Република Србија

КРИМИНАЛИСТИЧКО-ПОЛИЦИЈСКИ УНИВЕРЗИТЕТ

Департман информатике и рачунарства

Славиша Ж. Илић

Унапређење динамичке анализе злонамерног софтвера засновано на проширеном скупу обележја и машинском учењу

– докторска дисертација –

Београд, 2025.

Republic of Serbia

UNIVERSITY OF CRIMINAL INVESTIGATION AND POLICE STUDIES

Faculty of Computer Science and Information Technology

Slaviša Ž. Ilić

Improvement of dynamic malware analysis based on extended feature set and machine learning

– Doctoral Dissertation –

Belgrade, 2025

Ментор:

проф. др Милан Гњатовић, редовни професор, Криминалистичко-полицијски универзитет у Београду, Департман информатике и рачунарства.

Комисија за одбрану докторске дисертације:

Датум одбране:

Захвалница

За разлику од истраживачког задовољства, вишегодишњи напор уложен у истраживања и израду ове дисертације није био само мој. Обухватио је одрицања свих вас, којима је мој живот употпуњен, на којима сам вам захвалан.

Хвала мојој драгој супрузи и деци, брату и родитељима, чија велика љубав и потпуно разумевање су били велики извор снаге.

Мојим колегама, које су несебично покривале ватром док сам напредовао, велико, велико хвала!

Мом ментору, за бројне и дуге сате, отете од његове породице и сна, који превазилазе очекивано менторско ангажовање. Хвала на људскости, стручности и вођењу кроз ово комплексно истраживање!

Захвалан сам Богу на свима вама, као и пријатељима и професорима које овде нисам могао појединачно поменути. Наша интеракција, размена идеја, подршка, у општем смислу – љубав и саборност, су уграђени у ово скромно дело и тиме је оно и ваш успех.

НАСЛОВ ДОКТОРСКЕ ДИСЕРТАЦИЈЕ:

Унапређење динамичке анализе злонамерног софтвера засновано на проширеном скупу обележја и машинском учењу

САЖЕТАК:

Динамичка анализа злонамерног софтвера у контексту машинског учења доминантно се заснива на АПИ-позивима. Ова дисертација предлаже унапређење динамичке анализе злонамерног софтвера засновано на проширеном скупу динамичких обележја софтвера. Почетна тачка овог истраживања јесте та да постоје динамичка обележја софтвера која су релевантна за аутоматско детектовање злонамерних софтвера, а не припадају скупу АПИ-позива, и да проширени скуп динамичких обележја софтвера може да унапреди тачност система за аутоматско детектовање злонамерних софтвера у односу на систем који би био заснован само на скупу обележја која се односе на АПИ-позиве. Доприноси ове дисертације јесу следећи. (i) Генерисан је нов корпус за динамичку анализу злонамерног софтвера, који садржи извештаје о резултатима динамичке анализе узорака спроведене у наменски подешеном окружењу Куку. Број узорака у корпусу (22200 узорака), његова величина (969GB) и број изведених обележја (више од 25 милиона динамичких обележја софтвера) знатно превазилазе уобичајене корпусе сличне намене сачињене само од секвенци АПИ-позива. Осим тога, тренутно не постоје други јавно доступни корпуси са оваквим карактеристикама који би били прикладни за анализу бинарног класификовања злонамерног софтвера засновану на скупу динамичких обележја која, по свом типу, превазилазе АПИ-позиве. (ii) Развијен је прототипски систем за аутоматску динамичку анализу софтвера у контексту бинарног класификовања софтвера на злонамерни и безопасни, заснованог на динамичким обележјима из датог корпуса и ансамблу стабала одлучивања. Експериментално је демонстрирано да проширени скуп динамичких обележја унапређује тачност оваквог система (99,74%) у односу на систем који би био засновани само на скупу обележја која се односе на АПИ-позиве (95,56%). (iii) Коначно, утврђено је који су типови динамичких обележја софтвера, поред типа који се односи на АПИ-позиве, најзначајнији за откривање злонамерног софтвера.

КЉУЧНЕ РЕЧИ: злонамерни софтвер, динамичка анализа, бинарно класификовање, ансамбл стабала одлучивања, корпус, АПИ-позиви.

НАУЧНА ОБЛАСТ: Рачунарске науке.

УЖА НАУЧНА ОБЛАСТ: Рачунарство.

DOCTORAL DISSERTATION TITLE:

Improvement of dynamic malware analysis based on extended feature set and machine learning

ABSTRACT:

Dynamic malware analysis in the context of machine learning is dominantly based on API calls. This dissertation proposes an improvement in dynamic malware analysis based on an extended set of dynamic software features. The point of departure for this research is that there are dynamic software features not belonging to the set of API call-based features that are relevant for automatic malware detection, and that an extended set of dynamic software features may improve the accuracy of an automatic malware detection system, when compared to a system based only on a set of features related to API calls. The contributions of this dissertation are the following. (i) A new corpus for dynamic malware analysis is generated, containing reports on the results of dynamic analysis of samples conducted in the carefully configured Cuckoo sandbox system. The number of samples in the corpus (22200 samples), its size (969GB), and the number of derived features (more than 25 million dynamic software features) significantly exceed common corpora of similar purpose containing only API call sequences. In addition, there are currently no other publicly available corpora with such characteristics that would be suitable for binary malware classification analysis and based on a set of dynamic features that go beyond API calls. (ii) A prototype system for automatic dynamic software analysis in the context of binary (i.e., malware vs benign) classification, based on the given corpus and a random forest model, has been developed. It has been experimentally demonstrated that the extended set of dynamic features improves the accuracy of such a system (99.74%) compared to a system based only on the set of features related to API calls (95.56%). (iii) Finally, it has been determined which types of dynamic software features, in addition to the type related to API calls, are the most significant for malicious software detection.

KEYWORDS: malware, dynamic analysis, binary classification, random forest, dataset, API calls.

SCIENTIFIC FIELD: Computer Science.

NARROW SCIENTIFIC FIELD: Scientific Computing.

Садржај

Попис изабраних скраћеница

Скраћеница	Назив	Превод
APT	Advanced Persistent Threats	Напредне претње
ANN	Artificial Neural Network	Вештачка неуронска мрежа
API	Application Programming Interface	Програмски интерфејс апликације
BiLSTM	Bidirectional Long Short-Term Memory	Бидирекциона мрежа са дугом краткотрајноом меморијом
CNN	Convolutional Neural Network	Конволутивна неуронска мрежа
CNN-LSTM	Convolutional Neural Network – Long Short-Term Memory	Конволутивна неуронска мрежа са дугом краткотрајном меморијом
CV	Count Vector	Врећа речи имплементирана као низ
DEVOPS	Development and Operations	Развој и операције
XGBoost	Extreme Gradient Boosting	Екстремно градијентно појачавање
GAT	Graph Attention Networks	Графовске мреже са механизмом пажње
GBRT	Gradient Boosted Regression Trees	Градијентно појачана регресиона стабла
GNB	Gaussian Naive Bayes	Гаусовски наивни бајесовски модел

GNN	Graph Neural Network	Графовска неуронска мрежа
HTTP	Hypertext Transfer Protocol	Протокол за пренос хипертекста
HTTPS	Hypertext Transfer Protocol Secure	Сигурносни протокол за пренос хипертекста
IP	Internet Protocol	Интернет протокол
KNN	K-Nearest Neighbors	Алгоритам *k* најближих суседа
LSTM	Long Short-Term Memory	Мрежа са дугом краткотрајноом меморијом
LR	Logistic Regression	Логистичка регресија
MISP	Malware Information Sharing Platform	Платформа за дељење информација о злонамерном софтверу
MLP	Multi-Layer Perceptron	Вишеслојни перцептрон
NB	Naive Bayes	Наивни бајесовски модел
NLP	Natural Language Processing	Обрада природних језика
RNN	Recurrent Neural Network	Рекурентна неуронска мрежа
SVM	Support Vector Machine	Модел потпорних вектора
Self-attention	Self-attention mechanism	Механизам само-пажње
SMB	Server Message Block	Серверска блок-порука
TF-IDF	Term Frequency – Inverse Document Frequency	Учесталост термина – инверзна учесталост докумената

TextCNN	Textual Convolutional Neural Network	Текстуална конволутивна неуронска мрежа
TCNN-LSTM	Temporal Convolutional Neural Network - Long Short-Term Memory	Темпорална конволутивна неуронска мрежа – дуга краткотрајна меморија
WMI	Windows Management Instrumentation	Софтвер за управљање системом Виндоуз
WPE	Windows Portable Executable	Извршни фајл за оперативни систем Виндоуз
Word2Vec	Word to Vector	Реч у вектор

Попис табела

Попис слика

1 Увод

Злонамерни софтвери представљају изазов у области информационе безбедности[1], јер узрокују материјалне трошкове, губитак информација и штету по углед жртве. Због константног унапређења функционалности злонамерних софтвера, класични технички механизми одбране (попут антивирусних програма, софтверских агента са реактивном компонентом, мрежних баријера и др.), не спречавају, у општем случају, доспеће злонамерних узорака у радно окружење рачунара, о чему сведочи растући број злонамерних софтвера уочен у актуелним извештајима компанија: Мандиант (*Mandiant*, 2024), АВ-Атлас (*AV-ATLAS*, 2025) и Стејшн икс (*Station X*, 2025).

Напредне претње (енгл. *Advanced Persistent Threats - APT*), као високо софистициране и веома специфичне, чине идентификацију изазовнијом због примене шифроване комуникације, софистицираних техника напада, континуираног надгледања ресурса жртве, брисања или маскирања трагова итд. Да би се боље сакрио прави циљ злонамерног софтвера, користе се полиморфне и метаморфне технике за избегавање откривања. Ове технике мењају код при сваком извршавању, при чему основна сврха кода остаје иста. Компресија и шифровање су два најчешћа начина за скривање кода, али се користе и преименовање регистара, пермутација кода, проширење кода непотребним деловима, маскирање кода итд., што објашњава тако брз и константан раст броја злонамерних софтвера, чинећи анализу дуготрајном, скупом и сложеном, о чему, између осталих, извештавају Хиберт, Матју и Планис (*Gibert, Mateu, Planes*, 2020).

1.1 Проблем научног истраживања

Због пораста броја и сложености злонамерних софтвера, њихова динамичка анализа постала је важан део сајбер-одбране. Ова анализа представља процес испитивања понашања софтвера у контролисаном изолованом окружењу да би се открили његови стварни ефекти и функције. За разлику од статичке анализе, која укључује проучавање кода злонамерног софтвера без његовог извршавања, динамичка анализа омогућава истраживачима да посматрају како се он понаша током свог рада. Овај приступ укључује покретање злонамерног софтвера у виртуелним машинама за анализу (енгл. *sandbox*), што

[1]Термин „информациона безбедност” се у овој дисертацији употребљава као синоним термину „информациона сигурност”, по узору на законске и друге акте који регулишу ову област у Републици Србији. Треба приметити да се у литератури могу наћи различите дефиниције ових термина, без консензуса о њиховом интерпретирању.

