Univerzitet u Sarajevu
Elektrotehnički fakultet u Sarajevu
Odsjek za računarstvo i informatiku

Simulacija DDoS napada i strategija odbrane korištenjem SimEvents alata

Ime i prezime: Amer Mujalo

Broj indexa: 19101

Datum: 14.01.2024

1. Uvod

Distribuirani napadi uskraćivanja usluge (DDoS) predstavljaju ozbiljnu prijetnju sigurnosti informacijskih sistema u modernom dobu. Ovi napadi, koji ciljaju na preopterećenje mrežnih resursa ili servera, često rezultiraju velikim ekonomskim gubicima i ozbiljnim narušavanjem dostupnosti usluga.

DDoS napadi su postali česti zbog povećanja broja uređaja povezanih na internet i nedostatka adekvatnih sigurnosnih mjera u mnogim organizacijama. Ovi napadi koriste razne metode za iscrpljivanje resursa servera, onemogućavajući ga da pruži usluge legitimnim korisnicima.

U ovom seminarskom radu, simulacija DDoS napada je izvedena korištenjem alata SimEvents u MATLAB-u. Cilj ove simulacije je kvantitativno mjeriti uticaj DDoS napada na žrtvu, tj. server koji je predmet napada. Eksperimenti provedeni u ovom istraživanju pokazuju da server, u svojim normalnim radnim uvjetima, ima visoku dostupnost i nisku prosječnu iskorištenost, jer prima samo legitimne zahtjeve od korisnika. Međutim, kada napadač pokrene DDoS napad, iskorištenost servera naglo raste, što dovodi do smanjenja dostupnosti servera, jer je preplavljen nelegitimnim zahtjevima od strane napadača i "zombi" računara unutar mrežnog domena. Dodatna istraživanja pokazuju da, kada se simulira faza zagrijavanja za server u uslovima napada, iskorištenost naglo raste jer napadač iskorištava sporo oporavljanje servera, dodatno ga preplavljujući i na kraju dostižući tačku zasićenja.

Motivacija

Jedan od glavnih motiva za istraživanje DDoS napada je potreba za razvojem efikasnih alata za odbranu od tih napada. DDoS napadi dovode do smanjenja ili potpunog uklanjanja dostupnosti usluge za legitimne korisnike. Ako znamo ponašanje sistema u njegovim normalnim uvjetima, možemo prepoznati devijacije i na vrijeme prepoznati DDoS napad. To je osnova za istraživanje u ovom radu.

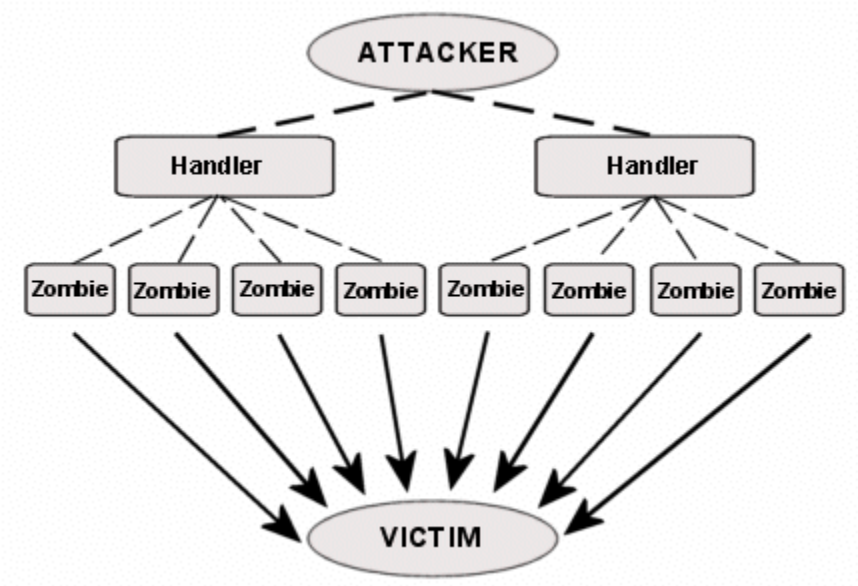
2. Opis problema

Napadi uskraćivanja usluge (DoS) i njihova distribuirana varijanta (DDoS) predstavljaju oblik cyber napada koji su usmjereni na dostupnost sistema ili mreža.

U DDoS napadu, napadač organizuje mrežu kompromitovanih uređaja, poznatih kao "zombiji", koji simultano preplavljuju ciljani sistem velikim brojem zahtjeva. Ovi zahtjevi izgledaju kao legitimni, što otežava njihovo razlikovanje od stvarnih korisničkih zahtjeva.

Komponente DDoS napada

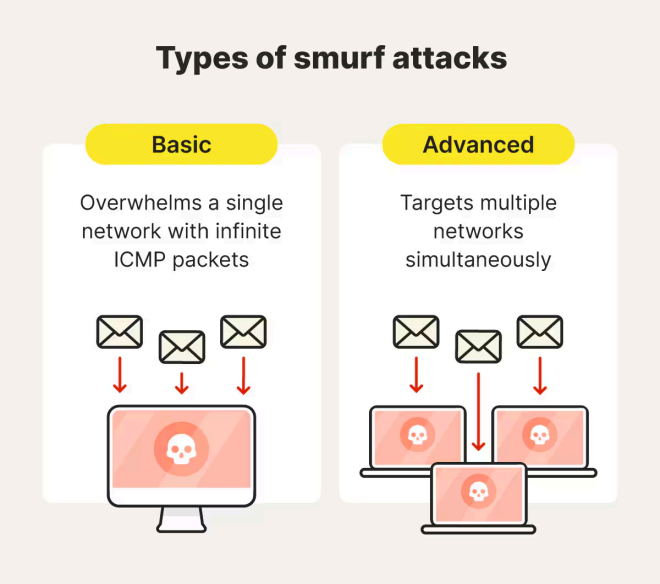
1. **Napadači:** To su subjekti ili grupe koji planiraju i izvršavaju napade. Njihova uloga je da kreiraju maliciozni softver i kompromituju mrežne uređaje.
2. **Rukovatelji (Handler):** Ovo su sistemi koje je kompromitirao ili hakirao napadač na mreži, on koristi sumnjive metode za instaliranje DDoS napadačkih alata na njihov sistem.
3. **Zombie uređaji (Agenti):** Ovo su uređaji čiji su vlasnici nesvjesni da su kompromitovani. Napadači koriste ove uređaje za slanje velikog broja zahtjeva prema ciljanom serveru.
4. **Žrtva:** Žrtva je server, mrežna infrastruktura ili aplikacija koju napadač želi onemogućiti.



