Zbornik radova Fakulteta tehničkih nauka, Novi Sad

UDK: 621.39
DOI: https://doi.org/10.24867/05BE28Gavric

MODEL DDOS NAPADA APLIKACIONOG SLOJA *GAMING* SERVERA EKSPLOATACIJOM KLIJENTSKE APLIKACIJE

MODEL OF APPLICATION LAYER BASED DDOS ATTACK ON A GAMING SERVER BY EXPLOITING THE CLIENT APPLICATION

Nikola Gavrić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – *U radu se analiziraju problemi i posledice DdoS napada na aplikacionom sloju gaming servera. Postavljena je hipoteza, da deo tih napada nastaje eksploatacijom klijentske aplikacije. Radi potvrde postavljene hipoteze, izvršena je simulacija napada u laboratijskim uslovima. Dobijeni su rezulati u formi logova na serveu koji su prikazani u grafičkoj formi. Njihovom evaluacijom došlo se do određenih zaključaka na osnovu kojih je predloženo je rešenje za zaštitu od ovih napada, koje se bazira na primeni machine learning tehnike na mrežnim firewall-ima.*

Ključne reči: *DDoS, protocol-specific DDoS, DDoS u gaming industriji, firewall, machine learning*

Abstract – *This paper analyzes problems and consequences of application layer based DDoS attacks on gaming servers. The hypothesis is that some of these attacks are caused by exploitation of the client application. A simulation was performed in a laboratory in order to prove these assumptions. The obtained logs from the server are plotted. There are several conclusions based on the data gathered through the simulation process and there's a suggested solution relying on machine-learning based network firewalls.*

Keywords: *DDoS, protocol-specific DDoS, DDoS in gaming industry, firewall, machine learning.*

1. UVOD

Na samom početku računarstva, pojam bezbednosti se uglavnom vezivao za zaštitu podataka na izolovanim računarima. Nastankom računarskih mreža i njihovim permanentnim širenjem i razvojem, naročito sa aspekta primene novih, naprednijih tehnologija i servisa, pojavili su se brojni i veoma sofisticirani izazovi u oblasti zaštite podataka koji se razmenjuju različitim komunikacionim kanalima. Motivi za napade su razni, od hakerskog aktivizma do iznude i potiskivanja konkurencije. Napadi su u početku bili usmereni ka krajnjim korisnicima, sa ciljem da im se ubacivanjem malicioznog softvera degradiraju performanse računara i ukradu važni podaci (passwordi, brojevi kreditnih kartica i sl.). Drugim rečima, ubacivanjem malicioznog softvera preuzima se kontrola nad korisničkim računarima, radi izvođenja koordinisanih napada. S druge strane, napadi usmereni ka velikim sistemima realizuju se najčešće u formi DDoS (eng. *Distributed Denial of Service*) napada, čiji je osnovni cilj da izazovu prekid korišćenja određenih servisa i naruše performanse mreže, a da se pri tome sakrije identitet napadača.

Sa promenama u društvu, menjali su se motivi, ali i metode koje su napadači koristili u kreiranju DDoS napada, ali je cilj ostao isti, da se korisnicima onemogući ili oteža pristup određenim servisima ili resursima i da se sakrije identitet. Prvobitni napadi ovog tipa bili su DoS (eng. *Denial of Service*) napadi i bili su usmereni na mreže, sisteme i određene servise.

Generalno posmatrano, DDoS napadi se mogu izvesti na bilo kojem sloju mrežne komunikacije, ali zavisno od toga na šta su usmereni napadi i kako se izvode mogu podeliti u sledeće tri kategorije:

1. Napadi na propusni opseg (eng. *bandwidth*) - usmereni su na preopterećenje tj. narušavanje performansi mrežnih resursa u pogledu protoka ili propusne moći uređaja, a u slučaju napada velikog intenziteta mogu dovesti do zagušenja i prekida linka između provajdera i korisnika (kod zagušenja dolazi i do usporavanja korisničkog saobraćaja, pojave retransmisija koje proizvode dodatni saobraćaj i kašnjenja).
2. Protokolski napadi - izvode se preko tačno određenih protokola, koristeći njihove nesavršenosti.
3. Aplikacioni napadi - ciljaju softverske nedostatke aplikacija na kojima se baziraju korisnički servisi, kao npr. nedostatke web servera.

