Zaštita informacijskog sustava od malicioznog koda

Mikičić, Dorian

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Dorian Mikičić

ZAŠTITA INFORMACIJSKOG SUSTAVA OD MALICIOZNOG KODA

ZAVRŠNI RAD

Varaždin, 2024.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Dorian Mikičić

Matični broj: 0016148800

Studij: Primjena informacijske tehnologije u poslovanju

ZAŠTITA INFORMACIJSKOG SUSTAVA OD MALICIOZNOG KODA

ZAVRŠNI RAD

Mentorica:

Izv. prof. dr. sc. Petra Grd

Varaždin, rujan 2024.

Dorian Mikičić

Izjava o izvornosti

Izjavljujem da je moj završni rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor potvrdio prihvaćanjem odredbi u sustavu FOI-radovi

Sažetak

Maliciozni kod predstavlja ozbiljnu prijetnju informacijskim sustavima, a obuhvaća razne oblike napada poput virusa, trojanskih konja, crva, ransomware-a i spyware-a. Kako bi se zaštitili sustavi i osigurali povjerljivi podaci korisnika, potrebno je primijeniti učinkovite mjere zaštite. Ovaj rad analizira suvremene metode napada malicioznim kodom, uključujući tehnike kao što su širenje putem mreža i datoteka, iskorištavanje ranjivosti te napadi putem ransomware-a. Glavni cilj je istražiti karakteristike ovih napada te razumjeti kako funkcioniraju te identificirati najučinkovitije strategije zaštite sustava od takvih napada kroz analizu učinkovitosti antivirusnih programa, vatrozida, sustava za detekciju napada i sandboxing tehnologija.

Ključne riječi: maliciozni kod, tehnike zaštite, sigurnosne mjere, antivirusni program, vatrozid, tehnike detekcije napada, virtualno okruženje**.**

Sadržaj

1. Uvod

Maliciozni kod, poznat i kao malware, obuhvaća niz zlonamjernih softverskih prijetnji poput virusa, crva, trojanskih konja, ransomwarea i spywarea, a svaki od ovih oblika ima specifične metode napada na računalne sustave. S obzirom na brzinu širenja ovih prijetnji i njihovu sposobnost prilagođavanja sigurnosnim mjerama, zaštita od malicioznog koda predstavlja jedan od ključnih prioriteta u području kibernetičke sigurnosti.

Motivacija za odabir ove teme proizlazi iz potrebe za razumijevanjem modernih tehnika zaštite koje se koriste za otkrivanje i sprječavanje malicioznih napada. Tehnološki napredak omogućuje napadačima korištenje sve sofisticiranijih metoda za izbjegavanje detekcije, što predstavlja značajan izazov za stručnjake za sigurnost. Ovaj završni rad istražuje različite tehnike zaštite, uključujući antivirusne programe, sustave za detekciju napada, vatrozide i sandboxing, te analizira njihovu učinkovitost kroz praktično testiranje u virtualnom okruženju. Cilj rada je pružiti cjelovit pregled postojećih alata i tehnika te procijeniti njihovu sposobnost otkrivanja i uklanjanja malicioznog koda.

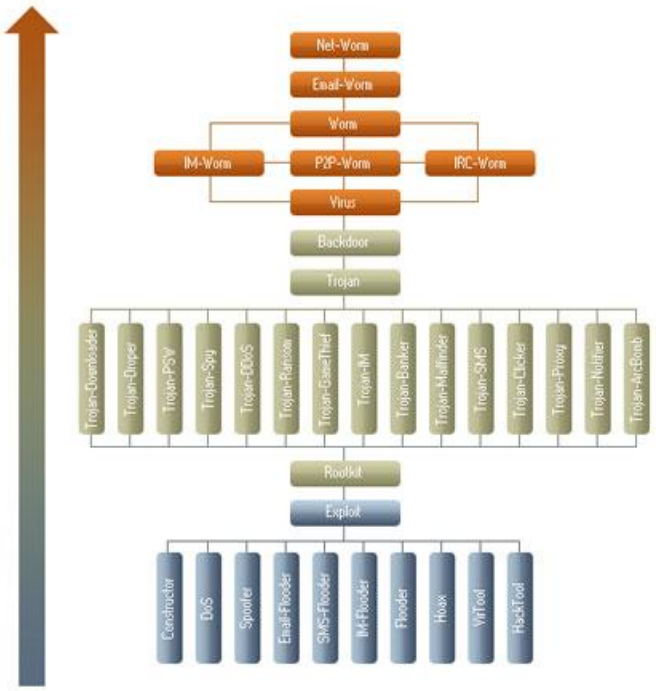
Važnost teme očituje se u stalnoj prijetnji koju maliciozni kod predstavlja za sigurnost podataka, financijske sustave i privatnost korisnika. Svaki napad može uzrokovati značajne financijske gubitke, narušavanje ugleda i gubitak povjerljivih informacija, što stvara potrebu za implementacijom učinkovitih obrambenih mjera. Ovaj rad pokušava odgovoriti na pitanje koje tehnike pružaju najvišu razinu sigurnosti te kako ih implementirati na način koji neće ometati poslovne procese, već će ih podržati i zaštititi.

2. Definicija i vrste malicioznog koda

Maliciozni kod, poznatiji kao i malware, je oblik zlonamjernog softvera čija je namjena oštećenje informacijskog sustava ili neovlašteno pristupanje različitim podacima i mrežama. Zlonamjerni kod može oštetiti informacijski sustav u različitim oblicima te može izvoditi različite vrste napada. Obilježen je svojom sposobnošću reprodukcije, širenja, automatskog pokretanja i nanošenja štete računalnom sustavu. Pogođeni računalni sustav može biti izložen gubicima povjerljivosti informacija, narušavanju integriteta i uskraćivanju pružanja usluga (Saeed, Selamat i Abuagoub, 2013).

Širenje zlonamjernog koda često se odvija preko različitih zaraženih datoteka, e-mailova ili web stranica s kojima se korisnik računala stalno susreće. Na njegovo djelovanje najveći utjecaj ima sam korisnik koji prilikom pokretanja određene datoteke ili posjećivanja we-mjesta preuzima zlonamjerni softver koji se zatim proširuje na cijeli računalni sustav gdje korisnik u principu nije ni svjestan da ga preuzima.Ovakvi napadi mogu biti vrlo opasni s obzirom na velike financijske gubitke gdje se napadaju i institucije viših razina. Malware ne nastaje sam od sebe, već ga kreiraju hakeri i cyber kriminalci čime se maliciozni kod svrstava u dio cyber kriminala. Oni su veoma stručni u pogledu računalnih tehnologija te koriste napredne tehnike kreiranja softvera kako bi se otežalo njihovo otkrivanje.

Postoji nekoliko vrsta malicioznih kodova, a neki od glavnih su virusi, trojanski konji, crvi, ransomware, spyware, adware i rootkit. Na slici 1. prikazane su sve vrste malicioznog koda u obliku klasifikacijskog dijagrama. U donjem dijelu dijagrama prikazani su zlonamjerni sofveri koji predstavljaju najmanju prijetnju, dok su u gornjem dijelu prikazani softveri koji predstavljaju najopasniju prijetnju za informacijski sustav (Kaspersky, bez dat.).



Slika 1. Klasifikacijski dijagram (izvor:Types of Malware Threats, bez dat.)

2.1. Virusi

Virusi predstavljaju jednu vrstu malicioznog koda koja se može replicirati unutar informacijskog sustava koji je zaražen. Virus se prenosi iznimnom lakoćom te se najčešće pojavljuje unutar datoteka ili dokumenata. Karakterizira ih mogućnost kopiranja u sami kod gdje se njegovim pokretanjem može proširiti na neki drugi kod te se tako prenosi s jednog sustava na drugi. To je moguće pomoću lokalnog dijeljenja datoteka, preko prijenosnih uređaja kao što su USB, CD ili DVD te u današnje vrijeme najčešće putem interneta.

Virusi djeluju na razne načine, međutim osnovno se virusi prema načinu djelovanja dijele na nerezidentne i rezidentne. Ako je virus postavljen u RAM-u (Random Access Memory) prilikom pokretanja tada se govori o nerezidentnom virusu, a ako je virus učitan u samu memoriju računala i zadržava se u memoriji tijekom rada računala tada se radi o rezidentnom virusu.(„Computer Emergency Response Team [CERT]“ bez dat.).

U krugovima sigurnosti informacijskih sustava poznata su tri osnovna tipa virusa („CERT“ bez dat.):

- **Boot sektor virusi:** Oni kopiraju svoj maliciozni kod u glavni boot sektor i tako osiguravaju da se maliciozni kod izvršava svaki put kada se računalni sustav p okrene
- **Programski virusi:** tipično, ove vrste zlonamjernog softvera postaju aktivne kada se pokrene zaražena izvršna datoteka, što je obično naznačeno ekstenzijama .exe ili .com.
- **Makro virusi**: ovi virusi napisani u programskom makro jeziku visoke razine imaju mogućnost kopiranja i brisanja sami sebe te izmjene dokumenata

Jedan od poznatijih primjera zaraze virusom jest Melissa virus. Ovo je vrsta makro virusa nastala 1999. godine. Glavne žrtve su bile korisnici Windows programa Microsoft Word. Virus je u trenutku ulaska u program koristio takozvane makro naredbe pomoću kojih je dolazio do privatnih informacija korisnika. Glavni cilj je bio „oteti“ ime i lozinku elektroničke pošte žrtve. Ako je virus uspješno pristupio e-mail adresi, u daljnjem procesu širenja slao je poruke eksplicitnog sadržaja na prvih 50 adresa koji su se nalazili na popisu korisnika. Poruke su sadržavale privitke gdje su bile prikvačene videozapisi i slike. Virus je djelovao poput automatizirane poruke. Svakim otvaranjem privitka poruke pokrenuo bih se programski kod kojim se poruka opet prenosila na prvih 50 primatelja korisnikove e-pošte. Kod nije bio u mogućnosti ugroziti bitne datoteke i dokumente, ali je uspio pritom onemogućiti globalne mreže velikih korporacija poput Mornaričkog zbora Sjedinjenih država (eng. *United States Marine Corps* – USMC) i Microsoft-a. Sprječavanje daljnjeg širenja potrajalo je nekoliko dana, a procijenjena šteta je iznosila 80 milijuna dolara za ponovnu stabilizaciju mreža i poslužitelja elektroničke pošte (Federal Bureau of Investigation [FBI], 2019).

2.2. Trojanski konj

Ovo je vrsta malicioznog koda koja se ponaša poput stvarne aplikacije, programa ili elektroničke pošte pokušavajući natjerati korisnika na njegovo preuzimanje. To je softver koji koristi daljinski upravljač za upravljanje drugim računalom u skladu s određenim programom. Ovaj virus može prodrijeti u korisnikovo računalo svojom funkcijom implantacije ili svojstvom svog dodatka koji nosi virus kako bi ukrao lozinke i osobne podatke, promijenio podatke ili izbrisao datoteke (Zhenfang, 2015).

Postoji mnogo načina širenja ovog malicioznog koda, međutim najčešći oblik širenja ove vrste malware-a je putem elektroničke pošte. Korisnik često prima poruku e-pošte od osobe koje poznaje, međutim ne shvaća kako je u toj poruci i zlonamjerni privitak. Klikom na privitak trojanski konj će se instalirati na sam uređaj te će uređaj prilikom svakog drugog

pokretanja i dalje biti izložen zlonamjernim kodom. U najgorem slučaju, korisnik će vjerojatno i dalje bezobzirno koristiti svoj uređaj bez ikakvog znanja o instaliranom softveru. Trojanski konj se teško može detektirati. Ovisno o njegovoj vrsti, on može ostati aktivan sve do njegove detekcije ili se prilikom izvršavanja svih zadanih akcija može sam izbrisati.

Prema (Fortinet, bez.dat.) u cyber kriminalu nalaze se različite vrste „trojanaca“ koji su stvoreni od cyber kriminalaca, a najčešće se pojavljuju:

- **Backdoor trojanac**: Trojanac omogućuje napadaču ulazak na daljinu u računalo i preuzimanje njegovih operacija. To zlonamjernom akteru daje potpunu kontrolu nad uređajem, dopuštajući mu da radi što god želi, uključujući brisanje datoteka, ponovno pokretanje računala, krađu podataka ili učitavanje zlonamjernog softvera.
- **Trojanac s lažnim antivirusom:** uspješno pokazuje kako učinkovito radi antivirusni softver. Ovaj softver je napravljen da radi slično standardnom antivirusnom programu u smislu otkrivanja i uklanjanja prijetnji, ali također iznuđuje novac od korisnika u zamjenu za navodno uklanjanje rizika.
- **Trojanac DDoS (eng. *Distributed denial of service*):** Ovi trojanski konji pokreću napade koji uzrokuju preopterećenje mreže. Kako bi zatrpao ciljanu web adresu i rezultirao uskraćivanjem usluge, poslat će brojne zahtjeve s jednog računala ili s grupe računala.
- **Ransom Trojanac**: ima za cilj ometati korisnikov pristup, korištenje računala smanjenjem njegovih performansi ili blokiranjem podataka na sustavu. Osobu ili tvrtku tada će napadač držati kao taoca dok ne plati otkupninu za poništavanje oštećenja uređaja ili objavljivanje oštećenih podataka.
- **Špijun trojanac**: stvoreni su da ostanu na korisnikovom računalu i gledaju što rade. To uključuje snimanje njihovih pritisaka na tipke, prikupljanje snimaka zaslona, pristup programima koje koriste i praćenje podataka za prijavu

Zeus Trojan je jedan od prvih i najpoznatijih napada trojanskog konja na neki informacijski sustav. Otkriven je 2007. godine, stvoren od strane grupe hakera iz Istočne Europe. Ubrzo je kod postao poznat diljem Europe te su mnogi napadači pokušavali stvarati svoje inačice virusa. Funkcionalnost Zeusa započinje porukom e-pošte na kojoj se nalazi zlonamjerna poveznica. Prilikom klika žrtve na poveznicu, u lokalni sustav se implementira zlonamjerni softver kako bi se omogućila komunikacija sa glavnim softverom zvanim Zbot. Preko glavnog poslužitelja napadač ostvaruje kontrolu nad zaraženim sustavom. Sljedeće, Zeus trojanac na lokalni sustav postavlja aplikaciju za bilježenje pritisaka tipki sa tipkovnice korisnika (eng. *Keylogger*). Napadač ovom tehnikom pokušava otkriti korisničko ime ili lozinku

ciljanog korisnika tijekom svakog njegovog pisanja u Internet pregledniku. Zeus se u najviše slučajeva koristi za krađu bankarskih podataka (Proofpoint, bez dat.).

U posljednje vrijeme, Emotet Trojan je jedan od najvećih malicioznih incidenta. Emotet je vrsta bankarskog „trojanca“ koji se širi putem phishing e-mail-ova, često se predstavljajući kao lažni računi, obavijesti o dostavi ili dokumenti koji izgledaju legitimno. Kada korisnici otvore zaraženi privitak ili kliknu na poveznicu, malware se instalira na njihov uređaj, što omogućava hakerima daljnje napade. Najveći incident dogodio se 2020. godine kada se program proširio diljem svijeta ciljajući financijske sektore, banke, zdravstvo i državne institucije širući informacije o „COVID-19“ pandemiji. U siječnju 2021. godine, međunarodna policijska operacija, predvođena Europolom, FBI-jem i drugim agencijama, uspješno je srušila infrastrukturu Emoteta. Ova koordinirana akcija rezultirala je zapljenom ključnih servera koji su omogućavali upravljanje Emotet mrežom, što je značajno smanjilo njegovu prisutnost na globalnoj razini.(Europol, 2021)

2.3. Crvi

Prema svojim obilježjima crvi su vrlo slični virusima. Računalni crvi poput virusa imaju mogućnost samostalnog repliciranja unutar informacijskog sustava. Lalić (2021) navodi kako su crvi reproduktivni računalni programi koji koriste računalne mreže da se kopiraju na druga računala, često bez ljudske intervencije. Ovaj izrazito opasan malware može se umnožavati u beskonačnost te postoje i neke vrste koje komuniciraju međusobno stvarajući komunikacijsku mrežu. Često nemaju neku drugu funkciju osim razmnožavanja, ali kao i druge vrste malicioznog koda mogu brisati datoteke, mijenjati konfiguraciju računala te u najgorem slučaju preuzeti cjeloukupnu kontrolu. Većinom se šire preko „stražnjih vrata“ (engl. backdoor) kako bi napadač mogao kontrolirati sustav na daljinu ili pomoću DDoS (engl. distributed denial-of-service) gdje se prekida komunikacija između sustava i računalne mreže.

Jedan od prvih razvijenih računalnih crva bio je „Morris crv“ koji je nastao 1988. godine od strane Roberta Tappana Morrisa. Cilj njegovog napada je bio izbrojiti sva računala na internetu u tadašnje vrijeme. Problem se pojavio kada se na računalo crv mogao replicirati više puta sve do stanja neupotrebljivosti računala. Time je uzrokavana velika šteta i problemi u radu cijelog interneta.