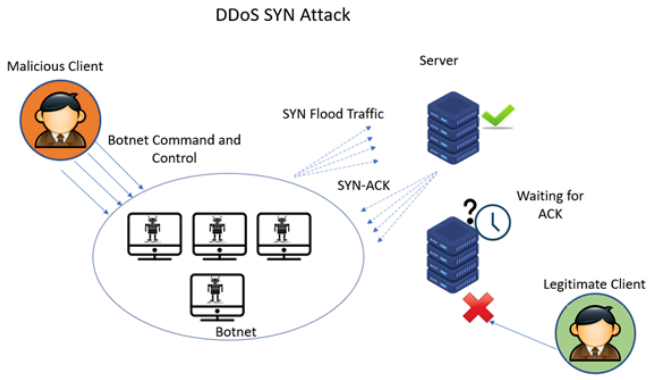
3. Metode DDoS napada

Postoji nekoliko poznatih metoda DDoS napada, uključujući:

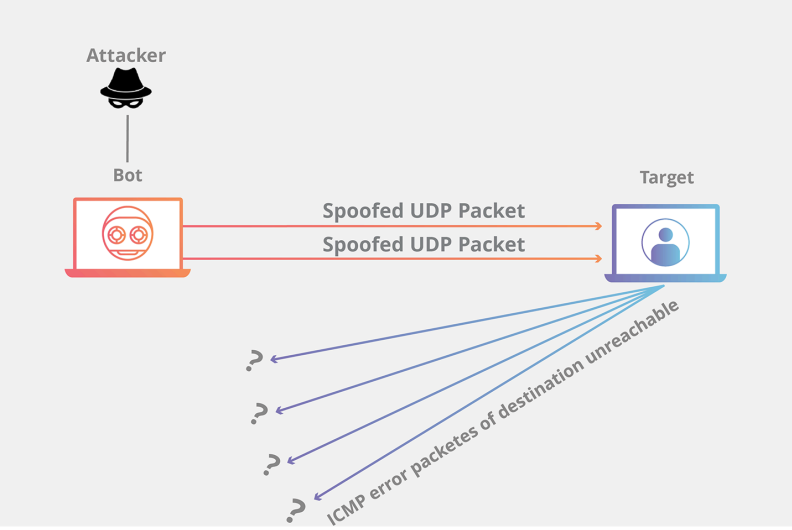
1. **Smurf napad**: Napadač šalje lažnu ICMP echo paket na broadcast adresu ranjivih mreža, što uzrokuje da svi sistemi na mreži odgovore na žrtvu, čime se iscrpljuje propusni opseg mreže i onemogućava pristup legitimnim korisnicima.



2. **TCP SYN napad**: Napadač iskorištava slabost u TCP trostrukoj vezi slanjem zahtjeva prema serveru s nedostupnim izvorom paketa. Server ne može završiti vezu, što rezultira potrošnjom resursa servera i njegovim eventualnim padom.



3. **UDP napad**: Napadač šalje UDP paket na nasumičnu port adresu na serveru žrtve. Kada server primi paket, pokušava pronaći aplikaciju koja čeka na tom portu. Ako aplikacija nije pronađena, server šalje ICMP paket, što može dovesti do pada sistema ako napadač pošalje dovoljno paketa.



4. DDoS napadi u Bosni i Hercegovini

U Bosni i Hercegovini zabilježeno je nekoliko značajnih DDoS napada, posebno usmjerenih prema medijskim portalima i institucijama:

Napadi na medijske portale

Tokom 2020. i 2021. godine, web stranice poput **Žurnal.info**, **Buka.com i Face.tv** bile su mete snažnih DDoS napada koji su ometali njihovo funkcionisanje. Ovi napadi su često povezivani s pokušajima gušenja slobode medija i političkim pritiscima. Također, **Nezavisne novine** su prijavile DDoS napad na njihov portal 10. 08. 2021. godine.

Institucionalne mete

Sistemi Parlamentarne skupštine BiH također su bili meta napada, što je ukazalo na ranjivost javnih institucija. Samo u novembru 2024. godine, zabilježeno je 3,8

miliona DDoS napada u BiH prema podacima **CSEC-a**. Ovi napadi uključivali su i pokušaje kontrole uređaja te zloupotrebu baza podataka i Android uređaja.

Nedostatak strategije za cyber sigurnost

Bosna i Hercegovina suočava se s nedostatkom sveobuhvatne strategije za cyber sigurnost, uključujući izostanak specijalizovanih timova za odgovor na sigurnosne incidente (CERT) i odgovarajućih zakona, što otežava borbu protiv ovakvih prijetnji.

5. Struktura modela

Simulacija DDoS napada zahtijeva jasno definisan model koji oponaša stvarne uslove mrežne komunikacije. Polazni model implementiran je koristeći SimEvents alat MATLAB-a, koji omogućava simulaciju tokova podataka kroz razne komponente sistema.

Model se sastoji od dva osnovna stanja:

1. **Normalno stanje:** Sistem funkcioniše u optimalnim uslovima. Korisnici šalju zahtjeve serveru u pravilnim vremenskim intervalima. Ti zahtjevi se procesuiraju prema redoslijedu dolaska.
2. **Napad:** Tokom napada, dodatni zahtjevi dolaze od napadača i zombi uređaja. Ovi zahtjevi preplavljuju server, uzrokujući pad njegove dostupnosti i smanjenje performansi.

Tok simulacije

U ovoj sekciji opisuje se kako će simulacija DDoS napada biti implementirana koristeći alat SimEvents u MATLAB-u. Simulacija uključuje ključne elemente potrebne za modeliranje mreže i simulaciju napada, koji zajedno omogućavaju praćenje toka podataka od generacije zahtjeva do njihovog procesuiranja na serveru. Ovdje su detaljno opisani svi elementi simulacije:

1. Generacija entiteta (Entity Generator)

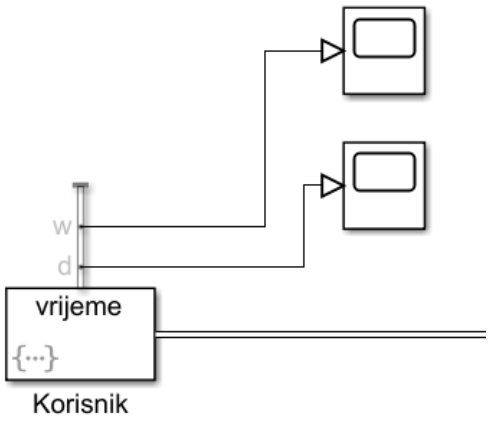
- **Opis**: Ovaj element simulira generaciju mrežnih zahtjeva. Normalni korisnici generiraju zahtjeve u nasumičnim vremenskim intervalima kako bi se oponašao realan saobraćaj u mreži.

