliranje povezanosti i omogućavaju opsežan pogled na povezanost uz apstrakciju detalja kao što su veličine paketa i dužine sjednice. Mnogi važni protokoli kao što je SMTP i protokoli usmjeravanja oslanjaju se na algoritme koji modeliraju mrežu pomoću grafova a atributi grafova omogućavaju identifikaciju kritičnih čvorova u mreži. Osnovni atributi elemenata u grafu su: jednosmjernost i dvosmjernost čvorova, usmjerenost i neusmjerenost veze, izvor i odredište usmjerene veze. Uobičajeno, graf se u potpunosti sastoji od usmjerenih veza ili u

potpunosti od neusmjerenih veza. Svaki čvor u usmjerenom grafu ima svoj stupanj – broj veza koje su povezane na taj čvor. Čvorovi imaju ulazni stupanj – broj veza kojima je taj čvor odredište i izlazni stupanj – broj veza kojima je taj čvor izvor. Put je skup veza koji spaja dva čvora u grafu. Na usmjerenom grafu put prati smjer poveznica dok se na neusmjerenom grafu može kretati u oba smjera. Vrlo važna stavka u analizi grafova je najkraći put – najkraći mogući skup poveznica između dva čvora na grafu. Najkraći put je temeljna stavka analize grafa. Pronalazak najkraćeg puta prisutan je u većini usluga usmjeravanja u obliku OSPF (engl. *Open Shortest Path First*) algoritma. Dijkstrin algoritam koristi težinske vrijednosti veza pomoću kojih određuje najkraći put i vrlo dobro funkcionira ako su težine veza samo pozitivne vrijednosti. S obzirom da je najkraći put temeljni blok za niz drugih atributa većina analiza grafova počine upravo pronalaskom istoga. Idući važan atribut grafa je centraliziranost. Centraliziranost je koncept koji vuče korijene iz analize društvenih veza koja modelira veze između entiteta pomoću grafova te rudari grafove s ciljem pronalaska poveznica. Centraliziranost pokazuje važnost čvora u grafu i postoje tri metrike koje ga određuju: stupanj, blizina i međupovezanost. Blizina predstavlja jednostavnost prijenosa informacije od pojedinog čvora do bilo kojeg drugog čvora u grafu, za što je potrebno izračunati najkraći put između čvorova. Čvor koji ima najmanju vrijednost najkraćeg puta posjeduje atribut najveće blizine. Međupovezanost čvora označava vjerojatnost da će promatrani čvor biti dio najkraćeg puta između bilo koja dva čvora. Izračunava se generiranjem Tablice svih najkraćih puteva i prebrojavanjem broja puteva koji sadrže taj čvor. Algoritmi za centraliziranost su relativne mjere i operativno ih je najbolje koristiti za rangiranje. Koeficijent grupiranja je još jedan mehanizam za određivanje veza među čvorovima u grafu. Koeficijent grupiranja je vjerojatnost da su bilo koja dva susjeda čvora promatranog čvora i sami međusobno susjedi jedan drugome. Graf čiste klijent – poslužitelj arhitekture će imati koeficijent grupiranja nula jer klijenti komuniciraju samo sa poslužiteljima i obrnuto [2].

4.5.1. Detekcija anomalija pomoću grafova

Već su spomenuti neki mehanizmi koji su se oslanjali na napadačevo neznanje o mreži kao što je skeniranje i slično. Povezane komponente mogu biti korisne pri modeliranju različitih tipova napadačevog neznanja. Napadač može znati gdje se nalaze različiti poslužitelji i sustavi na mreži ali ne i kako se odnose jedni prema drugima. Prema primjeru sa Slike 4.3., ako se korporativna mreža sastoji od dva diskretna dijela npr. Inženjeri i marketing i između njih postoji vrlo malo interakcije,

kada napadač uđe u mrežu kombinira ove dvije komponente i kreira jednu veliku komponentu koja ne postoji u normalnim uvjetima.



Slika 4.3. Napadač izaziva anomaliju povezivanjem dvije mrežne komponente bez interakcije [2]

Komponente je moguće odrediti pomoću veličine ili prateći entitete unutar komponenti. Ako se prati veličina potrebno je pratiti komponente po veličini što je jednostavno ali nedovoljno osjetljivo na suptilne napade [2].

4.5.2. Odziv pomoću grafa

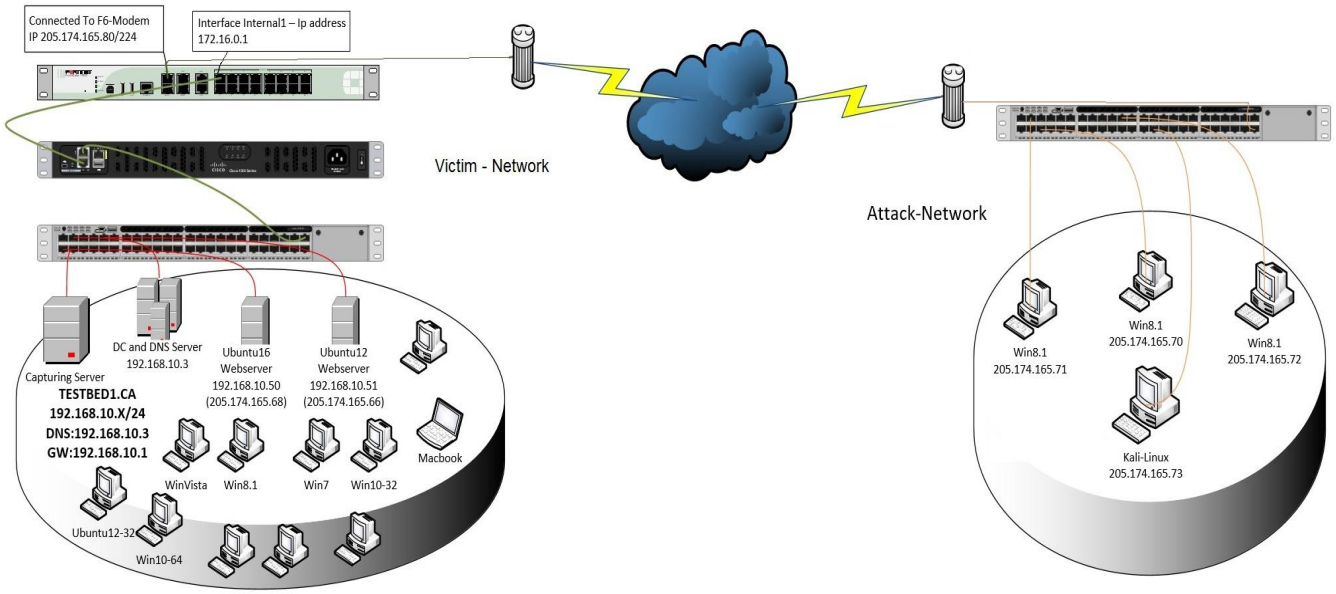
Koristeći centraliziranost moguće je dobiti uvid u zaraženo računalo u mreži koristeći metodu iz sljedećeg primjera. Npr. ako napadač zarazi jedno ili više računala u mreži, koja zatim komuniciraju sa serverima za kontrolu i upravljanje a prvotno zaraženo računalo pokazivati će veću centraliziranost od ostalih. Analiza se može provesti izolirajući promet u onaj prije i onaj nakon događaja čime je moguće dobiti uvid u novonastale čvorove u mreži. Idući korak je određivanje s kim inficirana računala komuniciraju. Ovaj postupak provodi se pomoću BFS (engl. *Breadth-First Search* – BFS) algoritma koji izvršava pretraživanje odabirući čvor, promatrajući sve njegove susjede i zatim promatrajući sve susjedne čvorove susjednih čvorova. Pomoću pretraživanja moguće je doći do drugih inficiranih računala, sumnjivih meta i ostalih dijelova mreže koje je potrebno dublje analizirati [2].

5. PRIMJENA MREŽNOG SIGURNOSNOG NADZORA U DETEKCIJI I ODZIVU RAZLIČITIH TIPOVA NAPADA

U ovome poglavlju analizirati će se načini na koje se različiti tipovi napada manifestiraju u obliku IDS alarma i vizualizacija. Odabrani tipovi napada su infiltracija preuzimanjem malicioznog dokumenta, XSS(engl. *Cross Site Scripting*) napad, DoS napad na razini usluge, DoS napad na TCP razini i kreiranje botneta pomoću ARES alata. Testovi su izvedeni na otvorenom setu podataka izrađenog u sklopu [13] rada s ciljem generiranja kvalitetne kolekcije podataka za potrebe testiranja IDS sustava. Podaci su preuzeti u obliku potpune snimke prometa (pcap format) a zatim je na njima provedena analiza. Paketi su procesuirani Snort IDS – om i PacketBeat alatom. Podaci su pohranjeni u ElasticSearch bazi podataka a pretražuju se i vizualiziraju pomoću Kibana sučelja.

5.1. Testni set podataka/okruženje

Infrastruktura je podijeljena u dvije u potpunosti odvojene mreže, mrežu žrtava i mrežu napadača. U mreži žrtava pokrivena je sva najčešća i nužna oprema kao što su usmjerivači, vatrozidi, preklopnici kao i računala sa tri najčešća operacijska sustava Windows, Linux, Macintosh. Sav ulazni i izlazni promet se nadzire na glavnom preklopniku mreže žrtava pomoću zrcaljenog porta kako je i prikazano Slikom 5.1. na kojoj se vidi i sveukupna arhitektura testne mreže.



Slika 5.1. Arhitektura testne mreže[13]

Osim malicioznog prometa ovaj set sadrži i legitiman benigni promet generiran pomoću algoritama strojnog učenja. Benigni promet profilira legitimne korisnike na temelju HTTP, HTTPS, FTP, SSH i protokola elektroničke pošte. Popis svih korištenih uređaja te pripadajućih IP adresa i operacijskih sustava nalazi se u Tablici 5.1.

Tablica 5.1. Popis korištenih uređaja

	Vrsta uređaja	Operacijski sustav	IP adresa
Mreža žrtava	Poslužitelj	Windows poslužitelj 2016 (DC i DNS)	192.168.10.3
		Linux Ubuntu 16 (Web poslužitelj)	192.168.10.50/ 205.174.165.68
		Linux Ubuntu 12	192.168.10.51/ 205.174.165.66
	PC	Linux Ubuntu 14.4. 32-bit	192.168.10.19
		Linux Ubuntu 14.4. 64-bit	192168.10.17
		Linux Ubuntu 16.4. 32-bit	192.168.10.16
		Linux Ubuntu 16.4. 64-bit	192.168.10.12
		Windows 7 Pro	192.168.10.9
		Windows 8.1 – 64 bit	192.168.10.5
		Windows Vista	192.168.10.8
		Windows 10 Pro 32-bit	192.168.10.14
		Windows 10 Pro 64-bit	192.168.10.15
		Macintosh	192.168.10.25
	Vatrozid	Fortinet	
Mreža napadača	PC	Kali Linux	205.174.165.73
		Windows 8.1	205.174.165.69
		Windows 8.1	205.174.165.70
		Windows 8.1	205.174.165.71

Promet je kontinuirano sniman pet dana u razdoblju od ponedjeljka 3.7.2017 09:00 h do petka 7.7.2017. 17:00 h. U radu je obrađen promet u rasponu dana u koje su se odvijali napadi koji će se analizirati. DoS napadi izvedeni su u četvrtak 5.7.2017., napadi vezani za Web i infiltraciju putem preuzimanja maliciozne datoteke izvedeni su četvrtak 6.7.2017., a botnet napadi su izvedeni u petak 7.7.2017.

5.2. XSS napad / Umetanje malicioznih skripti u web stranice

XSS napad je tip injekcije u kojemu se maliciozne skripte injektiraju u benigne i legitimne web stranice. Događaju se kada podaci ulaze u Web aplikaciju putem nepovjerljivog izvora, što je najčešće web zahtjev (engl. *request*) ili kada su podaci unutar dinamičkog sadržaja koji se šalje web korisniku bez da se prethodno provjeri postoji li maliciozni sadržaj. Maliciozni sadržaj koji se šalje web pretraživaču često je u obliku JavaScript koda ali može sadržavati i HTML, Flash ili bilo koji drugi oblik koda koji pretraživač može izvršiti. Varijacije u XSS napadima su gotovo beskonačne ali zajedničko im je da šalju privatne podatke, kao što su kolačići (engl. cookies) i drugi podaci o sjednici, napadaču koji žrtvu preusmjerava na web sadržaj koji je pod njegovom kontrolom ili izvodi druge maliciozne operacije na korisnikovom uređaju pod okriljem ranjivog web mjesta. Posljedice napada su jednake bez obzira na vrstu napada, razlika je u načinu dostave malicioznog koda do poslužitelja [14].

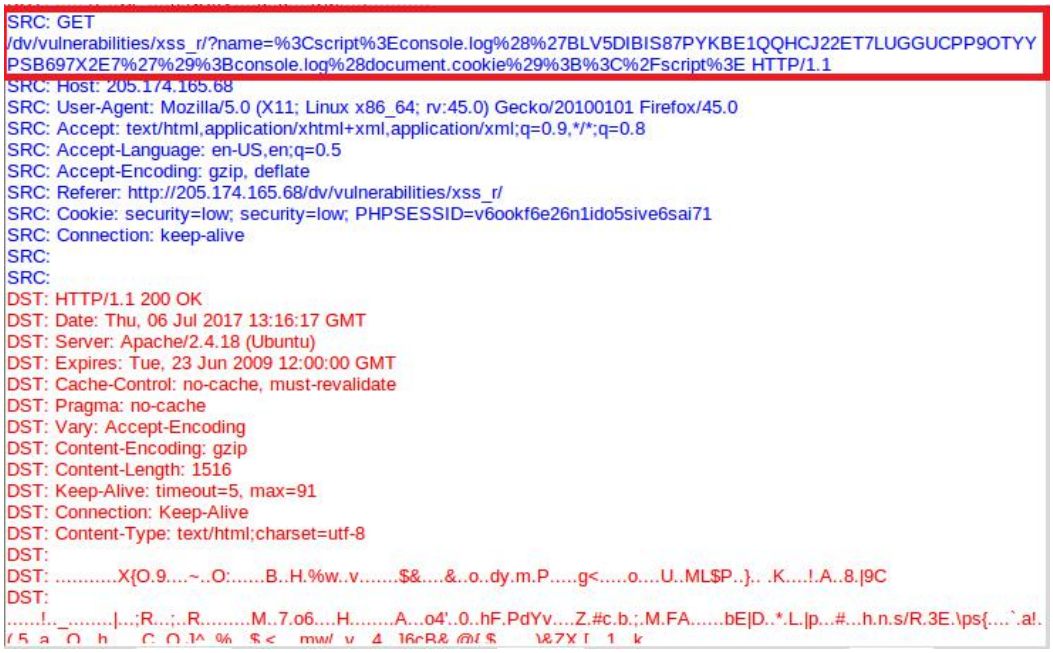
5.2.1. Detekcija događaja i anomalija u mreži

Izvršeni napad proizveo je vrlo malo buke osim same činjenice da je u URL smjestio HTML <script></script> oznake što je aktiviralo IDS alarm. Kako alarm izgleda možemo vidjeti na Slici 5.2. gdje su izlistani svi IDS alarmi u sguil programskom alatu za rukovanje IDS alarmima. Prvo što je bitno primijetiti je velik broj lažno pozitivnih upozorenja s kojima je okruženo upozorenje za napad koji se stvarno dogodio, a na Slici 5.2. je istaknut crvenom bojom.