омогућава безбедно праћење његових акција, као што су модификације системских фајлова, мрежне активности, интеракције са другим процесима итд.

Примена динамичке анализе злонамерног софтвера широка је и укључује циљеве као што су идентификација нових врста злонамерних софтвера, развој и унапређење алата за заштиту информационих система и подршка форензичким истрагама у случајевима сајбер-напада. Ова метода је нарочито корисна у откривању напредних претњи, које често користе сложене технике да би избегле детектовање традиционалним методама.

Резултат динамичке анализе софтвера јесте аутоматски генерисани извештај, који садржи доказе о радњама које је поднети узорак извршио. Међутим, тумачење ових извештаја може бити изазован задатак. На пример, понашање безопасног инсталационог пакета који генерише знатан број фајлова, уноса у регистре и других модификација циљног система, неће се значајно разликовати од понашања злонамерног софтвера који извршава сличне активности о чему пиши Сети и други (*Sethi et al.*, 2019) и Фико и други (*Ficco et al.*, 2022). Стога је за тумачење извештаја динамичке анализе често потребна стручна експертиза, што га чини дуготрајним и скупим.

Ова дисертација посматра специфични аспект овог проблема, који се односи се на унапређење аутоматске динамичке анализе злонамерног софтвера у контексту машинског учења.

1.2 Предмет научног истраживања

Примена машинског учења представља значајно унапређење динамичке анализе софтвера. Алгоритми машинског учења који се примењују аутоматски анализирају велике количине података добијених динамичком анализом, идентификујући обрасце и аномалије које указују на присуство злонамерног софтвера. То не само да повећава брзину и ефикасност анализе, већ и смањује потребу за интервенцијом стручњака.

Међутим, релевантни приступи динамичкој анализи софтвера засновани на машинском учењу доминантно су фокусирани на обележја која се односе на АПИ-позиве. АПИ-позив (енгл. *API – Application Programming Interface*) јесте поступак којим један софтверски систем или апликација комуницира са другим системом или апликацијом путем дефинисаног интерфејса. АПИ је скуп правила и протокола који омогућавају различитим апликацијама да размењују податке или функције. Анализом досадашњих публикација у овој области утврђено је да је мали број истраживача разматрао могућности проширења

овог скупа обележја или анализе целих извештаја генерисаних од стране окружења за динамичку анализу што је присутно и у Мира и други (*Mira et al.*, 2019).

Два главна недостатка приступа фокусираних искључиво на обележја која се односе на АПИ-позиве јесу следећа. Динамичка обележја која не припадају скупу АПИ-позива, попут обележја која рефлектују приступање фајловима на диску, промене у систему регистара оперативног система, мрежни саобраћај итд., јесу занемарена, иако потенцијално могу да буду релевантна за детектовање злонамерног софтвера. Значајан број злонамерних и безопасних софтвера не генерише АПИ-позиве (Илић и други, 2024а), што упућује на извођење додатних динамичких обележја. Предмет научног истраживања у овој дисертацији, односи се на проширење скупа динамичких обележја, у циљу унапређења перформанси аутоматског детектовања злонамерног софтвера у контексту машинског учења.

1.3 Хипотетички оквир

Општа хипотеза овог истраживања јесте да постоје динамичка обележја софтвера која су релевантна за аутоматско детектовање злонамерних софтвера, а не припадају скупу АПИ-позива.

Посебна хипотеза овог истраживања јесте да проширени скуп динамичких обележја софтвера унапређује тачност система за аутоматско детектовање злонамерних софтвера у односу на систем који би био заснован само на скупу обележја која се односе на АПИ-позиве.

1.4 Доприноси истраживања

Доприноси научног истраживања у овој дисертацији односе се на унапређење динамичке анализе злонамерног софтвера у контексту машинског учења:

1. Произведен је нови корпус података за динамичку анализу злонамерног софтвера који садржи 22200 узорака, од који је сваки узорак означен као безопасан или злонамеран. Сваки узорак представља извештај о бинарном фајлу, аутоматски генерисан од стране окружења отвореног кода за динамичку анализу софтвера Куку сендбокс (енгл. *Cuckoo Sandbox*).
2. Из произведеног корпуса изведен је скуп динамичких обележја софтвера који се односе на АПИ-позиве, али и друге аспекте понашања узорака, попут модификовања фајлова, мрежног саобраћаја итд. За издвајање обележја примењене су две технике:

ЦВ (енгл. *Count Vector, CV*) и ТФ-ИДФ (енгл. *Term Frequency – Inverse Document Frequency, TF-IDF*).

3. Практично је показано да проширење скупа обележја побољшава перформансе аутоматске динамичке анализе софтвера у контексту бинарног класификовања софтвера на злонамерни и безопасни, заснованог на ансамблима стабала одлучивања (енгл. *Random Forest - RF*). Ово укључује поређење перформанси модела машинског учења заснованих искључиво на АПИ-позивима с перформансама модела машинског учења заснованих на проширеном скупу динамичких обележја изведених из датог корпуса.
4. Извршени су издвајање најважнијих динамичких обележја, њихово интерпретирање и класификовање по типовима. Саопштена је расподела ових обележја по типовима, у контексту бинарног класификовања злонамерног софтвера. Тиме је указано на типове динамичких обележја који су од значаја за класификовање злонамерних софтвера, а не припадају класи АПИ-позива.

1.5 Структура дисертације

Ова дисертација је структурирана на следећи начин. Поглавље 2 даје преглед релевантних истраживања у овој области. Поглавље 3 описује производњу новог корпуса у изабраном окружењу за динамичку анализу злонамерних софтвера. Поглавље 4 извештава о скупу модела машинског учења дизајнираних за бинарно класификовање софтвера у контексту динамичке анализе и развијених на произведеном корпусу. Посебна пажња је посвећена поређењу перформанси модела развијених на скупу података који садржи проширени скуп динамичких обележја са перформансама модела развијених на изведеном скупу података који садржи само обележја која се односе на АПИ-позиве. У поглављу 5 дата је анализа издвојених динамичких обележја. Поглавље 6 представља дискусију о резултатима истраживања. Поглавље 7 закључује дисертацију.

2 Преглед релевантних истраживања

2.1 Увод

Увидом у релевантна истраживања у области динамичке анализе софтвера, могу се уочити два главна смера у вези са издвајањем обележја. Први смер, који је доминантно заступљен и заснован на анализи обележја која се односе на АПИ-позиве, размотрен је у секцији 2.2. Други смер, који је знатно мање заступљен и полази од претпоставке да скуп релевантних обележја превазилази скуп АПИ-позива, размотрен је секцији 2.3. У обе секције фокус разматрања примарно је стављен на корпусе динамичких обележја и моделе машинског учења. Преглед перформанси модела размотрених у овом поглављу дат је у секцији 2.4.

2.2 Динамичка обележја која се односе на АПИ-позиве

У информационој безбедности, секвенце АПИ-позива сматрају се индикативним за откривање злонамерног софтвера, што описују Диор и други (*Deore et al.*, 2023). Главна идеја овог приступа јесте та да секвенце АПИ-позива одражавају понашање злонамерног софтвера, и да различите варијанте истог софтвера генеришу сличне секвенце. За опсежни преглед откривања злонамерног софтвера на основу секвенци АПИ-позива, читалац може да консултује Мира и друге (*Mira et al.*, 2019). Овде су размотрена релевантна актуелна истраживања.

Дизгин и други (*Duzgun et al.*, 2022) извештавају о корпусу података који је креиран коришћењем окружења Куку сендбокс. Корпус података који је описан у раду садржи секвенце АПИ-позива генерисаних од стране 24411 категорисаних узорака злонамерних софтвера, али не укључује безопасне узорке. Овај корпус настао је спајањем два скупа података: први садржи 14616 узорака преузетих од платформе *VirusShare*, а други 9795 узорака преузетих од платформе *VirusSample*. На овим подацима примењено је седам различитих модела машинског учења: ансамбл стабала одлучивања, модел потпорних вектора (*SVM*), екстремно градијентно појачавање (*XGBoost*) и градијентно појачавање засновано на хистограму, неуронска мрежа са дугом краткотрајном меморијом (*LSTM*), као и два претходно обучена трансформер-модела: *BERT* (*Bidirectional Encoder Representations from Transformers*) и *CANNIE* (*Character Architecture with No tokenization In Neural Encoder*). Модели који су дали најбоље резултате, зависно од корпуса, јесу: модел потпорних вектора, екстремно градијентно појачавање, мрежа са дугом краткотрајном меморијом и трансформер-модел *CANINE*.

Алсмарњи и Елаиди (*Alchmarni, Alliheedi,* 2023) примењују окружење Куку на два корпуса: први садржи податке о 1080 безопасних и 2576 злонамерних узорака, а други 1079 безопасних и 42797 злонамерних узорака. Као и у претходном истраживању, представљени скуп података садржи скуп обележја ограничен на секвенце АПИ-позива. На овом корпусу обучавани су и тестирани различити алгоритми машинског учења: модел потпорних вектора, ансамбл стабала одлучивања, модел k најближих суседа (KNN), екстремно градијентно појачавање, конволутивна неуронска мрежа (CNN) и рекурентна неуронска мрежа (RNN). Сви модели су показали тачност која превазилази 90%. Рекурентне неуронске мреже показале су тачност у опсегу од 95% до 99%. Ансамбл стабала одлучивања показао је тачност од 95%.

Саида и Асгар (*Syeda, Asghar*, 2024) извештавају о корпусу који садржи узорке из извршних фајлова оперативног система Виндоуз (енгл. *Windows Portable Executable, WPE*). Овај корпус је релативно мали (582 злонамерних и 438 безопасних узорака) и садржи само АПИ-позиве. Шест модела машинског учења процењено је на овом скупу података, укључујући логистичку регресију (*LR*), модел потпорних вектора, наивни бајесовски модел (*NB*), ансамбл стабала одлучивања, екстремно градијентно појачавање и модел *k* најближих суседа, при чему је модел ансамбл стабала одлучивања показао највећу тачност класификовања.

Ђанг, Ву, Ђанг и Јанг (*Zhang, Wu, Zhang, Yang,* 2023), предлажу нови модел, *Mal-ASSF*, за детектовање злонамерних софтвера, који комбинује семантичке и секвенцијалне карактеристике АПИ-позива. За репрезентовање секвенци АПИ-позива користе графовски заснован метод (*API2Vec*), који редукује димензионалност обележја. Ова репрезентација обележја комбинована је са бидирекционим мрежама са дугом краткотрајном меморијом и механизмом пажње. Резултати експеримената показују да *Mal-ASSF* остварује тачност од 94,49%, што за 3% до 5% превазилази тачност других разматраних модела: текстуалне конволутивне неуронске мреже (*TextCNN*), конволутивне неуронске мреже са дугом краткотрајноом меморијом (*CNN-LSTM*) и темпоралне конволутивне неуронске мреже са дугом краткотрајном меморијом (*TCNN-LSTM*).