Na aplikacionom sloju, DDoS napadi se mogu podeliti u tri grupe [1]:

1. Napadi koji se izvode pravom „poplavom" novih sesija na resursima u mreži (eng. Session flooding) - izvodljivi su isključivo na jednostavnim aplikacijama koje ne podrazumevaju verifikaciju i autorizaciju korisnika (npr. veb sajtovi) i nisu predmet ovog rada, iz razloga što to nije svojstveno za gaming servere.
2. Poplava i pravo bombardovanje novim zahtevima (eng. Request flooding)
3. Asimetrični napadi – nastaju kada zahtev za izvršavanje neke operacije koristi znatno manje resursa na klijentskoj strani, naspram same operacije koja se izvršava na serveru (ovi napadi

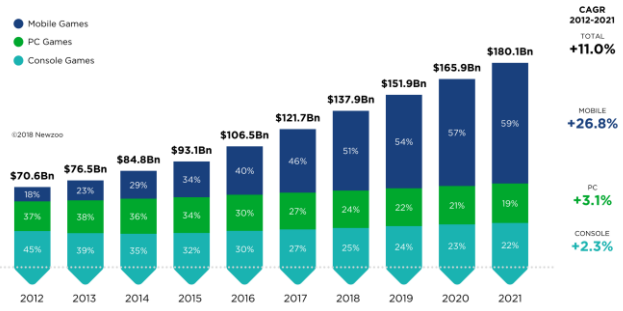
---

NAPOMENA:
Ovaj rad proistekao je iz master rada čiji mentor je bio doc. dr Živko Bojović.

su takve prirode da iziskuju znatne hardverske resurse servera).

2. DDoS NAPADI U INDUSTRIJI IGRICA

Većina napada na aplikacionom sloju cilja HTTP, pa je s toga *cybersecurity* industrija isključivo usmerena ka zaštiti web sajtova i web servisa. Ovaj rad se fokusira na aplikacioni sloj, specifično na *gaming* aplikacije, jer *gaming* industrija trenutno beleži velik porast i rapidnu ekspanziju. Biće demonstrirane potpuno nove metode DDoS napada koje proizilaze iz eksploatacije klijentskih aplikacija i metodi zaštite od istih, sa osvrtom na postojeće mehanizme i njihove nedostatke.



Slika 1. Rast gaming industrije [2]

Potrebno je naglasiti da je većina industrije igrica orijentisana ka tzv. *online gaming*-u. U ovom radu pažnja je usmerena prvenstveno ka *online gaming*-u na PC platofrmi, s time što su koncepti koji će biti prikazani, uz male modifikacije primenljivi na mobilnoj i konzolnoj platformi.

Aplikacije na *gaming* serverima su takve prirode da zahtevaju znatne hardverske i mrežne resurse. Tome doprinose novi trendovi, na primer da se dozvoljava pokretanje aplikacije bez posedovanja potpunog sadržaja iste, a da se zatim tokom korišćenja aplikacije ostatak podataka (ili čak samo neophodni podaci) preuzima sa nekog servera. Još jedan bitan faktor koji utiče na upotrebu resursa servera jesu dozvoljene modifikacije klijentske aplikacije (eng. *Mods/add-ons*), koje mogu da predstavljaju dodatan stepen slobode za napdače.

2.1. Eksploatacija klijentske aplikacije

Kada se govori o bezbednosti klijentske aplikacije, najčešće se misli na zaštitu od napada u smislu prevencije mogućnosti da korisnik aplikacije na nedozvoljen način ostvari prednost u odnosu na ostale korisnike ili drugu vrstu dobiti.