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	29	filip-hp-15...	5.161	2017-07-06 12:13:34	192.168.10.12	48286	91.189.95.83	80	6	ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management
RT	105	filip-hp-15...	5.198	2017-07-06 12:17:14	192.168.10.17	33675	91.189.95.83	80	6	ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management
RT	35	filip-hp-15...	5.304	2017-07-06 12:17:47	192.168.10.50	43772	91.189.91.23	80	6	ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management
RT	1	filip-hp-15...	5.339	2017-07-06 12:19:23	174.36.232.136	443	192.168.10.14	59419	6	ET POLICY Self Signed SSL Certificate (SomeOrganizationalUnit)
RT	10	filip-hp-15...	5.375	2017-07-06 12:30:00	192.168.10.9	60843	192.168.10.3	53	17	ET INFO Observed DNS Query to .biz TLD
RT	86	filip-hp-15...	5.380	2017-07-06 12:30:52	192.168.10.19	38885	91.189.91.26	80	6	ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management
RT	68	filip-hp-15...	5.474	2017-07-06 12:34:48	192.168.10.51	48630	91.189.91.26	80	6	ET POLICY GNU/Linux APT User-Agent Outbound likely related to package management
RT	44	filip-hp-15...	5.543	2017-07-06 12:37:19	192.168.10.15	55141	192.168.10.3	53	17	ET DNS Query for .to TLD
RT	69	filip-hp-15...	5.545	2017-07-06 12:37:19	192.168.10.3	60157	192.168.10.1	53	17	ET DNS Query for .to TLD
RT	22	filip-hp-15...	5.562	2017-07-06 12:41:41	192.168.10.19	37951	192.168.10.3	53	17	ET INFO Observed DNS Query to .biz TLD
RT	4	filip-hp-15...	5.568	2017-07-06 12:41:41	192.168.10.19	36429	192.168.10.3	53	17	ET DNS Query for .su TLD (Soviet Union) Often Malware Related
RT	12	filip-hp-15...	5.593	2017-07-06 12:44:39	192.168.10.16	7974	192.168.10.3	53	17	ET DNS Query for .cc TLD
RT	41	filip-hp-15...	5.597	2017-07-06 12:44:39	192.168.10.3	62523	192.168.10.1	53	17	ET DNS Query for .cc TLD
RT	891	filip-hp-15...	5.622	2017-07-06 13:16:18	172.16.0.1	52298	192.168.10.50	80	6	ET WEB_SERVER Script tag in URI Possible Cross Site Scripting Attempt
RT	46	filip-hp-15...	5.719	2017-07-06 13:18:20	192.168.10.17	35332	192.168.10.3	53	17	ET DNS Query for .cc TLD
RT	28	filip-hp-15...	5.1365	2017-07-06 13:31:53	192.168.10.16	51135	192.168.10.3	53	17	ET DNS Query for .to TLD
RT	2	filip-hp-15...	5.1378	2017-07-06 13:31:55	192.168.10.9	60088	192.168.10.3	53	17	ET DNS Query for .cc TLD
RT	4	filip-hp-15...	5.1479	2017-07-06 13:34:00	192.168.10.17	3941	192.168.10.3	53	17	ET DNS Query for .to TLD

Slika 5.2. Prikaz IDS alatma u sguil konzoli

Poruka IDS sustava Snort je u ovom slučaju jasna radi se anomaliji pojave Javascript skripte u URL-u web mjesta. Kako bi dublje istražili alarm iz sguil konzole pivotiramo desnim klikom na „Alert ID“ tog događaja i zatim odabiremo „Transcript“ te dobivamo prikaz sa Slike 5.3. na kojem je crvenim pravokutnikom istaknut GET zahtjev koji dolazi na web poslužitelj s privatnom IP adresom 192.168.50 tj. javnom IP adresom 205.174.165.68.



Slika 5.3. Prijepis komunikacije između napadača i web poslužitelja

Iz ovoga HTTP GET zahtjeva možemo razlučiti što je napadač napravio. Dakle, putem GET metode u sklopu „name“ atributa predan je tekst koji je na prvi pogled pomalo nečitljiv jer je heksadekadski kodiran. Razlog tomu je to što se URL može slati internetom koristeći samo ASCII znakove a s obzirom da često sadrži znakove izvan ASCII kodne sheme ti znakovi se kodiraju s znakom „%“ iza kojeg slijede dvije heksadekadske znamenke. Kodna shema za znakove koji se pojavljuju nalazi se u Tablici 5.2.

Tablica 5.2. URL kodiranje znakova posebnih znakova

Znak	Heksadekadski oblik
<	%3C
>	%3E
/	%2F
(	%28
)	%29
;	%3B

Pivotiranjem u Wireshark alat iz sguil konzole možemo dobiti detaljan uvid u promet tj. razumjeti opasnost napada. Prema Slici 5.4. vidimo da se uneseni niz znakova ispisuje u HTML dokumentu što omogućava izvršavanje koda unutar „<script></script>“ oznaka. Ovakav tip napada klasificira se kao reflektirani XSS napad jer se umetnuta skripta reflektira od web poslužitelja u obliku poruke o pogrešci, rezultata pretrage ili bilo kojeg drugog oblika koji sadrži dio ili sav sadržaj unosa poslanog na poslužitelj kao dio zahtjeva. Napadu obično prethodi „phishing“ tj. metoda slanja digitalne poveznice, koja se čini legitimna ali je zapravo namijenjena da namami potencijalnu žrtvu na pružanje nekih osobnih informacija a može se poslužiti elektroničkom porukom, bilo kojom porukom ili web stranicom. Kada je korisnik uvjeren da klikne na maliciozni link i preda specifično izrađenu formu za unos ili čak samo da posjeti malicioznu stranicu, umetnuti kod putuje do ranjive stranice što reflektira napad do korisnikova pretraživača koji tad izvršava kod koji jer je došao od „povjerljivog“ poslužitelja. U ovom primjeru izvršeni kod je pokaznog karaktera i samo pokazuje mogućnost pristupa kolačićima koje bi u malicioznoj namjeri mogao poslati na server pod napadačevom kontrolom i na taj način pristupiti povjerljivim informacijama [14].

Slika 5.4. Rekonstrukcija HTML koda web stranice nakon umetanja JavaScript koda

5.2.2. Odziv na XSS

Za zaustavljanje napada potrebne su hitni popravci u programskom kodu. Popravci su vrlo jednostavni, potrebno je provesti provjeru teksta unesenog u obrazac. Prvo je potrebno ispraviti ranjivosti vezane za napad koji je u tijeku tj. spriječiti unos teksta koji sadrži posebne znakove „<“, „>“ i „/“, kao i njihove heksadekadske pretvorbe zbog mogućnosti ponovno istog napada korištenjem dvostrukog kodiranja teksta. Zatim je preporučljivo pratiti i ostala pravila za prevenciju XSS napada preporučena od strane OWASP -a (engl. *Open Web Application Security project*) [15].

5.3. Infiltracija preuzimanjem malicioznog dokumenta

Napadi infiltracijom preuzimanjem malicioznog dokumenta tipično započinju primjenom socijalnog inženjeringa s ciljem navođenja žrtve da preuzme malicioznu datoteku na svoje računalo. Maliciozni kod je obično prikriven unutar nekog programa ili dokumenta koji obavlja radnje koju su korisne za žrtvu (tj. žrtva ih smatra korisnima), ali u pozadini izvodi niz zadanih naredbi i uspostavlja vezu s napadačevim računalom ili serverom za kontrolu i upravljanje. Infiltracija obično iskorištava ranjivosti programa kao što su Adobe Acrobat Reader, Microsoft Word i sl.

5.3.1. Detekcija događaja i anomalija u mreži

Najprije je potrebno obratiti pozornost na upozorenja koja nam daje IDS. Prema Slici 5.5. vidimo događaje u IDS sustavu putem squil sučelja.

ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	1	filip-hp-15...	5.1868	2017-07-06 17:28:44	192.168.10.8	54119	205.174.165.73	444	6	ET TROJAN Windows dir Microsoft Windows DOS prompt command exit OUTBOUND
RT	1	filip-hp-15...	5.15071	2017-07-06 17:32:48	192.168.10.8	54573	205.174.165.73	444	6	ET TROJAN Windows dir Microsoft Windows DOS prompt command exit OUTBOUND
RT	1	filip-hp-15...	5.29002	2017-07-06 18:05:02	192.168.10.8	1260	205.174.165.73	444	6	ET TROJAN Windows dir Microsoft Windows DOS prompt command exit OUTBOUND

Slika 5.5. IDS događaj u squil sučelju

IDS sustav okida upozorenje s porukom „TROJAN Windows Microsoft Windows DOS prompt command exit OUTBOUND“. Upozorenja se okida se prilikom svakog spajanja javne IP adrese na računalo u privatnom rasponu IP adresa, u ovom slučaju privatne IP adrese 192.168.10.8 i javne IP adrese 205.174.165.73. Iako ovo može biti indikator kompromitacije, zaključak ovisi organizaciji koju se nadzire. Ovaj događaj vrlo lako može biti lažno pozitivan alarm jer je nekim slučajevima normalno da se zaposlenik spaja na mrežu s udaljene lokacije. Međutim pogledamo li daljnji tijek događaja sa Slike 4.6. vidimo da daljnja upozorenja ukazuju na to da upravo računalo s IP adresom 192.168.10.8 skenira cijelu mrežu pomoću Nmap alata. Sveukupno kao posljedica skeniranja genreira se otprilike 500 upozorenja koja su u prikazu sa Slike 5.6. sažeta prema tipu upozorenja ali pivotiranjem sa „CNT“ polja možemo doći i do popisa svakog događaja posebno kao što je prikazano na Slici 5.7.

ST	CNT	Sensor	Alert ID	Date/Time △	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	254	filip-hp-15...	5.18533	2017-07-06 17:33:43	192.168.10.8	38868	192.168.10.5	34906	17	ET SCAN NMAP OS Detection Probe
RT	34	filip-hp-15...	5.28938	2017-07-06 17:53:08	192.168.10.14	52759	192.168.10.3	53	17	ET DNS Query for .cc TLD
RT	2	filip-hp-15...	5.29001	2017-07-06 18:04:20	192.168.10.8	1259	137.254.120.31	80	6	ET POLICY Vulnerable Java Version 1.8.x Detected
RT	75	filip-hp-15...	5.29042	2017-07-06 18:08:18	192.168.10.8	1442	192.168.10.9	445	6	GPL NETBIOS SMB-DS IPC$ share access
RT	4	filip-hp-15...	5.29058	2017-07-06 18:09:12	192.168.10.12	38116	192.168.10.3	445	6	GPL NETBIOS SMB-DS IPC$ unicode share access
RT	4	filip-hp-15...	5.29110	2017-07-06 18:10:30	192.168.10.8	46935	192.168.10.15	135	6	ET SCAN Behavioral Unusual Port 135 traffic Potential Scan or Infection
RT	4	filip-hp-15...	5.29124	2017-07-06 18:10:52	192.168.10.16	48692	192.168.10.3	445	6	GPL NETBIOS SMB-DS IPC$ unicode share access
RT	4	filip-hp-15...	5.29140	2017-07-06 18:11:51	192.168.10.17	37000	192.168.10.3	445	6	GPL NETBIOS SMB-DS IPC$ unicode share access
RT	4	filip-hp-15...	5.29154	2017-07-06 18:12:46	192.168.10.19	56319	192.168.10.3	445	6	GPL NETBIOS SMB-DS IPC$ unicode share access
RT	96	filip-hp-15...	5.29186	2017-07-06 18:13:54	192.168.10.8	1528	192.168.10.50	80	6	ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)
RT	96	filip-hp-15...	5.29187	2017-07-06 18:13:54	192.168.10.8	1528	192.168.10.50	80	6	ET SCAN Possible Nmap User-Agent Observed
RT	4	filip-hp-15...	5.29227	2017-07-06 18:13:55	192.168.10.50	38996	192.168.10.3	445	6	GPL NETBIOS SMB-DS IPC$ unicode share access
RT	2	filip-hp-15...	5.29400	2017-07-06 18:20:25	192.168.10.12	38018	52.23.111.175	3478	17	ET INFO Session Traversal Utilities for NAT (STUN Binding Request)
RT	2	filip-hp-15...	5.29401	2017-07-06 18:20:25	52.23.111.175	3478	192.168.10.12	38018	17	ET INFO Session Traversal Utilities for NAT (STUN Binding Response)
RT	2	filip-hp-15...	5.29467	2017-07-06 18:23:01	192.168.10.8	51700	192.168.10.50	3389	6	ET SCAN Behavioral Unusually fast Terminal Server Traffic Potential Scan or Infection (...
RT	2	filip-hp-15...	5.29468	2017-07-06 18:23:01	192.168.10.8	51700	192.168.10.50	3389	6	ET SCAN Behavioral Unusually fast Terminal Server Traffic Potential Scan or Infection (...
RT	6	filip-hp-15...	5.29611	2017-07-06 18:28:50	192.168.10.51	51801	192.168.10.3	53	17	ET DNS Query for .cc TLD