Хуан, Чен, Шао и Сан (*Huang, Chen, Hsiao, Sun,* 2022) примењују алгоритам за груписање секвенци АПИ-позива које су карактеристичне за уобичајено понашање породица злонамерног софтвера, и на којима потом обучавају неуронску мрежу. Истраживање је спроведено на корпусу секвенци АПИ-позива издвојених динамичком анализом 6585 узорака злонамерног софтвера. Коришћени су следећи модели: рекурентна неуронска

мрежа са механизмом пажње (*RasNN*), трансформер-модел (*BERT*), енкодер мрежа (*Sent2Vec*) са механизмом пажње (*Self-attention*) и рекурентна мрежа са ГРУ-неуронима (*Gated Recurrent Unit*). Најбољи резултати су постигнути применом рекурентне неуронске мреже са механизмом пажње.

Хуан, Сан и Чен (*Huang, Sun, Chen,* 2022) предлажу модел *TagSeq* заснован на неуронској мрежи, који користи динамичке податке о понашању злонамерног софтвера, у коме су АПИ-позиви коришћени у проширеном смислу, тј., са свим параметрима који се прослеђују уз сам позив. Окружење које је коришћено преузето је од Шаоа, Сана и Чена (*Hsiao, Sun, Chen,* 2020), и у њему је извршено 11677 узорака који припадају различитим породицама софтвера. У експериментима су коришћени модели: модел *TagSeq* – заснован на дугој краткотрајној меморији и методи за процењивање максималне веродостојности (*MLC*), модел *TagSeq* заснован на моделу енкодера-декодера (*Seq2Seq*), гаусовски наивни бајесовски модел (*GNB*), стабла одлучивања, алгоритам *k* најближих суседа, ансамбл стабала одлучивања, линеарни модел потпорних вектора (*LSVM*) и конволутивне неуронске мреже. Најбољи резултати су постигнути применом *TagSeq* (*LSTM + MLC*).

Чен, Јагмен и Даунин (*Chen, Yagemann, Downing,* 2019), предлажу графичку интерпретацију АПИ-позива, АПИ-секвенци и учесталости АПИ-позива, на којој врше обучавање дубоке неуронске мреже. Аутори користе мрежу са дугом краткотрајном меморијом и неуронске мреже VGG (енгл. *Visual Geometry Group – VGG*) са трансфером учења, како би класификовали генерисане дигиталне слике које репрезентују злонамерне софтвере.

Да би побољшали перформансе својих модела, неки аутори разматрају проширење АПИ-позива параметрима који се унутар њих прослеђују. У наставку су размотрени актуелни радови који усвајају ову идеју.

Алхашми и други (*Alhashmi et al.,* 2023) примењују технику ТФ-ИДФ за издвајање обележја (в. секцију 4.2.2) из АПИ-позива добијених применом статичке анализе извршних фајлова и динамичке анализе АПИ-позива. Обрађени корпус садржи 6877 злонамерних и 4835 безопасних узорака. У раду је презентован хибридни модел за детекцију варијанти злонамерног софтвера који примењује екстремно градијентно појачавање и вештачку неуронску мрежу *(ANN)*.

Ли, Чејон, Ли и Чо (*Lee, Jeon, Lee, Cho,* 2022) примењују окружење за динамичку анализу под називом *ARBDroid*, дизајнирано за анализирање злонамерних софтвера (приближно 10000 узорака) који користе технике прикривања (енгл. *obfuscation*) и издвајање

програмских инструкција, стрингова и АПИ-позива на платформи Андроид. У раду није коришћено машинско учење за процену квалитета издвојених обележја у контексту класификовања софтвера.

Чен, Санг, Алми и Су (*Chen, Zeng, Lv, Zhu,* 2024) проширују скуп АПИ-позива њиховим параметрима и спољашњим индикаторима компромитовања (енгл. *Indicators of Compromise - IOC*). Први корпус коришћен у овом истраживању садржи 30000 секвенци АПИ-позива прикупљених извршавањем Виндоуз-фајлова у сендбокс-систему, од којих је 10000 злонамерно. Други корпус садржи 12000 АПИ-секвенци од којих је 8000 злонамерно. Додатна база података, која садржи 232397 индикатора компромитовања, направљена је на основу 17341 документа, преузетих од три компаније које пружају онлајн-услуге информисања о индикаторима компромитовања. У раду се уводи модел *CTIMD* који је направљен помоћу три неуронске мреже.

Јао и други (*Yau et al.*,2023), осим назива АПИ-позива користе и његове аргументе и друге – статистичке карактеристике, као што су повратне вредности и број АПИ-позива у узорку. Због разноврсности АПИ-позива у односу на број аргумената, назив и повратне вредности, аутори примењују хеш-функције за генерисање вектора обележја фиксне дужине. Корпус је прикупљен од компаније за антивирусну заштиту и укључује 14860 узорака, од којих 7398 злонамерних и 7462 безопасна. У овом раду користе се следећи модели: лако градијентно појачање (*LightGBM*), екстремно градијентно појачавање, ансамбл стабала одлучивања, резидуална неуронска мрежа *ResNet1D,* и потпуно повезана мрежа са пропагацијом унапред. Модели су експериментално оптимизовани.

Шу и Чен (*Xu, Chen,* 2023) моделују инстанцу злонамерног софтвера као стабло понашања (енгл. *Behavioral Tree*), генерисано из секвенце АПИ-позива. Израчунавање сличности између појединих узорака и класа злонамерног софтвера врши се на основу вектора обележја изведених применом технике ТФ-ИДФ. На ове векторе обележја примењује се наивни бајесовски модел. Испитивање је извршено на скупу података који садржи 10620 узорака злонамерног софтвера из 43 породице, а тачност класификовања применом оваквог скупа обележја већа је за 10% од тачности добијене на основу обележја заснованих само на АПИ-позивима.

Ли и други (*Li et al.*, 2022) предлажу хибридни приступ издвајању семантичких карактеристика из назива АПИ-позива и њихових аргумената, помоћу којих се генерише граф АПИ-позива. Примењују више техника за кодовање обележја укључујући приступ

Word2Vec. Након генерисања графа АПИ-позива, обучавају се графовска неуронска мрежа (*GNN*), преузета од Хуа и других (*Hu et. al.,* 2020) и графовска мрежа са механизмом пажње (енгл. *Graph Attention Networks – GAT*), приказана у раду Величковића и других (2018).

Ли и други (*Li et al.,* 2023) баве се проблемом отклањања шума насталог применом техника прикривања извршавања од стране злонамерног софтвера. Примењују динамичку анализу уз увођење механизма меког прага у резидуалној мрежи да би се постигло филтрирање компоненти шума у секвенцама АПИ-позива. Обучавају бидирекциону неуронску мрежу са дугом краткотрајном меморијом *(BiLSTM*).

Нунес и други (*Nunes et al.,* 2019) наглашавају разлику између прикупљања АПИ-позива на нивоима корисника и кернела, у односу на способност класификатора да открије злонамерни софтвер. Коришћен је ансамбл стабала одлучивања који постиже најбоље перформансе комбиновањем обележја са оба нивоа. Обележја на корисничком нивоу ослањају се на „анти-дебаг“ технике, док обележја на кернел нивоу одражавају опште понашање система, чиме се ова два приступа међусобно допуњују. Представљени резултати истичу значај података прукупљених на кернел-нивоу за класификовање злонамерног софтвера.

Ли, Лу, Ма, Чен и Дон (*Li, Lu, Ma, Chen, Don,* 2024) представљају алгоритам *PrefixSpan* за издвајање честих АПИ-позива генерисаних од стране злонамерног софтвера, од којих је створена база правила. АПИ-позиви који се испитују класификују се применом модела, а по потреби се прослеђују текстуалној конволутивној неуронској мрежи (*TextCNN*), која представља додатни модел за класификовање.

Ерера-Силва и Ернандес-Алварес (*Herrera-Silva, Hernández-Álvarez,* 2023), генерисали су корпус који садржи динамичке карактеристике ренсомвера (енгл. *ransomware*), који су тестирани и одабрани на основу релевантности и корелације. Корпус садржи 50 најважнијих обележја, који су тестирани применом ансамбла стабала одлучивања, градијентно појачаног регресионог стабла *(GBRT),* гаусовског наивног бајесовског модела и неуронске мреже са унакрсном-валидацијом. Најбоље резултате је показао модел градијентно појачаног регресионог стабала.

Иршад и Дота (*Irshad, Dutta,* 2020) користе приступ обраде узорака у окружењу Куку, али избор обележја се врши на основу пет најважнијих функција. За класификовање су коришћена два класификатора: ансамбл стабала одлучивања, који је постигао тачност од 85% и стабло одлучивања, са тачношћу од 83%.

Сров и Кумар (*Sraw, Kumar*, 2022) посматрају проблем одређивања ауторства злонамерног софтвера. Корпус који се користи у овом истраживању садржи приближно 120000 узорака преузетих са портала *VirusShare*, додатне податке са портала *VirusTotal*, и податке о мрежном саобраћају добијене од стране сендбокс-система. Скуп обележја АПИ-позива је додатно обогаћен скупом од осам ручно одабраних обележја, преузетих са портала *VirusTotal* и извештаја о динамичкој анализи софтвера. Аутори су обучавали и тестирали следеће моделе: мултиноминални наивни бајесовски класификатор, модел потпорних вектора и ансамбл стабала одлучивања. Посебна пажња је посвећена оптимизовању модела. Осим класификације спроведен је и експеримент са визуелизацијом злонамерног софтвера.

Сети и други (*Sethi et al.*, 2019) проширују скуп обележја АПИ-позива обележјима која су аутоматски изведена из извештаја о динамичкој анализи софтвера генерисаних од стране сендбокс-система Куку. Додатне карактеристике су издвојене применом χ^2 теста и ансамбла стабала одлучивања.

2.3 Проширени скуп динамичких обележја

Ниједно од претходно размотрених истраживања не користи потпуне извештаје о динамичкој анализи софтвера генерисане од стране сендбокс-система као извор обележја. Према најбољем сазнању аутора у овом тренутку, доступне су само две публикације у којима је описан овај приступ.