Ovo je bio prvi slučaj neke tužbe za računalni napad čime je Morris postao prvi svjetski haker. Računalni crv „VOLIM TE“ (eng. I LOVE YOU) je jedan od najpoznatijih slučajeva zaraze 20. stoljeća. Ovaj crv se širio putem elektroničke pošte, a u prilogu se nalazila datoteka „LOVE-LETTER-FOR-YOU.TXT.vbs“. Njezinim otvaranjem korisnik bi pokrenuo virus koji bi potom tražio slikovne i video datoteke s namjerom da ih zamjeni vlastitom kopijom. Također, instalirao je program za krađu lozinki koji bi se aktivirao prilikom svakog resetiranja računala.

Procijenjena cjelokupna šteta ovog slučaja širenja malware-a je bila 10 milijardi dolara (NoypiGeeks, 2023) .

2.4. Ransomware

Ransomware predstavlja skupinu različith programskih rješenja koji omogućuju dohvaćanje osobnih podataka na različitim uređajima žrtve napada gdje napadači iznuđuju žrtvu odšetom kojoj se zauzvrat vraćaju podaci u njihovo vlasništvo. Oblik odštete koji cyber-kriminalci najčešće traže jesu kriptovalute jer se ovim oblikom iznude otežava praćenje i identifikacija napadača. Zlonamjerni kod se pojavljuje tijekom 90-ih godina 20. stoljeća u obliku malicoznog koda naziva AIDS kojeg je programirao Joseph Popp. Liska i Gallo (2016.) navode da je kod je bio zamišljen u obliku virusa gdje je pritom omogućavao da se sustav koji je zaražen može pokrenuti 90 puta prije nego što bi se izbrisao i sakrio sve datoteke i mape sa sustava te prilikom tog procesa i kriptirao preostale datoteke. Izgled zlonamjernog koda prikazan je na slici 2.

Slika 2. AIDS Ransomware (Case Study: AIDS Trojan Ransomware, bez dat.)

Prema Lessing (2020.) nakon šifriranja datoteka, zlonamjerni bi softver prikazao poruku o otkupnini koja navodi da je zakup softvera od računalne kibernetičke korporacije („PC Cyborg Corporation“) istekao. Korisnik bi zatim dobio upute da izvrši plaćanje kako bi ga obnovio, uz naknade od 189 američkih dolara za godišnji najam ili 378 američkih dolara za doživotni najam. Općenito proces rada ransomware-a započinje infekcijom sustava preko ranjivih sigurnosnih kanala kao što su privitci, mail-ovi ili posjete sumnjivim web stranicama. U većini slučajeva preuzimanje koda, to jest zaraza se događa bez korisnikovog znanja.Kada zlonamjerni softver dospije u sustav računala, tada kreće njegova instalacija. Instalacija se automatski pokreće u

trenutku kada ransomware dospije u sustav. Većina korisnika koristi Windows operacijski sustav što ova vrsta malware-a iskorištava na način da u registru operacijskog sustava postavlja set ključeva koji omogućuju rad virusa svakim sljedećim pokretanjem uređaja (Liska & Gallo, 2016). Nakon instalacije slijedi analiza sustava gdje kod pokušava pronaći važne podatke koji bi bili vrijedni kriptiranja. Pronalaskom podataka oni se šifriraju i postaju nedostupni korisniku sustava sve dok ne bude imao pristup ključu za dekriptiranje. Jednom kada su podaci ukradeni prikazuje se obavijest na ekranu sustava koja obavještava korisnika da su njihovi podaci zaključani i da mogu biti otključani samo uz plaćanje određenim sredstvima. Unutar obavijesti se često nalazi i način na koji je potrebno kontaktirati napadača. Kao što je već spomenuto, napadač najčešće traži otkupninu u obliku kirptovaluta čime osigurava svoju anonimnost. Treba napomenuti da se plaćanje napadaču ne preporučuje jer ono ne uvjetuje vraćanje osobnih podataka ili ključa za dešifriranje.

Ransomware WannaCry jedan je od poznatijih ransomware napada u posljednje vrijeme. Prvi put se pojavljuje 2017. godine gdje napada računala sa Windows operacijskim sustavom. Računala sa Windows operacijskim sustavima bili su savršeni za napad jer su koristila računalni softver „EternalBlue“ koji je kasnije procijenjen kao ranjivost Windows sustava. Softver je bio razvijen od strane Nacionalne sigurnosne agencije (eng. *National security agency* – NSA). Napadači su preko ovog softvera mogli uspostaviti kontrolu nad računalom i šifrirati podatke. Šifriranje se odvijalo pomoću snažne enkripcije koja je zabranjivala korisnicima da pristupe podacima. Nakon enkripcije prikazivala se poruka na računalu u kojoj se nalazio zahtjev za otkupninu u obliku Bitcoin-a. Vrijednost otkupnine kretala se između 300 i 600 američkih dolara. Ransomware se ovom tehnikom ubrzo proširio svijetom gdje su najviše pogođene bile bolnice koje nisu imale pristup medicinskim podacima pacijenata te kompanije poput FedEx-a i Renault-a. Tvrtka Microsoft je pronašla odgovor za napade u obliku sigurnosnih zakrpavanja zastarjelih verzija Windowsa kako bi se spriječilo daljnje širenje. Ukupna financijska šteta iznosila je preko 100 milijuna dolara (TechTarget, bez dat.).

Također, u travnju 2022. godine dogodio se jedan od najpoznatijih ransomware napada ovog desetljeća. Grupa „Conti ransomware“ izvela je seriju sinkroniziranih napada na nekoliko vodećih vladinih institucija u Kostarici, uključujući i Ministarstvo financija. Napadi su uzrokovali potpuno isključivanje sistema koji su podržavali ključne državne servise, uključujući i zaustavljanje plaćanja poreza i carinskih pristojbi, uzrokujući ozbiljne ekonomske slomove i kašnjenja u državnim aktivnostima. Conti grupa je tražila otkupninu u cijeni od 10 milijuna dolara. Kibernetički napad je uglavnom bio usmjeren na javni sektor gdje je vlada Kostarike, kako bi spriječila daljnje posljedice, proglasila izvanredno stanje (CM Alliance,2022).

2.5. Spyware i adware

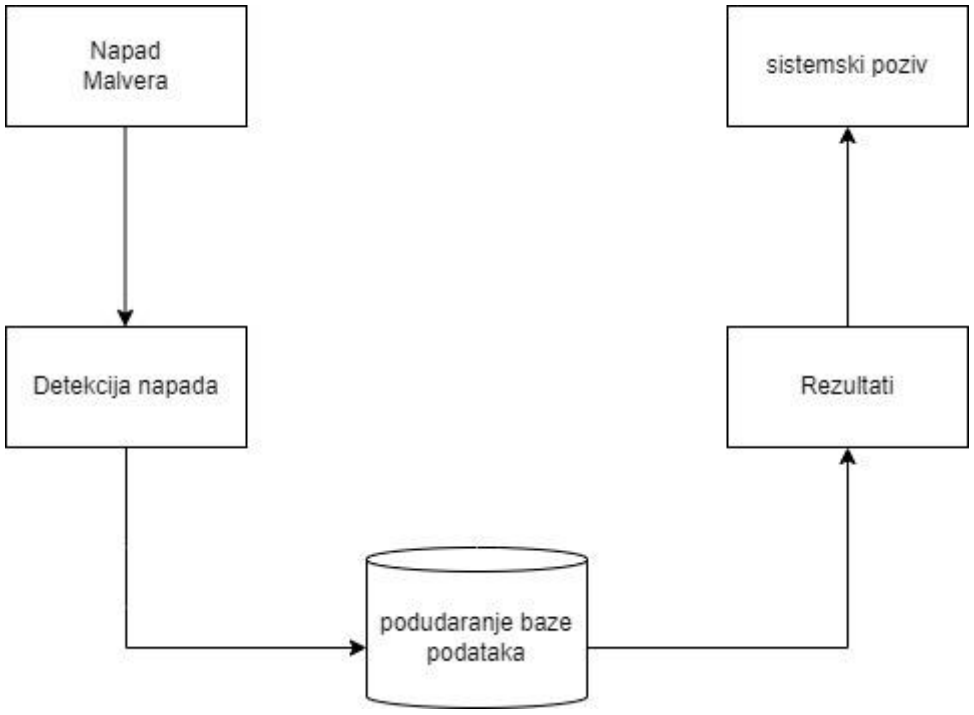
Zlonamjerni softver špijunski kod (eng. *Spyware*) služi za pribavljanje osjetljivih informacija korisnika kako bi na kraju preuzeo kontrolu nad korisnikovim računalom bez njegova znanja. Zaraženost špijunskim kodom se može prepoznati po preopterećenosti procesora sustava ili većom zauzetošću prostornog diska. Ovaj program uglavnom koriste korsinici treće strane jer se informacije prikupljene špijunskim kodom pretežno šalju na drugu lokaciju. Spyware prikuplja osobne informacije na razne načine i ovdje su nabrojane neke od primjera ponašanja špijunskog koda (Aycock, 2011, str. 2):

- bilježenje pritisaka tipki, pokreta i klikova mišem
- snimanje ekrana
- snimanje pomoću mikrofona ili kamere računala
- praćenje korisnikove aktivnosti na internetu
- krađa datoteka sa uređaja
- izbjegavanje deinstalacije sa uređaja

Ovo su samo neke od karakteristika prema kojima kompanije orijentirane izgradnji softvera prepoznaju zaraženost softvera špijunskim kodom. Sličan spyware-u, oglašavajući softver (eng. *Adware*) se definira kao vrsta malicioznog koda kojoj je glavna zadaća prikaz oglasa na ekranu korisnika. Nije opasan poput špijunskog koda, međutim oglasima može navoditi korisnika na preusmjeravanje prema nesigurnim stranicama. Pomoću svojih oglasa, pruža opciju pratiti statistiku posjeta određenim web stranicama za koje se reklama pojavljuje. Osim toga, može se pojavljivati prilikom instalacije i korištenja računalnih igara na računalu. Ovo dovodi do instalacije softvera za prikaz reklama gdje u najgorem slučaju se iskorištava za ugradnju mogo opasnijeg malware-a poput trojanskog konja ili spyware-a (Cisco, 2024.)

3. Tehnike zaštite od malicioznog koda

Sve većim napredovanjem tehnologije, razvijaju se mnoge tehnike napada na različite informacijske sustave koje je potrebno u što kraćem roku spriječiti. Bitno je da metode zaštite sustava linearno prate razvitak tehnologije, stoga mnoge tvrtke s digitalnim okruženjem kao jednu od glavnih zadaća postavljaju razvijanje i primjenu tehnike zaštita. Napadi sustava malicioznim kodom su vrlo nepredvidljivi te se kod može širiti na razne načine koje ponekad čak ni najzaštićeniji sustavi ne mogu otkriti velikom brzinom. Razvoj zaštite ne odvija se samo na tehnološkoj razini, već je uključen u proceduralnim radnjama i edukaciji korisnika. Pod proceduralne aspekte spadaju redovito ažuriranje softvera, operativnih sustava te postavljanja politika sigurnosti. Edukacija uključenih djelatnika u informacijski sustav ima ključnu ulogu kako bi individualno mogli prepoznati sumnjive aktivnosti unutar sustava ili prepoznati elemente socijalnog inženjeringa čime bi napadi bili što manje uspješni. Prepoznavanje ovih napada sastoje se od procesa i koraka prikazanih na slici 4.



Slika 3 Detekcija napada malicioznog koda (Malicious Software Detection, Protection & Recovery Methods: A Survey., 2014)

Kao što je prikazano i na slici, u radu *Malicious Software Detection, Protection & Recovery Methods: A Survey.,* autori Mirza i sur. (2014.) napominju pet koraka prilikom detektiranja napada malicioznog koda na sustav:

- **Napad malvera (eng. *Malware attack*)**: prepoznavanje opasnosti započinje napadom malverskog softvera na informacijski sustav. To uključuje napade već spomenutih tehnika malicioznog koda poput virusa, crva i trojanskih konja.

- **Detekcija napada (eng. *Attack detection*)**: ovo uključuje identificiranje sumnjivih radnji unutar sustava ili mreže sustava koji mogu implicirati na prisutnost zlonamjernog softvera. Identificranje ovih napada omogućuju bržu reakciju u sprječavanju prijetnji, ali i sprječavanje sličnih napada u budućnosti.

- **Podudaranje baze podataka (eng. *Database matching*)**: tijekom ovog koraka uspoređuju se datoteke malicioznog softvera koji imaju obilježja zlonamjernog koda s onima koji su pohranjeni u bazi podataka. Utvrdi li se da je kod zlonamjeran, datoteka se označava kao visokorizična te se ona briše ili blokira. Najveću funkcionalnost ima prilikom korištenja anitivirusnog programa.

- **Rezultati (eng. *Results*):** podrazumijeva dobivanje krajnjeg rezultata od strane sigurnosnih alata. Ovdje se nalaze svi mogući detalji i analiza napada te izbacivanjem rezultata alati preporučuju daljnje korake u sprječavanju mogućih opasnosti.

- **Sistemski poziv (eng. *System call*):** omogućuje programu ili aplikaciji da pristupi hardverskoj strani sustava preko samog operacijskog sustava,odnosno služi kao posrednik između aplikacije i operacijske jezgre. Praćenjem poziva mogu se otkriti sumnjive aktivnosti poput promjena sadržaja datoteka, oštećivanje memorije ili povezivanje aplikacija na druge udaljene servere.

3.1. Antivirusni programi

Antivirusni programi u današnjem svijetu su jedan od najraširenijh oblika zaštite od računalnih napada na sustav. Oni omogućuju redovito pretraživanje, odnosno skeniranje memorija ili datoteka na računalu kako bi pokušali otkriti uzorke ponašanja zlonamjernog softvera. Poželjno je za korisnika računala da redovito ažurira ove vrste programa jer

antivirusni softveri prilikom svake pretrage računala otkrivaju nove i optimizirane viruse. Računalo se pomoću programa može skenirati ručno ili automatski gdje je većina programa podešena na automatsko skeniranje uzimajući u obzir da se ona izvršavaju periodički.

Prema Choudary, Saroha i Beniwal (2013.) rad svakog antivirusnog programa započinje informacijom koja se kreće od izvornog sustava te putuje sve do odredišnog sustava. Glavno izvorište informacije unutar računala može biti tvrdi disk dok odredište može biti poslužitelj internetskih usluga koja sprema određenu informaciju i prosljeđuje je daljnjem korisniku. Prije dolaska informacije na odredišni sustav potrebno je utvrditi njezin sadržaj. Stoga, antivirusni softver može analizirati informaciju kao zaseban program ili može biti implementiran u operacijski sustav kao što je to slučaj unutar *Windows-a*. Primjerice, operacijski sustav Windows sastoji se od antivirusnog programa čiij je cilj praćenje aktivnosti tvrdog diska sustava. Pokuša li informacija pristupiti tvrdom disku, antivirusni softver će presresti njezino čitanje i pisanje kako bi se izvršilo skeniranje podataka. Nakon skeniranja, proces tumačenja koji može ili ne mora biti dostupan antivirusnom sustavu. U slučaju nedostupnosti potrebno je iskoristiti alternative koje prikupljaju sadržaj informacije te je prosljeđuju programu na daljnu analizu. Proces tumačenja se može odvijati kroz različite tehnike kao što su analiza potpisa ili analiza ponašanja. Naposljetku, informacije koje ne sadržavaju nikakvu vrstu malicioznosti odlaze na interpretaciju sadržaja kako bi stigle do odredišnog sustava, a korisniku se prikazuje obavijest upozorenja na ekranu pomoću korisničkog sučelja.

3.2. Sustavi za detekciju upada

Sustavi za detekciju napada (eng. *Intrusion Detection System, IDS*) ponajviše se fokusiraju na sprječavanje mrežnih napada i upada u sustav. Pokušavaju spriječiti sve napade ili neobične aktivnosti na sustav u stvarnom vremenu kako bi se informacija o napadu mogla dalje proslijediti ostalim sudionicima u zaštiti sustava. Oni su na raspolaganju računalu gdje mu pomažu suočiti se sa već prije spomenutim napadima na mrežu. Sustav za detekciju napada mora imati pristup cjelokupnom računalnom sustavu jer prikuplja potrebne informacije koje uspoređuje sa prijašnjim prijetnjama. Autori (Asmaa Shaker & Sharad, 2011).) navode pet glavnih funkcija sustava za detekciju napada:

- Analiza aktivnosti korisnika i sustava
- Analiza ranjivosti sustava i analiza njegove konfiguracije
- Procjena ispravnosti datoteka
- Mogućnost prepoznavanja obrazaca koji bi mogli asocirati na napad sustava

- Praćenje kršenje pravila korisničke politike

Jedan od načina za otkrivanje neželjenih aktivnosti unutar sustava jest pregledavanje logova. Oni se automatski kreiraju unutar operacijskog sistema. Logovi zapisuju sve aktivnosti i poduzete radnje od strane korisnika. Ovi zapisi mogu biti pregledani ručno od strane administratora cjelokupnog sustava, međutim ovaj postupak iziskuje veliku količinu vremena zbog samih veličina datoteka gdje su zapisane aktivnosti korisnika. (Hrvatska akademska i istraživačka mreža (CARNet), 2000). Uzimajući ovo u obzir, detekcija potpisima i detekcija anomalija su najčešće metode za otkrivanje napada sustava.

