Univerzitet u Zenici
Politehnički fakultet
Softversko inženjerstvo

# Detekcija DDoS napada pomoću K-Means algoritama za analizu saobraćaja u mreži

Vještačka inteligencija

*Ajla Brdarević, br. indeksa II-120*

Zenica, decembar 2024.

# Uvod

Razvoj modernih mrežnih tehnologija donio je niz prednosti, ali istovremeno povećao rizik od cyber napada. DDoS napadi (Distributed Denial of Service) predstavljaju jedan od najučestalijih oblika prijetnji koji ciljano opterećuju mrežne resurse prekomjernim brojem zahtjeva, čime se narušava dostupnost usluga.

Zbog rastuće složenosti ovih napada, klasične metode detekcije često nisu dovoljne. Ovaj projekat sam odabrala jer me posebno zainteresovalo korištenje umjetne inteligencije u oblasti cyber sigurnosti, s fokusom na detekciju i analizu napada. Implementacijom algoritma K-Means klasteriranja, kao i razmatranjem drugih AI metoda, cilj je prikazati efikasan pristup prepoznavanju anomalija koje ukazuju na DDoS napade, što ima poseban značaj za sigurnost savremenih mrežnih sistema.

# Detekcija DDoS napada pomoću K-Means algoritama za analizu saobraćaja u mreži

## 1. K-Means

K-means algoritam je algoritam nenadgledanog učenja koji se koristi za grupisanje podataka u unaprijed definisan broj klastera (K). Cilj algoritma je da podijeli skup podataka na K grupa, gdje podaci unutar svake grupe imaju slične karakteristike.

Princip rada K-means algoritma

- Inicijalizacija centara klastera: Algoritam nasumično bira K početnih centara klastera iz skupa podataka.
- Dodjela podataka klasterima: Svaki podatak se dodjeljuje najbližem centru klastera na osnovu izračunate udaljenosti (najčešće euklidske distance).
- Ažuriranje centara klastera: Nakon dodjele, centar svakog klastera se ažurira kao prosjecna vrijednost svih podataka koji pripadaju tom klasteru.
- Iteracija: Proces dodjele i ažuriranja se ponavlja sve dok centri klastera ne prestanu značajno da se mijenjaju ili dok se ne dostigne maksimalni broj iteracija.
- Rezultat: Kao rezultat, podaci su grupisani u K klastera, a svaki podatak dobija oznaku koja predstavlja pripadnost određenom klasteru.

## 2. Ddos_attack projekat

- main.py

```
from src.simulate_data import generate_data
from src.kmeans_detection import detect_ddos
from src.visualize import plot_results

def main():
    time, latency, packet_size = generate_data()
    labels = detect_ddos(latency, packet_size)
    plot_results(time, latency, packet_size, labels)

if __name__ == "__main__":
    main()
```

*generate_data*: Ova funkcija je odgovorna za generisanje simuliranih podataka o mrežnom saobraćaju. Ovi podaci uključuju vrijeme, latenciju i veličinu paketa. To su ključne varijable koje omogućuju analizu i detekciju napada.

*detect_ddos*: Ova funkcija koristi K-means algoritam za grupisanje podataka o mrežnom saobraćaju u klastere. Prepoznavanje napada temelji se na faktoru koji je generiran kroz analizu latencije i veličine paketa.

*plot_results*: Ova funkcija omogućava vizualizaciju rezultata analize. Grafički prikaz pomaže korisnicima u boljoj interpretaciji podataka i olakšava detekciju sumnjivih obrazaca koji mogu ukazivati na DDoS napad.

Pretpostavimo da imamo simulirane podatke za saobraćaj na mreži koji uključuju vrijeme, latenciju i veličinu paketa. Na temelju tih podataka, K-means algoritam će pokušati grupirati saobraćaj u dva klastera. Jedan klaster može predstavljati normalan saobraćaj, dok drugi može ukazivati na DDoS napad. Vizualizacija rezultata prikazuje kako se saobraćaj mijenja u vremenu, uočavajući vrhove ili abnormalne promjene koji signaliziraju napad.

Na primjer, ako primijetimo da se broj paketa odjednom povećava u vrlo kratkom vremenu ili da se paketi značajno povećavaju u veličini, to bi mogao biti signal da je u pitanju DDoS napad.