Ђиндел и други (*Jindal et al.*, 2019) користе узорке који долазе из два извора: корпус преузет од власника комерцијалног окружења за анализу софтвера (13760 безопасних и 13760 злонамерних) и корпус изведен насумичним одабиром узорака из јавно доступног скупа података *EMBER* за статичку анализу узорака, који су генерисали Андерсон и Ротл (*Anderson, Rothl,* 2018). Анализом наведених узорака, у окружењу Куку и додатном комерцијалном окружењу за динамичку анализу софтвера добијена су четири корпуса. Међутим, ниједан од ових корпуса није јавно доступан. У фази обраде корпуса, аутори врше токенизацију извештаја и задржавају првих 10000 најчешћих токена. Коришћен је модел, који комбинује конволутивну неуронску мрежу, бидирекциону мрежу са дугом краткотрајном меморијом и механизмом пажње.

Други приступ заснован на комплетним извештајима о динамичкој анализи софтвера применили су Бошански и други (*Bosansky et al.*, 2022). Они извештавају о скупу података са 48976 узорака злонамерног софтвера, генерисаног у окружењу за динамичку анализу Кејп (енгл. *CAPE*). Иако је сваки узорак означен у односу на фамилију злонамерног

софтвера којој припада, корпус не садржи бенигне узорке и разматра само шест породица злонамерног софтвера. На пример, узорци рансомвера нису заступљени, што објашњава чињеницу да максимална величина појединачних извештаја о разматраним узорцима не прелази један гигабајт (величина целог некомпресованог скупа података је 563GB). Класификовање је засновано на парадигми хијерархијског учења са више инстанци (*Hierarchical Multi-Instance Learning - HMIL*).

Иако су у наведеним радовима (Ђиндел и други, 2019; Бошански и други, 2022), коришћени извештаји из окружења за динамичку анализу злонамерног софтвера, корпус из првог истраживања није јавно доступан, док корпус из другог истраживања не укључује безопасне узорке и садржи злонамерне узорке из само шест породица злонамерног софтвера.

У овој дисертацији приказано је генерисање скупа података који садржи извештаје о динамичкој анализи безопасних и злонамерних узорака генерисаних применом окружења Куку. Разлика у односу на Ђиндела и друге (2019), осим у јавној доступности корпуса, јесте у томе што се у овој дисертацији не разматра само подскуп најчешћих обележја. Полазећи од претпоставке да обележја са малом учесталошћу појављивања могу да имају незанемарљиву информациону добит, разматрају се комплетни и непромењени извештаји. За разлику од Бошанског и других (Bosansky et al., 2022), у истраживање су укључени и безопасни узорци и додатне породице злонамерних софтвера. Треба приметити да тренутно не постоје други јавно доступни корпуси комплетних извештаја о динамичкој анализи софтвера са тако великим бројем узорака као корпус представљен у овој дисертацији.

2.4 Преглед перформанси модела

У овој секцији дат је преглед перформанси модела (табела 1 и 2), размотрених у овом поглављу.

Табела 1: Преглед перформанси модела машинског учења у релевантним истраживањима заснованим на АПИ-позивима (*A* – тачност, *P* – прецизност, *R* – одзив, F_1 – балансирана Ф-мера, AUC – површина испод ROC криве).

Рад	Модел		Мере квалитета				
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
Дизгин и други (*Duzgun et al.*, 2022) Корпус: *VirusShare*	Вишекласна	RF				60,20	93,34
		SVM				73,43	92,26
		XGBoost				71,78	96,66
		HGBoost				69,52	95,82
		LSTM				70,07	93,59
		BERT				70,68	94,32
		CANINE				69,55	92,61
Дизгин и други (*Duzgun et al.*, 2022) Балансирани корпус: *VirusShare*	Вишекласна	RF				66,09	93,04
		SVM				75,81	94,04
		XGBoost				75,25	95,77
		HGBoost				74,70	94,47
		LSTM				70,07	93,59
		BERT				74,47	88,43
		CANINE				72,84	90,45
Дизгин и други (*Duzgun et al.*, 2022) Корпус: *VirusSample*	Вишекласна	RF				55,10	89,62
		SVM				73,31	96,40
		XGBoost				73,51	98,16
		HGBoost				73,15	97,76
		LSTM				77,88	96,82
		BERT				72,53	95,90

Рад	Модел		Мере квалитета				
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
		CANINE				71,82	96,21
Дизгин и други (*Duzgun et al.*, 2022) Балансирани корпус: *VirusSample*	Вишекласна	RF				83,91	96,88
		SVM				89,75	97,82
		XGBoost				90,31	99,41
		HGBoost				88,02	98,41
		LSTM				84,00	94,78
		BERT				89,48	97,08
		CANINE				90,86	98,28
Алсмарњи и Елаиди (*Alchmarni, Alliheedi*, 2023) Корпус 1080 безопасних и 2576 злонамерних узорака.	Бинарна	CNN	98,00	98,00	98,00	98,00	97,00
		RNN	99,00	98,00	99,00	99,00	98,00
		SVM	91,00	91,00	91,00	91,00	96,00
		KNN	92,00	92,00	92,00	92,00	95,00
		XGB	93,00	92,00	93,00	92,00	97,00
		RF	95,00	95,00	95,00	95,00	98,00
		GBC	95,00	95,00	96,00	95,00	96,00
Алсмарњи и Елаиди (Alchmarni, Alliheedi, 2023) Корпус 1079 безопасних и 42797 злонамерних.	Бинарна	CNN	98,00	98,00	98,00	98,00	98,00
		RNN	99,00	99,00	99,00	99,00	99,00
		SVM	95,00	96,00	94,00	95,00	96,00
		KNN	99,00	99,00	99,00	99,00	98,00
		XGB	96,00	97,00	95,00	96,00	97,00
		RF	98,00	98,00	98,00	98,00	98,00
		GBC	99,00	99,00	98,00	99,00	99,00
Саида и Асгар (*Syeda, Asghar*, 2024)	Вишекласна	LR	88,00	92,00	92,00	92,00	93,00
		SVM	72,00	98,00	51,00	67,00	88,00
		NB	84,00	96,00	73,00	83,00	84,00
		RF	96,00	99,00	93,00	96,00	98,00
		XGB	93,00	96,00	96,00	96,00	97,00
		KNN	92,00	97,00	87,00	91,00	93,00
		KNN	86,46	87,85	85,76	86,81	

Рад	Модел		Мере квалитета				
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
Ђанг, Ву, Ђанг и Јанг (*Zhang, Wu, Zhang, Yang*, 2023)	Вишекласна	SVM	88,46	89,85	85,76	87,75	
		DT	89,78	89,17	89,13	89,14	
		RF	90,45	88,95	91,54	90,23	
		XGB	91,84	91,95	91,54	91,77	
		TEXTCNN	88,69	88,59	88,69	88,64	0,97
		CNN-LSTM	90,23	89,64	90,23	89,93	0,99
		TCNN-LSTM	92,13	92,17	92,13	92,15	0,99
		Mal-ASSF	94,49	94,01	94,19	94,02	0,99
Хуан, Чен, Шао у Сан (*Huang, Chen, Hsiao, Sun*, 2022) 10% пуног корпуса.	Вишекласна	RasNN		88,15	86,56	87,35	
		BERT		85,81	86	85,91	
		Sent2Vec		70,95	85,12	77,37	
		Self-attention		77,99	40,01	52,89	
		GRU		63,93	73,13	68,22	
Хуан, Чен, Шао у Сан (*Huang, Chen, Hsiao, Sun*, 2022) Класификовање у 10 класа.	Вишекласна	DL-SVM		52,7	44,05	47,78	
		RasNN		58,69	57,89	57,21	
Хуан, Чен, Шао у Сан (*Huang, Chen, Hsiao, Sun*, 2022) Класификовање са непознатим узорцима.	Вишекласна	DL-SVM		42,28	40,46	40,06	
		RasNN		54,68	56,02	54,68	
Хуан, Сан и Чен (*Huang,*	Вишекласна	GaussianNB		56,06	25,21		

Рад	Модел		Мере квалитета				
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
Sun, Chen, 2022) Корпус само имена АПИ функција.		Decision Tree		40,34	55,13		
		Kneighbors		40,94	56,33		
		Random Forest		39,85	74,13		
		LinearSVC		41,26	68,86		
		CNN		42,72	69,32		
		TagSeq (LSTM + MLC)		45,50	72,39		
		TagSeq (Seq2Seq)		57,10	53,02		
Хуан, Сан и Чен (*Huang, Sun, Chen*, 2022) Корпус имена АПИ функција и повратних вредности.	Вишекласна	GaussianNB		58,10	26,35		
		Decision Tree		41,96	53,06		
		Kneighbors		43,46	57,35		
		Random Forest		39,90	74,85		
		LinearSVC		41,80	70,01		
		CNN		40,35	72,27		
		TagSeq (LSTM + MLC)		44,66	74,10		
		TagSeq (Seq2Seq)		56,25	52,39		
Хуан, Сан и Чен (*Huang,*	Вишекласна	GaussianNB					

Рад	Модел		Мере квалитета				
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
Sun, Chen, 2022) Корпус имена АПИ функција, повратних вредности и параметара.		Decision Tree					
		Kneighbors					
		Random Forest					
		LinearSVC		40,82	69,63		
		CNN		46,4	72,1		
		TagSeq (LSTM + MLC)					
		TagSeq (Seq2Seq)		57,18	53,05		
Чен, Јагмен и Даунин (*Chen, Yagemann, Downing*, 2019), Корпус ПДФ фајлова.	Бинарна	VGGNet	100,00				
Чен, Јагмен и Даунин (*Chen, Yagemann, Downing*, 2019), Случај употребе Виндоуз извршних фајлова.	Бинарна	VGGNet	94,00				
Алхашми и други	Бинарна	API-dynamic	94,00	89,00	89,00	89	

Рад	Модел		Мере квалитета				
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
(*Alhashmi et al.*, 2023)		API-static	91,00	83,00	85,00	84,00	
		API-HMVD	95,00	93,00	94,00	93,00	
Чен, Санг, Алми и Су (*Chen, Zeng, Lv, Zhu,* 2024) Тренирање на 80% узорака сваког од два корпуса и валидација на 20% супротног корпуса.	Бинарна	LR	88,4	84,2	80,8	82,4	
		NB	83,6	73,5	79,5	76,4	
		DT	87,6	83,2	78,6	80,8	
		RF	93,6	90,8	90	90,3	
		TextCNN	98,2	98,6	96,1	97,3	
		ABiLSTM	98,2	97,7	96,8	97,2	
		Transformer	97,1	96,4	95,1	95,6	
		CTIMD-TC	98,3	97,8	97	98,4	
		CTIMD-AB	98,2	98,4	96,3	97,3	
		CTIMD-TF	97,1	96,9	94,4	95,7	
Чен, Санг, Алми и Су (*Chen, Zeng, Lv, Zhu,* 2024) Тренирање на оба корпуса заједно.	Бинарна	LR	64,10	64,20	63,70	63,90	
		NB	72,00	82,50	56,00	66,70	
		DT	50,50	50,20	98,80	66,60	
		RF	73,20	80,90	60,90	69,50	
		TextCNN	93,90	99,40	91,40	95,20	
		ABiLSTM	98,40	99,40	98,20	98,80	
		Transformer	89,60	98,80	85,40	91,60	
		CTIMD-TC	99,50	99,40	99,80	99,60	
		CTIMD-AB	99,40	99,30	99,90	99,60	
		CTIMD-TF	96,00	99,30	94,60	96,90	
	Бинарна	LightGBM	97,42±0,35	97,75±0,27	97,05±0,54		99,57±0,09