3.2.1. Detekcija potpisima

Detekcija potpisima koristi unaprijed definirane obrasce ili potpise za prepoznavanje zlonamjernih aktivnosti u mreži ili na sustavima. Ova metoda je široko korištena jer omogućava brzo prepoznavanje poznatih prijetnji uz relativno nisku stopu lažnih alarma. Glavna značajka ove detekcije uspoređivanje mrežnog prometa ili različitih aktivnosti zlonamjernih potpisa kako bi se napravile promjene nad bazom podataka. Ako detekcija potpisima pronađe podudaranje prilikom analize, generira aktivirajuća upozorenja. Prednost ovakvog načina detekcije napada jest brzo otkrivanje poznatih prijetnji gdje se one u većini slučajeva sprječavaju i prije nego što mogu izvršiti napad. Glavni nedostatak ove metode detekcije je nemogućnost prepoznavanja nepoznatih prijetnji koje prijašnje nisu izvršavale napade na sustav (Kwon eisur., 2022).

Drugi problem metode potpisima očitava se u velikom korištenju računalnih resursa. Prema istraživanjima CARNet-a (2000.) pokazano je da neovlašteni korisnici izvode napad preko udaljenih izvora gdje se napadi ne izvršavaju odmah, već kroz duže vremensko razdoblje čime postupno i neprimjetno provode svoje aktivnosti. Time se uzrokuje zadržavanje velike količine podataka unutar sustava na duže vrijeme gdje se opterećuje velika količina memorije. Stoga, kako bi ova tehnika ostala učinkovita na najvišoj razini, potrebno je stalno ažuriranje baze podataka potpisima i eventualno kombiniranje zaštite s metodom za otkrivanje anomalija.

3.2.2. Detekcija anomalija

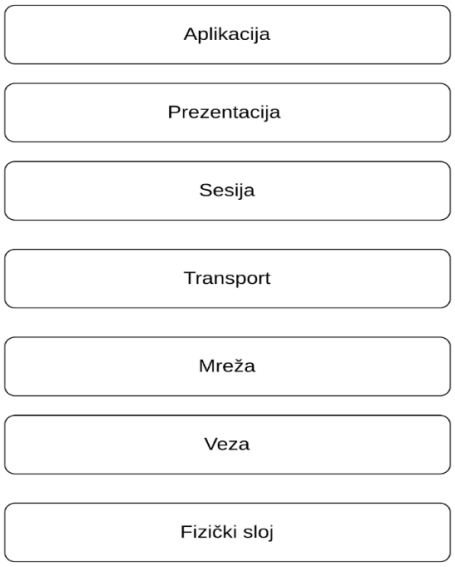
Detekcija anomalija bazira se na praćenje mrežnog prometa i računala u potrazi za sumnjivim aktivnostima. Analizom skupa podataka pokušava otkriti neuobičajene obrasce

aktivnosti korisnika i odvojiti od aktivnosti koje se smatraju normalnim. Za razliku od detekcije potpisa, ova metoda posjeduje mogućnost prepoznavanja nepoznatih prijetnji. Alat za detektiranje anomalija najprije proučava normalno ponašanje korisnika što uključuje prosječno korištenje mrežnog prometa, memorije i procesora. Prema prikupljenim podacima stvara se „normalni“ profil gdje se odvijaju uobičajene aktivnosti. Ove aktivnosti se konstantno prate tako da je uočavanje promjena u ponašanju izrazito učinkovito. Neke od uobičajenih anomalija mogu biti prijavljivanje korisnika u neuobičajeno vrijeme ili dodavanje novih uređaja na mrežu (SIS Wiki, bez dat.).Prema (N-able, 2021) ovo je jedan od nedostataka jer mnoge ne maliciozne aktivnosti mogu biti prijavljene samo zato što odstupaju od norme stoga je potrebno više vremena i novaca da se ispita svaka prijetnja koja se odnosi na opasnost jer detekcija anomalija ima veliku vjerojatnost da proizvede lažne rezultate.

3.3. Vatrozid

Glavna uloga vatrozida (eng. *firewall*) kao sigurnosne mjere unutar sustava je uspostava kontrole i reguliranja privatne mreže računala. Vatrozid služi kao posrednik između sustava i interneta gdje odlučuje koji promet usmjeren prema računalnom sustavu može propustiti, a koji blokirati. Kontrola mreže se provodi inspekcijom paketa koji prolaze kroz mrežu (Cisco, bez dat.). Paket sadrži podatke i informacije koje se preko mreže prenose od jednog sustava do drugog, a Noonan i Dubrawsky (2006.) inspekciju paketa definiraju kao presretanje podataka unutar tog paketa gdje se podaci analiziraju i u konačnici se odlučuje da li će se mrežni promet propustiti ili blokirati. Prilikom evaluacije paketa, primarni fokus je na analizi izvornog i odredišnog porta te njihovih IP (internet protokol) adresa. U današnjim suvremenim tehnologijama, vatrozidi se često svrstavaju u okvirima zaštite internetske povezanosti što u stvarnosti nije jedina funkcija njihovog djelovanja. Za rad vatrozida nije potrebna internetska konekcija i mnoge današnje korporacije ga koriste za ograničavanje povezivanja s internim mrežama osjetljivih sektora, poput računovodstva.

Većina današnjih vatrozidnih okruženja djeluju prema OSI referentnom modelu (eng. Open Systems Interconnection Basic Reference Model). Model služi za razvijanje komunikacija između informacijskog sustava i računalnih mreža (Wack i sur. 2002.). Podijeljen je u sedam slojeva koji su vizualno prikazani na slici 5.



Slika 4. OSI referentni model (autorski rad)

U prvom sloju, fizičkom sloju, stvara se fizička komunikacija između hardvera i različitih medija za prijenos kao što je kabel za Ethernet. Drugi sloj služi za prijenos podataka mrežnog prometa na lokalnu mrežu gdje vatrozidi preko MAC (eng. *Media Access Control)* adresa kontorliraju pristup svim lokalnim mrežama. Nadalje, na trećem mrežnom sloju koriste se IP adrese pomoću kojih se podaci paketa šalju različitim mrežama najboljim mogućim putem. Transportni sloj omogućuje uspješni protok podataka i informacija između dva sustava gdje vatrozidi pokušavaju otkirti greške prilikom slanja. Naposljetku, slojevi sesije, prezentacija i aplikacija su slojevi koji se odnose na krajnjeg korisnika. U ovim slojevima se upravljama vezama između aplikacija, prezentiraju se podaci te pružaju se različite usluge aplikacijama. Vatrozidna okruženja najviše djeluju u drugom, trećem, četvrtom i sedmom sloju prema kojima su određeni različiti tipovi vatrozida.

3.3.1. Tipovi vatrozida

Tipovi vatrozida određeni su prema OSI modulu, a razlikuju se prema načinu praćenja veza, načina filitriranja i načina praćenja logova (What is a firewall? Definition and explanation, bez dat.):

- **Vatrozid za filtriranje statičkih paketa (eng. *Static Packet-Filtering Firewall*):** nude praćenje podataka svakog paketa pojedinačno tijekom slanja preko mreže. Ovaj tip djeluje u trećem sloju spomenutog modela jer se filtiranje

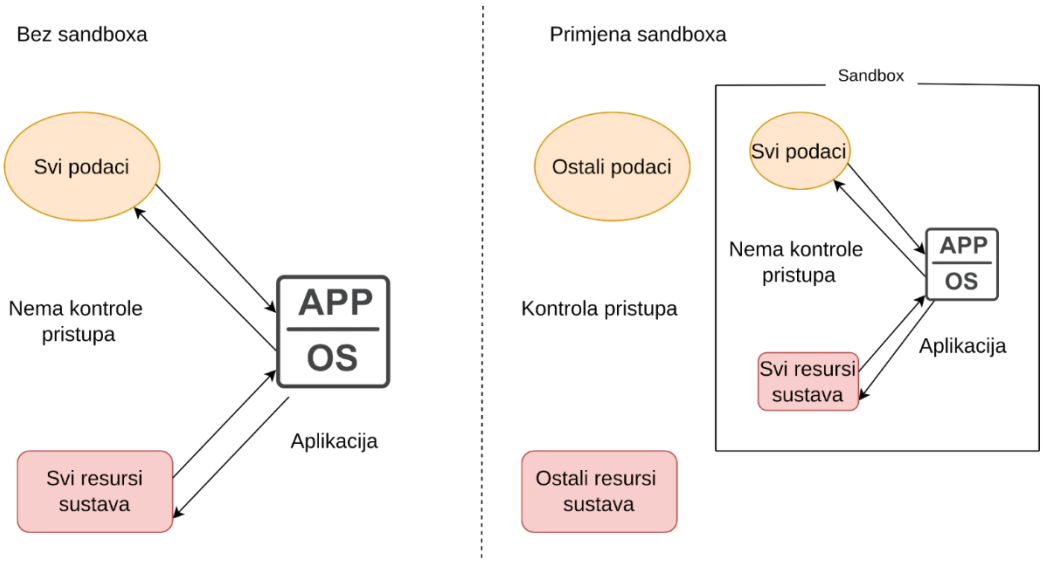
statičkih paketa odvija pomoću IP adresa i portova čime se sprječava da se dvije mreže istovremeno spajaju bez ikakvog dopuštenja.

- **Vatrozid pristupnika na razini kruga (eng. *Circuit-Level Gateway Firewall*):** djeluje na petom sloju modela. Omogoćuje otvorenu vezu prilikom povezivanja dviju mreža. Kako bi se otvorena veza održavala, potrebna je stalna provjera praktičnosti i pouzdanosti paketa koji se međusobno povezuju.
- **Vatrozid za inspekciju stanja (eng. *Stateful Inspection Firewall)*:** ovaj vatrozid se još naziva i vatrozid za filtriranje dinamičkih paketa. Poput vatrozida za statičko filtriranje, propušta mrežne pakete na temelju IP adresa te ulaznih i izlaznih portova. Imaju prednost u odnosu na statično filtriranje jer imaju mogućnost praćenja veza u stvarnom vremenu i pamćenja prošlih veza koje su se već odvijale. Djeluju na slojevima 4 i 7 OSI referentnog modela.
- **Proxy vatrozid (eng. *Proxy firewall*):** predstavlja vrstu vatrozida na razini aplikacije. Proxy vatrozid se ponaša poput stvarne fizičke pregrade. Njihovo djelovanje zasniva se na ulozi posrednika pri čemu provjerava komunikaciju koja se odvija između vanjske mreže i unutarnje mreže organizacije. S obzirom da je njegovo djelovanje na razini aplikacije, tako je i filriranje podataka temeljeno na aplikacijskoj razini. Pritom se čitaju i pišu razni aplikacijski protokoli koji omogućuju unakrsno filtriranje osobnih podataka.
- **Vatrozid sljedeće generacije (eng. *Next-Generation Firewall*):** dizajnirani su za sprječavanje specifičnih prijetnji, a najčešće se koristi u velikim organizacijama. Pruža detaljniju analizu napada s obzirom da stalnim napretkom tehnologije prijetnje postaju naprednije i teže se otkrivaju. Neke od značajki koje uključuje jesu sposobnost prepoznavanja aplikacija, identifikacija korisnika i dubinska analiza mrežnog prometa.
- **Hibridni vatrozid (eng. *Hybrid Firewall*):** zaštita sustava se odvija pomoću dvije ili više vrsta vatrozida na jednoj privatnoj mreži kombiniranjem njihovih tehnologija.

3.4. Sandboxing

Ideja sandboxing-a je postavljanje sumnivih aplikacija ili programa u izolirano okruženje unutar kojeg se izvode operacije za provjeru potencijalno zlonamjernog sadržaja. Cilj je detaljno ispitati program i njegove pojedinačne komponente zasebno, prikupiti mrežne događaje i maksimalno povećati obradu svih podataka prikupljenih iz bilo koje aplikacije ili sustava koji prenosi informacije u sandbox (Lenić, 2022).Postavljanjem u izolirano virtualno

okruženje se osigurava da zlonamjerni programi ne utječu na glavni sustav i izolirani resrusi su odvojeni od ostalih resursa što je detaljnije prikazano na slici 6.



Slika 5. Primjena sandbox-a (izvor: What Is Sandboxing, And Why Do We Need It?, 2022)

Pomoću sandboxa korisnik može otvariti datoteke, kodove i programe što mu omogućuje točan uvid u ono što promatrani objekt pokušava napraviti u zadanom okruženju. Sandboxing se može implementirati u različitim oblicima, a njegove karakteristike mogu varirati ovisno o namjeni i sloju na kojem radi. Jedan primarni oblik je sandboxing na razini operativnog sustava gdje OS provodi sigurnosne politike za sadržavanje aplikacija (Egele, Scholte, Kirda i Kruegel, 2012). Primjerice, značajka Windows Sandbox omogućuje korisnicima pokretanje privremenog i zasebnog okruženja radne površine. Drugi oblik je sandboxing na razini aplikacije, kojim upravljaju određene aplikacije, a ne operacijski sustav. Uobičajen primjer su web-preglednici koji koriste sandboxing za pokretanje skripti web-stranice u ograničenom okruženju kako bi spriječili skripte da pristupe lokalnom datotečnom sustavu ili drugim karticama (Rescorla, 2010.). Sandboxovi temeljeni na virtualizaciji stvaraju virtualne strojeve za izvršavanje i analizu nepoznatog softvera. Ova se vrsta često koristi u analizi zlonamjernog softvera, omogućujući stručnjacima za sigurnost da promatraju ponašanje zlonamjernog softvera u kontroliranom okruženju bez ugrožavanja stvarnih sustava.

Važno je napomenuti da iako su sandboxovi moćan alat za povećanje sigurnosti, oni nisu sigurni. Napadači mogu upotrijebiti sofisticirane tehnike za otkrivanje i bijeg iz sandbox okruženja. Zlonamjerni softver može otkriti tehnike kao što su spajanje i instrumentacija kako bi izmijenio svoje ponašanje unutar sandboxa, čime bi izbjegao otkrivanje. Stoga je ključno da

sandboxovi budu ispravno konfigurirani i njima se upravlja s razumijevanjem njihovih ograničenja.

3.5. Virtualizacija

Razvijanjem suvremene tehnologije koncept virtualizacije postaje jedna od ključnih tehnika zaštita gotovo svih poslovnih sustava. Pomoću tehnike virtualizacije stvaraju se virtualna okruženja u kojem se testiraju virtualne varijante operativnih i računalnih sustava. Primarni cilj tehologije je spriječiti opterećenost fizičkih sustava što omogućuje da svaka aplikacija neke organizacije ne mora imati svoje fizičko računalo, već se mogu nalaziti na virtualnim strojevima (eng. Virtual machine - VM). Virtualni stroj je oblik virtualnog računala sličan fizičkom u kojem se procesuira prividni operacijski sustav s time da treba napomenuti kako on ne postoji u fizičkom obliku (Jug, 2021.). Shodno tome, korištenjem virtualne mašine ne postoji razlika između stvarnog i prividnog sustava jer virtualno okruženje ima identične značajke kao i fizičko. Ovime se osigurava da krajnji korisnik koristi mašinu u potpunoj sigurnosti, bez pribojavanja da će uzrokovati pad sistema ili ga zaraziti. Uvođenjem virtualnih strojeva može pomoći pri uštedi resursa kao što je primjerice smanjenje fizičkih servera čime se i paralelno smanjuju troškovi poslovanja organizacije. Upravljanje i stvaranje mašina izvodi se pomoću hipervizora. U suštini, hipervizor se ponaša poput sloja između fizičkih resursa računala i virtualnog stroja. Resursi se hipervizorom skupljaju, raspoređuju i dodjeljuju mašinama. Postoji dva tipa hipervizora: tip 1 i tip 2. Tip 1 hipervizor je automatski ugrađen na fizičko računalo, dok se tip 2 konfigurira na operacijski sustav i time djeluje poput svake druge aplikacije (Katić, 2024)

Napredovanjem informatičkih struktura poslovnih organizacija, proces virtualizacije se ne koristi samo za stvaranje virtualnog operativnog sustava, već se ovaj proces pojavljuje i u drugim komponentama sustava. Vrste tehnika virtualiziranja, zbog sveobuhvaćenosti i broja, prikazani su u sažetom obliku pomoću tablice sa svojim prednostima i nedostacima po uzoru na istraživanje Nacionalnog Računalnog tima za hitne slučajeve (eng. Computer Emergency Response Team - CERT).

Tablica 1. Prednosti i nedostaci tehnika virtualizacije

	Prednosti	Nedostaci
Potpuna virtualizacija	Omogućuje instalaciju izvornog operacijskog sustava na virtualno računalo	Nije moguća na svim Sustavima
Djelomična virtualizacija	Omogućuje dijeljenje memorijskih sredstava među korisnicima	Sam dio programa može se virtualno pokretati
Paravirtualizacija	Omogućuje instalaciju operacijskih sustava na virtualno računalo	Zahtjeva izmjene u OS-ovima koji se instaliraju
Virtualizacija na razini OS-a	Učinkovito korištenje sredstava operacijskog sustava domaćina	Svi OS-ovi moraju biti iste vrste
Sklopovski potpomognuta virtualizacija	Brži i učinkovitiji rad za virtualne sustave	Moguća smanjena učinkovitost kod drugih primjena

(Izvor: Computer Emergency Response Team – CERT)

4. Implementacija tehnika zaštite

Za učinkovitu zaštitu informacijskih sustava od zlonamjernih kodova, važno je primijeniti sigurnosne prakse koje uključuju instalaciju i održavanje antivirusnog softvera, kontrolu pristupa sustavima i implementaciju dodatnih sigurnosnih mjera poput vatrozida i sandboxing tehnologije. Antivirusni softver se koristi za otkrivanje i neutralizaciju zlonamjernih programa. Posebnu pozornost treba posvetiti metodama za detekciju napada, poput analize ponašanja (heuristička analiza) i tehnologijama skeniranja potpisa, koje mogu otkriti i poznate i nepoznate prijetnje. Korištenje sandboxing tehnologije pruža dodatnu sigurnost omogućujući pokretanje sumnjivih programa u izoliranom okruženju bez ugrožavanja ostatka sustava. Korištenjem ovih strategija smanjuje se rizik od uspješnih napada, međutim važno je napomenuti da nijedna od ovih strategija ne pruža potpunu zaštitu, stoga je nužno koristiti slojevitu strategiju zaštite.