Slika 5.6. IDS događaji vezani uz skeniranje mreže

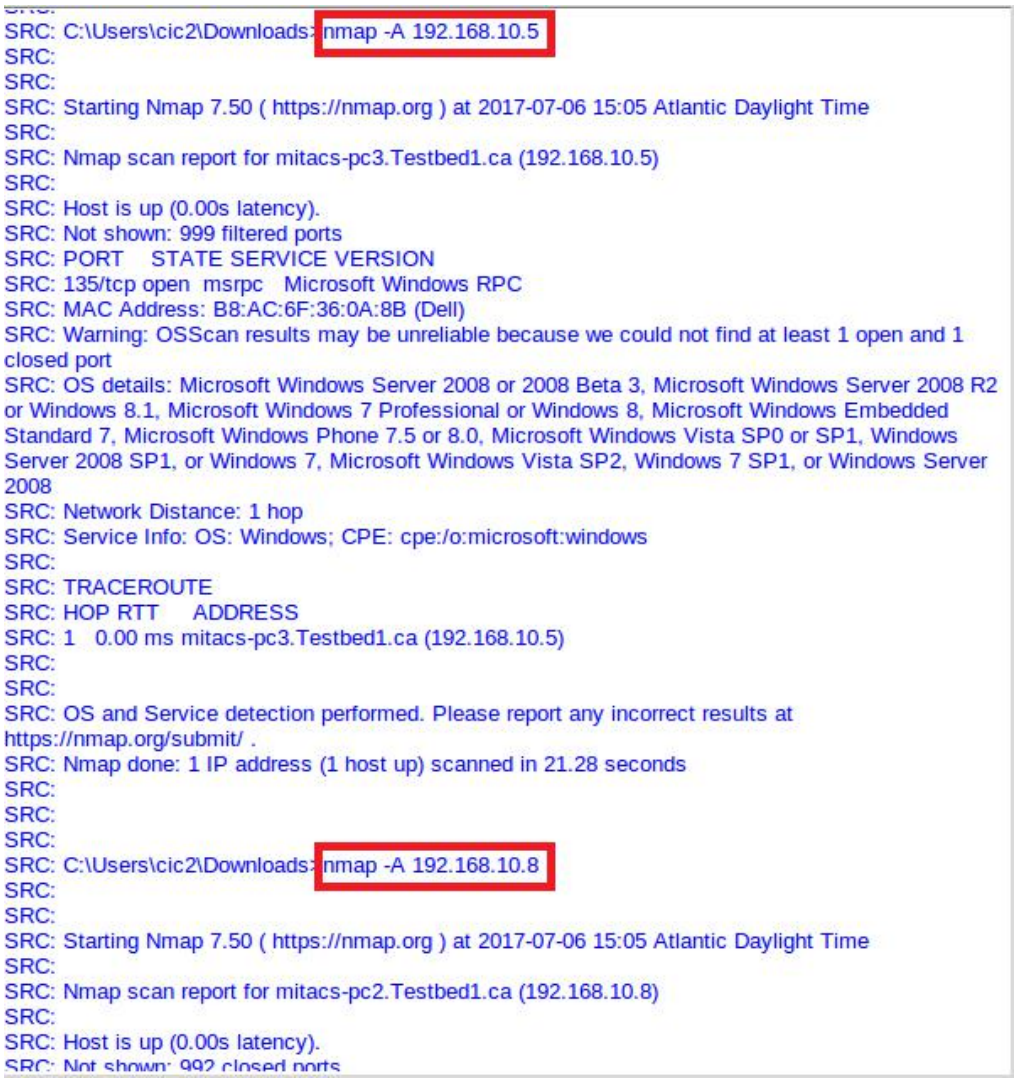
ST	CNT	Sensor	Alert ID	Date/Time △	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	1	filip-hp-15...	5.29026	2017-07-06 18:05:26	192.168.10.8	45257	192.168.10.5	44354	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29027	2017-07-06 18:05:27	192.168.10.8	45257	192.168.10.5	44354	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29028	2017-07-06 18:05:27	192.168.10.8	45257	192.168.10.5	44354	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29040	2017-07-06 18:08:12	192.168.10.8	45900	192.168.10.9	38372	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29041	2017-07-06 18:08:12	192.168.10.8	45900	192.168.10.9	38372	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29056	2017-07-06 18:09:10	192.168.10.8	35141	192.168.10.12	41776	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29057	2017-07-06 18:09:10	192.168.10.8	35141	192.168.10.12	41776	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29080	2017-07-06 18:10:05	192.168.10.8	40684	192.168.10.14	43157	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29081	2017-07-06 18:10:05	192.168.10.8	40684	192.168.10.14	43157	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29082	2017-07-06 18:10:06	192.168.10.8	40684	192.168.10.14	43157	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29083	2017-07-06 18:10:06	192.168.10.8	40684	192.168.10.14	43157	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29084	2017-07-06 18:10:06	192.168.10.8	40684	192.168.10.14	43157	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29085	2017-07-06 18:10:06	192.168.10.8	40684	192.168.10.14	43157	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29086	2017-07-06 18:10:06	192.168.10.8	40684	192.168.10.14	43157	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29087	2017-07-06 18:10:06	192.168.10.8	40684	192.168.10.14	43157	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29088	2017-07-06 18:10:08	192.168.10.8	40684	192.168.10.14	36576	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29089	2017-07-06 18:10:08	192.168.10.8	40684	192.168.10.14	36576	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29090	2017-07-06 18:10:09	192.168.10.8	40684	192.168.10.14	36576	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29091	2017-07-06 18:10:09	192.168.10.8	40684	192.168.10.14	36576	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29092	2017-07-06 18:10:09	192.168.10.8	40684	192.168.10.14	36576	17	ET SCAN NMAP OS Detection Probe
RT	1	filip-hp-15...	5.29093	2017-07-06 18:10:09	192.168.10.8	40684	192.168.10.14	36576	17	ET SCAN NMAP OS Detection Probe

Slika 5.7. Zaseban prikaz događaja povezanih s upozorenjem „SCAN NMAP OS detection Probe“

Sada možemo , kao i u prethodnom primjeru napada, zatražiti prijepis komunikacije između napadača i kompromitiranog računala. Prijepis se nalazi na Slici 5.8. i daje nam uvid u naredbe koje napadač izdaje kompromitiranom računalu.

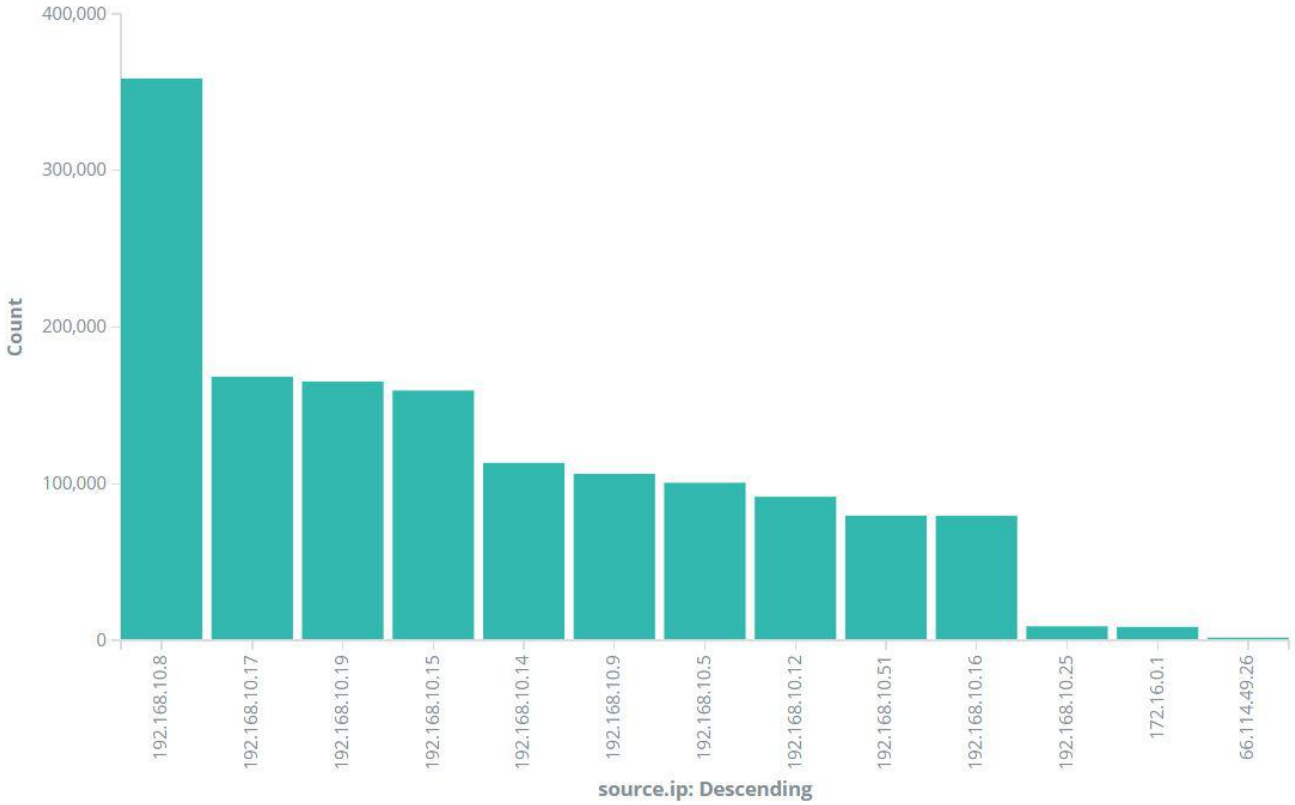
Slika 5.8. Prijepis komunikacije između napadača i kompromitiranog računala prvi dio

Prema Slici 5.8. prvo vidimo da je napadač uspostavio kontrolu nad žrtvinim računalom putem CLI nakon čega pokreće „aa.bat“ datoteku. Datoteka je „batch file“ tj. Skripta koja zadrži niz naredbi za izvršavanje na Microsoft Windows operacijskim sustavima. Prilikom pokretanja ona prvo pokreće Nmap alat za skeniranje mreže. Prvo horizontalno skenira cijeli opseg mreže 192.168.10.0/24 a nakon toga vertikalno skenira IP adrese za koje je dobio povratnu informaciju da su prisutne u mreži što je djelomično prikazano na Slici 5.9.



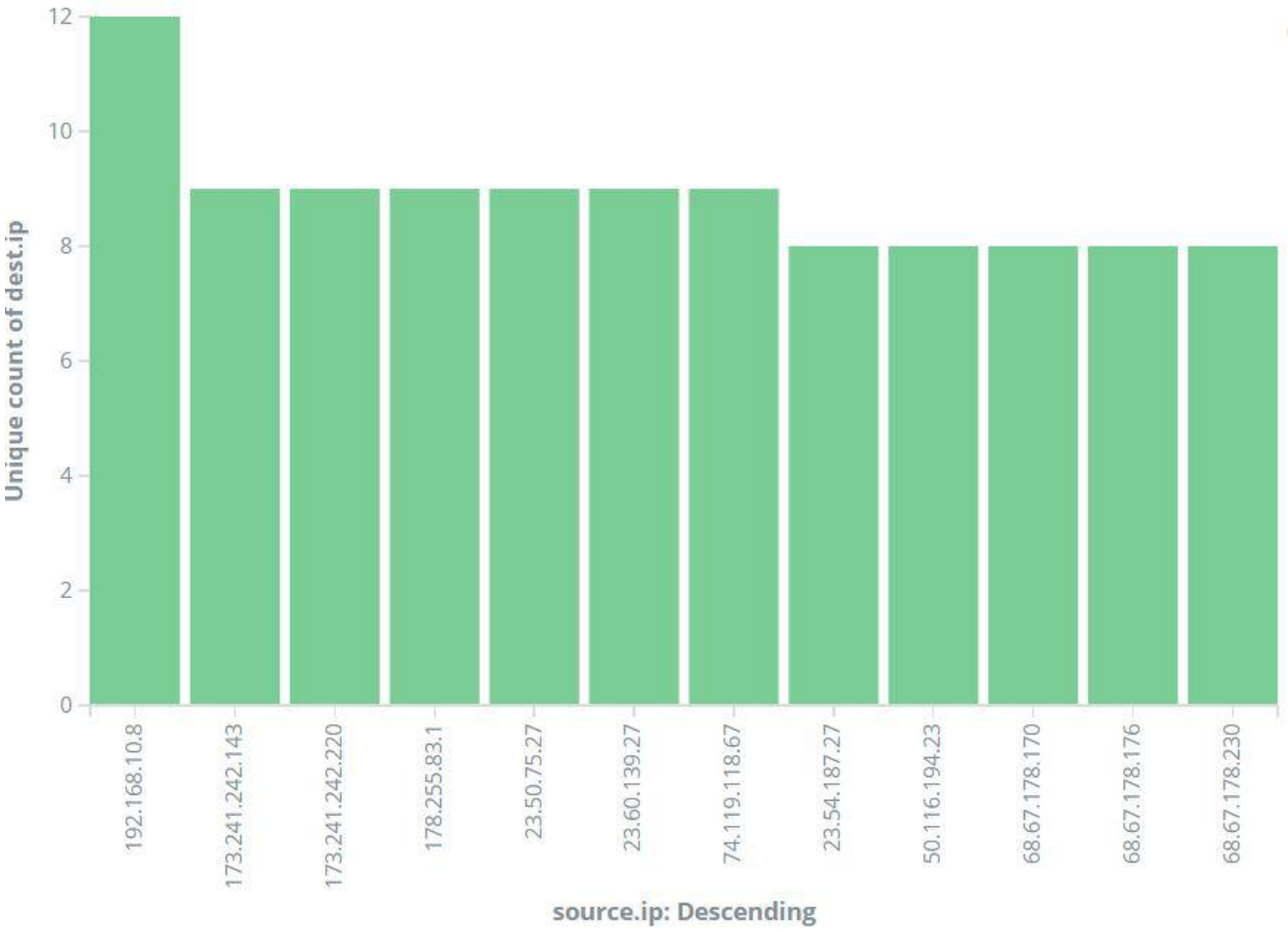
Slika 5.8. Prijepis komunikacije između napadača i kompromitiranog računala drugi dio

Indikatore skeniranja mreže moguće je vidjeti i na nekim grafovima koji su konstruirani na temelju nekih činjenica iz uvodnih razmatranja. Već je spomenuto kako ne postoji legitimna TCP sjednica koja sadrži manje od tri paketa, stoga konstruiramo li graf koji prikazuje broj TCP tokova u prometu koji imaju manje od tri paketa dobivamo prikaz sa Slike 5.9. Graf ističe kompromitirano računalo 192.168.10.8 kao ono koje je uspostavljalo dva do tri puta više puta više tokova s manje od tri paketa.