Рад	Модел		Мере квалитета				
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
Јао и други (*Yau et al.*,2023)		XGBoost	97,11±0,38	96,97±0,42	97,23±0,59		99,56±0,10
		CatBoost	96,94±0,21	96,69±0,32	97,19±0,47		99,56±0,08
		RF	96,59±0,26	95,85±0,35	97,38±0,36		99,47±0,10
		ResNet1D	96,55±0,24	96,88±0,36	96,17±0,30		98,42±0,42
		MLP	95,42±0,34	95,97±0,76	94,80±1,32		98,43±0,32
		KNN	94,89±0,15	94,61±3,01	95,16±0,22		98,07±0,14
		EfficientNetB0	73,76±1,13	92,37±0,92	51,55±2,56		6,85∓2,31
		LogisticRegression	58,47±0,78	54,90±0,46	92,90±1,26		87,93±0,40
Ли и други (*Li et al.*, 2022)	Бинарна	DMalNet	98,43				
	Вишекласна	DMalNet	91,42				
Нунес и други (*Nunes et al.*, 2019) Корпус настао анализом кроз Кернел	Бинарна	AdaBoost	94,10	93,40		94,10	98,30
		Decision Tree	92,30	90,60		92,50	94,40
		Linear SVM	90,30	87,30		90,60	94,50
		Nearest Neighbour	90,30	89,60		90,30	96,40
		Random Forest	95,20	96,00		94,40	98,60
Нунес и други (*Nunes et al.*, 2019) Корпус настао кроз	Бинарна	AdaBoost	91,8	91,10		92,00	97,30
		Decision Tree	87,8	91,80		91,30	94,30
		Linear SVM	86,9	83,50		87,00	93,20

Рад	Модел		Мере квалитета				
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
Куку сендбокс.		Nearest Neighbour	86,2	87,70		86,30	94,20
		Random Forest	94	95,80		94,20	98,40
Ли, Лу, Ма, Чен и Дон (*Li, Lu, Ma, Chen, Don*, 2024) Вредност н-грама: 2	Бинарна	Random Forest	90,05	90,22	90,10		
		Logistic Regression	89,28	89,50	89,28		
		KNN	88,56	88,56	88,30		
Ли, Лу, Ма, Чен и Дон (*Li, Lu, Ma, Chen, Don*, 2024) Вредност н-грама: 3	Бинарна	Random Forest	90,28	90,37	90,48		
		Logistic Regression	89,38	89,64	89,59		
		KNN	88,82	88,82	89,02		
Ли, Лу, Ма, Чен и Дон (*Li, Lu, Ma, Chen, Don*, 2024) Вредност н-грама: 4	Бинарна	Random Forest	89,88	90,29	90,03		
		Logistic Regression	89,10	89,33	89,49		
		KNN	88,56	88,82	85,58		
Ли, Лу, Ма, Чен и Дон (Li, Lu, Ma, Chen, Don, 2024)	Бинарна	TextCNN (1 3 3 5 5)[2]	90,73	90,57	90,73	90,47	
		TextCNN (1 3 5)	92,79	92,78	92,8	92,76	

[2] Вредности у заградама се односе на величину конволутивног језгра (кернела).

Рад	Модел		Мере квалитета				
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
		TextCNN (1 2 4)	91,79	91,73	91,79	91,63	
		TextCNN (2 3 4)	91,65	91,69	91,65	91,6	
		Прилагођени TextCNN	92,88	92,46	99,93	96,05	
Ерера-Силва и Ернандес-Алварес (*Herrera-Silva, Hernández-Álvarez*, 2023)	Бинарна	Naive Bayes	63,39	68,15	57,86		
		Neural Networks	98,06	98,78	94,71		
		Random Forest	92,88	96,39	65,28		
		Gradient Boosted Trees	99,68	99,81	98,11		
Сров и Кумар (*Sraw, Kumar*, 2022)	Вишекласна	SVM (Virtual Total корпус 95,000 samples)	84,99	83,98	84,99	83,72	
		SVM (Virtual Total и Куку корпус, 1936 узорака)	67,87	69,72	67,87	66,48	
		CNN (Класификаци	97,00				

Рад	Модел		Мере квалитета				
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
		ја слика)					
Сети и други (*Sethi et al.*, 2019)	Бинарна	K-Nearest Neighbor (108)[3]	94,68	95,00	95,00	91,00	85,00
		Decision Tree (108)	99,37	99,00	99,00	98,00	98,00
		Support Vector Machine (1000)	89,37	92,00	89,00	90,00	93,00
		Random Forest (92)[4]	95,93	96,00	96,00	96,00	81,00
	Вишекласна	K-Nearest Neighbor	96,46	97,00	96	96,00	
		Decision Tree	99,11	99,00	99,00	99,00	
		Support Vector Machine	86,72	88,00	87,00	87,00	
		Random Forest	88,23	90,00	88,00	86,00	

[3] У загради је број обележја на којима је модел коришћен.
[4] У загради је број обележја на којима је модел коришћен.

Табела 2: Преглед перформанси модела машинског учења у релевантним истраживањима заснованим на проширеном скупу динамичких обележја (*A* – тачност, *P* – прецизност, *R* – одзив, *F1* – балансирана Ф-мера, *AUC* – површина испод *ROC* криве).

Рад	Модел	Мере квалитета					
	Класификација	Назив	A %	P %	R %	F_1 %	AUC %
Ђиндел и други (*Jindal et al.*, 2019)	Бинарна	MalDy[5]	89,23	85,00	83,00	88,50	
		Counts Model	93,50	89,10	94,10	92,00	
		Ensemble Model	98	98,6	97,5	98	
		Raw Model	91,40	94,80	90,60	92,70	
		Neurlux[6]	96,80	96,70	96,60	96,80	
Бошански и други (*Bosansky et al.*, 2022) Модел је тестиран посебно на неколико класа злон. софтв.	Вишекласна	HMIL	86,54-100				

[5] Модел из рада Ђен и Син (*Jain*, *Singh*,2019).
[6] Модел дизајниран у раду.

3 Нови корпус за динамичку анализу злонамерног софтвера

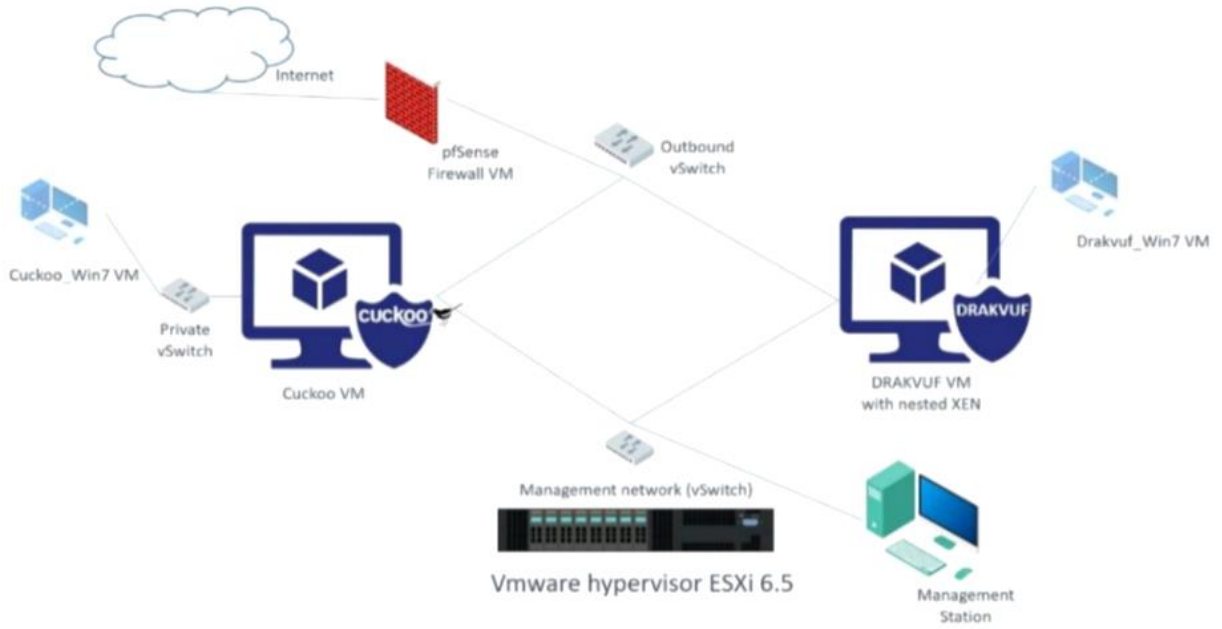
3.1 Увод

У претходном поглављу указано је на недостатак јавно доступних скупова података за динамичку анализу злонамерних фајлова чији је скуп обележја знатно шири од скупа АПИ-позива. За потребе овог истраживања створен је квалитативно и квантитативно одговарајући скуп података, настао динамичком анализом понашања реалних злонамерних фајлова преузетих из базе портала Вирус тотал (*Virus Total*, 2025) и безопасних фајлова добијених од компаније Техника КБ д.о.о. Београд. Генерисање овог скупа података (у даљем тексту: корпус) описано је у овом поглављу, са посебним освртом на пажљиво подешавање окружења за динамичку анализу, спроведеним да би се из добијених извештаја извео максималан број обележја. Пре припреме лабораторијског окружења за генерисање корпуса, било је неопходно одабрати окружење за динамичку анализу злонамерног софтвера, за шта су, као два најозбиљнија кандидата из области софтвера отвореног кода, разматрана окружења Куку сендбокс (енгл. *Cuckoo Sandbox*, у наставку Куку) и Драквуф сендбокс (енгл. *Drakvuf Sandbox*, у наставку: Драквуф).

3.2 Упоредна анализа окружења Куку и Драквуф

Упоредном анализом окружења за динамичку анализу злонамерног софтвера отвореног кода утврђено је да је окружење Куку прикладније за потребе овог истраживања (в. Илић, Гњатовић, Поповић и Мачек, 2022).