Prema Timoneri (2023.) ovdje su neke od sigurnosnih praksi koje se mogu implementirati kako bi se smanjili rizici povezani sa zlonamjernim kodom:

1. Instalacija i održavanje antivirusnog softvera. Antivirusni softver prepoznaje zlonamjerni softver i štiti računalo od njega. Instaliranje antivirusnog softvera renomiranog dobavljača važan je korak u sprječavanju i otkrivanju infekcija. Uvijek treba posjećivati izravno web-mjesta dobavljača umjesto klikati na reklame ili veze e-pošte. Budući da napadači neprestano stvaraju nove viruse i druge oblike zlonamjernog koda, važno je održavati antivirusni softver ažuriranim.
2. Oprez s poveznicama i privicima.
3. Blokiranje skočnih oglasa. Blokatori skočnih prozora onemogućuju prozore koji potencijalno mogu sadržavati zlonamjerni kod. Većina preglednika ima besplatnu značajku koja se može omogućiti za blokiranje skočnih oglasa.
4. Korištenje računa s ograničenim dopuštenjima. Pri surfanju, dobra je sigurnosna praksa koristiti račun s ograničenim dopuštenjima. Ako se računalo zarazi, ograničene dozvole sprječavaju širenje zlonamjernog koda i eskalaciju na administrativni račun.
5. Onemogućavanje značajki AutoRun i AutoPlay vanjskih medija. Onemogućavanje značajki AutoRun i AutoPlay sprječava automatsko pokretanje vanjskih medija zaraženih zlonamjernim kodom na računalu.
6. Promjena lozinki. Ovo uključuje sve lozinke za web stranice koje su možda bile spremljene u predmemoriju u web pregledniku.
7. Održavanje softvera ažuriranim.

8. Sigurnosno kopiranje podataka. Redovito sigurnosno kopiranje dokumenata, fotografija i važnih poruke e-pošte u oblak ili na vanjski tvrdi disk. U slučaju infekcije, podaci neće biti izgubljeni.
9. Instalacija ili omogućavanje vatrozida. Vatrozidi mogu spriječiti neke vrste infekcije blokiranjem zlonamjernog prometa prije nego uđe u računalo. Neki operativni sustavi uključuju vatrozid;
10. Korištenje anti-spyware alata. Špijunski softver čest je izvor virusa, ali infekcije se mogu svesti na minimum korištenjem programa koji identificira i uklanja špijunski softver. Većina antivirusnog softvera uključuje anti-spyware opciju.
11. Praćenje računa. Bilo kakva neovlaštena upotreba ili neuobičajena aktivnost na računima—osobito na bankovnim računima, valja odmah kontaktirati davatelja računa.
12. Izbjegavanje korištenja javnog Wi-Fi-ja. Neosigurani javni Wi-Fi može omogućiti napadaču da presretne mrežni promet uređaja i dobije pristup osobnim podacima.

4.1. Odabir tehnika zaštite sustava

Učinci zlonamjernog softvera na pojedince i organizacije mogu biti katastrofalni. Pojedinci mogu pretrpjeti ogromnu psihološku i emocionalnu štetu čak i od prijetnje da će njihovi privatni podaci biti otkriveni ili njihov teško zarađeni novac ukraden, dok organizacije mogu pretrpjeti financijske, pa čak i regulatorne posljedice od kibersigurnosnog incidenta. Za tvrtke oštećenje i gubitak podataka može rezultirati uništenjem važnog intelektualnog vlasništva, zastojem sustava i prekidom poslovanja, pa čak i prijetnjama kontinuitetu poslovanja zbog financijskih gubitaka i štete ugledu. Neovlašteni pristup i povrede podataka također mogu imati pravne posljedice. Stoga je odabir odgovarajućih metoda zaštite sustava vrlo bitan i zahtijeva razmatranje različitih faktora s ciljem razvoja učinkovitih obrambenih mjera protiv kibernetičkih prijetnji. Prvi korak u tom procesu je identifikacija prijetnji i procjena rizika, koji pomaže u otkrivanju mogućih prijetnji poput vanjskih napadača, zlonamjernih i tehničkih ranjivosti.

Kako bi se procijenila prijetnja, važno je izvesti dubinsku analizu ranjivosti sistema, uključujući stare sisteme, loše upravljanu politiku pristupa i slabe enkripcijske mjere. Dokazivanjem ranjivosti omogućuje se usmjeravanje resursa na kritične točke sistema koje zahtijevaju dodatnu zaštitu. (Egele i sur., 2012). Kada se postave prioriteti sigurnosnih mjera informacijske zaštite, važno je da se postave u skladu s poslovnim načinom rada organizacije. Sigurnosne mjere trebale bi služiti za podršku poslovnim procesima, a ne za ometanje, a važnost fleksibilnosti u rješenjima igra ključnu ulogu. Tehnološki kapaciteti i raspoloživost resursa također su važni pri odabiru tehnika zaštite. Organizacije sa manjim budžetom možda

neće moći priuštiti najskuplja rješenja, ali će moći koristiti pristupačnije mjere koje pružaju njima odgovarajuću razinu sigurnosti. Osim toga, potrebno je razmotriti proširivost odabrane tehnike zaštite kako bi se omogućilo prilagođavanje sigurnosnih mjera rastu i razvoju organizacije. Fleksibilnost i mogućnost integracije s novim tehnologijama osiguravaju da sustav ostane siguran i otporan na promjene u dinamičnom kibernetičkom okruženju (Pahl i sur., 2019).

5. Testiranje i analiza tehnika zaštite u virtualnom okruženju

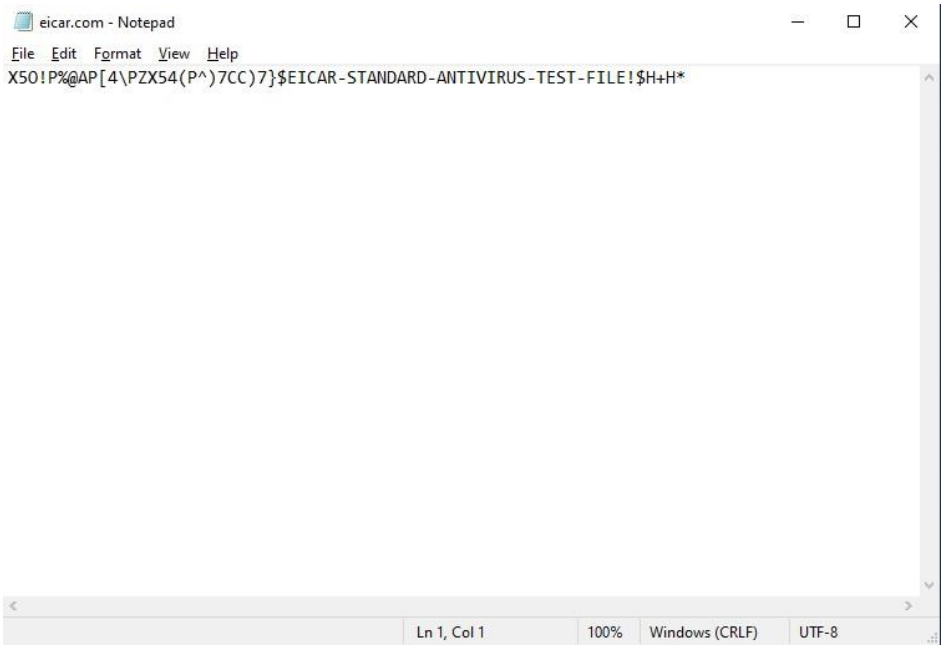
Za praktični dio rada, testirati će se tehnike zaštite malicioznog koda koji su navedeni u teoretskom dijelu rada preko primjene virtualnog okruženja. Virtualno okruženje koje će se koristiti je aplikacija Virtualbox. Aplikacija omogućuje stvaranje više operativnih sustava preko virtualnih mašina, poput Windows-a ili Linux-a, na jednom fizičkom računalu. Njezine značajke olakšavaju testiranje različitih sigurnosnih scenarija jer i omogućuje stvaranje snimki trenutnog stanja virtualne mašine kako bi se u slučaju pada virtualke korisnik mogao prethodno vratiti na prijašnje stanje.

Nakon instalacije aplikacije, izrada virtualne mašine je vrlo jednostavna. Za potrebe testiranja kreirati ćemo dvije virtualne mašine za različitim operativnim sustavima. Na jednoj mašini konfigurirat će se operativni sustav Windows 10, a na drugoj sustav Linux Ubuntu. Značajke i konfiguracija ovih mašina su iste na obje mašine.

Slika 6. Konfiguracija Windows 10 virtualne mašine

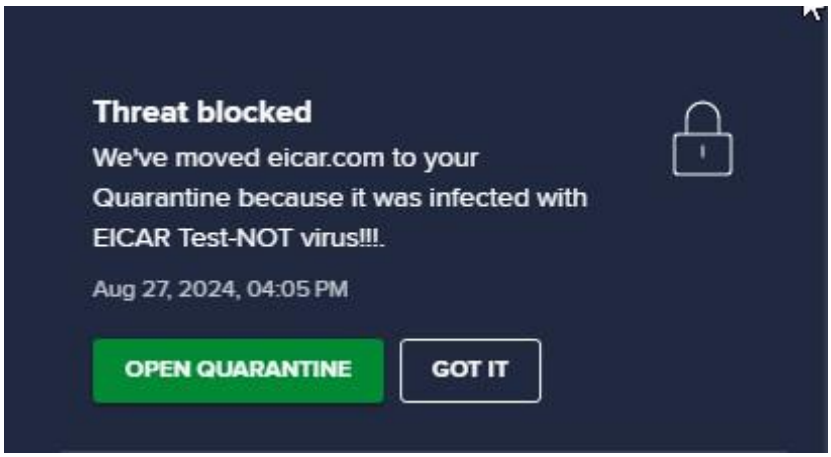
5.1. Testiranje antivirusnih programa u Windows sustavu

Testiranje antivirusnih programa odvijati će se u Windows virtualnoj mašini zbog jednostavnosti instalacije i konfiguracije različitih antivirusnih rješenja. Također, Windows operativni sustav sadrži ugrađeni antivirusni program koji pruža osnovnu zaštitu sustava čime je olakšano uspoređivanje sa ostalim antivirusnim rješenjima. U svrhu testiranja uspoređena su dva antivirusna programa: Microsoft Defender Antivirus, Avast i Virus Total. Kao što je već spomenuto Microsoft Defender Antivirus je ugrađen u operativni sustav Windows. Avast je jedan od najpopularnijih besplatnih antivirusnih alata koji se može preuzeti preko njihove službenih internetske stranice. Simulacija prisutnosti malicioznog koda obavljena je preko testne datoteke koje je razvilo Europski institut za istraživanje računalnih antivirusnih programa (eng. European Institute for Computer Antivirus Research – EICAR). Kako bismo testirali antivirusne programe najprije je potrebno kreirati tekstualnu datoteku pod imenom „eicar.com“ dok se unutar datoteke nalazi niz znakova koji simuliraju virus u sljedećem obliku: „X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*“.
Treba napomenuti da ova vrsta datoteke nije stvaran virus već je stvorena za potrebe testiranja, međutim antivirusna rješenja bi ju trebala prepoznati kao malicioznu prijetnju.



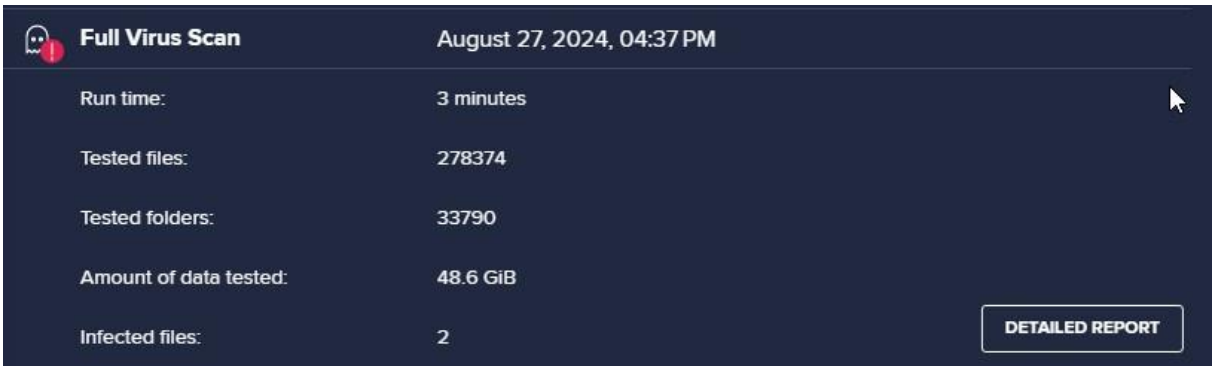
Slika 7. eicar.com testna datoteka

Tijekom korištenja Avasta, već prilikom pokušaja kreiranja tekstualne datoteke program onemogućava njezino stvaranje, izbacuje obavijest o blokiranju prijetnje i stavlja datoteku u karantenu.



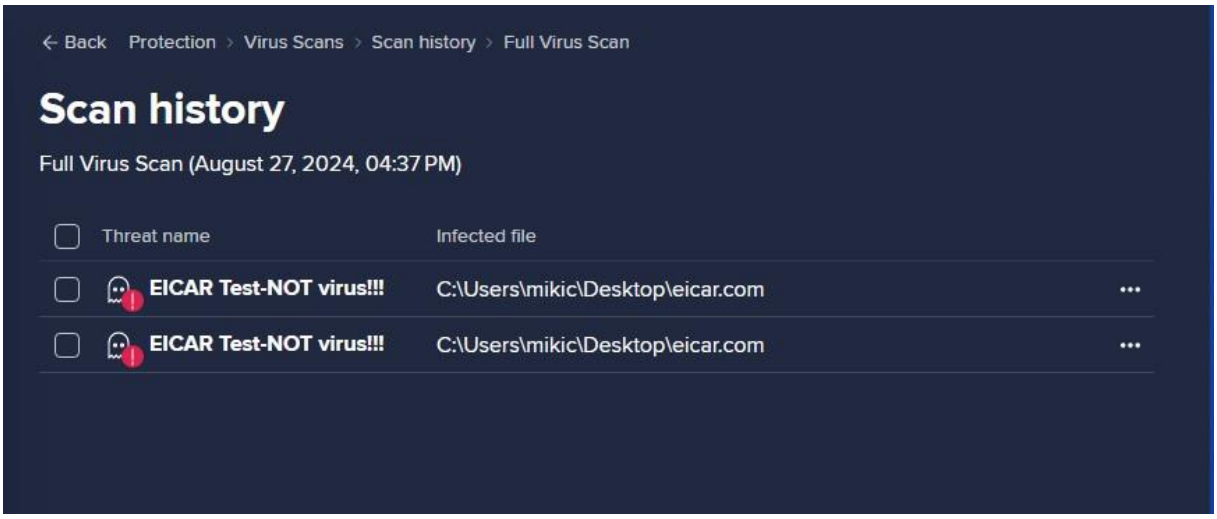
Slika 8.blokiranje prijetnje pomoću Avast antivirusa

Kako bi dodatno analizirali učinkovitost ovog alata, prilikom kreiranja maliciozne datoteke Avast antivirusni program je bio onemogućen kroz vremenski period od 10 minuta kako bi se tekstualna datoteka mogla kreirati unutar Windows virtualne mašine. Nakon kreiranja, unutar Avast programa nalazi se opcija cjelokupnog skeniranja virusa na računalu gdje tijekom pokretanja opcije program pregledava sve datoteke na računalu u potrazi za malicioznim prijetnjama.