Glavni cilj pri kreiranju aplikacije jeste njena funkcionalnost, a zatim sve ostalo. Jedan od glavnih uzroka ovakve strategije jeste činjenica da mnoge kompanije teže da aplikacije puste u prodaju pre krajnjeg roka (eng. *Deadline*) po svaku cenu, jer to igra veliku ulogu pri inicijalnoj prodaji. Ovakav pristup ne ostavlja dovoljno vremena za intenzivno testiranje i svodi se na popravke *bag*-ova i poboljšanje aplikacije u njenim kasnijim revizijama. Primenjeni odbrambeni mehanizmi u *online gaming* aplikacijama se mogu podeliti na:

1. Potpunu zaštitu na serverskoj strani (eng. *Server-sided checking/protection*)
2. Distribuiranu zaštitu

Potpuna zaštita na serverskoj strani podrazumeva da se kompletno procesiranje vrši na serverskoj aplikaciji. Ovakav pristup obezbeđuje bolju zaštitu od napada, ali po cenu veće hardverske zahtevnosti i većeg kašnjenja u procesiranju zbog provera dolazećih paketa, koji moraju biti u skladu sa pravilima. Suštinska prednost ovog metoda naspram distribuiranog je u tome što napadačima smanjuje stepen slobode.

Pod pojmom distribuirana zaštita se podrazumeva sistem kod koga se deo procesiranja vrši na klijentskoj strani (određene varijable koje se koriste u procesiranju prisutne su samo na klijentskoj aplikaciji), dok je uloga servera da koordiniše rad klijenata, obavlja deo procesiranja, vrši određene provere i dr.

2.2. Trenutno stanje u oblasti zaštite od DDoS napada

Problem zaštite od DDoS napada predstavlja danas jedan od gorućih problema u *gaming* industriji i šire. Metode kojima se pokušava obezbediti zaštita od DDoS napada se mogu podeliti u dve osnovne grupe (faze):

1. Prevencija DDoS napada
2. Ublažavanje DDoS napada (eng. *mitigation*) koji su u toku.

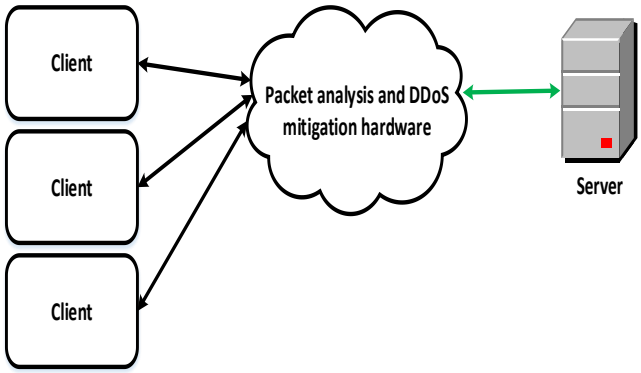
Prilikom dizajniranja aplikacije, odbrambeni mehanizmi nisu u prvom planu i postoje tendencije da se ceo *cybersecurity* domen prepusti provajderu takve usluge (npr. *Cloudflare*). Ovakva rešenja se nazivaju eksternim, ona su najjeftinija i najbrža, ali sa sobom povlače i određene nedostatke.

Najveći nedostatak ovakvih rešenja jeste to što su redno izvedena (slika 2). Naime, radi jednostavnosti implementacije, izdvoje se određeni resursi na cloudu koji se potom ponašaju kao *firewall* i *proxy* server za komunikaciju sa klijentima.

Loša strana redne implementacije jeste u dodatnom kašnjenju koje prouzrokuje fizička ili bolje rečeno geografska udaljenost između *game* servera i *cybersecurity* servera provajdera. Potrebno je istaći da dodatno kašnjenje mogu da unesu algoritmi za detekciju napada i algoritmi za druge vidove zaštite koji se ne odnose na DDoS. U HTTPu ovo nije veliki problem jer navedena kašnjenja gotovo da ne utiču na QoS i QoE.

Međutim, posmatrano sa aspekta *gaming*-a se najčešće (zbog *real-time* prirode) zahtevaju što manje vrednosti kašnjenja. Neke prihvatljive vrednosti kašnjenja su do 50ms, pa je svaka eliminacija kašnjenja dragocena. Rešenja ovog problema su uglavnom privatna i postoje razne implementacije koje su prilagođene specifičnim potrebama nekog preduzeća.