Куку и Драквуф су окружења намењена за динамичку анализу злонамерног софтвера и често се примењују у области професионалне анализе софтвера, а засновани су на алатима отвореног кода. Куку примењује софтверског агента дефинисаног у форми Пајтон-скрипте, у виртуелној машини за анализу, док Драквуф примењује приступ без агента. Усвојена архитектура лабораторије (Илић, Гњатовић, Поповић и Мачек, 2022) приказана је на слици 2, где је наглашено да су оба решења инсталирана и подешена унутар виртуалних машина (лево на слици: Куку; десно: Драквуф), док рачунари за анализу, у којима се извршавају потенцијално злонамерни фајлови, представљају засебно инсталиране виртуелне машине са оперативним системом Виндоуз 7. Све виртуелне машине засноване су на хипервизору „*vmware ESXi*“. Архитектура наведена на слици 2 коришћена је за генерисање корпуса коришћеног у овој дисертацији, при чему је, након извршене анализе, одабрано и примењено само окружење Куку.



Слика 1: Архитектура лабораторијског окружења за поређење система Куку и Дракву (преузето од Илића, Гњатовића, Поповић и Мачека, 2022)

3.2.1 Окружење Куку

Окружење Куку анализира понашање софтвера током извршавања у изолованом окружењу, тј., виртуелној машини. Прате се АПИ-позиви, мрежни саобраћај, промене у регистрима оперативног система Виндоуз, креирање фајлова итд. Препознавање злонамерних активности врши се коришћењем потписа, који представљају Пајтон-скрипте. Откривање злонамерног софтвера постиже се коришћењем динамичке библиотеке *cuckoomon.dll*, која је део процеса евидентирања понашања, а прикупљене информације преносе се главном процесу окружења Куку (Илић, Гњатовић, Поповић и Мачек, 2022).

Куку може да се интегрише са локалним решењима за електронску пошту и системима за спречавање упада (енгл. *Intrusion prevention system*), да би се идентификовали злонамерни фајлови у мрежном саобраћају. Окружењем се може управљати преко командне линије и веб-интерфејса, а у оквиру анализе доступни су узорци злонамерног софтвера и извештаји о њиховом понашању. Пре извршења злонамерног софтвера, виртуелна машина за анализу враћа се у почетно стање (енгл. *snapshot*), што обезбеђује да трагови претходних анализа не утичу на наредну. Пајтон-агент на виртуелној машини за анализу служи да изврши узорак злонамерног софтвера и пошаље извештај о његовом понашању управљачком механизму Куку-сендбокса.

У лабораторијском окружењу коришћена је верзија 2.0.7 окружења Куку, заснована на верзији 2 програмског језика Пајтон. У тренутку писања ове дисертације, доступна је верзија 3 окружења Куку (в. *Cuckoo Sandbox CERT Estonia*, 2025). Међутим, ова верзија окружења је још увек у развојној фази, због чега није коришћена у дисертацији, иако је потенцијално интересантна за будућа истраживања.

3.2.2 Окружење Драквуф

Окружење Драквуф, које су описали Ленгјел и други (*Lengyel et al*., 2014), анализира понашање злонамерног софтвера, али без коришћења софтверског агента. Уместо тога, ово окружење примењује тзв. интроспекцију виртуелне машине (енгл. *Virtual Machine Introspection, VMI*), чиме се врши праћење злонамерних активности софтвера на корисничком нивоу и на нивоу кернела оперативног система, о чему извештавају Мелвин и Кетрин (*Melvin, Kathrine*, 2020). На овај начин се, на нивоу хипервизора, могу надгледати извршавање процеса, операције над фајловима, системски позиви и функције кернела, са могућношћу уочавања напада „*kernel rootkit*“. Овим се умањује могућност злонамерног софтвера да открије окружење за анализу и, последично, активира технике избегавања (Мелвин и Кетрин, 2020). Уместо агента, окружење Драквуф користи технику уметања тачке прекида (енгл. *breakpoint*) у којој се инструкције уписују у меморију на локацијама од интереса, што детаљније разматрају Ленгјел и други (2014). Подешавањем инструкције „*VM-Exit*“ постиже се да, након извршења тачака прекида, хипервизор прослеђује ове податке управљачком делу окружења Драквуф, чиме се евидентира извршење било ког кода унутар виртуелне машине за анализу. Техника увођења тачака прекида, која се обично користи за откривање грешака у коду (енгл. *debug*), користи се за аутоматско праћење извршавања процеса оперативног система, укључујући и интерне функције кернела.

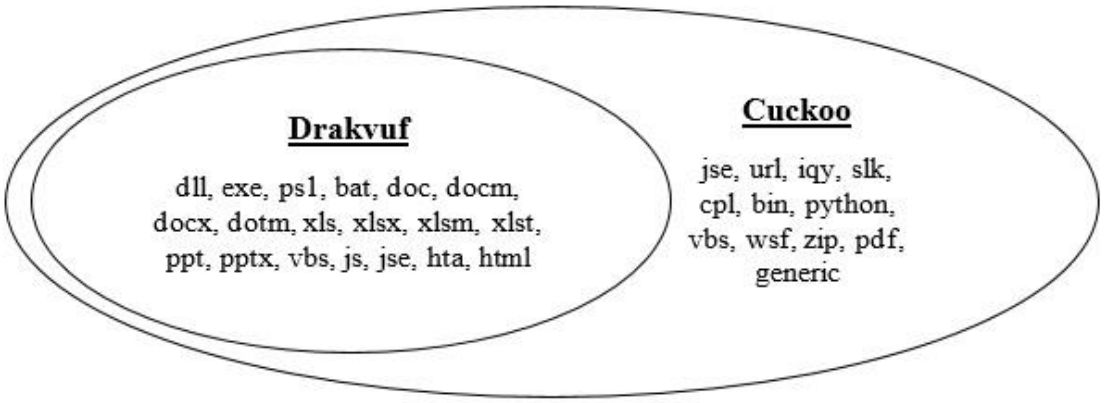
3.2.3 Упоредни преглед

У табели 3 приказане су категорије података садржаних у извештајима генерисаним од стране оба посматрана окружења. На основу овог упоредног приказа, закључено је да окружење Куку прикладније за примену у истраживању приказаном у оквиру ове дисертације. Први разлог односи се на релативно велики број обележја која издваја окружење Куку. Знатно је мањи део обележја која се могу извести из лог-фајлова окружења Драквуф.

Табела 3: Упоредни преглед садржаја извештаја генерисаних од стране окружења Куку и Драквуф (преузето од Илића, Гњатовића, Поповић и Мачека, 2022).

Куку	Драквуф
Сажетак	Метаподаци
Информације о извршењу	
Бодовање (функционалност у бета верзији)	-
Јара-правила (енгл. *Yara*)	-
Преузимање узорака	-
Потписи (за три ризичне категорије)	-
Слике екрана	-
Мрежни саобраћај (доступно у извештају)	Преузимање мрежног саобраћаја (сачуваног у фајлу)
Статичка анализа	-
Издвајање артефаката	-
Анализа понашања (информативнија)	Стабло процеса (напреднија, графичка репрезентација)
	Граф понашања (напреднија, графичка репрезентација)
Креирање фајлова	-
Креирање бафера	-
Меморија процеса	-
Додатне операције (нпр. поновно подношење узорака, поновна анализа са рестартовањем итд.)	-

Други разлог за избор окружења Куку јесте већи број типова фајлова који могу бити обрађени, у поређењу са окружењем Драквуф, што је приказано на слици 2.



Слика 2: Подржани типови фајлова у окружењима Куку и Драквуф (преузето од Илића, Гњатовића, Поповић и Мачека, 2022).

Упоредном анализом окружења (приказаном у табели 4) истакнуте су и друге предности окружења Куку. Претходно описане предности, које су важне за генерисање корпуса у смислу квантитета и квалитета обележја, сврстане су у категорије „Извештавање“ (бр. 3) и „Подржани типови фајлова“ (бр. 5). У техничком смислу, од великог значаја за аутоматско генерисање корпуса јесте предност окружења Куку наведена у категорији „Аутоматско подношење узорака и АПИ-позиви“ (бр. 9). Ова предност односи се на лакши поступак дефинисања Пајтон-скрипти као АПИ-клијентских софтвера у окружењу Куку.

Табела 4: Упоредна анализа окружења.

	Особина	Куку	Драквуф
1	Комплексност инсталације и подешавања		+
2	Скалабилност		+
3	Извештавање	+	
4	Време извршавања	+	
5	Подржани типови фајлова	+	
6	Превенција избегавања извршења узорака		+
7	Разноликост подржаних верзија машина за анализу, хардвера и хипервизора	+	
8	Интеграција са другим алатима и могућност	+	
9	Аутоматизовано подношење узорака и АПИ	+	
10	Потписи (статичка анализа, извршни фајлови итд.)	+	
11	Визуелизација		+

Предност окружења Драквуф је боља обрада злонамерних софтвера који имплементирају технике избегавања (енгл. *evasion techiques*), због непостојања софтверског агента на виртуелној машини за анализу и због особина које произлазе из саме архитектуре овог окружења, нпр., откривање трагова које злонамерни софтвер покушава да обрише, откривање софтвера који не остављају фајлове на диску рачунара, већ се налазе само у радној меморији) итд. Због тога, окружење Драквуф би било прикладније за узорке који имплементирају технике избегавања.

3.3 Подешавање окружења Куку

За генерисање корпуса проширених обележја, тј. за трансформисање почетног скупа извршних фајлова у корпус извештаја о њиховој динамичкој анализи, коришћена је верзија 2.0.7 окружења Куку. Пошто је ова верзија система заснована на верзији 2 програмског језика Пајтон, инсталирање и подешавање окружења у смислу одабирања адекватних

верзија програмских пакета од којих зависи његово исправно функционисање, захтевали су извесну пажњу.

3.3.1 Избор оперативног система виртуелне машине за анализу

Као виртуелна машина за анализу у којој се извршавају узорци могу се користити оперативни системи Виндоуз „XP“, Убунту, Виндоуз 7 и (са ограниченом функционалношћу) Виндоуз 10.

Изабрана је шездесетчетворобитна виртуелна машина Виндоуз 7, из три разлога: препоручена је у техничкој документацији, друго, због своје застарелости погодна је да се на њој успешно искористе рањивости које нису елиминисане од стране произвођача, и, према Лију и другима (*Li et al.*, 2022) више од 78% злонамерних софтвера креирано је за оперативни систем Виндоуз. Како би се рањивости овог оперативног система задржале, на наведену машину није инсталиран антивирусни програм (нити је омогућено функционисање постојећег), искључено је ажурирање путем интернета и нису инсталирани никакви додаци (енгл. *patches*), осим „*Service Pack 1*“, који је био неопходан да би постојала подршка за различите корисничке програме који су инсталирани за потребе истраживања.