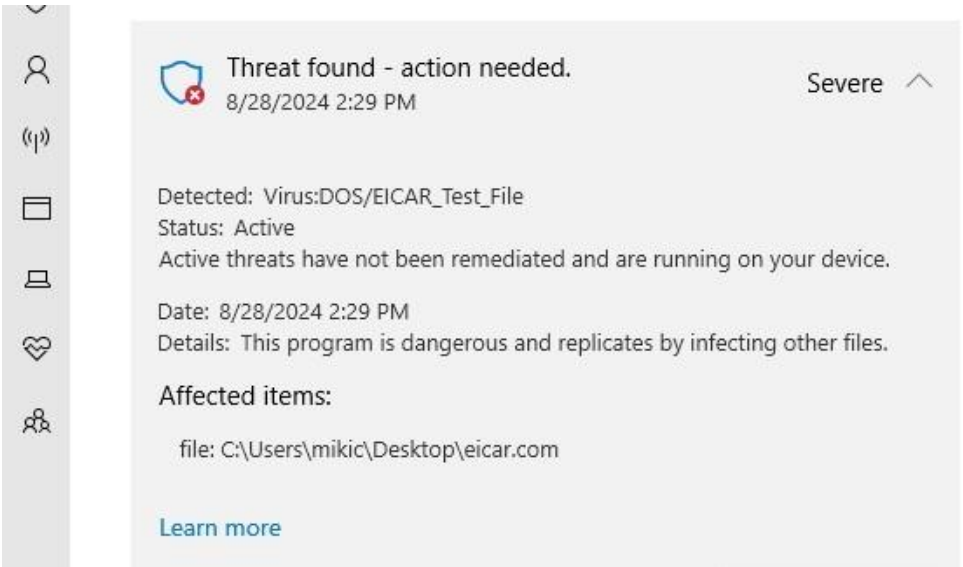
Slika 9. Avast Skeniranje virusa

Prema slici 9 može se uočiti svi detalji cjeloukupnog skeniranja. Proces skeniranja je trajao 3 minute gdje je testirano preko 25.000 datoteka i preko 30.000 mapa na računalu. Na kraju izvještaja prikazano je da se unutar sustava nalaze dvije zaražene datoteke. Prilikom klika na gumb „detaljno izvješće“ (eng. *Detailed report*) prikazuje se ime malicioznih datoteka i njihove putanje.



Slika 10. Pronađene maliciozne datoteke

Sljedeći test se odvijao preko Microsoft Defender Antivirus programa. Poput i Avast programa, Microsoft Defender Antivirus također sadrži zaštitu sustava u stvarnom vremenu te se prilikom kreiranja datoteke ona automatski stavlja u karantenu Windows sustava.Kako bi se maliciozna datoteka postavila na računalo potrebno je isključiti opciju zaštite u stvarnom vremenu. Ovo ugrađeno antivirusno rješenje nudi nekoliko vrsta skeniranja malicioznih datoteka poput brzog skeniranja, cjelokupnog skeniranja i skeniranje bez interneta. Tijekom testiranja iskorištena je opcija brzog skeniranja radi jednostavnosti procesa. Programu je bilo potrebno 24 sekunde da pronađe eicar.com datoteku gdje se prilikom otkrivanja skeniranje automatski prekida. Naposljetku, program pruža detaljne informacije o otkrivenim prijetnjama koje su prikazane na slici 11.



Slika 11. Pronađena prijetnja Microsoft Defender Antivirus-a

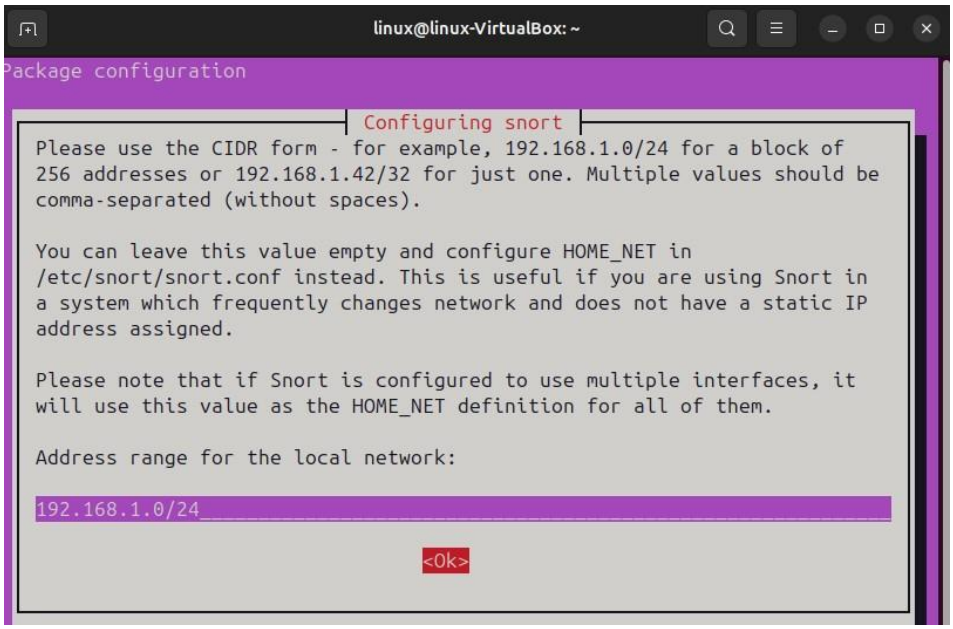
Uzimajući u obzir oba antivirusna rješenja, tijekom testiranja mogu se uočiti velike sličnosti u performansama. Oba programa vrlo brzo uočavaju prijetnje pomoću zaštite u stvarnom vremenu te stavljaju maliciozne datoteke u karantenu. Tijekom skeniranja, možemo uočiti razlike gdje Avast program može analizirati veliku količinu datoteka sustava u vrlo kratkom periodu, a Microsoft Defender Antivirus prilikom pronalaska prijetnje automatski zaustavlja daljnje skeniranje. Ugrađeni program sustava Windows omogućuje veću jednostavnost korištenja gdje nije potrebna instalacija i sučelje je jednostavno za interakciju s korisnikom. Avast, iako zahtjeva dodatnu instalaciju, nudi više naprednih opcija, uključujući detaljne izvještaje i dodatne postavke poput sandboxing-a i mrežne zaštite. Izbor antivirusne zaštite ovisi o korisnicima sustava – za osnovnu zaštitu i minimalan utjecaj na performanse, Microsoft Defender je adekvatan izbor, dok je Avast prikladniji za sustave koji traže naprednije značajke i detaljnije opcije zaštite.

5.2. Testiranje sustava za detekciju upada -Snort i Zeek

Testiranje mrežnog prometa odvijat će se unutar Linux Ubuntu virtualnog okruženja preko IDS alata Snort i Zeek. Ovo su dva vrlo poznata alata koja služe za nadzor mrežne sigurnosti. Snort predstavlja sustav za sprječavanje napada otvorenog koda unutar mreže pomoću niza pravila koja pokušavaju otkrivati zlonamjerne mrežne aktivnosti. Koristi se detekcijom potpisa te pomoću pravila, Snort može prepoznati i spriječiti pakete koji mogu ugroziti mrežni sustav te sukladno tome generira upozorenje za korisnika (Cisco Systems., bez dat.).Alat Zeek omogućuje duboku inspekciju mrežnog prometa gdje se osim sprječavanja prijetnji bavi i evidentiranjem događaja u mreži. Unutar mreža pokušava detektirati anomalije koje mogu ukazivati na sigurnosne prijetnje mrežnog prometa. (The Zeek Project., bez dat.)

5.2.1. Snort-detekcija potpisa

Instalacija Snort alata je vrlo jednostavna unutar Linux Ubuntu sustava te je i zbog tog razloga odabran kao okruženje u kojem će se testirati. Za instalaciju Snort-a potrebno je otvoriti terminal operativnog sustava i upisati naredbu „sudo apt-get install snort“. Nakon instalacije potrebnih paketa za alat slijedi konfiguracija Snort-a. Aplikacija automatski traži da se definira raspon mreže kako bi se lakše pratili i nadzirali mrežni paketi.



Slika 12. Konfiguracija mreže Snort-a

Nadalje, potrebno je preuzeti niz pravila sa službene web stranice pomoću naredbe „wget https://www.snort.org/downloads/community/snort3-community-rules.tar.gz“ unutar terminala. Ova poveznica uključuje veliku količinu postavljenih pravila kao što su pravila za otkrivanje mrežnih napada, zlonamjernog prometa, neželjenog prometa, specifičnih aplikacijskih napada i drugih. Kada su pravila preuzeta potrebno ih je raspakirati i premjestiti u direktorij „/etc/snort/rules/“. Snort je sada spreman za rad i detekciju potencijalnih prijetnji na mreži, a pokreće se naredbom: „sudo snort -A console -q -c /etc/snort/snort.conf -i enp0s3“. Naredba omogućuje da Snort radi u tihom načinu rada nadzirući enp0s3 mrežno sučelje s ispisom upozorenja direktno na konzolu.

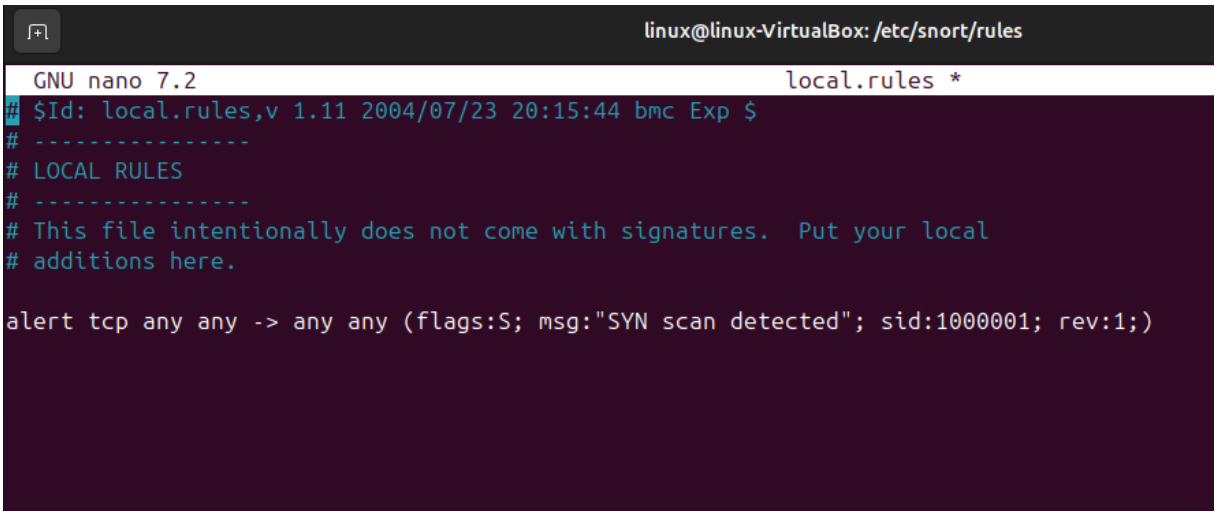
```
  ,,_     -*> Snort! <*-
 o"  )~   Version 2.9.20 GRE (Build 82)
  ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
          Copyright (C) 2014-2022 Cisco and/or its affiliates. All rights reserved.
          Copyright (C) 1998-2013 Sourcefire, Inc., et al.
          Using libpcap version 1.10.4 (with TPACKET_V3)
          Using PCRE version: 8.39 2016-06-14
          Using ZLIB version: 1.3

          Rules Engine: SF_SNORT_DETECTION_ENGINE  Version 3.2  <Build 1>
          Preprocessor Object: SF_DNS  Version 1.1  <Build 4>
          Preprocessor Object: SF_SSLPP  Version 1.1  <Build 4>
          Preprocessor Object: SF_FTPTELNET  Version 1.2  <Build 13>
          Preprocessor Object: SF_SDF  Version 1.1  <Build 1>
          Preprocessor Object: SF_POP  Version 1.0  <Build 1>
          Preprocessor Object: SF_SSH  Version 1.1  <Build 3>
          Preprocessor Object: SF_SMTP  Version 1.1  <Build 9>
          Preprocessor Object: SF_S7COMMPLUS  Version 1.0  <Build 1>
          Preprocessor Object: appid  Version 1.1  <Build 5>
          Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
          Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
          Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
          Preprocessor Object: SF_DCERPC2  Version 1.0  <Build 3>
          Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
          Preprocessor Object: SF_IMAP  Version 1.0  <Build 1>
          Preprocessor Object: SF_SIP  Version 1.1  <Build 1>
Commencing packet processing (pid=8302)
```

Slika 13. Pokretanje Snort-a (Autorski rad)

Nakon pokretanja, prikazano je i simulacija aktivnosti nad mrežnim sustavom pomoću nmap alata (eng. *Network Mapper*). Pomoću njega korisnici mogu skenirati mrežne sustave čime se otkrivaju otvoreni portovi i prikupljaju informacije o operativnim sustavima. Ovo je vrlo opasno za različite informacijske sustave jer napadači pomoću ovog sustava mogu izvoditi skrivena skeniranja u svrhu traženja osjetljivih informacija ili prepoznavanja ranjivih portova stoga je uloga alata poput Snort-a vrlo bitna. Instalacija nmap-a u Linux sustav se izvodi naredbom: „sudo apt-get install nmap“.

Za prikaz testiranja izvodi se sinkronizirano skeniranje (eng. *Synchronize scan* – SYN) ili poznatije kao poluotvoreno skeniranje (eng. *half-open scan*). Ovo je jedna od najčešćih korištenih tehnika za skeniranje i otkrivanje portova na mreži. SYN skeniranje iskorištava protokol s kontrolom prijenosa (eng. *Transmission Control Protocol* – TCP) kako bi uspostavio vezu na svaki postojeći port. Napadač šalje SYN paket na jedan od portova te ako dobije odgovor ACK paketom (eng. *Acknowledge*), dobiva informaciju da je port otvoren i ranjiv.(Hanna, 2021.).

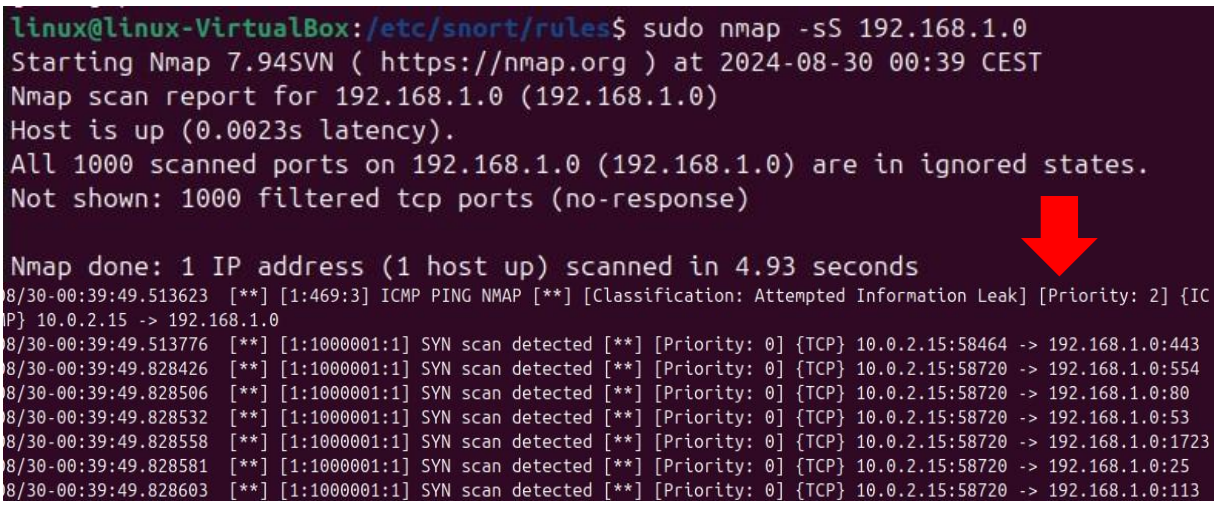
Za uočavanje aktivnosti na mrežama potrebno je postaviti pravilo za Snort program gdje je potrebno detektirati i upozoriti korisnika na sumnjive aktivnosti u mreži. Pravilo se postavlja preko putanje „/etc/snort/rules/local,rules/“ u datoteci „local.rules“ pomoću naredbe nano.



Slika 14. Postavljanje pravila za SYN skeniranje

Postavljeno pravilo služi za generiranje upozorenja u slučaju prepoznavanja paketa koji imaju postavljenu SYN zastavicu (flags:S). Unutar Snorta, generirat će se poruka „SYN scan detected“ prilikom otkrivanja SYN paketa. Pravilo vrijedi za bilo koji port kao izvor i kao odredište. Nakon postavljanje pravila, pokreće se SYN skeniranje preko nmap-a pomoću IP

adrese čiji je raspon zadan u početnoj konfiguraciji te isto vrijeme u drugom terminalu u kojem je pokrenut Snort, ispisuju se aktivnosti koje se odvijaju na mreži.



Slika 15. Pokretanje nmap-a i detekcija aktivnosti

Prema prikazanom, tijekom skeniranja nije otkriven nijedan port koji je otvoren sa zadane ip adrese što indicira na dobru zaštićenost sustava. Svi portovi su filtrirani ili blokirani od strane sigurnosnih mehanizama. Snort je uspješno detektirao sva SYN skeniranja. Na terminalu se prikazuje vrijeme detektiranja događaja, visina prioriteta 0 (visok prioritet) te broj porta s kojeg je poslano skeniranje.