Kod implementacije:

```
% Uniformna distribucija za interval dolaska paketa
% m: Minimum, M: Maximum
m = 0.1; M = 1;
dt = m + (M - m) * rand; % Interval između dolaznih paketa
```

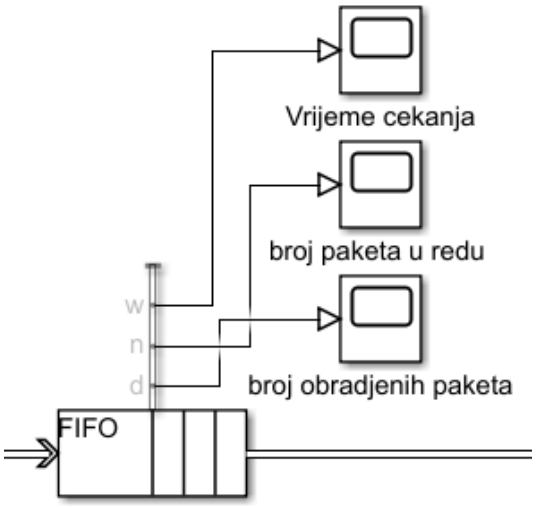
Atributi entiteta: Svaki generirani zahtjev sadrži atribut `vrijeme`, koji definiše koliko će dugo zahtjev provesti unutar servera. Vrijednost ovog atributa također je nasumično generisana unutar definisanog opsega:

```
m = 0.1; M = 0.5;
entity.vrijeme = m + (M - m) * rand;
```



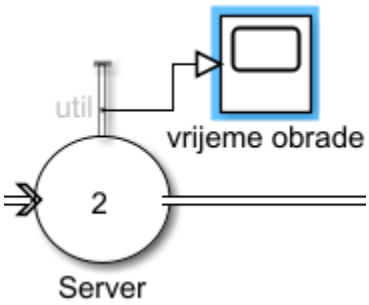
2. Red za čekanje (Entity Queue)

- **Opis**: Entiteti generisani u prethodnom koraku se dodaju u red za čekanje prije nego što budu poslani na server.
- **Karakteristike**:
  - Kapacitet reda: 100 entiteta.
  - Ako red dostigne puni kapacitet, dodatni zahtjevi se **odbacuju**, što simulira efekt zagušenja mreže tokom napada.



3. Server (Server Block)

- **Opis**: Server procesira dolazne zahtjeve prema njihovom redoslijedu dolaska. Ovdje se modelira ponašanje servera sa ograničenim resursima.
- **Karakteristike**:
  - Server ima 2 jezgra, što znači da može simultano procesirati do dva zahtjeva.
  - Vrijeme procesuiranja zahtjeva je definirano atributom `vrijeme`, koji je dodijeljen svakom entitetu tokom generacije.



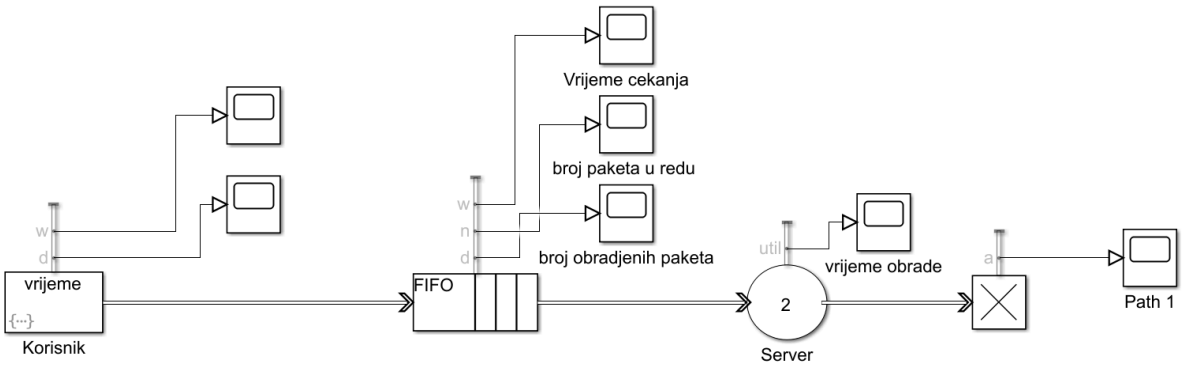
4. Dodavanje napadačkog saobraćaja

- **Opis**: Tokom simulacije napada, dodatni entiteti (napadački zahtjevi) generiraju se s visokom frekvencijom kako bi se oponašala preplavljenost servera. Napadački saobraćaj dolazi od zombi uređaja i koristi istu infrastrukturu kao i regularni saobraćaj.

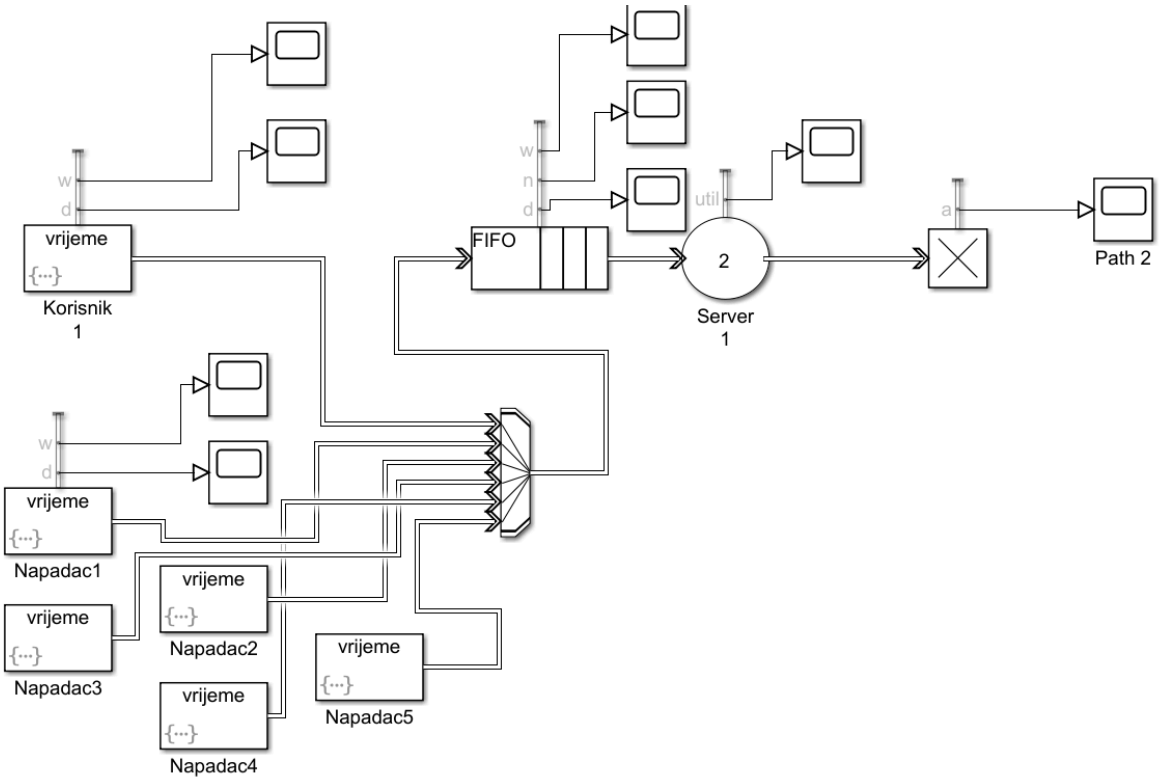
```
m_attack = 0.005; M_attack = 0.02;

dt_attack = m_attack + (M_attack - m_attack) * rand;
```