Инсталиран је софтверски пакет „*Microsoft Office 2013*“, чиме подржано учитавање различитих докумената уобичајених у пословној кореспонденцији, присутних у почетном скупу узорака. Такође је инсталирано и неколико уобичајених бесплатних софтверских алата који се често налазе на корисничким радним станицама: за обраду текста и слика, претраживање интернета, пуштање музике и слично.

3.3.2 Подешавање техника против избегавања извршења

У почетној фази тестирања окружења Куку, уочено је да поједини извештаји о понашању узорака садрже информације које указују на технике избегавања откривања у систему за анализу (енгл. *evasion techniques*). Творци злонамерних софтвера теже примени нових техника прикривања, али се до сада уочене технике могу поделити на следећи начин:

1. Детектовање виртуелних окружења (енгл. *VM detection*). Злонамерни софтвер покушава да открије да ли је извршен у виртуелном окружењу, као што су „*Vmware“*, „*VirtualBox“* и други. Уколико детектује виртуелно окружење, не извршава злонамерне активности. Најчешће технике којима се утврђује да ли је окружење у коме се софтвер извршава виртуелно јесу:

- проверавање специфичних регистара и уређаја повезаних са виртуелним машинама;
- препознавање виртуелних графичких и мрежних адаптера;
- детектовање специфичних фајлова и процеса који се односе на виртуелне машине.

2. Одлагање извршења (енгл. *delaying execution*). Извршавање злонамерног софтвера може да се одложи, чиме се избегава његово рано детектовање током анализе. Одлагање може укључивати чекање одређеног временског периода или покретање активности тек након што се систем сматра „стабилним“. У том смислу, могу се користити следеће технике:
    - касно извршавање (нпр. чекање неколико минута или сати пре него што почне штетна активност);
    - манифестација штетних активности само када се испуне специфични услови, као што су одређени временски периоди или одређене радње корисника.

3. Рад на нивоима кернела или преко системских позива (енгл. *syscall*). Злонамерни софтвер може да користи низак ниво комуникације са хардвером и системским ресурсима како би избегавао традиционалне технике детекције које се фокусирају на високе нивое комуникације (нпр. процеси и фајлови). За имплементацију таквих функционалности користи се:
    - рад посредством системских позива који нису обрађени у традиционалним системима за детекцију, што је и у овом истраживању присутно код одређеног броја узорака који не индукују АПИ-позиве при извршавању;
    - манипулација меморијом на ниском нивоу, као што су директни приступи физичким меморијским адресама.

4. Маскирање кода (енгл. *obfuscation*) подразумева скривање злонамерног софтвера кроз различите технике које анализу и разумевање кода чине тежим. Најпознатије технике маскирања јесу:
    - шифровање или замена дела кода,
    - специфично именовање променљивих или функција, фајлова и сл.,
    - замена делова легитимног и злонамерног кода.

5. Скривање у легитимним процесима (енгл. *process hollowing*) јесте техника где злонамерни софтвер користи простор у легитимном процесу, тако да изгледа као да је легитиман, чиме се избегава откривање. Злонамерни код се заправо смешта у меморију постојећег процеса (нпр. *web browser, windows explorer* итд.).

6. Технике усмерене против окружења за анализу (енгл. *anti-analysis techniques*), најчешће подразумевају детектовање тзв. „дибагера“ (нпр. *OllyDbg*). Ако злонамерни софтвер открије присуство „дибагера“ или других особина машина за анализу (попут ограничења у количини меморије, капацитету диска и броју језгара процесора, непостојања активности корисника и слично), може да престане са извршавањем или се изврши на начин који не манифестује злонамерне активности.

7. Успоравање анализе (енгл. *speed-bumping*) изводи се намерним индуковањем кашњења у процесу или појединим фазама извршавања, да би се успорила или отежала анализа, невезано за то да ли је претходно детектовано извршавање у окружењу за анализу.

8. Коришћење постојећих системских апликација (енгл. *living off the land*), као што су „*PowerShell“* или „*WMI“* (*Windows Management Instrumentation*), да би се избегло детектовање. Те апликације су легитимне и често примењиване у пословима администрације система, а њима се могу постићи различите злонамерне активности, које не морају бити имплементиране у сам код злонамерног софтвера.

9. Престанак извршавања када се детектује окружење за анализу (енгл. *termination on detection*).

Како би се умањила могућност да злонамерни софтвер прикрије своје извршавање, уведен је скуп техника против избегавања у окружењу Куку:

- Пајтон-агент који надгледа извршавање узорака подешен је да ради у позадини и именован је као скрипта која ради уобичајени административни посао на рачунару.
- Виртуелна машина за анализу подешена је са историјом коришћења и подацима, прилагођеним корисничким именом и софтвером. Посебна пажња

је посвећена историји претраживача, јер је раније примећена честа провера активности интернет претраживача коју су спроводили неки узорци злонамерног софтвера.

- Виртуелизациони алати и драјвери намерно су изостављени, а ресурси виртуелне машине за анализу подешени су да буду необично велики за окружење за анализу и одговарали су уобичајеном корисничком рачунару.
- Омогућена је техника имитације рада корисника (померање и притискање миша, притискање тастера на тастатури итд.) интегрисана у окружење Куку (алат *pywinauto*).
- Време извршавања анализе повећано је на два минута по узорку, а за симулацију одлагања, тј., продужетка времена, да би се спречило избегавање извршења чекањем, коришћени су уграђени алати и функције.
- Виртуелна машина за анализу имала је ограничен приступ интернету са мрежном баријером подешеном испред лабораторијског окружења, како би се осигурали надзор и контрола над окружењем.

3.4 Подешавање окружења Пајтон

Сви поступци подношења фајлова на анализу, прикупљања извештаја, истраживања, анализе и генерисања корпуса, изведени су применом скрипти написаних у програмском језику Пајтон (верзија 3.10, тј., актуелна верзија у тренутку генерисања корпуса). Пошто су поједини узорци били специфични, као нпр., „*ransomware*“, који генерише велике извештаје због великог броја активности, било је технички изазовно обрађивати их. Да би се обрадиле велики улазни фајлови, осим велике количине меморије, у окружењу Пајтон било је потребно повећати максималну дубину рекурзије за учитавање извештаја добијених од окружења за анализу.

Осим тога, када је корпус из појединих фајлова на диску једном учитан у заједничку логичку целину (представљену структуром података „*pandas dataframe*“), морао се поново чувати на диску у компримованом облику. У супротном би за сва извршавања и све верзије које су настајале у току истраживања било потребно много више ресурса на физичком диску сервера, а учитавање би трајало неприхватљиво дуго и доводило би до непланираних прекида.

Велики извештаји су индуковали знатна времена извршавања, па је осим оптимизовања кода, посебна пажња посвећена чувању међуверзија корпуса. Овде се мисли на фазе када је

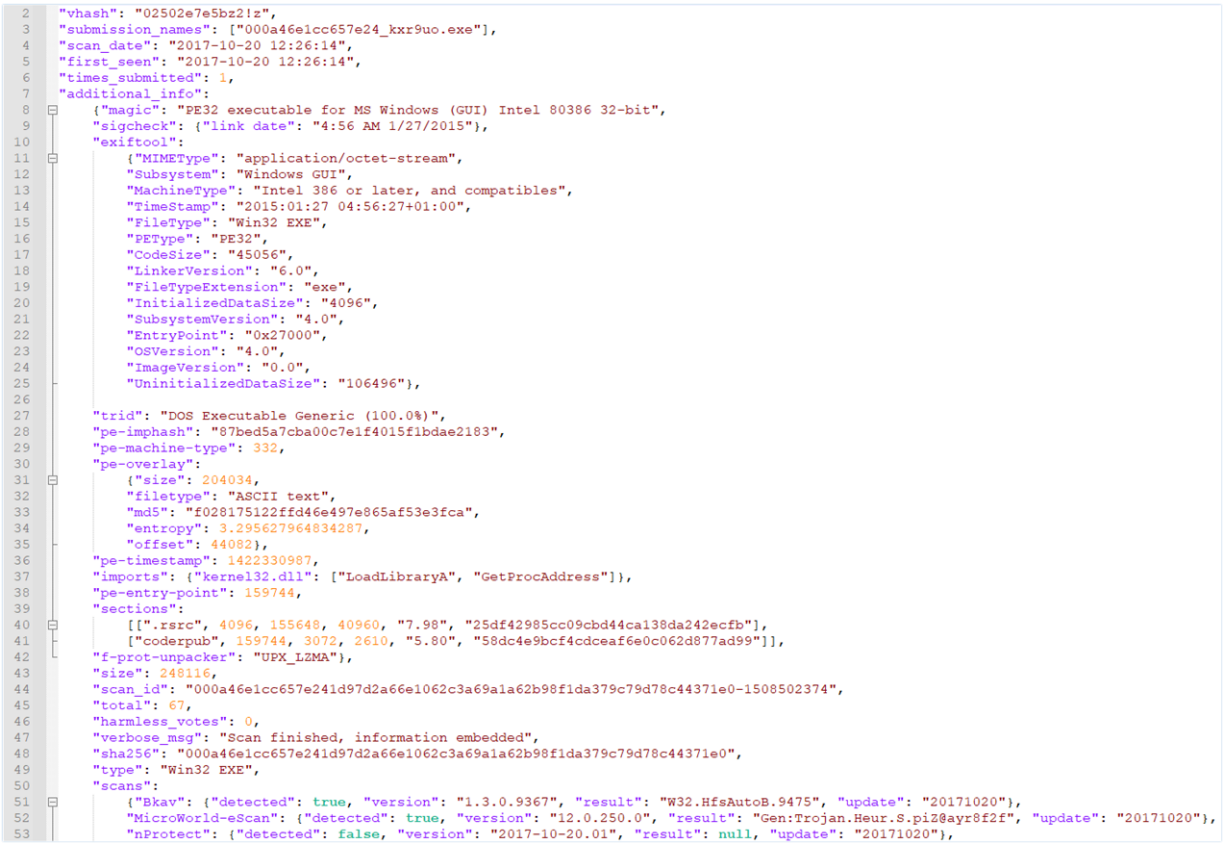
корпус трансформисан применом техника ТФ-ИДФ (енгл. *Term Frequency – Inverse Document Frequency*) и ЦВ (енгл. *Count Vector*), у циљу издвајања обележја, о чему ће више бити речи у наредним секцијама. Показало се да је издвајање обележја на великом корпусу временски захтевно, па су добијене информације чуване у неколико фајлова који су учитавани по потреби.

3.5 Избор узорака и генерисање корпуса

Један од циљева ове дисертације јесте да се произведе нов корпус података за динамичку анализу, који садржи знатан број узорака безопасних и злонамерних софтвера представљених скупом обележја изведених из извештаја динамичке анализе спроведене од стране окружења Куку. Генерисање корпуса и његове особине описани су у наредним секцијама.