5.2.2. Zeek-detekcija anomalija

Instalacija Zeek-a za otkrivanje anomalija unutar mreža je kompleksnija u odnosu na instalaciju Snorta. Za korištenje programa potrebni su dodatni paketi za pravilan rad te automatska konfiguracija paketa za instalaciju. Radi jednostavnosti, proces instalacije prikazan je u obliku koda čije se naredbe upisuju unutar terminala prema uputama dolje:

```
wget https://download.zeek.org/zeek-6.0.5.tar.gz
sudo apt-get install cmake make gcc g++ flex bison libpcap-dev
libssl-dev python3-dev swig zlib1g-dev

cd Downloads
tar -xzf zeek-<verison>.tar.gz


cd zeek-<version>
```

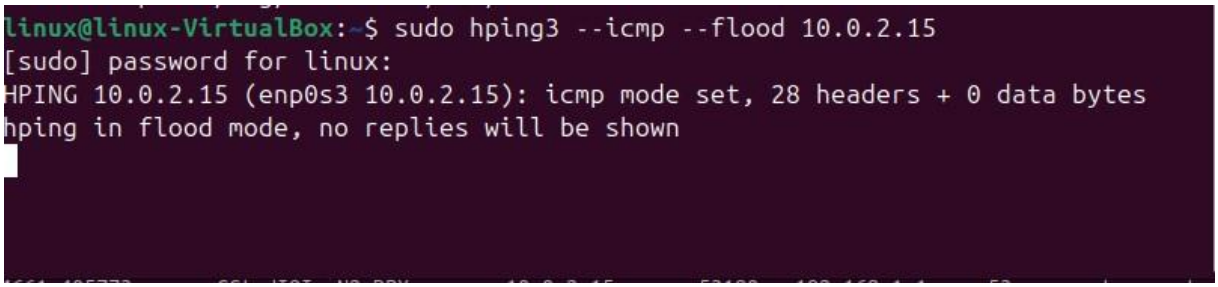
```
./configure

make
make install

source ~/.bashrc
which zeek
zeek –version

cd /usr/local/zeek/etc

zeekctl check

zeekctl deploy
```

Nakon procesa instalacije, Zeek se pokreće pomoću naredbe „zeekctl deploy“ te se pomoću naredbe „zeekctl status“ provjerava status alata, odnosno da li je pokrenut unutar Linux sustava.

```
root@linux-VirtualBox:/usr/local/zeek/etc# zeekctl deploy
checking configurations ...
installing ...
creating policy directories ...
installing site policies ...
generating standalone-layout.zeek ...
generating local-networks.zeek ...
generating zeekctl-config.zeek ...
generating zeekctl-config.sh ...
stopping ...
stopping zeek ...
starting ...
starting zeek ...
root@linux-VirtualBox:/usr/local/zeek/etc# zeekctl status
Name         Type       Host          Status    Pid    Started
zeek         standalone localhost     running   51005  02 Sep 11:56:29
root@linux-VirtualBox:/usr/local/zeek/etc#
```

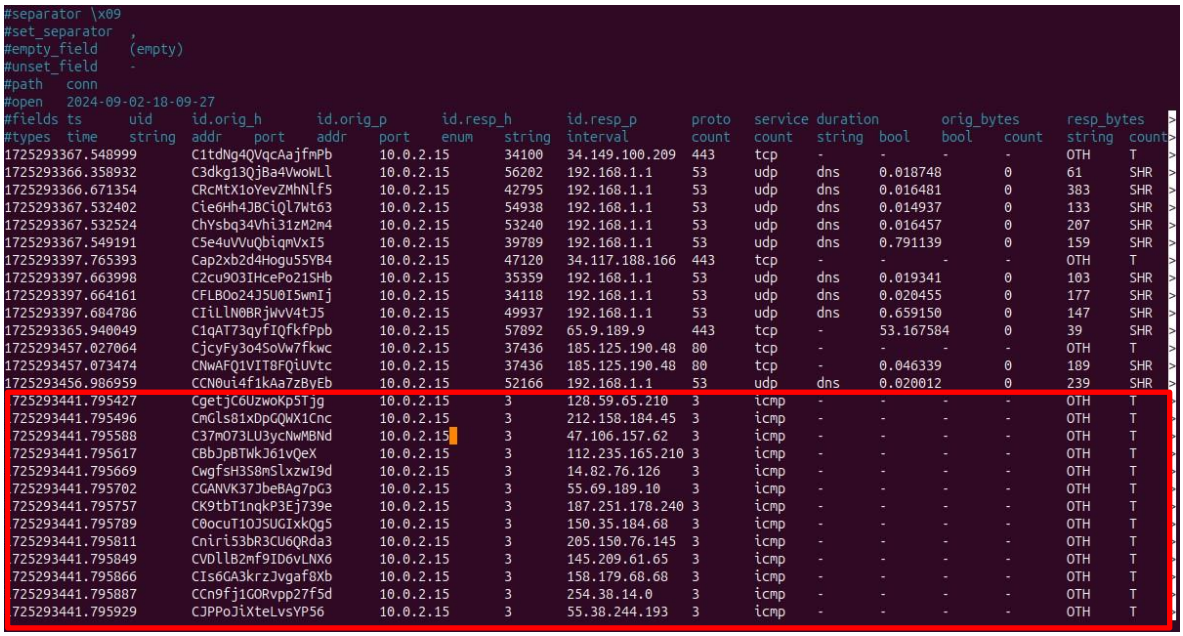
Slika 16. Pokretanje Zeek-a

Funkcionalnost Zeek-a testirana je pomoću hping alata. Hping omogućuje generiranje i slanje mrežnih paketa za potrebe procjene mrežnih performansi. Također, kao jednu od glavnih opcija sadrži „flood“ funkcionalnost gdje se generira veliki broj paketa te se opterećuje određena mreža slanjem velikog broja paketa ciljanom serveru. Ovo je primjer DoS (eng. *Denial of Service*) napada jer se opterećenjem mreže pokušava usporiti rad sistema ili ga učiniti potpuno nedostupnim. Prilikom testiranja prikazan je ICMP (eng. *Internet Control Message Protocol*) napad paketima. Ovaj protokol je dizajniran za razmjenu poruka između uređaja o stanju mreže i za prikaz upozorenja o greškama u komunikaciji. Za pokretanje napada koristi se naredba „sudo hping3 --icmp --flood 10.0.2.15“.



Slika 17. Pokretanje ICMP DoS napada

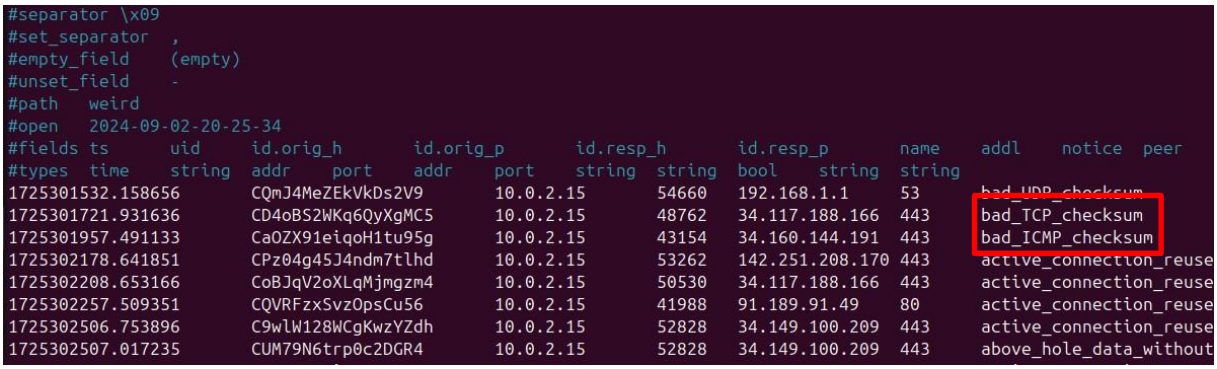
Ova naredba šalje veliki broj ICMP paketa prema ciljanoj IP adresi 10.0.2.15. Nakon pokretanja napada potrebno je unutar drugog terminala gdje je pokrenut Zeek postaviti se na putanju „/usr/local/zeek/logs/current“. Unutar mape nalaze se datoteke o zapisima mrežnih veza prema adresi koju Zeek nadgleda. Za potrebnu analizu najbitnije su datoteke „conn.log“ i „wierd.log“. U datoteci „conn.log“ prikazane su sve veze koje prolaze kroz zadanu IP adresu te uključuje podatke kao što su izvorna i odredišna IP adresa, portovi, vrijeme početka i završetka veze te veličinu prenesenih podataka. „Weird log“ bilježi sve događaje za koje alat smatra sumnjivim ili nepravilnim u mrežnom prometu. Ovi zapisi su vrlo bitni jer unutar njih korisnik identificira i analizira maliciozne napade na mrežu. Naredbom „nano“ ili „cat“ pregledava se „conn log“ zapisi, a potom i „weird log“.



```
#separator \x09
#set_separator  ,
#empty_field    (empty)
#unset_field    -
#path   conn
#open   2024-09-02-18-09-27
#fields ts      uid     id.orig_h       id.orig_p       id.resp_h       id.resp_p       proto   service duration        orig_bytes      resp_bytes
#types  time    string  addr    port    addr    port    enum    string  interval        count   count   string  bool    bool    count   string  count
1725293367.548999       C1tdNg4QVqcAajfmPb      10.0.2.15       34100   34.149.100.209  443     tcp     -       -       -       -       OTH     T
1725293366.358932       C3dkg13QjBa4VwoWLl      10.0.2.15       56202   192.168.1.1     53      udp     dns     0.018748        0       61      SHR
1725293366.671354       CRcMtX1oYevZMhNlf5      10.0.2.15       42795   192.168.1.1     53      udp     dns     0.016481        0       383     SHR
1725293367.532402       Cie6Hh4JBCiQl7Wt63      10.0.2.15       54938   192.168.1.1     53      udp     dns     0.014937        0       133     SHR
1725293367.532524       ChYsbq34Vhi31zM2m4      10.0.2.15       53240   192.168.1.1     53      udp     dns     0.016457        0       207     SHR
1725293367.549191       C5e4uVVuQbiqmVxI5       10.0.2.15       39789   192.168.1.1     53      udp     dns     0.791139        0       159     SHR
1725293397.765393       Cap2xb2d4Hogu55YB4      10.0.2.15       47120   34.117.188.166  443     tcp     -       -       -       -       OTH     T
1725293397.663998       C2cu9O3IHcePo21SHb      10.0.2.15       35359   192.168.1.1     53      udp     dns     0.019341        0       103     SHR
1725293397.664161       CFLBOo24J5U0I5wmIj      10.0.2.15       34118   192.168.1.1     53      udp     dns     0.020455        0       177     SHR
1725293397.684786       CIiLlN0BRjWvV4tJ5       10.0.2.15       49937   192.168.1.1     53      udp     dns     0.659150        0       147     SHR
1725293365.940049       C1qAT73qyfIQfkfPpb      10.0.2.15       57892   65.9.189.9      443     tcp     -       53.167584       0       39      SHR
1725293457.027064       CjcyFy3o4SoVw7fkwc      10.0.2.15       37436   185.125.190.48  80      tcp     -       -       -       -       OTH     T
1725293457.073474       CNwAFQ1VIT8FQiUVtc      10.0.2.15       37436   185.125.190.48  80      tcp     -       0.046339        0       189     SHR
1725293456.986959       CCN0ui4f1kAa7zByEb      10.0.2.15       52166   192.168.1.1     53      udp     dns     0.020012        0       239     SHR
1725293441.795427       CgetjC6UzwoKp5Tjg       10.0.2.15       3       128.59.65.210   3       icmp    -       -       -       -       OTH     T
1725293441.795496       CmGls81xDpCQWX1Cnc      10.0.2.15       3       212.158.184.45  3       icmp    -       -       -       -       OTH     T
1725293441.795588       C37mO73LU3ycNwMBNd      10.0.2.15       3       47.106.157.62   3       icmp    -       -       -       -       OTH     T
1725293441.795617       CBbJpBTWkJ61vQeX        10.0.2.15       3       112.235.165.210 3       icmp    -       -       -       -       OTH     T
1725293441.795669       CwgfsH3S8mSlxzwI9d      10.0.2.15       3       14.82.76.126    3       icmp    -       -       -       -       OTH     T
1725293441.795702       CGANVK37JbeBAg7pG3      10.0.2.15       3       55.69.189.10    3       icmp    -       -       -       -       OTH     T
1725293441.795757       CK9tbT1nqkP3Ej739e      10.0.2.15       3       187.251.178.240 3       icmp    -       -       -       -       OTH     T
1725293441.795789       C0ocuT1OJSUGIxkQg5      10.0.2.15       3       150.35.184.68   3       icmp    -       -       -       -       OTH     T
1725293441.795811       Cniri53bR3CU6QRda3      10.0.2.15       3       205.150.76.145  3       icmp    -       -       -       -       OTH     T
1725293441.795849       CVDllB2mf9ID6vLNX6      10.0.2.15       3       145.209.61.65   3       icmp    -       -       -       -       OTH     T
1725293441.795866       CIs6GA3krzJvgaf8Xb      10.0.2.15       3       158.179.68.68   3       icmp    -       -       -       -       OTH     T
1725293441.795887       CCn9fj1GORvpp27f5d      10.0.2.15       3       254.38.14.0     3       icmp    -       -       -       -       OTH     T
1725293441.795929       CJPPoJiXteLvsYP56       10.0.2.15       3       55.38.244.193   3       icmp    -       -       -       -       OTH     T
```

Slika 18. "conn log" zapisi

Unutar zapisa identificirani su ICMP paketi koji su rezultat pokretanja DoS napada označeni u crvenom pravokutniku na slici 18. U zapisima je naznačeno preko „service count“ kolone da je riječ o ICMP protokolu. Nadalje, prikazane su izvorna i odredišna IP adresa te su svi portovi ovih zapisa postavljeni statusom broja 3. Tijekom slanja ICMP zahtjeva portovi nisu bitni jer promatrani protokol ih ne koristi za komunikaciju. Paketi imaju stalnu odredišnu adresu dok se zahtjevi šalju različitim odredištima. Nadalje, paketi se šalju i primaju malim količinama bajtova što je karakteristično za ICMP zahtjeve. Unutar kolone „resp_bytes“ prikazuje se status „0TH“ što označava da nije bilo odgovora na poslani zahtjev prema ciljanom uređaju. Ovo može implicirati na DoS napad gdje uređaj ne odgovara na zahtjeve zbog velike preopterećenosti paketima. Za daljnu analizu, na slici 19 prikazan je sadržaj „weird.log“ zapisa.



```
#separator \x09
#set_separator  ,
#empty_field    (empty)
#unset_field    -
#path   weird
#open   2024-09-02-20-25-34
#fields ts      uid     id.orig_h       id.orig_p       id.resp_h       id.resp_p       name    addl    notice  peer
#types  time    string  addr    port    addr    port    string  string  bool    string  string
1725301532.158656       CQmJ4MeZEkVkDs2V9       10.0.2.15       54660   192.168.1.1     53      bad_UDP_checksum
1725301721.931636       CD4oBS2WKq6QyXgMC5      10.0.2.15       48762   34.117.188.166  443     bad_TCP_checksum
1725301957.491133       CaOZX91eiqoH1tu95g      10.0.2.15       43154   34.160.144.191  443     bad_ICMP_checksum
1725302178.641851       CPz04g45J4ndm7tlhd      10.0.2.15       53262   142.251.208.170 443     active_connection_reuse
1725302208.653166       CoBJqV2oXLqMjmgzm4      10.0.2.15       50530   34.117.188.166  443     active_connection_reuse
1725302257.509351       CQVRFzxSvzOpsCu56       10.0.2.15       41988   91.189.91.49    80      active_connection_reuse
1725302506.753896       C9wlW128WCgKwzYZdh      10.0.2.15       52828   34.149.100.209  443     active_connection_reuse
1725302507.017235       CUM79N6trp0c2DGR4       10.0.2.15       52828   34.149.100.209  443     above_hole_data_without
```

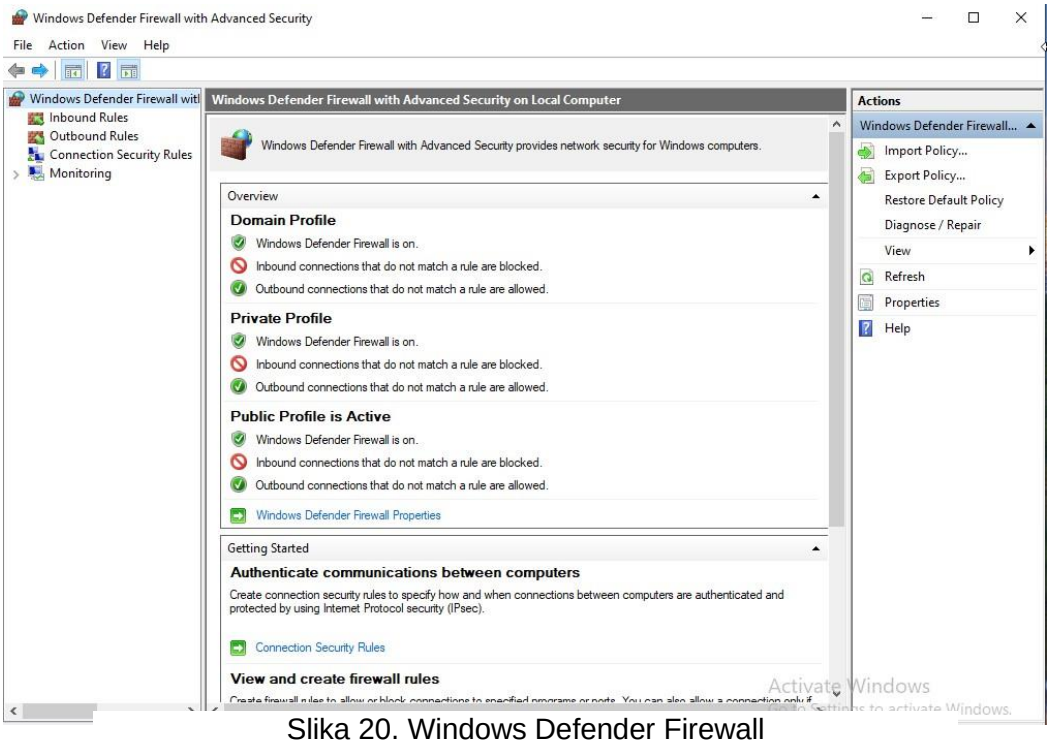
Slika 19. "weird log" zapisi

Ovdje su zapisane sve anomalije na promatranoj mreži. Najvažniji zapisi su označeni crvenom bojom, a sadržavaju upozorenja „bad_TCP_checksum“ i „bad_ICMP_checksum“. Oba upozorenja imaju ista značenja, ali različite protokole. Ove anomalije dodatno ukazuju na ICMP napad jer sugeriraju na svjesno oštećenje ili modificiranje mrežnih paketa u svrhu izbjegavanja sigurnosnih mjera i oštećivanja korištenog sustava. Svjesno oblikovanje ICMP poruka s nevaljanim kontrolnim zbrojem (eng. *checksum*) može učiniti obrambene mehanizme mreže zbunjenima, što može rezultirati nepravilnim tretiranjem prometa te dodatno izazvati ICMP odgovore ili fragmentaciju prometa. Ova anomalija, u kombinaciji s povećanim ICMP prometom, može biti jasan indikator prisutnosti ICMP napada koji je usmjeren na izazivanje poremećaja u mreži.