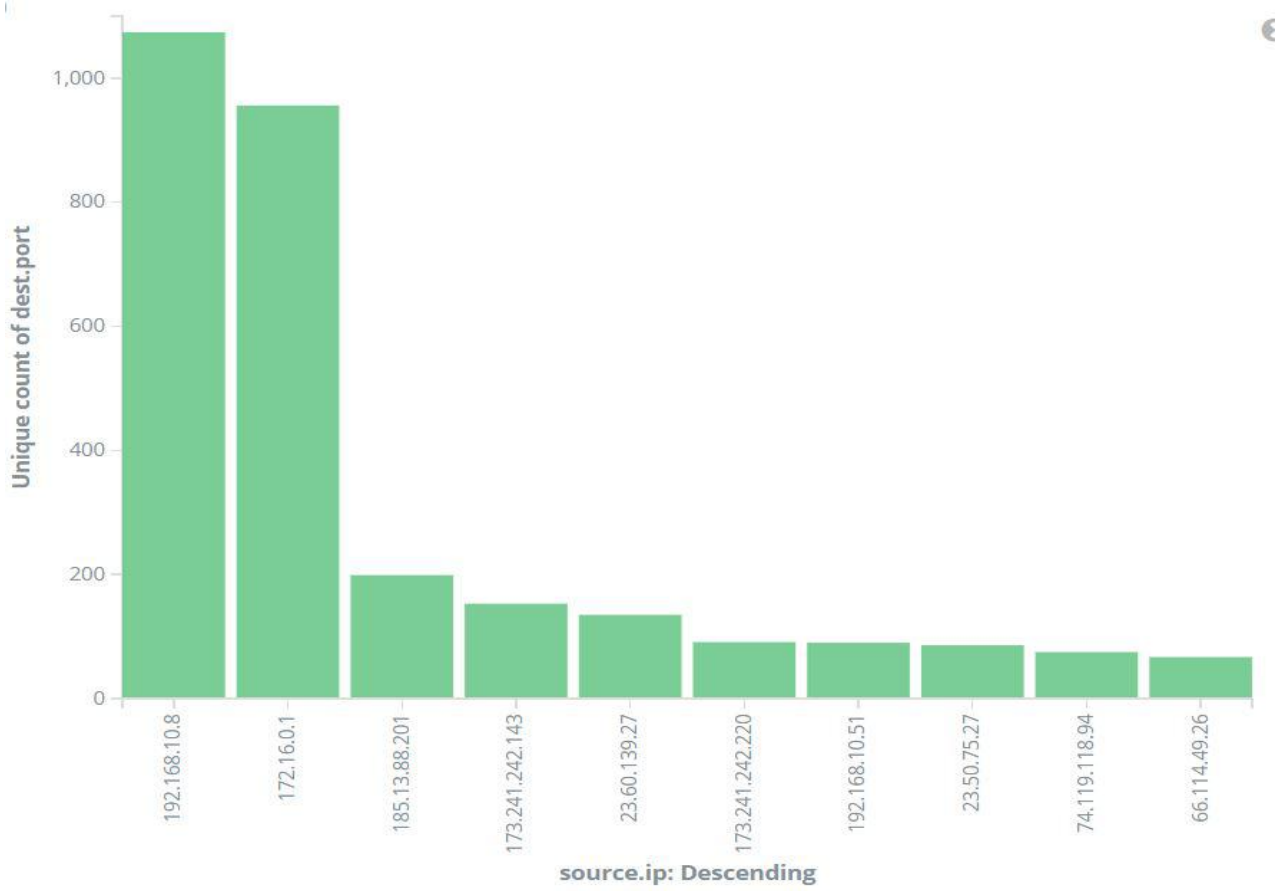
Slika 5.9. Raspodjela broja TCP tokova s manje od tri paketa po izvorišnim IP adresama

Iduću vizualizaciju izvesti ćemo iz sklonost entiteta u mreži da komuniciraju s malim brojem pretežno istih IP adresa i činjenice da napadači najčešće horizontalno skeniraju mrežu. Konstruiramo li graf na kojem pratimo s koliko računala iz lokalne mreže je neka od IP adresa, iz sveukupnog spektra IP adresa koje se pojavljuju u komunikaciji toga dana, uspostavila komunikaciju dobivamo graf sa Slike 5.10. Kao i u prethodnom slučaju kompromitirano računalo se ističe kao jedino koje je komuniciralo sa svim računalima u lokalnoj mreži tj. sa svim računalima.



Slika 5.10. Graf ovisnosti broja jedinstvenih odredišnih adresa iz lokalne mreže u ovisnosti o izvorišnim IP adresama

Napadači mogu mrežu skenirati i vertikalno kao što je učinjeno i u ovome primjeru. Pritom se događa anomalija komunikacije jedne IP adrese s puno odredišnih portova što nije uobičajeno. Koliko je to zapravo neuobičajeno vidljivo je sa grafa na Slici 5.11. koji prikazuje s koliko jedinstvenih odredišnih portova razgovaraju odredišne IP adrese u tokovima podataka.



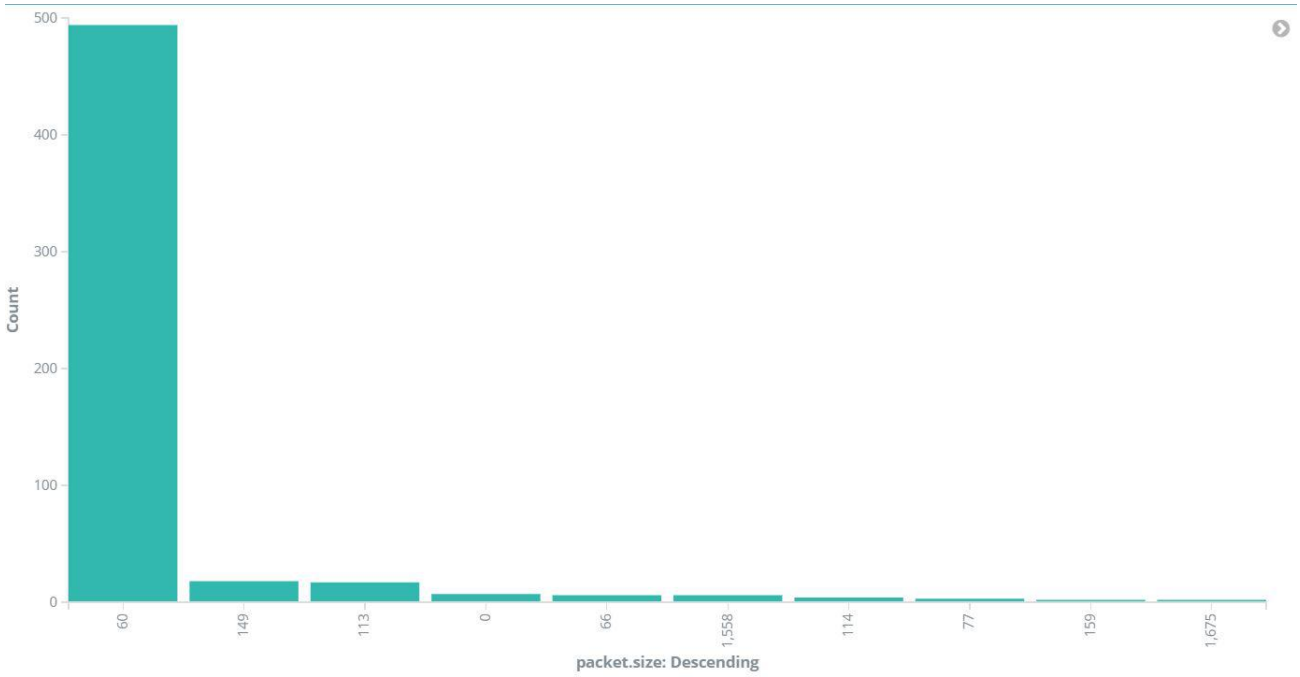
Slika 5.11. Graf raspodjele broja jedinstvenih odredišnih portova prema izvorišnim IP adresama

Bitno je primijetiti, prema Slici 5.1. da druga po redu IP adresa 172.16.0.1 predstavlja adresu izlaznog sučelja usmjerivača prema vanjskoj mreži te stoga posjeduje toliko velik broj različitih portova s kojima komunicira, a u usporedbi s drugim IP adresama komunicira s 5 do 10 puta više portova.

5.3.2. Odziv na infiltraciju preuzimanjem malicioznog dokumenta

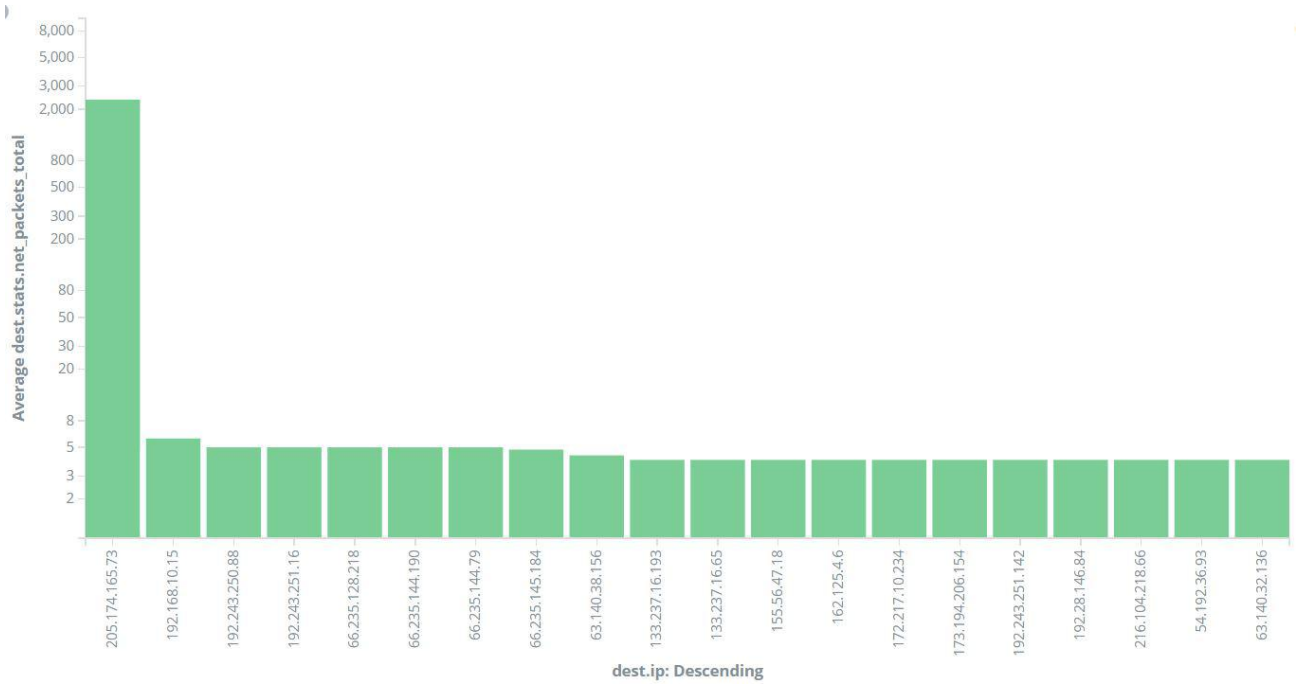
U fazi zaraženosti sustava kada je došlo doidentifikacije odziva prema malicioznoj domeni iz IDS alarma i indikatora skeniranja mreže iznutra, potrebno je blokirati javnu IP adresu koja izdaje naredbe kompromitiranom računalu. Kompromitirano računalo s IP adresom 192.168.10.8 potrebno je fizički isključiti iz mreže te provesti dublju forenziku na razini operacijskog sustava i memorije računala s ciljem otkrivanja drugih malicioznih radnji koje su prethodile ili bi tek mogle nastupiti te

izvesti čišćenje. Analizom mrežnog prometa zaključeno je da je komunikacija između maliciozne domene i kompromitiranog računala i sadržavala veliki broj paketa veličine 60 bajta. Razdiobu broja paketa po veličini za tokove kojima je odredišna adresa maliciozna domena 205.174.165.73 prikazana je na Slici 5.12.



Slika 5.12. Grafički prikaz razdiobe broja veličine paketa za odredišnu IP adresu 205.174.165.73

Na temelju ovoga zaključka uspoređen je prosječan broj paketa od 60 bajta u TCP tokovima za druge odredišne IP adrese te je konstruiran graf sa Slike 5.13. Graf pokazuje da je prosječan broj paketa veličine 60 bajta za malicioznu domenu oko 400 puta veći u odnosu na ostale odredišne IP adrese. Ovaj podatak se sada može iskoristiti proaktivno za buduću detekciju istog napada ukoliko se pojavi u varijaciji bez skeniranja mreže.