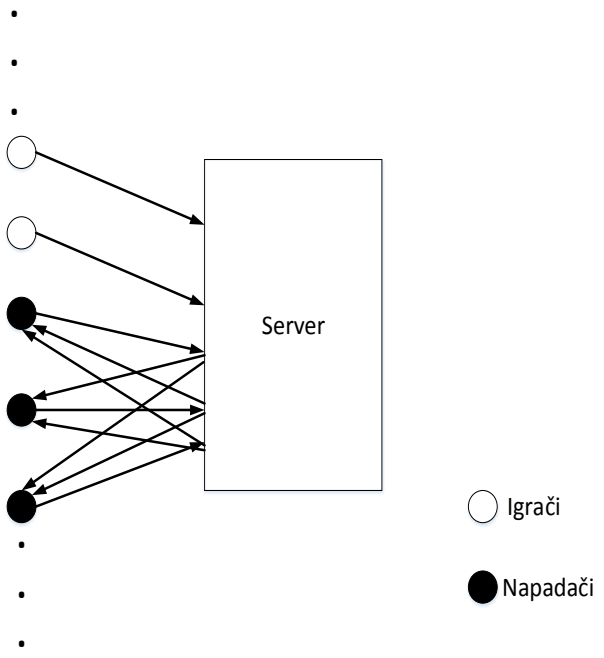
Izazovan deo u borbi protiv DDoS napada na aplikacionom jeste detekcija, s obzirom na to da se napadi lako mogu stopiti u regularan saobraćaj, pritom poštujući sva pravila bezbednosnih mehanizama.



Slika 2. Redno izvedena DDoS zaštita

3. SIMULACIJA

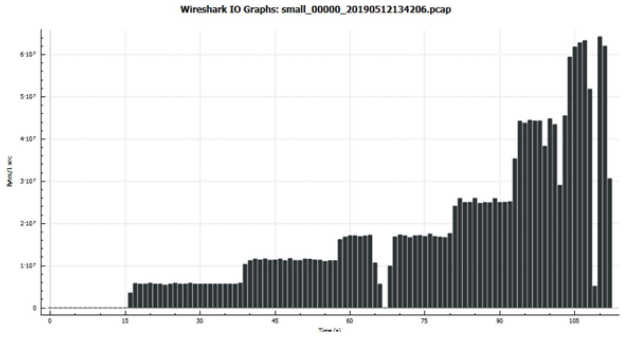
Radi potvrđivanja pretpostavki vezanih za DDoS napad na aplikacionom sloju, izvršena je simulacija u laboratoriji sa 30 računara na Računarskom fakultetu u Beogradu. Relevantne specifikacije računara su: osmojezgrani procesor AMD FX-8350, 16GB DDR3 RAM, Microsoft Windows 10 OS i svi su povezani na 1GB/s ethernet LAN (*Local Area Network*). Za demonstraciju napada specifičnog za aplikacioni protokol (eng. *Protocol specific)* korišćena je igra *World of Warcraft* (verzija 12340) kao klijentska aplikacija, a serverska aplikacija je emulator dostupan kao open source kod pod nazivom Trinity Core [3]. Napad se sastojao iz slanja velike količine poruka (paketa), putem kanala koji su rezervisani za razmenu informacija između *Add-ona* (*Addon Channels)*. *Add-ons* su softverske ekstenzije klijentske aplikacije koje su najčešće open source i prave se isključvo iz razloga što omogućavaju dodatne funkcionalnosti koje se mogu isprogramirati. Za napad je korišćena API funkcija *SendAddonMessage* [4].