5.3. Testiranje vatrozida - Windows Defender Firewall

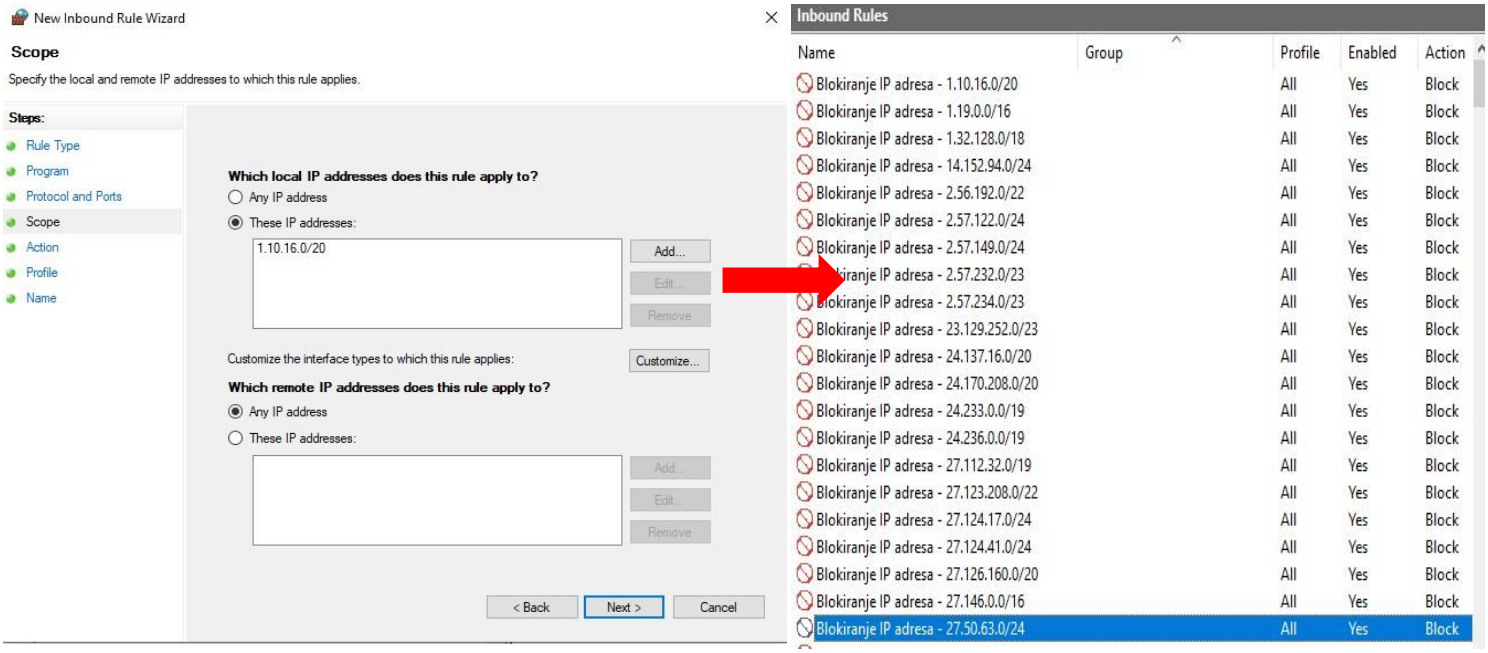
Testiranje vatrozida prikazano je pomoću Windows Defender Firewall sustava koji je automatski ugrađen unutar Windows operativnih sustava. Jedna od glavnih prednosti Windows Defender Firewall-a je njegova integracija s ostatkom Windows sustava, što omogućava lako upravljanje i ažuriranje kroz centar za sigurnost sustava Windows (eng. *Windows Security Center*). Korisnici mogu prilagoditi postavke, blokirati ili dopuštati određene aplikacije i portove, te konfigurirati pravila za određene mrežne profile (javne, privatne, ili domene). Osim osnovne zaštite, Windows Defender Firewall pruža napredne mogućnosti poput logiranja sigurnosnih događaja, što korisnicima i administratorima omogućava detaljan uvid u potencijalne prijetnje.

Za testiranje učinkovitosti firewall-a prikazan je test blokiranja pristupa poznatim zlonamjernim domenama ili IP adresama. Windows Defender Firewall-u se pristupa pomoću Windows tražilice u donjem lijevom kutu gdje se u tražilicu upisuje „Windows Defender Firewall with Advanced Security“. Klikom na nju, otvara se početno grafičko sučelje vatrozida.



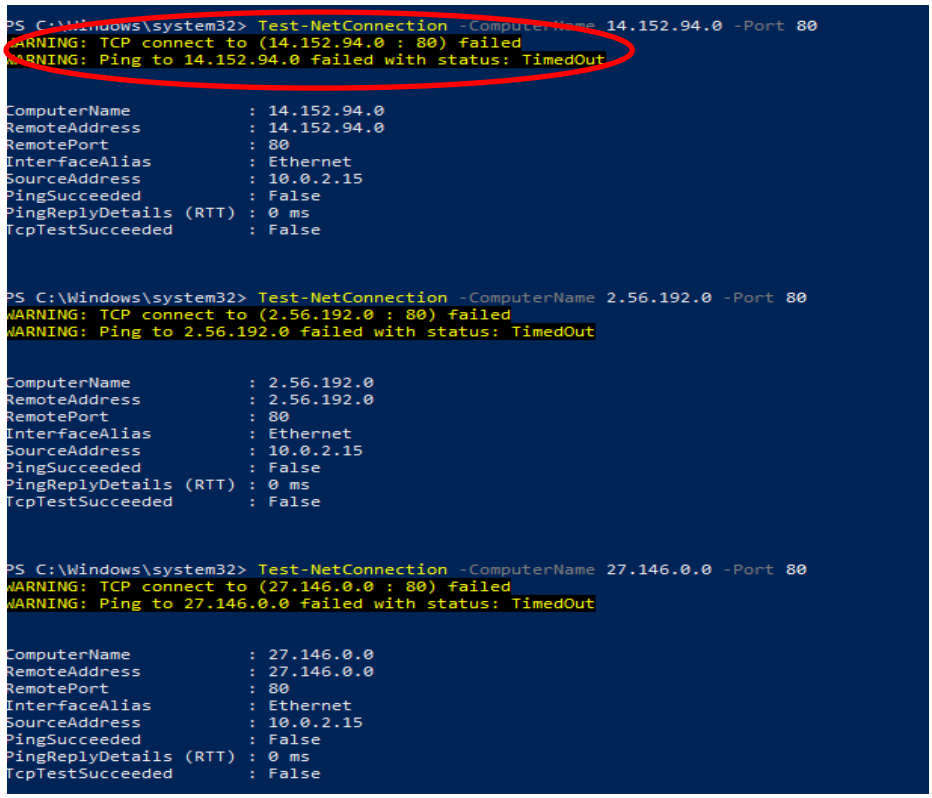
Slika 20. Windows Defender Firewall

Na početnom sučelju prikazane su osnovne informacije poput dostupnosti vatrozida, uključenosti pravila i sigurnosti komunikacije s mrežom. U svrhu testiranja, fokus je na stvaranja novih ulaznih pravila (eng. *Inbound rules*) čija se opcija nalazi u gornjoj lijevoj sekciji aplikacije. Pomoću ove opcije dodaju se maliciozne IP adrese čiji je popis preuzet sa službene stranice „Proofpoint“. Maliciozne IP adrese na popisu sadrže razne zlonamjerne aktivnosti poput botneta, phishing i DDoS napada. Zbog veličine sadržaja liste, dodano je prvih 20 IP adresa. Klikom na opciju „Inbound rules“ potrebno je zatim kliknuti na „New Rule“ opciju. Zatim, odabire se opcija „Custom“i „All Programs“. Nadalje, odabire se opcija „This IP adress or subnet“ gdje korisnik unutar polja dodaje željenu adresu koju želi blokirati. Korisnik po potrebi može dodati više adresa pomoću opcije „Add“ u „Scope“ odjeljku. Na kraju je potrebno odabrati opciju „Block the Connection“ te odabrati stavke za blokiranje adrese na domeni, lokalnoj i javnoj mreži.



Slika 21. Dodavanje IP adresa na "blocklistu"

Nakon dodavanja pravila za blokiranje određenih adresa, potrebno je testirati da li je pristup ovim adresama zaista blokiran. Pregled blokiranosti IP adresa može se provjeriti „ping metodom ili naredbom „Test-NetConnection“ unutar Windows PowerShell-a. Također, provjera prometa prema blokiranim adresama može se provjeriti pomoću službenih zapisa vatrozida u logovima. Za provjeru koristit će se „Test-NetConnection“ metoda. Unutar Windows PowerShella potrebno je upisati naredbu: „Test-NetConnection –ComputerName [IP Adresa]-Port 80“. Ovim se provjerava može li računalo uspostaviti TCP vezu s uređajem na odabranoj IP adresi. Prema popisu blokiranih IP adresa uzeto je nekoliko nasumičnih adresa u svrhu provjere povezanosti.



Slika 22. Provjera povezanosti blokiranih IP adresa

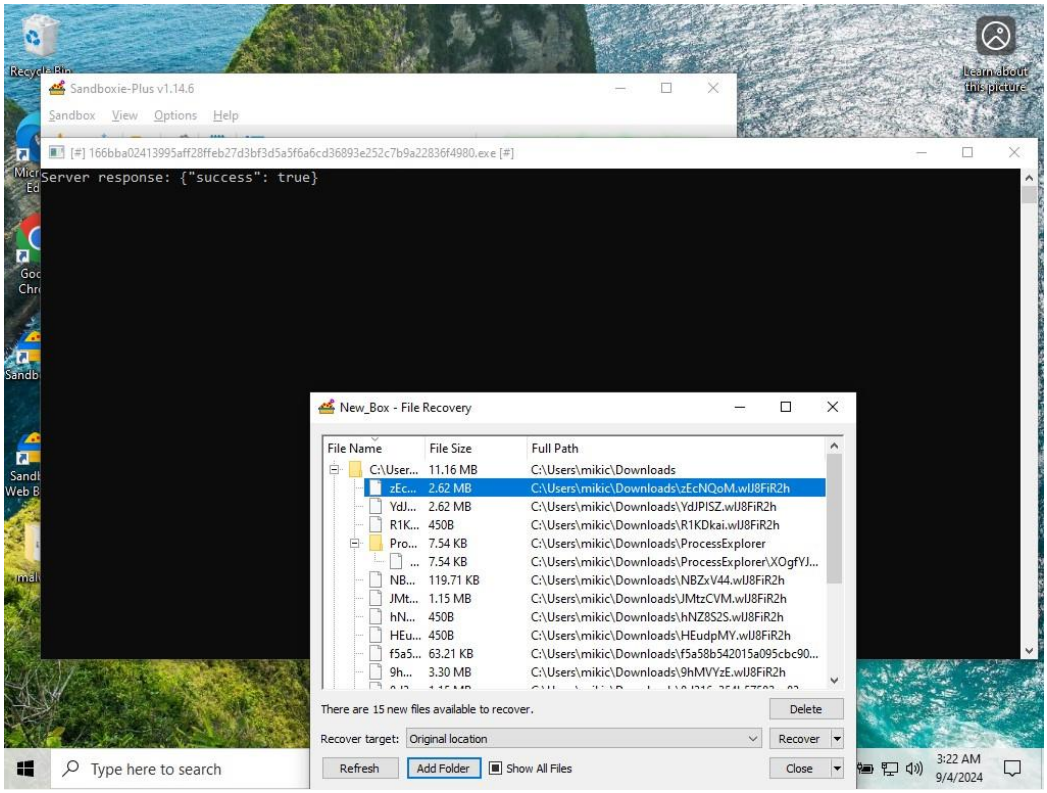
Prema slici 22. crvenom bojom je označen status povezanosti prema zadanoj adresi. Nakon pokušaja povezivanja, sustav izbacuje upozorenja o isteku TCP konekcije („TCP connect to [Naziv IP Adrese] failed“) i upozorenje o isteklom vremenu čekanja odgovora na ping zahtjev („Ping to [Naziv IP Adrese] failed with status: TimedOut“). Ovi pokazatelji utvrđuju da je Windows Firewall Defender uspješno blokirao i onemogućio povezivanje s malicioznim adresama.

5.4. Testiranje sandbox-a - Sandboxie

Testiranje malware-a u virtualnom okruženju prikazati će se pomoću Sandboxie alata. On omogućava postavljanje malicioznih prijetnji unutar izoliranog okruženja što znači da sve promjene koje maliciozni kod napravi dok je pokrenut unutar sandboxa ostaje unutar ograničenog prostora bez utjecaja na ostatak sustava. Sandboxie program se može preuzeti preko službene stranice, a instalacija se odvija preko čarobnjaka za instalaciju. Nakon instalacije, postupak testiranja malware-a uključuje kreiranje novog izoliranog okruženja unutar Sandboxie sučelja. Unutar sučelja mogu se konfigurirati različite postavke poput ograničavanja mrežnog prometa ili praćenja specifičnih datoteka i procesa. Pokretanjem napada unutar okruženja aplikacija prati sve promjene koje malware pokušava napraviti. Za prikaz rada

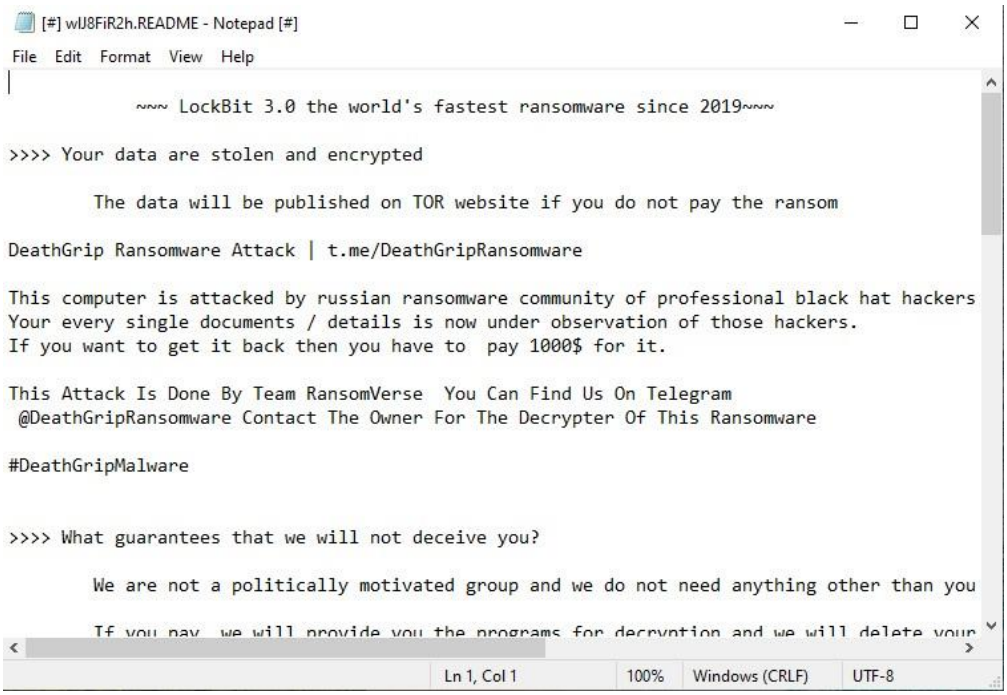
programa koriste se uzorak ransomware napada preuzet sa službene stranice „Malware Bazzar“ baze podataka.

Napad se pokreće preko početnog sučelja tako da se najprije stvara novo izolirano okruženje „New Box“ te desnim klikom se pokreće opcijom „Run“ i zatim „Run program“.Izolirano okruženje je pokrenuto kada su prikazani žute linije oko pokrenutog programa. Otvara se prozor preko kojeg se traži putanja datoteke koju želimo otvoriti. U ovom slučaju, najprije se otvara ransomware datoteka. Na slici 23 prikazano je pokretanje ransomware napada unutar izoliranog okruženja.