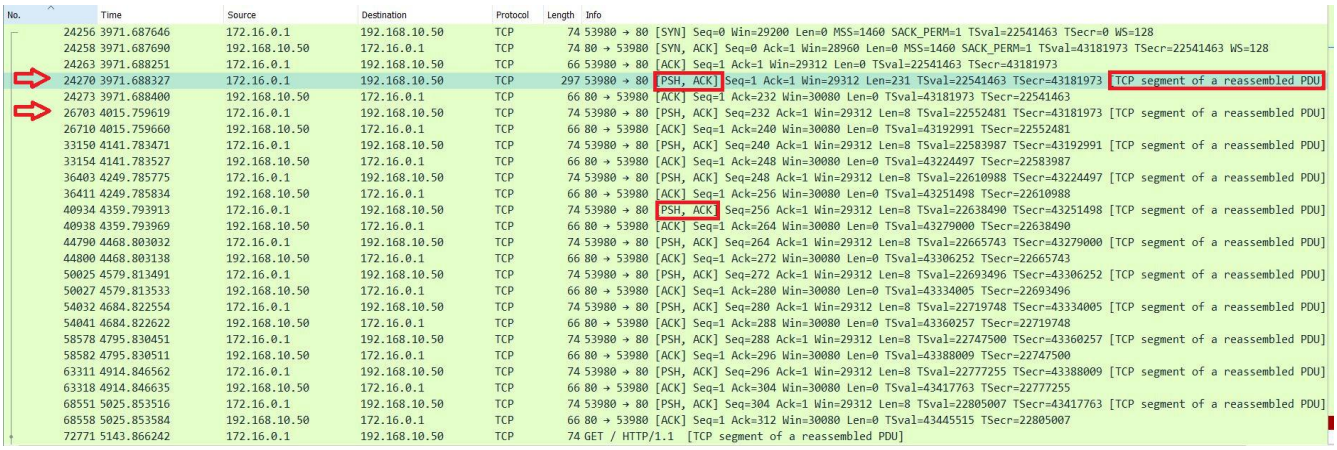
Slika 5.13. Raspodjela prosječnog broja paketa veličine 60 bajta po IP adresama

5.4. Slowloris napad

S obzirom na prije navedenu kategorizaciju DoS napada na prema načinu izvođenja, tj. iscrpljivanje na razini usluge, SYN poplava, iscrpljenje propusnosti i jednostavni fizički napadi, analizirana su dva tipa DoS napada. Prvi analizirani napad izveden je Slowloris alatom i predstavlja iscrpljivanje na razini usluge. Ovaj alat omogućava da se samo jednim računalom i uz veliku prikrivenost onemogući web poslužitelj. Efektivnost postiže istovremenim otvaranjem velikog broja djelomičnih HTTP zahtjeva koje nikada ne dovršava i pokušava ih održati otvorenim što je duže moguće slanjem djelomičnih zaglavlja zahtjeva zbog čega poslužitelj ne zatvara vezu jer „smatra“ kako je još uvijek legitimna ali je samo spora. Napad se kreće relativno sporo jer napadač mora čekati da se oslobode veze legitimnih korisnika kako bi ih preuzeo i iscrpio puni kapacitet. Zbog slanja djelomičnih a ne deformiranih zahtjeva, kao neki drugi alati, ostaje u potpunosti nezamijećen od strane IDS sustava.

5.4.1. Detekcija događaja i anomalija u mreži

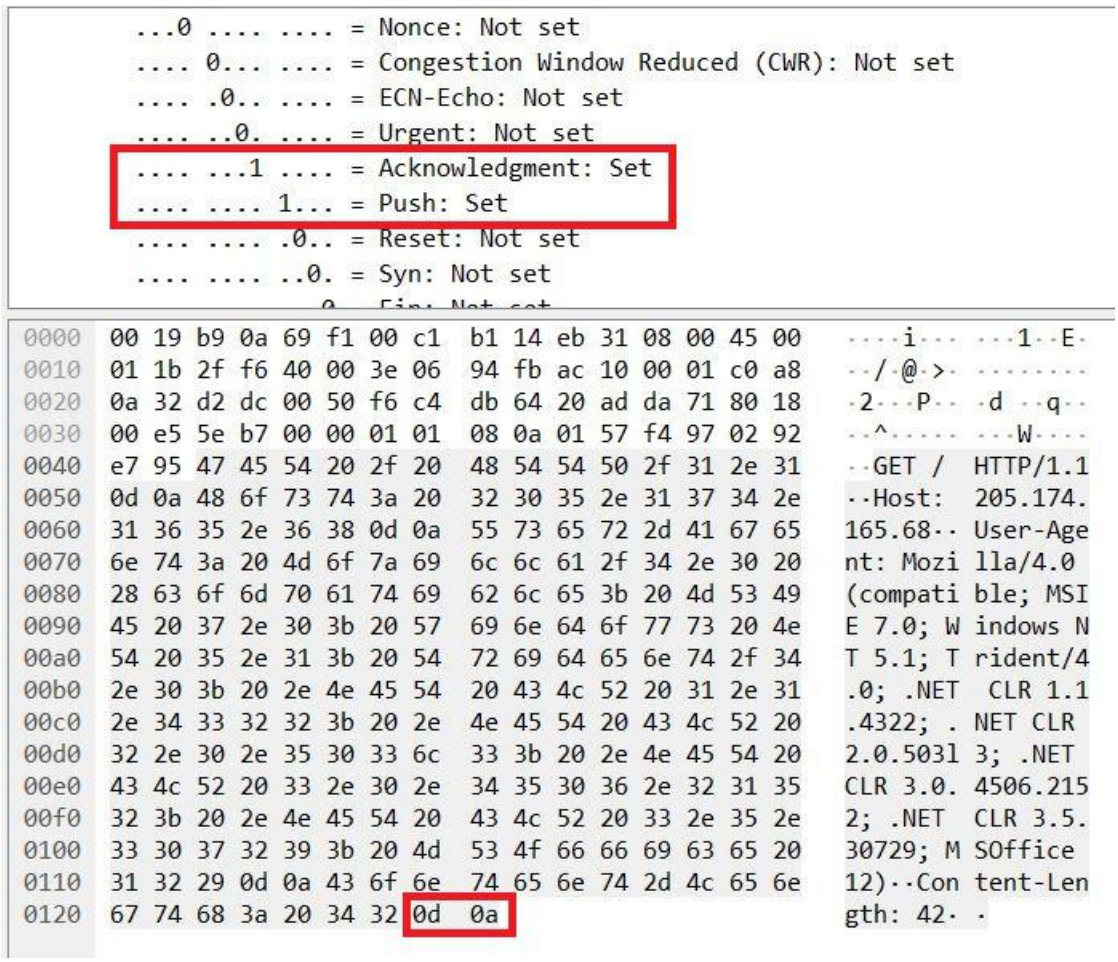
Snort IDS sustav nije detektirao nikakvu anomaliju vezanu za Slowloris napad. Ovakav napad je teško detektirati pomoću metapodataka o prometu. Uzimajući u obzir i jedinu anomaliju povezanu s ovim napadom, velik broj veza s jedne IP adrese potrebno je uzeti u obzir velike organizacije iz koje veliki broj korisnika Internetu pristupaju putem NAT (engl. *Network Adress Translation*). usmjerivača tj. sa jednom javnom adresom. Drugi razlog otežane detekcije ovog fenomena je pozicioniranje mrežnog senzora iza NAT usmjerivača lokalne mreže koju nadziremo pri čemu javne IP adrese mrežnih konekcija koje se uspostavljaju izvana prema unutra nisu dostupne. Analiza je stoga usmjerena otkrivanju slowloris napada na razini potpune mrežnog prometa.



No.	Time	Source	Destination	Protocol	Length	Info
24256	3971.687646	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=22541463 TSecr=0 WS=128
24258	3971.687690	192.168.10.50	172.16.0.1	TCP	74	80 → 53980 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=43181973 TSecr=22541463 WS=128
24263	3971.688251	172.16.0.1	192.168.10.50	TCP	66	53980 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=22541463 TSecr=43181973
24270	3971.688327	172.16.0.1	192.168.10.50	TCP	297	53980 → 80 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=231 TSval=22541463 TSecr=43181973 [TCP segment of a reassembled PDU
24273	3971.688400	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=232 Win=30080 Len=0 TSval=43181973 TSecr=22541463
26703	4015.759619	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [PSH, ACK] Seq=232 Ack=1 Win=29312 Len=8 TSval=22552481 TSecr=43181973 [TCP segment of a reassembled PDU]
26710	4015.759660	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=240 Win=30080 Len=0 TSval=43192991 TSecr=22552481
33150	4141.783471	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [PSH, ACK] Seq=240 Ack=1 Win=29312 Len=8 TSval=22583987 TSecr=43192991 [TCP segment of a reassembled PDU]
33154	4141.783527	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=248 Win=30080 Len=0 TSval=43224497 TSecr=22583987
36403	4249.785775	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [PSH, ACK] Seq=248 Ack=1 Win=29312 Len=8 TSval=22610988 TSecr=43224497 [TCP segment of a reassembled PDU]
36411	4249.785834	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=256 Win=30080 Len=0 TSval=43251498 TSecr=22610988
40934	4359.793913	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [PSH, ACK] Seq=256 Ack=1 Win=29312 Len=8 TSval=22638490 TSecr=43251498 [TCP segment of a reassembled PDU]
40938	4359.793969	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=264 Win=30080 Len=0 TSval=43279000 TSecr=22638490
44790	4468.803032	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [PSH, ACK] Seq=264 Ack=1 Win=29312 Len=8 TSval=22665743 TSecr=43279000 [TCP segment of a reassembled PDU]
44800	4468.803138	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=272 Win=30080 Len=0 TSval=43306252 TSecr=22665743
50025	4579.813491	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [PSH, ACK] Seq=272 Ack=1 Win=29312 Len=8 TSval=22693496 TSecr=43306252 [TCP segment of a reassembled PDU]
50027	4579.813533	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=280 Win=30080 Len=0 TSval=43334005 TSecr=22693496
54032	4684.822554	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [PSH, ACK] Seq=280 Ack=1 Win=29312 Len=8 TSval=22719748 TSecr=43334005 [TCP segment of a reassembled PDU]
54041	4684.822622	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=288 Win=30080 Len=0 TSval=43360257 TSecr=22719748
58578	4795.830451	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [PSH, ACK] Seq=288 Ack=1 Win=29312 Len=8 TSval=22747500 TSecr=43360257 [TCP segment of a reassembled PDU]
58582	4795.830511	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=296 Win=30080 Len=0 TSval=43388009 TSecr=22747500
63311	4914.846562	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [PSH, ACK] Seq=296 Ack=1 Win=29312 Len=8 TSval=22777255 TSecr=43388009 [TCP segment of a reassembled PDU]
63318	4914.846635	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=304 Win=30080 Len=0 TSval=43417763 TSecr=22777255
68551	5025.853516	172.16.0.1	192.168.10.50	TCP	74	53980 → 80 [PSH, ACK] Seq=304 Ack=1 Win=29312 Len=8 TSval=22805007 TSecr=43417763 [TCP segment of a reassembled PDU]
68558	5025.853584	192.168.10.50	172.16.0.1	TCP	66	80 → 53980 [ACK] Seq=1 Ack=312 Win=30080 Len=0 TSval=43445515 TSecr=22805007
72771	5143.866242	172.16.0.1	192.168.10.50	TCP	74	GET / HTTP/1.1 [TCP segment of a reassembled PDU]

Slika 5.14. Prikaz jednog toka podataka prema web poslužitelju tijekom slowloris napada pomoću alata Wireshark

Ono što se može primijetiti iz prometa je puno otvorenih dolaznih veza prema poslužitelju koje se ne zatvaraju tj. ne sadrže pakete sa FIN ili RST zastavicama. Prema Slici 5.14. prikazana komunikacija izgleda kao da se prenosi velika količina podataka kroz puno TCP segmenata a ono što se zapravo događa prikazano je na Slici 5.15..



Slika 5.15. Izgled paketa prilikom slowloris napada

Posebno skladani paket ima postavljene ACK i PSH zastavice te nedovršeni HTTP zahtjev. Prema RFC 2616 specifikaciji zaglavlje HTTP protokola mora završiti s 0d0a 0d0a, međutim u ovom slučaju napadač izostavlja jedan 0d0a i tim postiže da web poslužitelj nakon primitka ovog paketa i dalje očekuje posljednji 0d0a misleći da je veza spora. Slika 5.16. prikazuje komunikaciju između poslužitelja i napadača u tekstualnom obliku. Vidljivo je kako izgledaju prividna zaglavlja koja se šalju te glavna stavka napada tj. zadržavanje zauzete konekcije što se postiže slanjem istog paketa sa Slike 5.15 jedanaest puta prije nego server ogovori s porukom koda 400 – „Bad request“ i zatvori ju.

```
GET / HTTP/1.1
Host: 205.174.165.68
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; Trident/4.0; .NET CLR 1.1.4322; .NET CLR 2.0.503l3; .NET CLR 3.0.4506.2152;
.NET CLR 3.5.30729; MSOffice 12)
Content-Length: 42
X-a: b
X-a: b
X-a: b
X-a: b
X-a: b
X-a: b
X-a: b
X-a: b
X-a: b
X-a: b
X-a: b
HTTP/1.1 400 Bad Request
Date: Wed, 05 Jul 2017 12:49:12 GMT
Server: Apache/2.4.18 (Ubuntu)
Content-Length: 301
Connection: close
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
<p>Your browser sent a request that this server could not understand.<br />
</p>
<hr>
<address>Apache/2.4.18 (Ubuntu) Server at 127.0.1.1 Port 80</address>
</body></html>
```

Slika 5.16. Prijepis komunikacije između napadača i poslužitelja

5.4.2. Odziv na slowloris napad

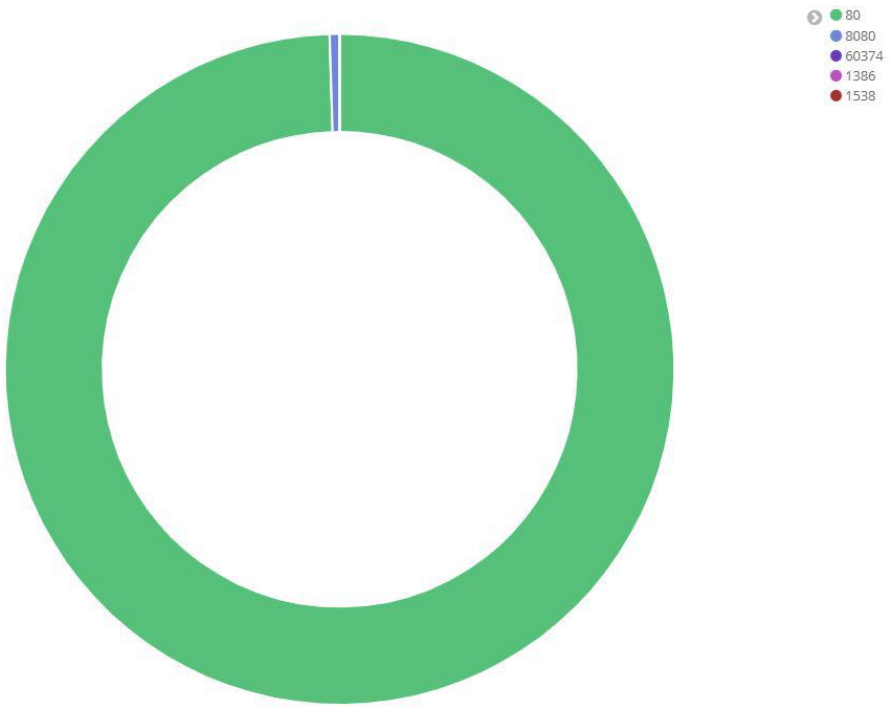
Uzevši u obzir da domena s koje se izvršava napad nije dostupna iz snimke prometa jer je mrežni senzor „zasljepljen“ NAT usmjerivačem potrebno je do javne IP adrese doći pregledavanjem NAT zapisa na usmjerivaču te ju blokirati. S obzirom da slowloris napad iskorištava ranjivost Apache poslužitelja pa pitanju čega nije moguće puno napraviti osim ograničiti broj konekcija s jedne IP adrese npr. na 50 (potrebno je pripaziti da se ne ugrožava pristup usluzi nekoj organizaciji iza NAT poslužitelja). Puno veću sigurnost omogućilo bi korištenje „reverse proxy“ poslužitelja s implementiranim nekim od Varnish, Nginx ili HAProxy alata koji će prevenirati dolazak ovakvih konekcija do primarnog Apache poslužitelja.