Slika 3. Šematski prikaz DDoS napada tokom simulacije

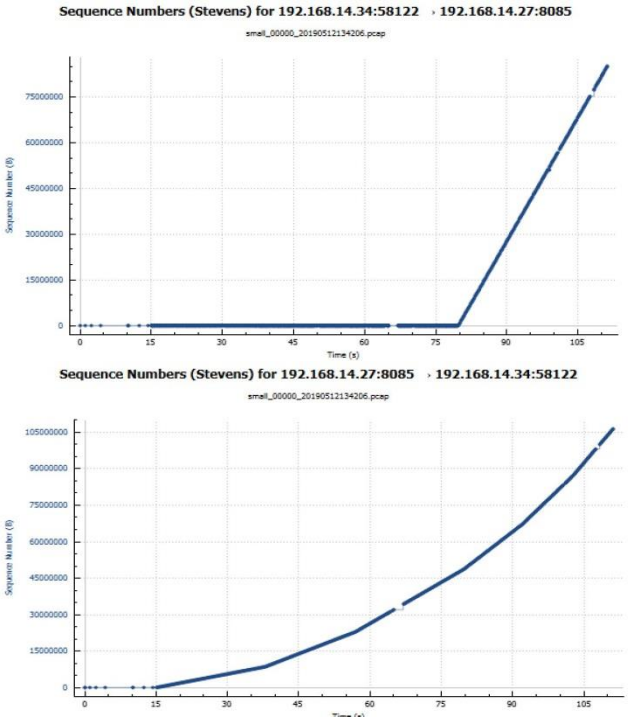
3.1. Efekti napada

Izmerene vrednosti potvrđuju hipotezu da je *protocol-specific* napad na aplikacionom sloju ozbiljna pretnja po bezbednost servera. Na slici 4. može se videti da server ne poseduje adekvatan mehanizam zaštite i da je ukupan protok u vrlo kratkom roku dostigao maksimum linka.



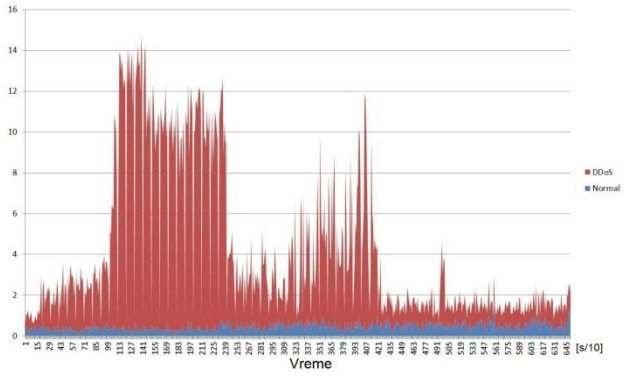
Slika 4. Protok na serveru pre (do 15s) i tokom DDoS napada

Intuitivniji uvid u broj poslatih i obrađenih paketa se može dobiti praćenjem rednih brojeva paketa, što je ekvivalentno njihovoj kumulativnoj sumi kao što je prikazano na slikama 5a i 5b.



Slika 5. Broj paketa između jednog od napadača i servera (a) i ukupna količina paketa na serveru (b)

Na slici 6. je prikazana iskorišćenost procesora u *Kernel-space*, što predstavlja sistemske pozive na linuxu inicirane većinski od strane programa koji se nalaze u *User-space*. Iz *Kernel-space*-a se upravlja mrežnom karticom i ostalim ulazno izlaznim operacijama (npr. poput pisanja na HDD (Hard Disk Drive) ili RAM). Važno je naglasiti, da se čak i sam *scheduler* (proces koji bira naredni proces za obradu od strane procesora) nalazi u *Kernel-space*.



Slika 6. Opterećeonst procesora u *Kernel-Space*

4. ZAKLJUČAK

Rezultati jasno ilustruju nemogućnost nezaštićenog servera da se nosi sa ovakvom vrstom napada. Prediktivnom analizom možemo zaključiti da standardni metodi odbrane poput ograničenja protoka ili zaštite na nižim slojevima TCP/IP stack-a ne mogu da zaštite server.

Iz rezultata se vidi da je količina generisanog saobraćaja linearno proporcionalna količini primljenog saobraćaja. Tj za svaki poziv *SendAddonMessage* usmeren ka grupi korisnika, poruka koju u vidu argumenta ove funkcije pošalje napadač se retransmituje ka svima u navedenoj grupi. Dakle ako je x broj korisnika koji primaju poruke poslate od strane jednog napadača i ako je ukupan broj poslatih malicioznih poruka y opisan funkcijom $f: \mathbb{N} \rightarrow \mathbb{N}$, tada u slučaju jednog napadača važi sledeće:

$$y = f(x) = x + 1 + c \quad (1)$$

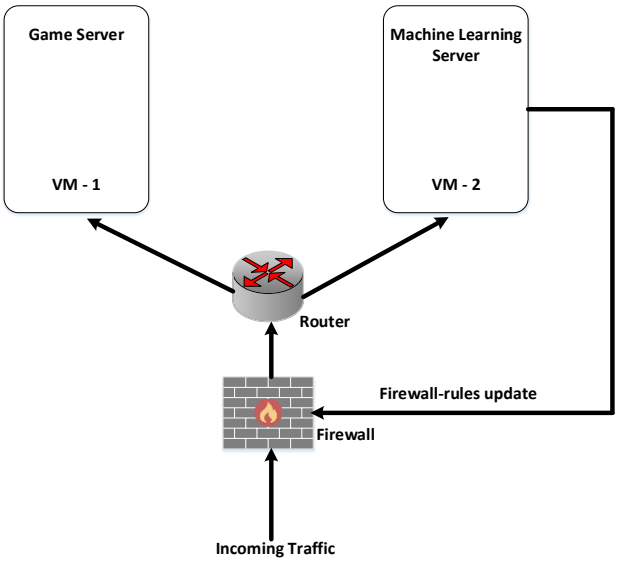
Izdvojena poruka plus 1 se pojavljuje kao poruka koja se inicijalno šalje serveru, dok x označava broj poslatih poruka od strane servera. Faktor c označava dodatne poruke prouzrokovane ovim saobraćajem, poput potvrde prijema itd.

Ukoliko dodamo više napadača kao tokom simulacije napada, u situaciji kada su svi primaoci malicioznih poruka takođe i predajnici istih i definišemo funkciju $g: \mathbb{N} \rightarrow \mathbb{N}$ koja opisuje ukupan broj poslatih poruka između servera i napadača, onda je:

$$y = g(x) = x * f(x) = x * (x + 1) + cx \quad (2)$$

Drugim rečima, ukupan broj poslatih poruka raste srazmerno kvadratu broja napadača. Iako su napadi bili različitih intenziteta, ovakav zaključak je i dalje evidentan na slici 5b. Intuitivniji opis jeste da zbog konstantne brzine slanja, kumulativna suma poruka raste linearno tokom vremena. Međutim kada tokom vremena dodajemo još napadača, jer nismo bili u mogućnosti da pokrenemo više napada istovremeno, dobijemo kumulativnu funkciju koja je nelinearna tokom vremena, a čije restrikcije su linearne funkcije kojima tokom vremena raste faktor skaliranja (Nagib). Uopšteno to predstavlja dobru ilustraciju kvadratne zavisnosti količine poslatih poruka od broja napadača.

Rešenje za ovakvu vrstu DDoS napada nazire se u machine-learning klasifikatorima. Predloženi mehanizam zaštite (slika 7) bi, uz uvođenje metrike ξ kojom bi se opisala „Cena" poziva API funkcija, bio u stanju da zaštiti server od navedenih DDoS napada. Problemi koji se javljaju pri upotrebi ovakvog modela su vezani isključivo za pristrasnost ka aplikaciji kao i činjenica da se oslanja na dobru procenu parametra ξ.



Slika 7. Predloženi mehanizam zaštite servera

mogućnosti za pravljenje modela zaštite su gotovo neograničene, ali je neophodno uzeti u obzir realna ograničenja, pogotovo kompleksnost detektora, a samim tim i brzinu detekcije koja se izuzev hardverskih i softverskih ograničenja svodi na *Uncertainty principle*. Drugim rečima detektor nema neograničeno vreme da donese odluku o detekciji, ali istovremeno treba da prikupi dovoljnu količinu informacija, što iziskuje vreme. Detektor takođe treba da demotiviše napadače, time što će ih naterati da traže još kompleksnije načine za napad. Iz prethodno navedenih razloga, problem detekcije DDoS napada na aplikacionom sloju u *gaming* industriji ostaje izazovna oblast, gde je cilj napraviti izvodljiv model zaštite koji će biti u stanju da u potpunosti ili makar u velikoj meri onemogući manifestaciju štetnih efekata napada, a da pritom ne naruši QoS i QoE regularnih igrača.

5. LITERATURA

[1] Gaurav Somani, „DDoS attacks in cloud computing: Issues, taxonomy, and future directions," Computer Communications, 107 30-48, 2017.
[2] https://newzoo.com/insights/articles/global-games-market-reaches-137-9-billion-in-2018-mobile-games-take-half/
[3] https://github.com/TrinityCore/TrinityCore
[4] https://wow.gamepedia.com/API_C_ChatInfo.SendAddonMessage

Kratka biografija:

Nikola Gavrić rođen je u Novom Sadu 1994. god. Master rad na Fakultetu tehničkih nauka iz oblasti Elektrotehnike i računarstva – Telekomunikacioni sistemi odbranio je 2019.god.
kontakt: nikolagavric021@gmail.com