Slika 23. Ransomware napad

Ransomware je pokrenuo „powershell“ terminal kako bi dohvatio osobne podatke korisnika. U ovom slučaju pokušano je preuzimanje podataka sa korisnikove privatne usluge u oblaku. U donjem prozoru prikazane su sve datoteke koje su pokrenute prilikom napada. Nadalje, program je generirao poruku u obliku bloka za pisanje namijenjeno kao upozorenje korisniku o krađi osobnih podataka te plaćanja otpremnine za vraćanje osobnih informacija u korisnikovo vlasništvo.



Slika 24. Ransomware upozorenje

Na kraju provođenja testiranja i pokretanja napada, Sandboxie omogućuje automatsko brisanje svih datoteka i direktorija prouzrokovani malicioznim napadom. Ovo onemogućuje daljnje širenje malicioznih datoteka na sustav. Brisanje datoteka može se obaviti i ručno. Desnim klikom na izolirano okruženje odabiremo opciju „Delete content“ te nakon toga virtualno okruženje nije više izloženo napadu. Sandboxie je uspješno izolirao prijetnju i prikazao simulaciju napada s ciljem prevencije korisnika od korištenja ransmoware aplikacije.

6. Zaključak

U radu je prikazana problematika zaštite informacijskih sustava od zlonamjernog koda, uz poseban naglasak na njegove oblike, detekcijske tehnike i metode zaštite. Zlonamjerni kôd, poput virusa, trojanaca, crva, ransomwarea, spywarea i adwarea, predstavlja značajnu prijetnju za sigurnost suvremenih informacijskih sustava. U ovom dokumentu ispitane su vrste zlonamjernog softvera, načini napada, kao i prepreke koje sigurnosni stručnjaci susreću prilikom identifikacije i suzbijanja tih opasnosti. Testirane su i metode obrane, uključujući antivirusne programe, vatrozide, sustave za detekciju napada (IDS) i sandboxing tehnologiju, od kojih su sve evaluirane u virtualnim okruženjima.

Rezultati testiranja različitih alata u virtualnom okruženju, dali su jasnu sliku o njihovoj učinkovitosti u stvarnom okruženju. Kombinacija ovih metoda rezultirala je visokom razinom sigurnosti. Antivirusni alati pružili su osnovnu zaštitu, sustavi za detekciju napada (Snort i Zeek) osigurali su nadzor i zaštitu od mrežnih prijetnji, dok je vatrozid osigurao sigurnost mrežnog prometa kroz filtriranje IP adresa. Sandboxing je dodatno osigurao izolaciju potencijalno opasnih aplikacija, smanjujući rizik za glavni sustav. Sve metode zajedno pružaju sveobuhvatan sustav zaštite koji značajno smanjuje rizik od malicioznih napada na informacijske sustave.

Imajući na umu brzinu kojom se zlonamjerni kodovi razvijaju i prilagođavaju, od izuzetne je važnosti neprekidno unapređivati metode obrane i tehnološke alate. Ključna komponenta u smanjenju rizika od napada jest edukacija korisnika, budući da mnogi napadi iskorištavaju ljudske pogreške ili nedovoljno znanje. Uvođenje sigurnosnih politika, redovito ažuriranje softvera i obrazovanje zaposlenika ključni su koraci u smanjenju prijetnji zlonamjernog koda.

Ovaj rad pokazuje da je obrana od zlonamjernog koda u informacijskim sustavima proces koji zahtijeva dubok i dinamički pristup. Iako su testirani alati pružili zadovoljavajuću razinu sigurnosti, napadači neprestano razvijaju nove i sofisticiranije metode zaobilaženja obrambenih mehanizama. Stoga je nužno kontinuirano ulagati u istraživanje i razvoj novih tehnologija koje će moći pružiti višu razinu zaštite i efikasno otkriti nove prijetnje. U budućnosti bi se istraživanja mogla usmjeriti na primjenu umjetne inteligencije i strojnog učenja u prepoznavanju anomalija i predviđanju novih oblika zlonamjernog koda, što bi povećalo otpornost sustava na nepoznate prijetnje.

Literatura

[1] *Adware vs. Spyware: What Is the Difference?* (bez dat.). Preuzeto 8. travnja 2024. iz Cisco: https://www.cisco.com/c/en/us/products/security/adware-vs-spyware.html

[2] Aycock, J. a. (2011). *Spyware and Adware.* Njemačka: Springer US. Preuzeto 8. travnja 2024. iz https://books.google.hr/books?hl=hr&lr=&id=UKNgoM3nLe0C&oi=fnd&pg=PR3&ots=IWxDW_9Y-p&sig=82MDCs0gRsghxVot5LOwgLLXUos&redir_esc=y#v=onepage&q&f=false

[3] Choudary, S., Saroha, R., & Mrs. Beniwal, S. (4. Travanj 2013). How Anti-virus Software Works??. *International Journal(3(4))*, 483-484. Preuzeto 8. travnja 2024. iz https://www.researchgate.net/publication/308800880_How_Anti-virus_Software_Works

[4] CSF tools: Preuzeto 17. lipnja 2024 https://csf.tools/reference/nist-sp-800-171/r2/3-14/3-14-2/

[5] Cybersecurity and Infrastructure Security Agency . (27. rujan 2019). *Understanding Anti-Virus Software*. Preuzeto 8. travnja 2024. iz Cybersecurity and Infrastructure Security Agency: https://www.cisa.gov/news-events/news/understanding-anti-virus-software

[6] *Fortinet*. (bez dat). Preuzeto 10. travanja 2024. iz *Trojan Horse Virus*: https://www.fortinet.com/resources/cyberglossary/trojan-horse-virus

[7] *kaspersky*. (bez dat.). Preuzeto 10. travnja 2024. iz *Types of Malware*: https://www.kaspersky.com/resource-center/threats/malware-classifications

[8] L. Allen, T. Heriyanto, S. Ali. (2014) *Kali Linux – Assuring Security by Penetration Testing, Packt Publishing*

[9] Lalić, D. (2021). EUROPSKI PROJEKT SAFETY4RAILS. *Željeznice 21*(20), str. 27-32. Preuzeto 10.travnja 2024. iz https://hrcak.srce.hr/265201

[10] NoypiGeeks. (2023, 5. Svibanj). *The story behind the ILOVEYOU virus that caused $10 billion in damages worldwide*. Preuzeto 10. travnja 2024. iz https://www.noypigeeks.com/featured/iloveyou-virus/

[11] Lessing, M. (Rujan 2020). *Case Study: AIDS Trojan Ransomware*. Preuzeto 10. travnja iz sdxcentral: https://www.sdxcentral.com/security/definitions/what-is-ransomware/case-study-aids-trojan-ransomware/

[12] Liska, A., & Gallo, T. (2016). *Ransomware: Defending Against Digital Extortion.* Sjedinjene Američke Države: O'Reilly Media. Preuzeto 12. travnja 2024. iz https://books.google.hr/books?hl=hr&lr=&id=IIORDQAAQBAJ&oi=fnd&pg=PR2&dq=+Ransomware:+Defending+Against+Digital+Extortion&ots=EwnmzE6Xfe&sig=mB8UKD-MW5TvqZWuN5CPlARWGME&redir_esc=y#v=onepage&q=Ransomware%3A%20Defending%20Against%20Digital%20Extortion&f=false

[13] Mirza, M. B. (2014). Malicious Software Detection, Protection & Recovery Methods: A Survey. *BRIS Journal of ADV S&T*, 14-23. Preuzeto 12. travnja 2024. iz BRIS Journal of ADV S&T.

[14] Raffa, G. (2021). Testing Antivirus in Linux: An Investigation on the Effectiveness of Solutions Available for Desktop Computers; *Information Security Group Royal Holloway University of London Egham, Surrey, TW20 0EX United Kingdom*

[15] Saeed, A. I., Selamat, A., & Abuagoub, M. A. (Travanj 2013). Preuzeto 11. svibnja 2024. iz *A Survey on Malware and Malware Detection Systems:* https://www.researchgate.net/profile/Imtithal Saeed/publication/272238656_A_Survey_on_Malwares_and_Malware_Detection_Systems/links/566284c608ae192bbf8cf1a5/A-Survey-on-Malwares-and-Malware-Detection-Systems.pdf

[16] Team, C. E. (bez dat.). *O virusima*. Preuzeto 11. svibnja 2024. iz CERT.hr: https://www.cert.hr/virusi/

[17] Zhenfang, Z. (Kolovoz 2015). *International Journal of Engineering and Applied Sciences.* Preuzeto 11. svibnja 2024. iz Study on Computer Trojan Horse Virus and

Its Prevention: https://www.neliti.com/publications/257840/study-on-computer-trojan-horse-virus-and-its-prevention

[18] Asmaa Shaker, A., & Sharad, G. (2011). *Importance of Intrusion Detection System (IDS)*. *2*(1), 1–4. Preuzeto 12. svibnja 2024. iz https://portal.arid.my/Publications/f3da7cd3-5bab-4294-94d1-6a22c1d4235d.pdf

[19] Hrvatska akademska i istraživačka mreža (CARNet). (2000*). Detekcija neovlaštenih upada (IDS)*. Preuzeto 12. svibnja 2024. iz https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2000-09-04.pdf

[20] Virus a retrospective: Preuzeto 19. lipnja 2024. iz https://cs.stanford.edu/people/eroberts/cs201/projects/viruses/anti-virus.html

[21] Kwon, H., Kim, T., & Lee, M. (2022). *Advanced intrusion detection combining signature-based and behavior-based detection methods.* Electronics, 11(6), 867.Preuzeto 20. lipnja 2024. iz https://www.mdpi.com/2079-9292/11/6/867

[22] SIS Wiki. (bez dat.). *Detekcija i prevencija upada - IDS/IPS.* Security.foi.hr. Preuzeto 20.lipnja 2024. iz https://security.foi.hr/wiki/index.php/Detekcija_i_Prevencija_upada_-_IDS/IPS.html

[23] N-able. (2021, ožujak 15). *Intrusion Detection System (IDS): Signature vs. Anomaly-Based*. Preuzeto 13. svibnja 2024. iz https://www.n-able.com/blog/intrusion-detection-system

[24] Wack, J., Cutler, K., & Pole, J. (2002). *Guidelines on firewalls and firewall policy.* NIST special publication, 800, 41. Preuzeto 13. svibnja 2024. iz https://corpora.tika.apache.org/base/docs/govdocs1/663/663364.pdf

[25] Cisco. (bez dat.). *What is a firewall?* Preuzeto 20. lipnja 2024. iz https://www.cisco.com/site/us/en/learn/topics/security/what-is-a-firewall.html

[26] Katić, D. (2024, April 16*). Uloga hipervizora u virtualizaciji*. Retrieved September 4, 2024, from https://www.dalibor-katic.com/2024/04/16/uloga-hipervizora-u-virtualizaciji/

[27] Tomas, K. (2022). *Sigurnosni aspekti korporativnih podataka mobilnih uređaja u privatnom vlasništvu* (Undergraduate thesis). Zagreb: University of Zagreb, Faculty of Transport and Traffic Sciences. Preuzeto 13. svibnja 2024. iz https://urn.nsk.hr/urn:nbn:hr:119:851527

[28] Borate, I., & Chavan, R. K. (2016). Sandboxing in linux: From smartphone to cloud. *International Journal of Computer Applications*, *148*(8). Preuzeto 13. svibnja 2024. iz https://www.researchgate.net/profile/Raosaheb-Chavan/publication/306127865_Sandboxing_in_Linux_From_Smartphone_to_Cloud/links/580f45df08aef2ef97afc02f/Sandboxing-in-Linux-From-Smartphone-to-Cloud.pdf

[29] Egele, M., Scholte, T., Kirda, E., & Kruegel, C. (2008). A survey on automated dynamic malware-analysis techniques and tools. *ACM computing surveys (CSUR)*, *44*(2), 1-42. Preuzeto 14. svibnja 2024. iz https://sites.cs.ucsb.edu/~chris/research/doc/acmsurvey12_dynamic.pdf

[30] Raffetseder, T., Kruegel, C., & Kirda, E. (2007). Detecting system emulators. In *Information Security: 10th International Conference, ISC 2007, Valparaíso, Chile,* October 9-12, 2007. Proceedings 10 (pp. 1-18). Springer Berlin Heidelberg. Preuzeto 14. svibnja 2024. iz https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=5d6c145bd91829229fbc63e18130e135eedf7132

[31] Rescorla, E. (2005). Is finding security holes a good idea?. *IEEE Security & Privacy*, 3(1), 14-19. Preuzeto 14. svibnja 2024. iz https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=07738e922945a8e818de04f329ea3d206012830c

[32] Čisar, P., & Joksimović, D. (2019). Heuristic scanning and sandbox approach in malware detection. *Archibald Reiss Days*, *9*(2). Preuzeto 15. svibnja 2024. iz https://www.researchgate.net/profile/Raosaheb-Chavan/publication/306127865_Sandboxing_in_Linux_From_Smartphone_to_Cloud/links/580f45df08aef2ef97afc02f/Sandboxing-in-Linux-From-Smartphone-to-Cloud.pdf

[33] Lenić, M. (2022). *Sigurnost krajnjih uređaja u korporativnom okruženju* (Diplomski rad). Osijek: Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike,

računarstva i informacijskih tehnologija Osijek. Preuzeto 15. svibnja 2024. iz https://urn.nsk.hr/urn:nbn:hr:200:470361

[34] T. Kalsi (2018) *Practical Linux Security Cookbook: Secure your Linux environment from modernday attacks with practical recipes*, 2nd Edition, Packt Publishing

[35] Egele, M., Scholte, T., Kirda, E., & Kruegel, C. (2012). *A Survey on Automated Dynamic Malware Analysis Techniques and Tools*. ACM Computing Surveys, 44(2), 1-42. Preuzeto 22. kolovoza 2024. iz: https://dl.acm.org/doi/10.1145/2089125.2089126

[36] Pahl, C., Brogi, A., Soldani, J., & Jamshidi, P. (2019). Cloud Container Technologies: A State-of-the-Art Review. IEEE Transactions on Cloud Computing, 7(3), 677-692. Preuzeto 22. kolovoza 2024. iz: https://ieeexplore.ieee.org/document/8422755

[37] Schultz, M. G., Eskin, E., Zadok, E., & Stolfo, S. J. (2001). Data mining methods for detection of new malicious executables. In Proceedings 2001 IEEE Symposium on Security and Privacy. S&P 2001 (pp. 38-49). IEEE. Preuzeto 22. kolovoza iz: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=ba889041ff5ed801b4d5e14a740a35c3c4254ee0

[38] Lazarevic, A., Ertoz, L., Kumar, V., Ozgur, A., & Srivastava, J. (2003,). *A comparative study of anomaly detection schemes in network intrusion detection*. In Proceedings of the 2003 SIAM international conference on data mining, 25-36. Society for Industrial and Applied Mathematics. Preuzeto 22. kolovoza 2024. iz https://epubs.siam.org/doi/pdf/10.1137/1.9781611972733.3

[39] Federal Bureau of Investigation (FBI). (2019, 25 ožujak). *Melissa Virus: 20th Anniversary of Fast-Moving Computer Virus*. Preuzeto 24. kolovoza iz https://www.fbi.gov/news/stories/melissa-virus-20th-anniversary-032519

[40] Proofpoint. (n.d.). Zeus Trojan (Zbot). Proofpoint. Preuzeto 25. kolovoza iza https://www.proofpoint.com/us/threat-reference/zeus-trojan-zbot

[41] TechTarget. (bez dat.). WannaCry ransomware. TechTarget. Preuzeto 25. kolovoza iz https://www.techtarget.com/searchsecurity/definition/WannaCry-ransomware

[42] Cisco Systems. (bez.dat.). *Snort - Network intrusion detection & prevention system*. Preuzeto 30. kolovoza 2024. iz https://www.snort.org/

[43] The Zeek Project. (bez dat.). *Network security monitoring*. Preuzeto 30. kolovoza 2024 iz https://zeek.org/

[44] Hanna, K. T. (2021, srpanj). *SYN scanning*. TechTarget. Preuzeto 30. kolovoza 2024. iz https://www.techtarget.com/searchnetworking/definition/SYN-scanning

[45] Proofpoint. (bez dat.). *Emerging Threats Block IPs*. Preuzeto 31.kolovoza iz https://rules.emergingthreats.net/fwrules/emerging-Block-IPs.txt

[46] Timonera, K. P. (2023, 24. Ožujak). *How to prevent malware: 15 best practices for malware prevention*. eSecurityPlanet. preuzeto 2. rujna 2024. iz https://www.esecurityplanet.com/threats/how-to-prevent-malware/

[47] Europol. (2021, January 27). *World's most dangerous malware Emotet disrupted through global action*. Europol. Preuzeto 3. rujna 2024 iz https://www.europol.europa.eu/media-press/newsroom/news/world%E2%80%99s-most-dangerous-malware-emotet-disrupted-through-global-action

[48] CM Alliance. (2022, July 14). *5 major ransomware attacks of 2022*. Cyber Management Alliance. Preuzeto 3. rujna 2024 iz https://www.cm-alliance.com/cybersecurity-blog/5-major-ransomware-attacks-of-2022

Popis slika

Popis tablica