5.5. Mreža zaraženih računala

Botnet mreža sastoji se od niza povezanih računala, koja međusobno surađuju i kojima upravlja jedan napadač ili više napadača koji se naziva „botmaster“. Bot je krajnje računalo (ili poslužitelj), koji je član botnet mreže. Isto tako, taj se naziv koristi i za zlonamjerno oblikovane izvršne datoteke koje služe za dobivanje kontrole nad računalom i njegovo uključivanje u botnet mrežu. Računala su upravljana, najčešće putem IRC ili HTTP protokola, od strane neke zaražene javne domene koja se pod kontrolom napadača a ovakva struktura se često naziva i C2 ili C&C (engl. *Command and Control*). Tijek napada je sljedeći: napadač prvo postavlja C2 poslužitelj, zatim širi bot viruse koji uzrokuju povezivanje s C2 poslužiteljem i čekaju daljnje naredbe. Botnet-i se mogu koristiti za krađu podataka, širenje zaraze na druga računala, slanje neželjene elektroničke pošte, DDoS napade [16]. U ovome poglavlju prikazan je postupak detekcije i odziva na ARES botnet.

5.5.1. Detekcija mreže zaraženih računala

Ranije je već spomenut fenomen koji se naziva „beaconing“ a uzrokovan je periodičnom komunikacijom zaraženih računala s C2 poslužiteljem. Postoje brojne legitimne usluge koje redovno komuniciraju u pravilnim vremenskim intervalima, stoga je bitno pripaziti na veliki broj lažno pozitivnih alarma. Razmotrit ćemo neke činjenice vezane za HTTP protokol s obzirom da se često (kao i u slučaju ARES botneta) koristi za komuniciranje bot-ova s C2 poslužiteljem. Sukladno onome opisanome u poglavlju 4.4.3. i činjenici da web pretraživači imaju ugrađeno ograničenje da se spajaju na port 80 web poslužitelja osim ako to nije eksplicitno naglašeno u URL-u npr. www.ferit.hr:81 možemo zaključiti da će broj izlaznih HTTP veza prema nekom drugom portu biti sumnjiv ali ne i direktan indikator malicioznog prometa. S obzirom na ovu činjenicu od samoga početka bilo bi dobro staviti naglasak na HTTP promet koji je iz mreže upućen prema portovima koji nisu broj 80. Na Slici 5.17. možemo vidjeti i omjer događaja koji su povezani s HTTP prometom i poslužiteljskim portom. Port 80 povezan je s 99.52% događaja, port 8080 povezan je s 0.48% događaja dok je događaja vezanih uz ostale portove neznatno malo.



Slika 5.17. Grafički prikaz udjela broja događaja vezanih za HTTP promet na različitim poslužiteljskim portovima

Filtriramo li HTTP događaje sa poslužiteljskim i klijentskim portovima različitim od 80 tijekom analiziranog dana 7.7.2017. možemo uočiti veliku količinu pretežno istih događaja koji se ponavljaju za istu domenu 205.174.165.73 na portu 8080 i pet računala iz unutarnje mreže 192.168.10.8, 192.168.10.9, 192.168.10.14, 192.168.10.15, 192.168.10.5. Na računalo 205.174.165.73 pomoću GET zahtjeva šalje se dva parametra, botid i sysinfo koji zapravo predstavljaju „beaconing“ koji se odvija u pormjenjivim vremenskim razmacima što se može dogoditi kod naprednijih botneta. U komunikaciji se još mogu primjetiti POST /api/upload zahtjevi i povratne informacije C2 poslužitelju o sastavu direktorija kao i rezultati izvršavanja netstat naredbe koja prikazuje sve aktivne veze sa računalom koji se šalju sa POST /api/report. Primjer jednog od zahtjeva u kojem se šalje sastav direktorija je:

botid=mitacspc6&output=+Volume+in+drive+C+has+no+label.%0A+Volume+Serial+Number+is+FC9F-8423%0A%0A+Directory+of+C%3A%5CUsers%5Ccic1%5CDownloads%0A%0A07%2F07

%2F2017++10%3A03+AM++++%3CDIR%3E++++++++++.%0A07%2F07%2F2017++10%3A03+AM++++%3CDIR%3E++++++++++..%0A23%2F03%2F2017++02%3A50+PM++++%3CDIR%3E++++++++++2017-03-22%0A23%2F03%2F2017++05%3A22+PM++++%3CDIR%3E++++++++++2017-03-23%0A07%2F07%2F2017++10%3A03+AM++++++++17%2C951%2C465+agent.exe%0A22%2F03%2F2017++12%3A28+AM++++++++++112%2C284+baby.jpg.jpg%0A22%2F03%2F2017++03%3A14+PM++++++++15%2C749%2C664+DB.Browser.for.SQLite-3.9.1-win64.exe%0A21%2F03%2F2017++09%3A53+PM++++++++++889%2C416+dotNetFx40_Full_setup.exe%0A21%2F03%2F2017++09%3A52+PM++++%3CDIR%3E++++++++++DriveMakerSetup%0A21%2F03%2F2017++09%3A52+PM++++++++++397%2C684+DriveMakerSetup.zip%0A21%2F03%2F2017++08%3A57+PM++++++++47%2C754%2C208+eclipse-inst-win64.exe%0A21%2F03%2F2017++10%3A39+PM++++++++++219%2C159+FDinst.exe%0A22%2F03%2F2017++12%3A16+AM+++++++++7%2C028%2C976+FileZilla_3.25.1_win64-setup_bundled.exe%0A21%2F03%2F2017++08%3A58+PM+++++++205%2C004%2C856+jdk-8u121-windows-x64.exe%0A23%2F03%2F2017++08%3A51+PM++++%3CDIR%3E++++++++++New+folder%0A22%2F03%2F2017++03%3A17+PM+++++++++2%2C982%2C992+npp.7.3.3.Installer.exe%0A21%2F03%2F2017++09%3A05+PM+++++++++8%2C540%2C160+swish-0.8.2.exe%0A15%2F06%2F2017++09%3A08+AM++++++++++++++++79+windump+command.txt%0A22%2F03%2F2017++06%3A32+PM++++++++++569%2C344+WinDump.exe%0A22%2F03%2F2017++06%3A22+PM++++++++++915%2C128+WinPcap_4_1_3.exe%0A++++++++++++++14+File%28s%29++++308%2C115%2C415+bytes%0A++++++++++++++++6+Dir%28s%29++209%2C831%2C686%2C144+bytes+free%0A. Kada ga dekodiramo URL shemom kodiranja posebnih znakova iz poglavlja 5.3.1. dobivamo uredniji prikaz :

botid=mitacs-pc6&output= Volume in drive C has no label.

Volume Serial Number is FC9F-8423

Directory of C:\Users\cic1\Downloads

```
07/07/2017  10:03 AM    <DIR>          .
07/07/2017  10:03 AM    <DIR>          ..
23/03/2017  02:50 PM    <DIR>          2017-03-22
23/03/2017  05:22 PM    <DIR>          2017-03-23
07/07/2017  10:03 AM        17,951,465 agent.exe
22/03/2017  12:28 AM           112,284 baby.jpg.jpg
22/03/2017  03:14 PM        15,749,664 DB.Browser.for.SQLite-3.9.1-win64.exe
21/03/2017  09:53 PM           889,416 dotNetFx40_Full_setup.exe
21/03/2017  09:52 PM    <DIR>          DriveMakerSetup
21/03/2017  09:52 PM           397,684 DriveMakerSetup.zip
21/03/2017  08:57 PM        47,754,208 eclipse-inst-win64.exe
21/03/2017  10:39 PM           219,159 FDinst.exe
22/03/2017  12:16 AM         7,028,976 FileZilla_3.25.1_win64-setup_bundled.exe
21/03/2017  08:58 PM       205,004,856 jdk-8u121-windows-x64.exe
23/03/2017  08:51 PM    <DIR>          New folder
22/03/2017  03:17 PM         2,982,992 npp.7.3.3.Installer.exe
21/03/2017  09:05 PM         8,540,160 swish-0.8.2.exe
15/06/2017  09:08 AM                79 windump command.txt
22/03/2017  06:32 PM           569,344 WinDump.exe
22/03/2017  06:22 PM           915,128 WinPcap_4_1_3.exe
              14 File(s)    308,115,415 bytes
               6 Dir(s)  209,831,686,144 bytes free.
```

Slanje rezultata netstat naredbe :

botid=mitacs-pc3&output=%0AActive+Connections%0A%0A++Proto++Local+Address++++++++++Foreign+Address++++++++State%0A++TCP++++127.0.0.1%3A53448++++++++mitacs-pc3%3A53449+++++++ESTABLISHED%0A++TCP++++127.0.0.1%3A53449++++++++mitacs-pc3%3A53448+++++++ESTABLISHED%0A++TCP++++127.0.0.1%3A53450++++++++mitacs-pc3%3A53451+++++++ESTABLISHED%0A++TCP++++127.0.0.1%3A53451++++++++mitacs-pc3%3A53450+++++++ESTABLISHED%0A++TCP++++192.168.10.5%3A53486+++++93.184.216.172%3Ahttps+++ESTABLISHED%0A++TCP++++192.168.10.5%3A53568+++++lga25s60-in-f2%3Ahttps+++TIME_WAIT%0A++TCP++++192.168.10.5%3A53569+++++lga25s62-in-f14%3Ahttps++TIME_WAIT%0A++TCP++++192.168.10.5%3A53592+++++lga25s61-in-f2%3Ahttps+++TIME_WAIT%0A++TCP++++192.168.10.5%3A53594+++++qh-in-f154%3Ahttps+++++++TIME_WAIT%0A++TCP++++192.168.10.5%3A53595+++++lga34s14-in-f1%3Ahttps+++TIME_WAIT%0A++TCP++++192.168.10.5%3A53610+++++lga25s40-in-f2%3Ahttps+++TIME_WAIT%0A++TCP++++192.168.10.5%3A53656+++++89%3Ahttps++++++++++++++++TIME_WAIT%0A++TCP++++192.168.10.5%3A53707+++++216.200.232.114%3Ahttps++ESTABLISHED%0A++TCP++++192.168.10.5%3A53716+++++ip-73-165-174-205%3A8080++TIME_WAIT%0A++TCP++++192.168.10.5%3A53717+++++CIC-WEBSERVER%3Assh++++++TIME_WAIT%0A++TCP++++192.168.10.5%3A53718+++++CIC-WEBSERVER%3Assh++++++TIME_WAIT%0A++TCP++++192.168.10.5%3A53719+++++ip-73-165-174-205%3A8080++TIME_WAIT%0A++TCP++++192.168.10.5%3A53720+++++ip-73-165-174-205%3A8080++TIME_WAIT%0A++TCP++++192.168.10.5%3A53721+++++ip-73-165-174-205%3A8080++TIME_WAIT%0A.