5. Scenariji simulacije

- **Normalno stanje**: Regularni korisnici generiraju zahtjeve u definisanim intervalima, server funkcioniše optimalno i svi zahtjevi se procesiraju unutar raspoloživih resursa.

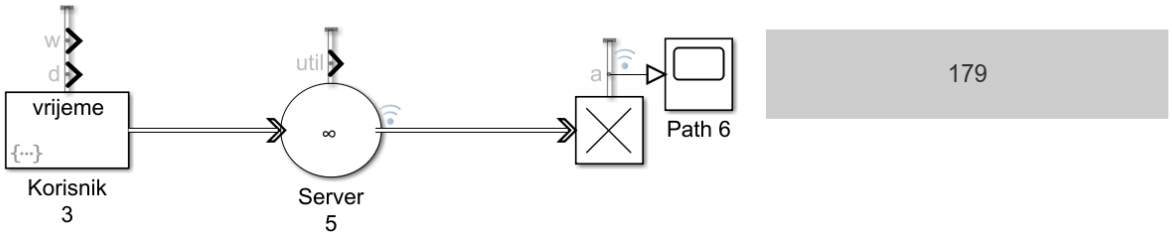


- **Napadno stanje**: Dodatni entiteti (napadački zahtjevi) generiraju se u visokoj frekvenciji. Red za čekanje se brzo puni, a server dostiže maksimalni kapacitet, što dovodi do pada performansi.



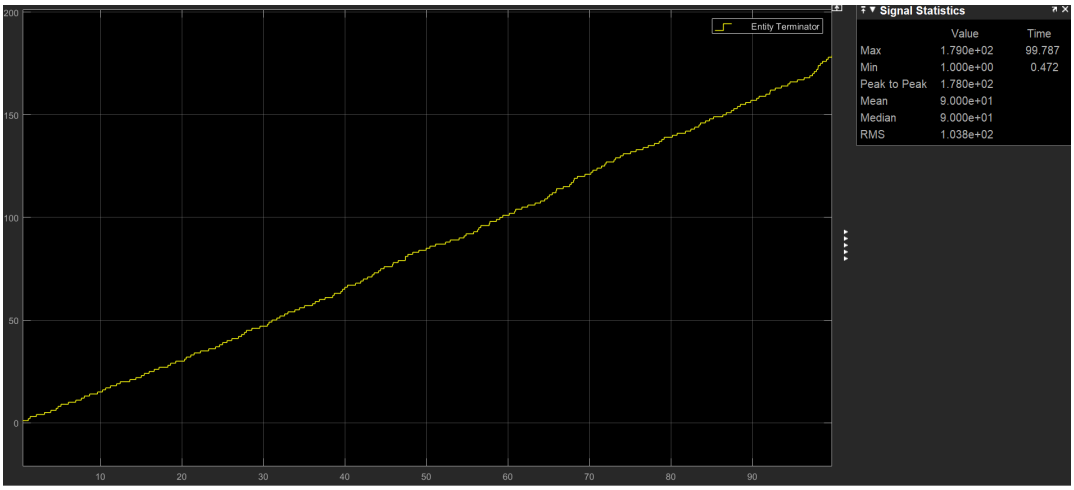
6. Mjerenje performansi

Za simulaciju apsolutnog slučaja, gdje brojimo ukupan broj paketa koje je korisnik poslao u periodu od 100 sekundi, eliminirali smo red za čekanje jer nam nije bio potreban u ovom jednostavnom modelu. Također, postavili smo kapacitet procesora (jezgra) na beskonačno, što znači da server može simultano procesirati neograničen broj zahtjeva bez ikakvih kašnjenja izazvanih zagušenjem.

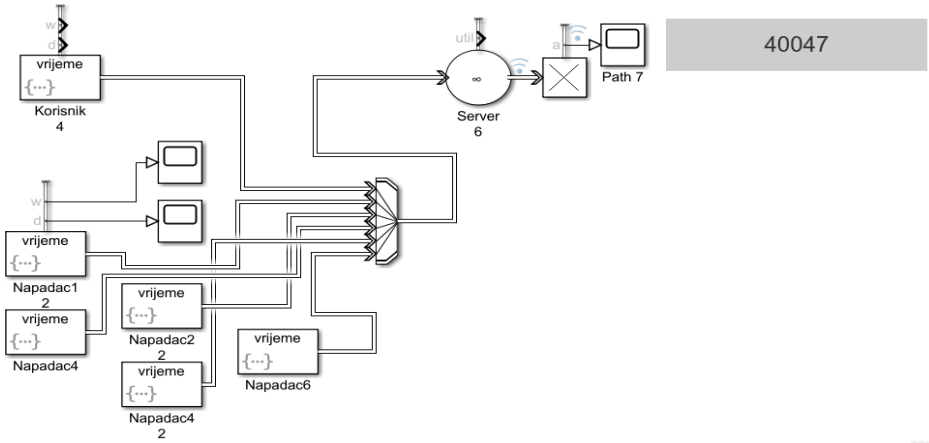


U ovom modelu, broj paketa koji korisnik šalje na server je **179** kroz interval od **100 sekundi**. Ovaj pristup omogućava preciznu analizu količine proslijeđenih podataka bez kompleksnosti uskih grla koja bi nastala u realnim uvjetima s ograničenim resursima.