- simulate_data.py

```
import numpy as np

def generate_data():
      latency = np.random.normal(10, 2, 100)
      packet_size = np.random.normal(500, 50, 100)

    latency[30:50] = np.random.normal(100, 10, 20)

   packet_size[30:50] = np.random.normal(5000, 500, 20)
   return time, latency, packet_size
```

*time*: Ovo je vremenska serija koja traje 100 vremenskih koraka (0 do 99). Svaka vrijednost u ovom nizu predstavlja jedan trenutak vremena u simulaciji.

*latency*: Latencija (kašnjenje) paketa generiše se iz normalne distribucije s prosječnom vrijednošću od 10 ms i standardnom devijacijom od 2 ms. To simulira "normalan" mrežni saobraćaj, gdje je latencija obično mala i konstantna.

*packet_size*: Veličina paketa generiše se iz normalne distribucije s prosječnom vrijednošću od 500 bajtova i standardnom devijacijom od 50 bajtova. Ovo predstavlja "normalne" veličine paketa koji se prenose na mreži.

Ovdje simuliramo DDoS napad tako što mijenjamo latenciju i veličinu paketa u određenom vremenskom intervalu (od 30. do 50. elementa u nizu). Ovo simulira abnormalne uvjete na mreži koji mogu nastati tokom napada. Latencija se povećava na prosječnu vrijednost od 100 ms, što je puno veće od standardne latencije (10 ms). DDoS napadi mogu uzrokovati znatno kašnjenje u mreži. Veličina paketa se također povećava na prosječnu vrijednost od 5000 bajtova, što može značiti da napadači šalju veće pakete kako bi preopteretili mrežu.

Ova funkcija generira "normalni" mrežni saobraćaj, a zatim simulira period u kojem se dešava DDoS napad. DDoS napad je simuliran promjenom latencije i veličine paketa u određenom vremenskom intervalu.

- kmeans_detection.py

```
from sklearn.cluster import KMeans
import numpy as np

def detect_ddos(latency, packet_size):
    data = np.column_stack((latency, packet_size))

    kmeans = KMeans(n_clusters=2, random_state=0).fit(data)
    labels = kmeans.labels_

    return labels
```

*data = np.column_stack((latency, packet_size))*: Ovdje spajamo dvije liste ili nizove (latency i packet_size) u jedan 2D niz.

*latency*: predstavlja latenciju paketa (kašnjenje), tj. vrijeme potrebno da paket stigne od izvora do odredišta.

*packet_size*: predstavlja veličinu paketa u bajtovima.

Ova dva podatka (latencija i veličina paketa) povezujemo u jednu matricu gdje je svaki redak jedan podatak (paket), a svaka kolona predstavlja jedan atribut (latenciju ili veličinu paketa).

*labels = kmeans.labels_*: Ovdje izvlačimo oznake (labels) klastera koje je K-Means dodijelio svakom podatku. Svaka oznaka u labels označava kojoj grupi podatak pripada. Ako je vrijednost 0, onda podatak pripada prvom klasteru (npr. normalan saobraćaj), a ako je vrijednost 1, podatak pripada drugom klasteru (npr. napad).

*return labels*: Vraćamo oznake klastera koje mogu biti korišćene za dalje analize ili vizualizacije. Na osnovu oznaka, možemo odrediti koji dio saobraćaja je normalan, a koji predstavlja DDoS napad. Ako je više podataka označeno sa 1 (napad), to znači da se u tom vremenskom periodu vjerojatno dogodio DDoS napad.