Primjenimo li URL dekodiranje dobivamo sljedeće:

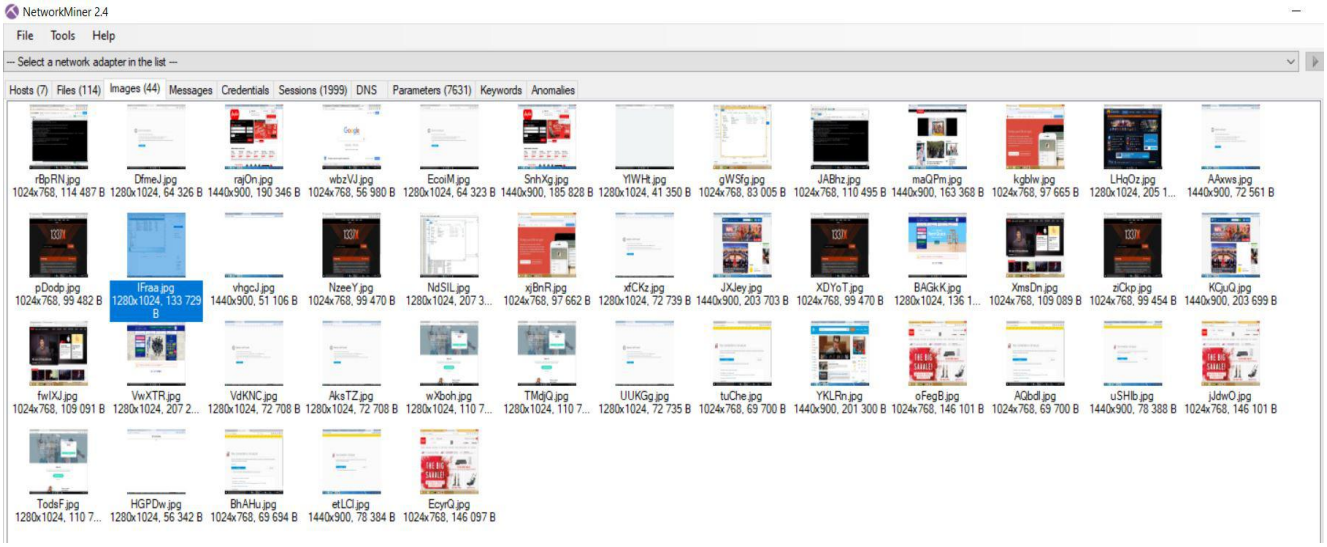
botid=mitacs-pc3&output=

Active Connections

Proto	Local Address	Foreign Address	State
TCP	127.0.0.1:53448	mitacs-pc3:53449	ESTABLISHED

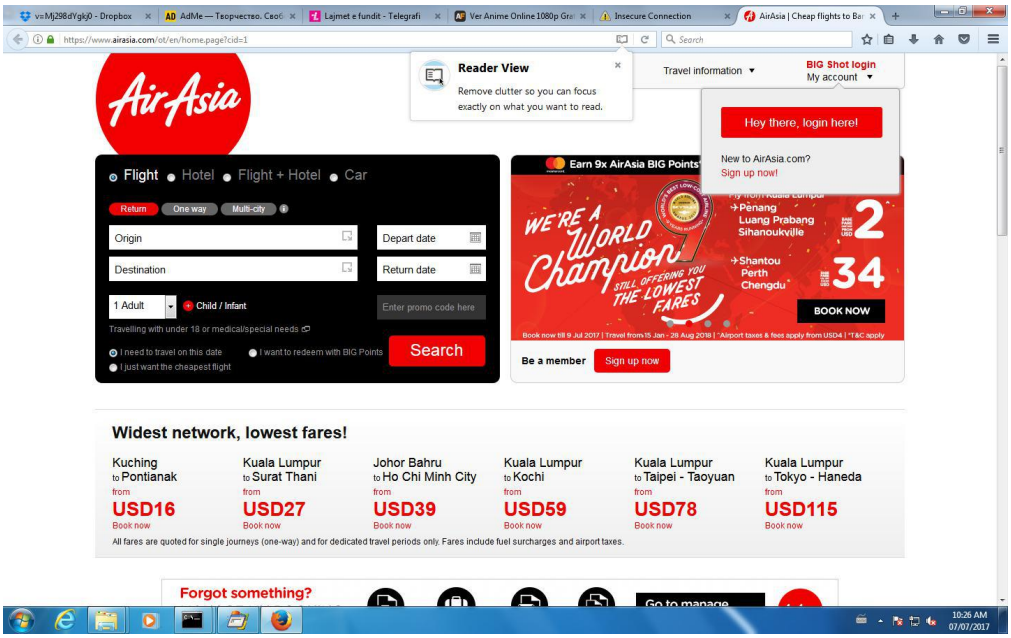
```
TCP    127.0.0.1:53449        mitacs-pc3:53448       ESTABLISHED
TCP    127.0.0.1:53450        mitacs-pc3:53451       ESTABLISHED
TCP    127.0.0.1:53451        mitacs-pc3:53450       ESTABLISHED
TCP    192.168.10.5:53486     93.184.216.172:https   ESTABLISHED
TCP    192.168.10.5:53568     lga25s60-in-f2:https   TIME_WAIT
TCP    192.168.10.5:53569     lga25s62-in-f14:https  TIME_WAIT
TCP    192.168.10.5:53592     lga25s61-in-f2:https   TIME_WAIT
TCP    192.168.10.5:53594     qh-in-f154:https       TIME_WAIT
TCP    192.168.10.5:53595     lga34s14-in-f1:https   TIME_WAIT
TCP    192.168.10.5:53610     lga25s40-in-f2:https   TIME_WAIT
TCP    192.168.10.5:53656     89:https               TIME_WAIT
TCP    192.168.10.5:53707     216.200.232.114:https  ESTABLISHED
TCP    192.168.10.5:53716     ip-73-165-174-205:8080  TIME_WAIT
TCP    192.168.10.5:53717     CIC-WEBSERVER:ssh      TIME_WAIT
TCP    192.168.10.5:53718     CIC-WEBSERVER:ssh      TIME_WAIT
TCP    192.168.10.5:53719     ip-73-165-174-205:8080  TIME_WAIT
TCP    192.168.10.5:53720     ip-73-165-174-205:8080  TIME_WAIT
TCP    192.168.10.5:53721     ip-73-165-174-205:8080  TIME_WAIT
```

Zbog POST /api/upload zahtjeva možemo pretpostaviti da bot-ovi šalju nekakve podatke na C2 server. Analizom prometa NetworkMiner alatom moguće je rekonstruirati datoteke koje se razmjenjuju u snimljenom prometu. Prema Slici 5.18. prikazane su fotografije koje su razmijenjene u komunikaciji.



Slika 5.18. Rekonstrukcija razmijenjenih datoteka pomoću NetworkMiner alata

Detaljnijom provjerom slike zaključuje se da se radi o snimkama ekrana (engl. screenshot) sa različitih računala, npr. Slika 5.19.



Slika 5.19. Primjer snimke ekrana koju je bot poslao na C2 server

5.5.2. Odziv na botnet napad

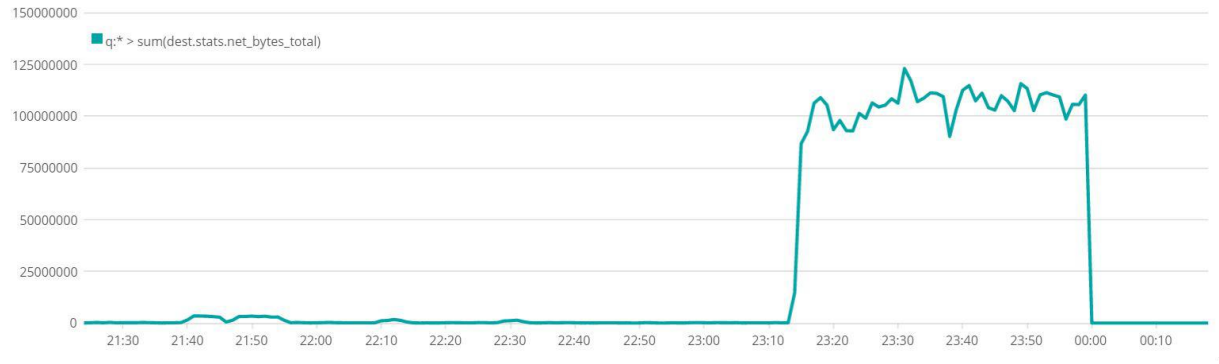
Na rubnim dijelovima mreže potrebno je blokirati IP adresu C2 poslužitelja. Zaražena računala fizički isključiti iz mreže i provesti čišćenje s ciljem uklanjanja botnet agenata i drugih potencijalno malicioznih datoteka i programa koje je botnet mogao preuzeti s neke druge domene. Isti napad može se ponovno pojaviti u istom obliku ali s druge IP adrese. U tom slučaju potrebno je filtrirati promet i pokrenuti alarme prilikom pojave nekog od karakterističnih HTTP zahtjeva koje ovaj botnet šalje; POST /api/upload ili POST /api/report.

5.6. DoS napad iscrpljivanjem resursa

Evaluacija ovog i sljedećeg napada izvršena je u samostalno napravljenom okruženju računalne mreže. Povezana 3 računala pri čemu će dva biti kompromitirana dok je jedno računalo žrtva. Nadzor je proveden „port mirroring“ funkcijom Mikrotik Cloud preklopnika i računalom sa Security Onion distribucijom Linx operacijskog sustva i više će se bazirati na detekciji Bro(Zeek) mrežnom senzoru/IDS – u. DoS napad izveden je UDP unicorn alatom za Windows operacijski sustav.

5.6.1. Detekcija DoS napada iscrpljivanjem resursa

DoS napad iscrpljivanjem resursa iscrpljulje količinu podataka koju je neko računalo, usmjerivač ili poslužitelj mogu primiti i obraditi. Temelji se na masovnom generiranju besmislenog prometa kao što je velika količina paketa sa SYN zastavicama (engl. SYN flood). U ovisnosti količine prometa o vremenu ovaj DoS napad manifestira se jasnim početkom i krajem napada s obzirom da se radi o mehničkom napadu pri kojem napadač samo pokreće ili zaustavlja napad kao što je prikazano u primjeru na Slici 5.20.



Slika 5.20. DoS napad iscrpljivanjem resursa

Bro mrežni senzor a ujedno i IDS prati anomalije u mreži i u usporedbi sa Snort IDS-om koji se bazira na potpisima može upozoriti na temelju pojave velike količine besmislenog prometa. Primjer upozorenja je prikazan na Slici 5.21. Vidljiva je i velika količina upozorenja u kratkom vremenskom razdoblju što igra bitnu ulogu u utvrđivanju lažno pozitivne detekcije. Osim upozorenja sa Slike 5.21. o tome kako se promet šalje bez prethodne uspostave veze, Bro upozorava i s porukama „DNS_Conn_count_too_large“ tj. prevelik broj DNS konekcija i „Connection attempt seen, no reply“ što možemo povezati sa SYN flood napadom. U principu UDP unicorn je konfiguriran da šalje pakete vezane uz više različitih protokolazato je moguće vidjeti upozorenja za različite tipove prometa. Volumetrički napadi titpično se izvode putem botneta ili dobrovoljnim sudjelovanjem veće broja računala s obzirom da je za velike distribuirane sustave to i potrebno. U ovome slučaju možemo razmotriti da je napad jednim računalom uzorkovao vrlo sporo učitavanje web stranica koje je nekad uzrokovalo prekid učitavanja. U napadu je došlo i do kolateralnih žrtava, prvotno mrežnog usmjerivača koji nije mogao podnijeti toliku količinu prometa, a zatim je usluga ometana i ostalim računalima u mreži.

Time	event_type	ips	connection_state_description
September 6th 2019, 16:23:49.354	bro_conn	192.168.1.5, 192.168.1.21	No SYN seen, just midstream traffic (a 'partial connection' that was not later closed)
September 6th 2019, 16:23:49.353	bro_conn	192.168.1.5, 192.168.1.21	No SYN seen, just midstream traffic (a 'partial connection' that was not later closed)
September 6th 2019, 16:23:49.348	bro_co	192.168.1.5, 192.168.1.21	No SYN seen, just midstream traffic (a 'partial connection' that was not later closed)
September 6th 2019, 16:23:49.343	bro_conn	192.168.1.5, 192.168.1.21	No SYN seen, just midstream traffic (a 'partial connection' that was not later closed)
September 6th 2019, 16:23:49.340	bro_conn	192.168.1.5, 192.168.1.21	No SYN seen, just midstream traffic (a 'partial connection' that was not later closed)
September 6th 2019, 16:23:49.339	bro_conn	192.168.1.5, 192.168.1.21	No SYN seen, just midstream traffic (a 'partial connection' that was not later closed)

Slika 5.21. Jedna od reakcija Bro IDS-a na volumetrički DoS napad

5.6.2. Odziv na DoS napad

Prije svega potrebno je blokirati IP adresu izvorišnog računala. Moguće je povećati kapacitet mreže kako se ovakav napad ne bi mogao izvesti tj. barem ne samo jednim računalom.

5.7. Napad otkrivanja lozinke pomoću rječnika

Ovakvi napadi izvode se pomoću aplikacija za otkrivanje lozinki ili aplikacija za napade grubom silom i rječnika koji sadrži sve lozinke koje će se isprobati dok se ne dođe do prave. Cilj napada je stjecanje kontrole korisničkim računom na nekom servisu ili aplikaciji. Napad je izveden pomoću Medusa alata i rječnikom koji se sastoji od 500 najčešće korištenih lozinki, a usmjeren je na SSH server. Prikaz izvođenja napada nalazi se na Slici 5.22.

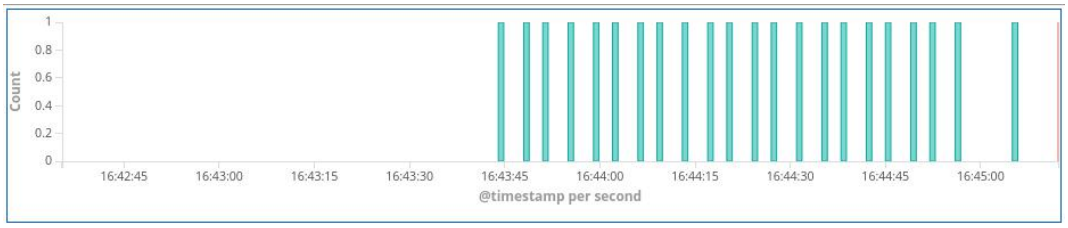
```
root@kali:~/Desktop# medusa -u Alice -P top-500-pass.txt -h 192.168.1.21 -M ssh
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.n
et>

ACCOUNT CHECK: [ssh] Host: 192.168.1.21 (1 of 1, 0 complete) User: Alice (1 of 1
, 0 complete) Password: 123456 (1 of 500 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.21 (1 of 1, 0 complete) User: Alice (1 of 1
, 0 complete) Password: password (2 of 500 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.21 (1 of 1, 0 complete) User: Alice (1 of 1
, 0 complete) Password: 12345678 (3 of 500 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.21 (1 of 1, 0 complete) User: Alice (1 of 1
, 0 complete) Password: 1234 (4 of 500 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.21 (1 of 1, 0 complete) User: Alice (1 of 1
, 0 complete) Password: pu[redacted] (5 of 500 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.21 (1 of 1, 0 complete) User: Alice (1 of 1
, 0 complete) Password: 12345 (6 of 500 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.21 (1 of 1, 0 complete) User: Alice (1 of 1
, 0 complete) Password: dragon (7 of 500 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.21 (1 of 1, 0 complete) User: Alice (1 of 1
, 0 complete) Password: qwerty (8 of 500 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.21 (1 of 1, 0 complete) User: Alice (1 of 1
, 0 complete) Password: 696969 (9 of 500 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.21 (1 of 1, 0 complete) User: Alice (1 of 1
, 0 complete) Password: mustang (10 of 500 complete)
```

Slika 5.22. Izvođenje napada alatom Medusa

5.7.1. Detekcija napada riječnikom

Bro mrežni senzor unutar datoteke zapisa pod nazivom ssh.log posebno sprema anomalije vezane za SSH protokol. Na Slici 5.23. prikazani su zapisi iz ssh.log datoteke u kojoj je zabilježen svaki pokušaj proboja u korisnički račun.