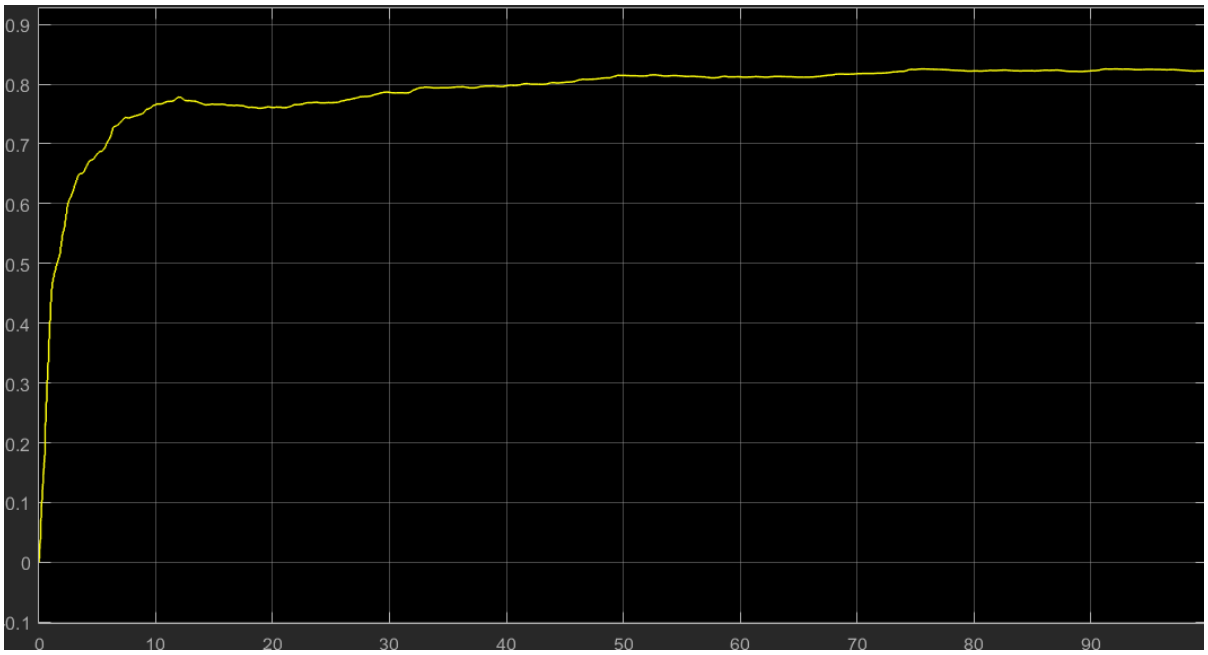
Kada smo testirali model u normalnom stanju, broj paketa koji je korisnik poslao bio je jednak apsolutnom broju paketa, što nam potvrđuje da je model adekvatan za simulaciju u uvjetima bez napada. U ovom stanju, red čekanja je bio prazan (jednak nuli), jer server nije bio preopterećen, te je mogao procesirati sve zahtjeve odmah, što potvrđuje da su svi generirani paketi uspješno prošli kroz sistem bez zastoja. Ovaj rezultat ukazuje na ispravno funkcionisanje modela u normalnim uvjetima i da su svi parametri postavljeni ispravno za ovu simulaciju.

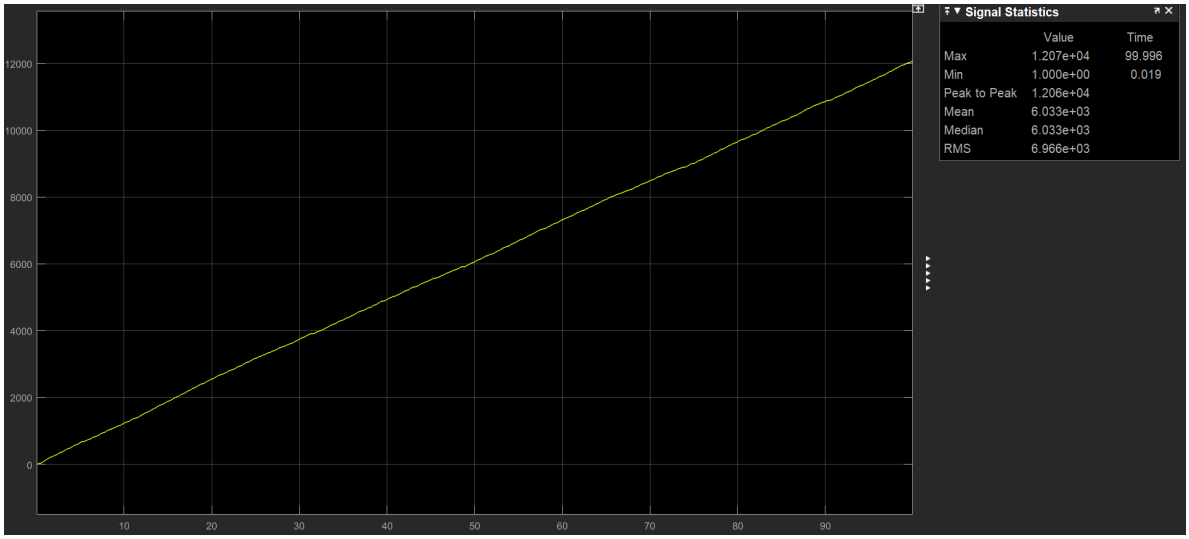


U napadnom stanju, osim regularnog korisnika, napadači također generiraju zahtjeve koji ulaze u server putem Entity Input Switch. Server, sa neograničenim brojem jezgri, može simultano procesirati sve zahtjeve, uključujući napadačke. Iako resursi servera nisu ograničeni, veliki broj napadačkih zahtjeva uzrokuje preopterećenje mreže, što smanjuje performanse sistema. Broj paketa koji je poslao korisnik i napadači se prati tokom simulacije, omogućavajući analizu utjecaja DDoS napada na mrežnu infrastrukturu.