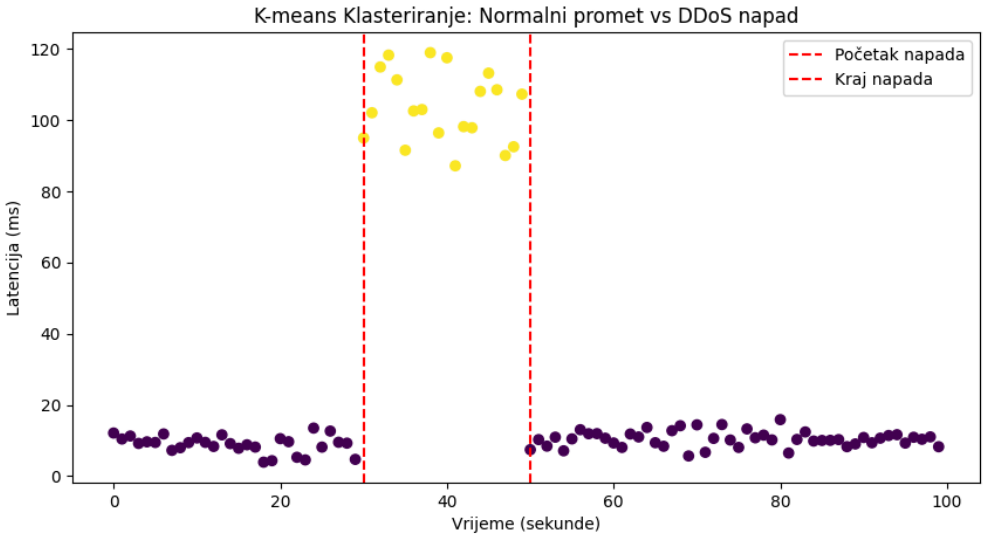
K-Means algoritam koristi podatke o latenciji i veličini paketa da bi podijelio podatke u dva klastera: jedan za normalni saobraćaj i drugi za DDoS napad. Algoritam vraća oznake koje ukazuju na to kojoj grupi (klasteru) svaki podatak pripada.

- visualize.py

```
import matplotlib.pyplot as plt
```

```
def plot_results(time, latency, packet_size, labels):
    plt.figure(figsize=(10, 5))
    plt.scatter(time, latency, c=labels, cmap='viridis')
    plt.axvline(x=30, color='r', linestyle='--', label="Početak napada")
    plt.axvline(x=50, color='r', linestyle='--', label="Kraj napada")
    plt.title("K-means Klasteriranje: Normalni promet vs DDoS napad")
    plt.xlabel("Vrijeme (sekunde)")
    plt.ylabel("Latencija (ms)")
    plt.legend()
    plt.show()
```

Ovim grafikom vizualizuje se razlika između normalnog prometa i DDoS napada pomoću boja koje dodjeljuje K-means algoritam. Vertikalne linije označavaju vremenski okvir napada, što olakšava identifikaciju anomalnog ponašanja u mrežnom saobraćaju.




## 3. K-Means algoritam za detekciju DDoS napada i prepoznavanje karakteristika napada

- Prekomjeran broj zahtjeva (DDoS napad)

DDoS napad generiše veliki broj paketa u kratkom vremenskom periodu, što se manifestuje kroz povećanu latenciju i veličinu paketa. K-Means algoritam uočava ovu anomaliju jer prepoznaje obrasce koji značajno odstupaju od normalnog saobraćaja i smješta ih u zaseban klaster.

Ključni dio koda:

U funkciji generate_data simulira se DDoS napad naglim povećanjem latencije i veličine paketa:

```
latency[30:50] = np.random.normal(100, 10, 20)
packet_size[30:50] = np.random.normal(5000, 500, 20)
```

K-Means razdvaja podatke u klastere gdje su anomalije (napadi) prepoznate kao zaseban klaster zbog ekstremnih vrijednosti.

- Varijabilnost saobraćaja

Raznolikost saobraćaja ogleda se u različitim veličinama paketa, vremenima dolaska i ponašanju tokom mrežnog protoka. K-Means uočava varijabilnost tako što grupiše slične obrasce u jedan klaster, dok anomalni obrasci (npr. naglo povećanje latencije i veličine paketa) pripadaju posebnom klasteru.

Ključni dio koda:

U funkciji generate_data, varijabilnost se simulira putem različitih distribucija latencije i veličina paketa:

```
latency = np.random.normal(10, 2, 100)
packet_size = np.random.normal(500, 50, 100)
```

Nagla odstupanja (tokom napada) jasno se izdvajaju od normalnog saobraćaja kroz K-Means grupisanje.