Time	event_type	ips
September 6th 2019, 16:45:05.707	bro_ssh	192.168.1.16, 192.168.1.21
September 6th 2019, 16:44:56.600	bro_ssh	192.168.1.16, 192.168.1.21
September 6th 2019, 16:44:52.982	bro_ssh	192.168.1.16, 192.168.1.21
September 6th 2019, 16:44:49.329	bro_ssh	192.168.1.16, 192.168.1.21
September 6th 2019, 16:44:45.754	bro_ssh	192.168.1.16, 192.168.1.21
September 6th 2019, 16:44:42.183	bro_ssh	192.168.1.16, 192.168.1.21
September 6th 2019, 16:44:38.612	bro_ssh	192.168.1.16, 192.168.1.21
September 6th 2019, 16:44:35.024	bro_ssh	192.168.1.16, 192.168.1.21
September 6th 2019, 16:44:31.469	bro_ssh	192.168.1.16, 192.168.1.21

Slika 5.23. Zapisi Bro IDS-a o pokušajima povezivanja na SSH server

Za indikaciju kompromisa bitno je da pokušaji dolaze sa iste IP adrese u kratkom vremenskom razdoblju. Unutar polja „authentication_attempts“ Bro bilježi broj pokušaja autentifikacije sa neke IP adrese.

5.7.2. Odziv na napad rječnikom

Nakon blokiranja IP adrese, potrebno je korisnike obavijestiti o napadu i zamoliti ih da promjene svoje lozinke. Korisnicima se mogu dati preporuke za izradu sigurne lozinke u smislu što su loše a što dobre lozinke. Primjeri savjeta što su loše lozinke su : lozinke koje sadrže slova, posebne

znakove i brojeve, lozinke koje sadrže samo brojeve, lozinke koje sadrže samo posebne znakove(&?*…), lozinke koje sadrže samo male ili samo velike znakove, lozinke koje sadrže samo slova/brojeve i posebne znakove te lozinke koje su kraće od jedanaest znakova [17].Na uslugama koje se pružaju potrebno je ograničiti broj mogućih krivih unosa lozinke na otprilike 3 puta kako bi se dopustio prostor za pogrešku ali opet spriječilo automatizirano isprobavanje lozinke.

6. ZAKLJUČAK

U ovome diplomskom radu cilj je bio provesti nadzor računalne mreže u svrhu detekcije i analize računalnih napada te oporavka nakon istih. Za analizu su korišteni podaci generirani u svrhu evaluacije IDS sustav pod nazivom „CICIDS2017“ i podaci generirani u vlastitoj lokalnoj mreži. Odabrano je sedam različitih tipova napada. Za svaki napad opisan je postupak detekcije, otpora, odziva i planiranja. Kvalitetni mrežni nadzor provodi se pomoću IDS sustava, zapisa tokova prometa i potpune snimke prometa. Podatke je potrebno brzo obrađivati s ciljem pravovremenog odziva, stoga su spremljeni u Elasticsearch bazu podataka, te analizirati i vizualizirati pomoću Kibana sučelja ili obrađivati drugim alatima kao što su Wireshark, Sguil i NetworkMiner. XSS, infiltracija preuzimanjem malicioznog dokumenta, Slowloris i ARES botnet napad izvdeni su u sklopu „CICIDS2017“ kolekcije podataka dok su volumetrički DoS i napad otkrivanja lozinke rječnikom provedeni u vlastito izrađenoj lokalnoj mreži.

Prvi napad usmjeren je prema web aplikaciji koja je ranjiva na reflektirani XSS napad. Napad je uspješno detektiran od strane IDS sustava okidanjem alarma za prisutnost „<script></script>“ oznaka u URL-u. U napadu je pokazano kako je moguće pristupiti kolačićima u pretraživaču ali nije napravljen korak dalje, gdje se ti podaci šalju napadaču što predstavlja nedostatak u izvršavanju napada. Zaustavljanje i prevencija reflektiranog XSS napada odvijaju se istovremeno s obzirom da je potrebno provesti validaciju podataka koji ulaze u aplikaciju. Bitno je naglasiti da je validaciju potrebno obaviti na strani poslužitelja.

Napad infiltracijom preuzimanjem malicioznog dokumenta teško je detektirati prije nego napadač poduzme neke druge akcije. U ovome primjeru nakon infiltracije, napadač izvodi naredbe s vanjske IP adrese što Snort IDS sustav detektira i kategorizira kao Trojanskog konja. Uspješna detekcija Snort IDS-om postignuta je i zbog napadačevog skeniranja mreže Nmap alatom. Iako je maliciozna IP adresa blokirana, napad se može ponoviti s istim programskim alatima ali s drugim tijekom događaja, stoga je u svrhu detekcije odmah nakon zaraze promet dublje analiziran. Analizom je utvrđena specifičnost velike količine paketa veličine 60 bajta u relativno kratkom vremenskom intervalu kao indikator buduće kompromitacije ovim tipom malicioznog koda.

Napad koji je zasigurno najteži za detekciju je slowloris napad. Ovaj napad posjeduje visoku razinu prikrivenosti zbog svog sporog napredovanja i vrlo vjerojatno će se detektirati tek nakon korisnici primijete uskraćivanje usluge. Analizom potpune snimke prometa utvrđena su indikatori kompromitacije kao što je struktura paketa i izgled sjednice. Karakteristični su paketi s postavljenim ACK i PSH zastavicama i HTTP zahtjevom koji izostavlja formalni završetak paketa i na taj način obmanjuje poslužitelj da je veza spora. Slowloris je karakteristična ranjivost Apache poslužitelja i može se izbjeći korištenjem „reverse proxy“ poslužitelja koji je otporan na ovaj napad.

Botnet zaraza u mreži detektirana je poznavanjem anomalija vezanih za dodjelu broja portova u mreži. Pretpostavka je da će većina legitimnih poslužiteljskih HTTP usluga raditi na portu 80, što je analizom i dokazano ali se ujedno i istaknula količina prometa na sporednom portu 8080. Port 8080 obično se dodjeljuje za HTTP posluživanje kada se ne posjeduju administratorske ovlasti na sustavu. Daljnjom analizom utvrđene su pojave periodičkog odziva sa pet računala prema malicioznoj domeni. Analizom prometa utvrđena je razmjena informacija o aktivnim mrežnim konekcijama i sadržaju direktorija. Zbog sumnjivih HTTP zahtjeva NetworkMiner alatom izrađena je rekonstrukcija razmijenjenih datoteka i utvrđeno slanje snimki ekrana zaraženih računala. S ciljem buduće prevencije predloženo je praćenje HTTP zahtjeva koje se šalju prema C2 poslužitelju.

DoS napad detektiran je Bro IDS-om koji detekciju temelji na anomalijama. S obzirom da se pri izvođenju ovakvog DoS napada koji za cilj ima iscrpljivanje resursa generira puno prometa bez nekakvog smisla, Bro IDS registrirao je pakete koji dolaze bez prethodno pravilno uspostavljene veze kao i pakete koji uspostavljaju vezu SYN zastavicom bez daljnjih akcija. Napad je utjecao i na usmjerivač a tako i na sva ostala računala u mreži koja predstavljaju kolateralne žrtve. Napad je teško moguće prevenirati ali povećanjem kapaciteta mreže može se izbjeći uskraćivanje usluge korištenjem manjeg broja računala .

Zbog vrlo čestog korištenja početnih postavki i neozbiljnog shvaćanja lozinki napad rječnikom koji sadrži nekakav popis popularnih lozinki može biti vrlo efektivan. Provedeni pokušaj otkrivanja i krađe računa na SSH poslužitelju uspješno je detektiran Bro IDS-om koji je zabilježio svaki pokušaj povezivanja kao i broj pokušaja povezivanja. Daljnja prevencija se provodi izradom kvalitetnih lozinki i ograničavanjem broja krivih unosa.

Slowloris napad je jedini analiziran i identificiran samo na razini snimke prometa jer se ne manifestira u metapodacima ili IDS alarmima. Svi ostali tipovi napada uspješno su detektirani pomoću istraživačke analize podataka i Bro ili Snort IDS sustava. Za svaki napad predložene su smjernice za oporavaka i sprječavanje ponovnog incidenta.

LITERATURA

[1] C. Z., Zimmerman, Ten Strategies of a World-Class Cybersecurity Operations Center, MITRE corporation, 2014.

[2] M. C., Collins, Network Security Through Data Analysis: From Data to Action, O'Reilly Media, Inc.2017.

[3] J, Bollinger, B. Enright, M. Valites, Crafting the InfoSec Playbook, O'Reilly Media, 2015.

[4] S. A., Axelsson, The Base-Rate Fallacy and its Implications for the Difficulty of Intrusion Detection, Detection Recent Advances in Intrusion, 1999.

[5] R. B., Bejtlich, The practice of Network Security Monitoring, No starch press, 2013.

[6] Committee on National Security Systems, "CNSS Instruction No. 4009," Committee on National Security Systems, Ft. Meade, 2010.

[7] I. B., Butun, S. D. M., Morgera, R. S., Sankar, A survey of intrusion detection systems in wireless sensor networks, IEEE Commun. Surveys Tuts., vol. 16, no. 1, pp. 266–282, 1st Quart., 2014.

[8] E.B., Benkhelifa, T.W.,Welsh, W. H., Hamouda, A Critical Review of Practices and Challenges in Intrusion Detection Systems for IoT: Toward Universal and Resilient Systems, IEEE communications surveys & tutorials, vol. 20, no. 4, fourth quarter 2018,

[9] S.A.R.S., Raza Shah, B. I., Biju, Intelligent intrusion detection system through combined and optimized machine learning, International Journal of Computational Intelligence and Applications, March 2018

[10] D.P., Papamartzivanos, F.G., Gomez, G.K., Kambourakis, Introducing Deep Learning Self-Adaptive Misuse Network Intrusion Detection Systems, 2018

[11] J. Dykstra, Essential Cybersecurity Science, O'Reilly Media, 2015.

[12] S.J. Son, Y. Kwon , Performance of ELK stack and commercial system in security log analysis, IEEE 13th Malaysia International Conference on Communications (MICC), 2017

[13] I. Sharafaldin, A. H. Lashkari,A. A. Ghorbani, Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization, 4th International Conference on Information Systems Security and Privacy (ICISSP), Portugal, 2018.

[14] https://www.owasp.org/index.php/Cross-site_Scripting_(XSS), pristupljeno 14.8. 11:12

[15] https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html, pristupljeno 14.8. 12:15

[16] Carnet, Botnet mreže, 2007.

[17] S.P. Oriyano, Penetration Testing Essentials, Sybex 2017.

SAŽETAK

U radu je izvršena detekcija događaja u mreži korištenjem Snort i Bro IDS – ova i raznih vizualizacija i obrada podataka dobivenih Packetbeat, Kibana, Sguil te NetworkMiner programskim alatima. Podaci su spremljeni u Elasticsearch bazu podataka i vizualizirani pomoću Kibana sučelja. Analizirano je šest različitih napada: XSS napad, napad infiltracijom preuzimanjem malicioznog dokumenta, Slowloris napad, napad ARES botnet-om, koji su generirani unutar CICIDS2017 kolekcije podataka i volumetrički DoS napad te napad otkrivanja lozinke rječnikom, generirani u vlastitoj lokalnoj mreži. Nakon detekcije i analize predložene su smjernice koje je potrebno slijediti za zaustavljanje i daljnju prevenciju napada.

KLJUČNE RIJEČI

Kibernetička sigurnost, IDS, incident, XSS, DoS, testiranje, analiza, mrežni nadzor

ABSTRACT

In this thesis network security monitoring was performed using Snort and Bro IDS, different visualizations and data processing using Packetbeat, Kibana, NetworkMiner, Wireshark and Sguil software. Data was stored in an Elasticsearch database and visualized or processed with Kibana interface. Six different attacks were analysed: XSS attack, Infiltration attack using malicious document, Slowloris attack, ARES botnet attack – which were generated in CICIDS2017 dataset, volumetric DoS attack and password cracking dictionary attack – which were generated in my own local network. After detection and analysis, policies and suggestion for further prevention and mitigation of the same attack were given.

KEYWORDS

Cybersecurity, IDS, incident, DoS, XSS, testing, analysis, network surveillance

ŽIVOTOPIS

Filip Žagar rođen je 2.3.1996. godine u Vinkovcima. Nakon završene Opće gimnazije u Vinkovcima, 2014. godine upisuje preddiplomski studij elektrotehnike na Fakultetu Elektrotehnike, Računarstva i Informacijskih Tehnologija u Osijeka. Tijekom preddiplomskog studija aktivno sudjeluje u nastavi u ulozi demonstratora na kolegijima Programiranje 2 i Elektronika 2 te dobiva priznanje za nagrađeni studentski rad povodom održavanja 39. godišnjice fakulteta. Preddiplomski studij završava 2017. godine i iste godine upisuje diplomski studij elektrotehnike, smjer komunikacije i informatika na Fakultetu Elektrotehnike, Računarstva i Informacijskih Tehnologija u Osijeka. Tijekom diplomskog studija aktivno sudjeluje u nastavi u ulozi demonstratora na kolegiju Mreže računala. Aktivni je član IEEE studentske udruge Osijek u kojoj obnaša dužnost tajnika i organizira predavanja iz područja računalne sigurnosti. Zvanje MTCNA (Mikrotik Certified Network Associate) stječe 2019. godine.