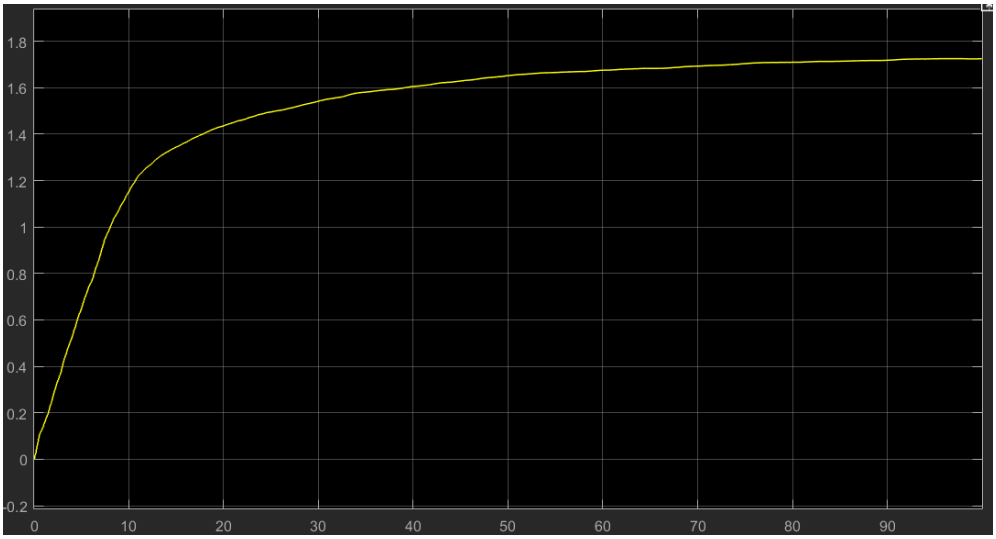
U standardnom stanju napada, gdje je kapacitet reda čekanja postavljen na 100 i server ima 2 jezgra, server je uspio obraditi oko 12,000 zahtjeva. Ovo predstavlja pad od približno **70%** u odnosu na apsolutni slučaj, gdje smo obradili oko 40,000 paketa. Ova značajna razlika ukazuje na drastičan pad u obradivosti zahtjeva zbog preopterećenja uzrokovanog napadačkim saobraćajem. Također, red čekanja konvergira prema vrijednosti 0.8, što dodatno pokazuje zagušenje sistema i smanjenje performansi tokom napada.

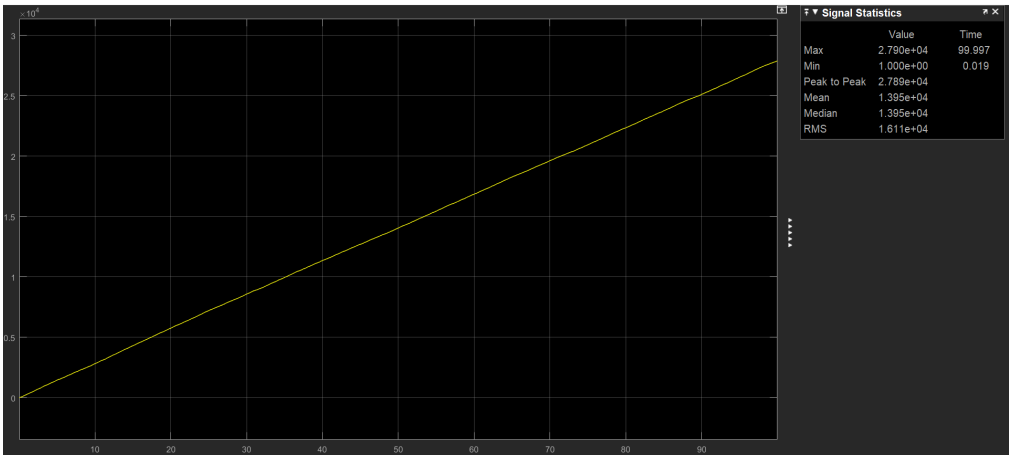




Jedan od načina za poboljšanje rezultata u ovom scenariju je povećanje kapaciteta reda čekanja, što bi omogućilo serveru da procesuira više zahtjeva prije nego što dođe do kašnjenja. Međutim, iako povećanje kapaciteta reda može donijeti određena poboljšanja, neće značajno utjecati na smanjenje zagušenja uzrokovanog napadom.

Drastično poboljšanje performansi postiglo bi se povećanjem broja jezgri na serveru, sa 2 na 4 jezgre. Ovo bi omogućilo paralelniju obradu većeg broja zahtjeva, čime bi se smanjio broj procesuiranih paketa i poboljšala ukupna efikasnost sistema. Povećanje broja jezgri bitno bi smanjilo preopterećenje i značajno unaprijedilo performanse tokom napada.





U ovom slučaju, kada smo povećali broj jezgri na serveru sa 2 na 4, postigli smo značajno poboljšanje u performansama, obradivši skoro 28,000 paketa. Ovo predstavlja povećanje od oko **133%** u odnosu na prethodni scenario, gdje je server obradio samo 12,000 paketa. Red čekanja je sada konvergirao prema vrijednosti 1.7, što pokazuje da je povećanje broja jezgri omogućilo bolju paralelnu obradu zahtjeva, iako još uvijek postoji zagušenje.

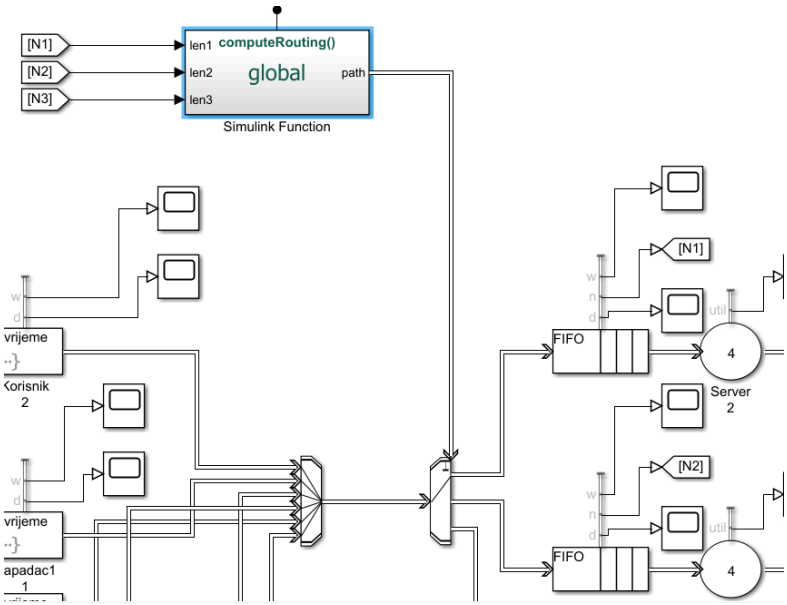
Međutim, i dalje nismo zadovoljni s trenutnim rezultatima, jer smo u apsolutnom slučaju, kada nije bilo nikakvih ograničenja, uspjeli obraditi čak 40,000 paketa. Razlika između trenutnog broja paketa (28,000) i apsolutnog slučaja (40,000) iznosi oko 30%, što ukazuje na to da i dalje postoji značajan prostor za daljnje poboljšanje performansi, posebno u uvjetima napada.