- Distribucija (DDoS napad iz botneta)

DDoS napad se generiše iz distribuirane mreže uređaja (botnet), pri čemu veliki broj uređaja šalje saobraćaj ka meti. Ovo rezultira naglom i široko rasprostranjenom anomalijom u mreži. K-Means algoritam grupiše podatke na osnovu sličnosti, prepoznajući distribuciju napada kroz povećane vrijednosti latencije i veličine paketa.

Ključni dio koda:

Distribuirani napad simuliran je putem niza anomalnih podataka unutar funkcije generate_data:

```
latency[30:50] = np.random.normal(100, 10, 20)
packet_size[30:50] = np.random.normal(5000, 500, 20)
```

Ove vrijednosti predstavljaju distribuciju anomalnog saobraćaja iz različitih "izvora".

- Nerealno ponašanje

Tokom napada, saobraćaj pokazuje neuobičajene obrasce poput ekstremno visoke latencije ili velikih paketa, što odstupa od uobičajenog saobraćaja. Algoritmi poput K-Means identifikuju ove obrasce automatskim grupisanjem podataka u klastere, pri čemu se anomalije smještaju u zaseban klaster.

Ključni dio koda:

K-Means grupiše podatke u funkciji detect_ddos, gdje neuobičajene vrijednosti postaju jasno prepoznate:

```
data = np.column_stack((latency, packet_size))
kmeans = KMeans(n_clusters=2, random_state=0).fit(data)
```

Kao rezultat, napadni saobraćaj se izdvaja u jedan od klastera zbog neuobičajenih vrijednosti.

# 4. Drugi Algoritmi za Analizu Saobraćaja u Mreži

Osim K-Means, postoje i drugi algoritmi koji se mogu koristiti za analizu mrežnog saobraćaja i detekciju DDoS napada:

## 4.1. Random Forest

Random Forest može se koristiti za klasifikaciju mrežnog saobraćaja na normalan saobraćaj i DDoS napad. Algoritam analizira kombinaciju različitih mjera kao što su latencija, veličina paketa i vrijeme dolaska, te na osnovu toga gradi više stabala odluke koja donose konačnu odluku. Random Forest je koristan jer može precizno prepoznati složene obrasce u podacima i pruža stabilne rezultate čak i uz veliku varijabilnost saobraćaja.

## 4.2. Support Vector Machines (SVM)

SVM se može koristiti za binarnu klasifikaciju saobraćaja. Na osnovu ulaznih karakteristika, poput latencije i veličine paketa, SVM pronalazi optimalnu hiper-ravninu koja razdvaja normalan saobraćaj od anomalnog (DDoS napada). Ovaj pristup je posebno koristan kada postoji jasna razlika između klasa u visoko dimenzionalnim podacima.

## 4.3. Neuronske mreže

Neuronske mreže mogu se koristiti za prepoznavanje kompleksnih anomalija u mrežnom saobraćaju, naročito u slučajevima kada podaci sadrže skrivene obrasce. Trening dubokih neuronskih mreža na podacima koji uključuju različite metrike (npr. vrijeme, latencija, veličina paketa) omogućava mreži da automatski prepozna nespecifične ili nelinearne anomalije koje ukazuju na DDoS napad.

## 4.4. Naivni Bayes

Naivni Bayes algoritam može se koristiti za brzo otkrivanje anomalnog saobraćaja na osnovu vjerovatnoće pripadnosti različitim klasama. Ovaj algoritam analizira karakteristike saobraćaja, poput povećane latencije i veličine paketa, te procjenjuje vjerovatnoću da podaci pripadaju normalnom saobraćaju ili DDoS napadu. Prednost Naivnog Bayesa je što je brz i efikasan za velike skupove podataka.

# Zaključak

Detekcija DDoS napada je ključna za održavanje sigurnosti mreža i sistema. K-Means je jedan od mnogih algoritama koji mogu pomoći u prepoznavanju takvih napada analizom karakteristika mrežnog saobraćaja. Iako su drugi algoritmi, poput Random Forest-a i SVM-a, također učinkoviti za detekciju DDoS napada, K-Means je jednostavan za implementaciju i efikasan za osnovnu analizu.

Nadalje, korištenje ovih algoritama u kombinaciji s naprednim tehnikama analize podataka može omogućiti kreiranje robusnih sistema za prevenciju i detekciju DDoS napada u stvarnim mrežnim okruženjima.