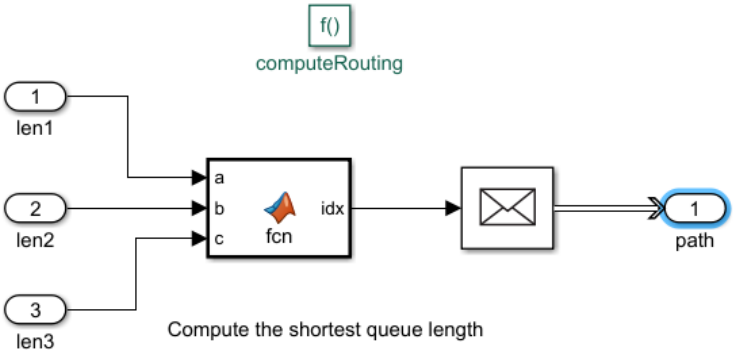
Naredno unapređenje u simulaciji uključuje korištenje **tri** servera umjesto jednog, svaki s **4** jezgra i kapacitetom reda čekanja od **500** paketa. Odabir servera za usmjerenje paketa vršit ćemo pomoću **rutiranja**, gdje će paket biti usmjeren na server s najmanjim brojem paketa u redu čekanja.

Rutiranje će se vršiti analizom broja paketa u svakom od triju servera. Za svaki server izračunavaćemo broj paketa u redu čekanja, a paket će biti usmjeren na onaj server koji ima najmanje paketa. Ovaj odabir ćemo implementirati pomoću Compute Routing funkcije u Simulinku, koja će proslijediti vrijednosti broja paketa za svaki server u MATLAB funkciju:

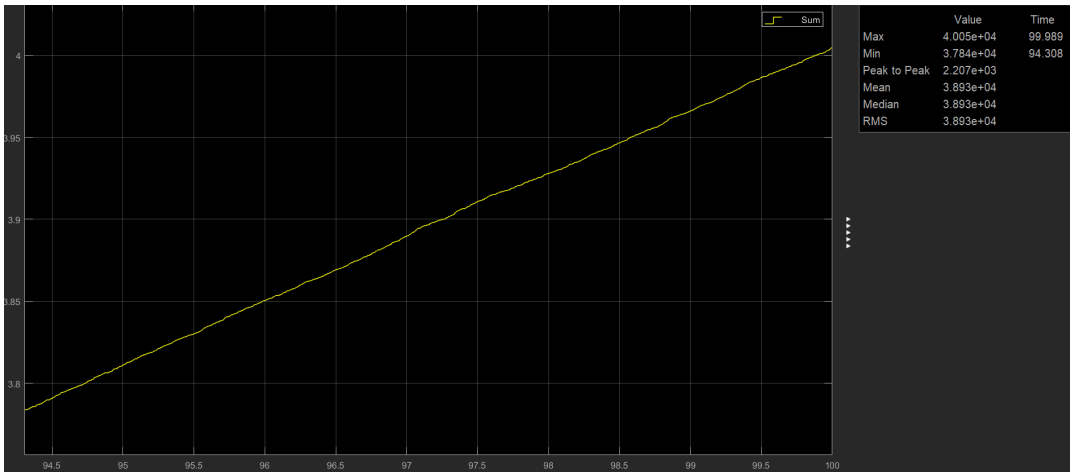
```
function idx = fcn(a,b,c)
    [~, idx] = min([a,b,c]);
end
```

Funkcija fcn uzima tri vrijednosti (broj paketa u redu čekanja za svaki od tri servera) i vraća indeks servera s najmanjim brojem paketa. Taj indeks se koristi za usmjeravanje paketa prema odgovarajućem serveru putem Entity Output Switch komponente, koja će paket odrediti prema serveru na temelju izračunatog indeksa.





Ovaj pristup omogućava bolju raspodjelu opterećenja među serverima, smanjujući preopterećenje na pojedinačnim serverima i poboljšavajući ukupnu efikasnost sistema. Time se postiže balansiranje opterećenja, što bi moglo značajno poboljšati performanse sistema, osobito u uvjetima DDoS napada, gdje dolazi do velikog broja simultanih zahtjeva.

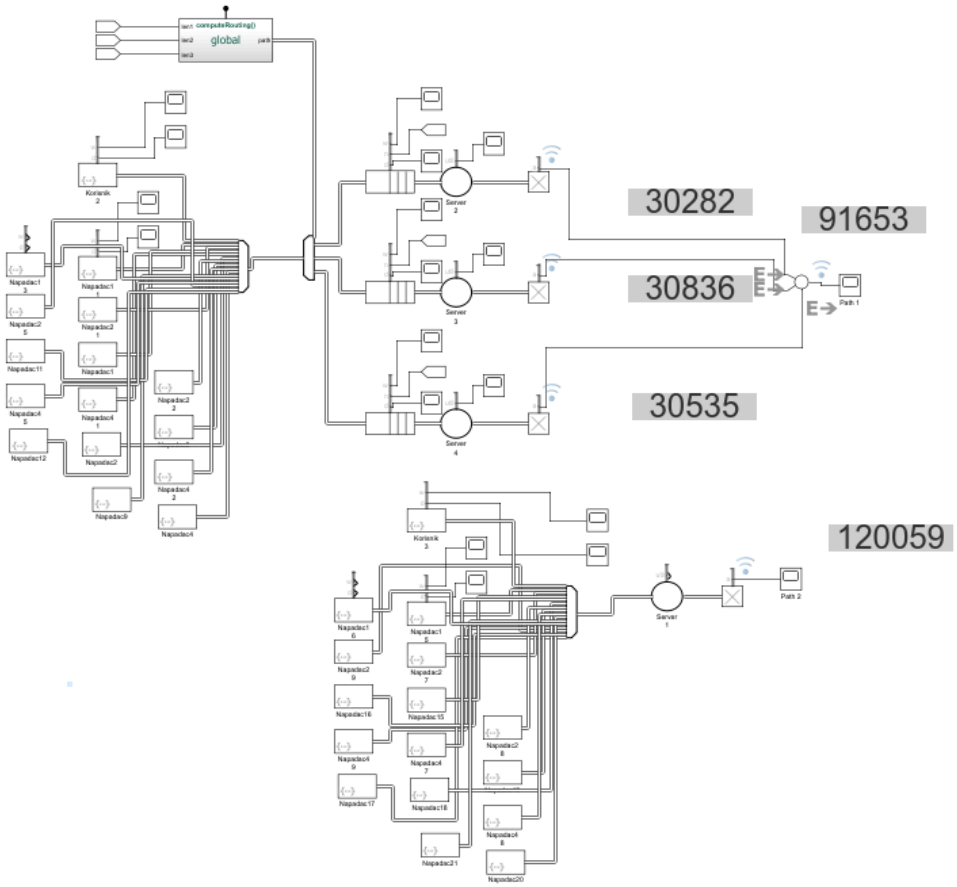


Rezultati ovog unapređenja, gdje smo implementirali tri servera s 4 jezgra i koristili rutiranje za raspodjelu paketa na servere, pokazuju značajna poboljšanja u performansama. U ovom scenariju, broj procesiranih paketa je gotovo isti kao u apsolutnom slučaju, gdje smo obradili oko 40,000 paketa. Ovaj rezultat pokazuje da smo postigli isti kapacitet kao u idealnim uvjetima, ali uz bolju raspodjelu opterećenja među serverima.

Red čekanja u ovom scenariju je ostao na nuli, što znači da nije došlo do zagušenja i svi paketi su obrađeni u realnom vremenu. Zbog efikasne raspodjele opterećenja putem rutiranja, serveri su uspjeli obraditi pakete bez kašnjenja, što ukazuje na to da smo postigli optimalan balans između kapaciteta servera i broja zahtjeva.

Ovaj pristup također omogućava daljnje povećanje opterećenja na servere, jer smo pokazali da čak i pri opterećenju od 40,000 paketa, sistem uspješno obrađuje zahtjeve bez problema sa zagušenjem.

Testiranjem sistema s 10 napadača nismo uočili nikakvu razliku u performansama u odnosu na apsolutni slučaj. Sistem je uspješno obradio isti broj paketa, oko **80,000**, što potvrđuje da je model dovoljno robustan da podnese ovo opterećenje bez degradacije performansi.



Međutim, kada smo povećali broj napadača na **15**, sistem je počeo pokazivati znakove preopterećenja. U apsolutnom slučaju, generisano je oko **120,000 paketa**, dok je naš model obradio približno **92,000 paketa**. Ovo predstavlja razliku od oko **23.33%**, što ukazuje na određeni pad performansi pod ekstremnim opterećenjem.

U ovom scenariju, svaki red čekanja u sistemu imao je prosječnu vrijednost reda čekanja od **1.5 sekundi**, što pokazuje da je došlo do zagušenja i da sistem nije mogao u potpunosti procesuirati sve dolazne zahtjeve u realnom vremenu.

Rješenje ovog izazova leži u daljnjem povećanju kapaciteta sistema, bilo povećanjem broja jezgri po serveru ili dodavanjem dodatnih servera. Povećanje broja jezgri omogućilo bi veću paralelnu obradu zahtjeva, dok bi dodavanje servera smanjilo opterećenje na svaki pojedinačni server putem efikasnijeg balansiranja opterećenja. Ove mjere bi značajno unaprijedile otpornost sistema na DDoS napade sa većim brojem napadača.

6. Zaključak

U ovom radu istražili smo simulaciju DDoS napada i strategija odbrane koristeći SimEvents alat u MATLAB-u. Kroz različite scenarije analizirali smo ponašanje sistema pod normalnim uvjetima, uvjetima napada i unapređenim konfiguracijama. Ključne lekcije koje smo naučili uključuju:

1. **Utjecaj kapaciteta servera:** Kapacitet servera, posebno broj procesorskih jezgri, ključan je za obradu zahtjeva u situacijama visokog opterećenja. Veći broj jezgri značajno smanjuje zagušenje i omogućava veću otpornost na napade.
2. **Efikasna raspodjela opterećenja:** Implementacija rutiranja zasnovanog na broju elemenata u redovima čekanja omogućila je optimalnu distribuciju zahtjeva i spriječila zagušenja, čak i pri visokim opterećenjima.
3. **Ograničenja sistema:** Iako smo uspjeli obraditi oko 92,000 zahtjeva pri opterećenju s 15 napadača, ovaj broj je još uvijek zaostajao za apsolutnim slučajem od 120,000 zahtjeva, što ukazuje na potrebu za dodatnim poboljšanjima.

U poređenju s realnim uvjetima, rezultati simulacije su vrlo blizu stvarnim scenarijima u kojima serveri s ograničenim resursima moraju balansirati između obrade legitimnih zahtjeva i obrane od napada. Naš model je pokazao tačnost u predviđanju kapaciteta i ponašanja sistema, ali treba imati na umu da simulacija pojednostavljuje određene aspekte, poput složenosti mrežnih protokola i dinamike napada.

Rezultati ukazuju na važnost strateškog povećanja resursa u stvarnim uslovima, bilo kroz povećanje procesorske moći ili raspoređivanje dodatnih servera s efikasnim balansiranjem opterećenja. Iako simulacija ne može u potpunosti replicirati sve aspekte stvarnih mreža, ona pruža vrijedan uvid u ponašanje sistema pod različitim uvjetima i omogućava identifikaciju ključnih tačaka za unapređenje.

Ova studija naglašava da adekvatno modeliranje i simulacija predstavljaju neophodan korak u dizajnu robusnih sistema otpornih na DDoS napade, pružajući smjernice za optimizaciju performansi i povećanje sigurnosti mreža u stvarnim okruženjima.

Reference

1. Abubakar Bala i Yahya Osais, "Modelling and Simulation of DDoS Attack using SimEvents." Dostupno na: https://www.researchgate.net/publication/249007957_Modelling_and_simulation_of_DDOS_Attack_using_SimEvents
2. "What is a Smurf DDoS Attack?" Cloudflare. Dostupno na: https://www.cloudflare.com/learning/ddos/smurf-ddos-attack/
3. "What is a SYN Flood DDoS Attack?" Cloudflare. Dostupno na: https://www.cloudflare.com/learning/ddos/syn-flood-ddos-attack/
4. "SimEvents for Operations Research," MathWorks. Video materijal. Dostupno na: https://www.mathworks.com/videos/simevents-for-operations-research-118566.html
5. MathWorks SimEvents Dokumentacija. Dostupno na: https://www.mathworks.com/help/simevents/index.html