3.5.1 Избор и безбедно управљање узорцима злонамерног софтвера

Колекција од приближно 430000 узорака злонамерног софтвера преузета је из базе портала Вирус тотал (*Virus Total*, 2025), који се у професионалној заједници сматра референтним извором. Скуп података је преузет у облику више архива заштићених лозинком, организованих у фолдере по годинама, од 2017. до 2021. Из ове колекције фајлова, без субјективне пристрасности, програмски је одабрано 10465 узорака који су укључени у корпус. Сваки узорак је сачуван у облику фајла без екстензије, са пратећим фајлом који га описује, у Џејсон-текстуалном формату (енгл. *JSON – Java Script Object Notation*), чији део је приказан на слици 3. Такав вид чувања узорака је спроведен да би се спречило да злонамерни фајл компромитује лабораторијско окружење, формирано у сврху генерисања корпуса.



```
"vhash": "02502e7e5bz2!z",
"submission_names": ["000a46e1cc657e24_kxr9uo.exe"],
"scan_date": "2017-10-20 12:26:14",
"first_seen": "2017-10-20 12:26:14",
"times_submitted": 1,
"additional_info":
    {"magic": "PE32 executable for MS Windows (GUI) Intel 80386 32-bit",
    "sigcheck": {"link date": "4:56 AM 1/27/2015"},
    "exiftool":
        {"MIMEType": "application/octet-stream",
        "Subsystem": "Windows GUI",
        "MachineType": "Intel 386 or later, and compatibles",
        "TimeStamp": "2015:01:27 04:56:27+01:00",
        "FileType": "Win32 EXE",
        "PEType": "PE32",
        "CodeSize": "45056",
        "LinkerVersion": "6.0",
        "FileTypeExtension": "exe",
        "InitializedDataSize": "4096",
        "SubsystemVersion": "4.0",
        "EntryPoint": "0x27000",
        "OSVersion": "4.0",
        "ImageVersion": "0.0",
        "UninitializedDataSize": "106496"},

    "trid": "DOS Executable Generic (100.0%)",
    "pe-imphash": "87bed5a7cba00c7e1f4015f1bdae2183",
    "pe-machine-type": 332,
    "pe-overlay":
        {"size": 204034,
        "filetype": "ASCII text",
        "md5": "f028175122ffd46e497e865af53e3fca",
        "entropy": 3.295627964834287,
        "offset": 44082},
    "pe-timestamp": 1422330987,
    "imports": {"kernel32.dll": ["LoadLibraryA", "GetProcAddress"]},
    "pe-entry-point": 159744,
    "sections":
        [[".rsrc", 4096, 155648, 40960, "7.98", "25df42985cc09cbd44ca138da242ecfb"],
        ["coderpub", 159744, 3072, 2610, "5.80", "58dc4e9bcf4cdceaf6e0c062d877ad99"]],
    "f-prot-unpacker": "UPX_LZMA"},
    "size": 248116,
    "scan_id": "000a46e1cc657e241d97d2a66e1062c3a69a1a62b98f1da379c79d78c44371e0-1508502374",
    "total": 67,
    "harmless_votes": 0,
    "verbose_msg": "Scan finished, information embedded",
    "sha256": "000a46e1cc657e241d97d2a66e1062c3a69a1a62b98f1da379c79d78c44371e0",
    "type": "Win32 EXE",
    "scans":
        {"Bkav": {"detected": true, "version": "1.3.0.9367", "result": "W32.HfsAutoB.9475", "update": "20171020"},
        "MicroWorld-eScan": {"detected": true, "version": "12.0.250.0", "result": "Gen:Trojan.Heur.S.piZ@ayr8f2f", "update": "20171020"},
        "nProtect": {"detected": false, "version": "2017-10-20.01", "result": null, "update": "20171020"},
```


Слика 3: Део описног Џејсон-фајла са информацијама о злонамерном узорку, преузет са портала Вирус тотал.

Тек када се узорак подноси на анализу, називу фајла се додаје одговарајућа екстензија. Податак о екстензији изворно се налази у у придруженом описном фајлу. Међутим, подаци о фајловима, укључујући и екстензију, прво су смештени у засебну базу података, из које се читају приликом аутоматског слања узорака на анализу, јер би парсирање описних Џејсон-фајлова у реалном времену успорило овај процес. Приликом слања узорака се релевантни подаци преузимају из базе чиме се узорак. Преглед података смештених у ову базу за потребе анализе дат је у табели 5.

Табела 5: Преглед података издвојених из описних Џејсон-фајлова (преузето од Илића, Кука, Стојановића и Петровића, 2024б).

Назив колоне у бази	Опис
Sha256	Хеш-вредност поднетог фајла.
ScanId	Јединствени идентификатор анализе.
ScanDate	Датум анализе.
FileTypeExtension	Екстензија фајла.
TimesSubmited	Број подношења узорка на анализу.
Positives	Број произвођача антивирусних софтвера који су узорак означили као злонамеран.
Total	Број антивирус произвођача који су учествовали у скенирању.
PermaLink	Хипервеза ка подацима о датом узорку на порталу *virustotal.com*.
Vendor	Произвођачи антивирусних софтвера који су анализирали фајл.

Након подношења фајла на анализу, вршена је поновна промена имена основног узорка у његов изворни облик (тј., без екстензије) ради даљег сигурног чувања. За аутоматизовање овог процеса развијене су Пајтон-скрипте у којима су за подношење узорака, као и за преузимање резултата, коришћени АПИ-позиви, у складу са упутствима из техничке документације окружења Куку. Преузети резултати су чувани у одговарајућем фолдеру на диску радне станице на којој су извршаване скрипте. Као јединствени идентификатор за одређени узорак, коришћена је хеш вредност поднетог фајла (*sha256*).

3.5.2 Избор безопасних фајлова

Осим злонамерних фајлова, за потребе посматраног истраживања, корпус мора да садржи адекватан број фајлова за које је потврђено да су безопасни. Скуп безопасних фајлова добијен је из архиве података о пословној кореспонденцији и инжењерским нацртима и документацији компаније Техника КБ д.о.о. Београд. Ова архива се састоји од различитих врста докумената, рачуна, електронске поште, цртежа, инсталационих фајлова различитог софтвера, као и од системских фајлова насталих инсталацијом Виндоуз 10 оперативног система (*.exe, .dll, .com*, итд.) на рачунару на којем није било претходних корисника ни приступа интернету.

Компанија је била опремљена алатима за заштиту, укључујући мрежну баријеру нове генерације (енгл. *Next Generation Firewall*), реномирани антивирусни софтвер и заштиту

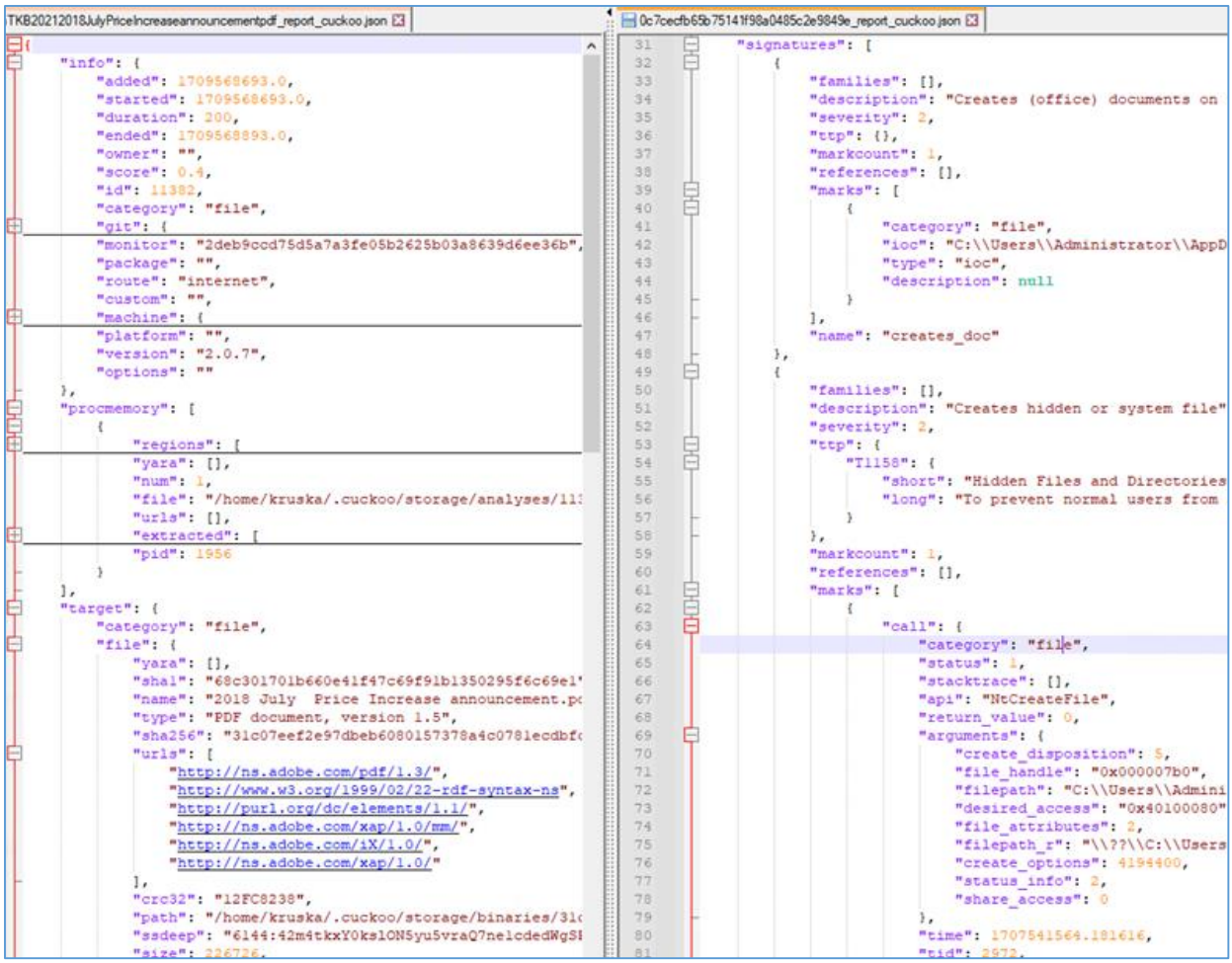
система електронске поште, уз стално праћење нивоа информационе безбедности од стране стручних лица и обезбеђену обуку за подизање нивоа свести запослених о претњама по информациону безбедност. Током неколико година није откривена ниједна компромитујућа активност злонамерног софтвера унутар окружења компаније, па се ова архива са великом поузданошћу може сматрати извором безопасних узорака.

Архива се састоји од приближно 74000 безопасних фајлова насталих у реалном окружењу, од којих је програмски и без субјективне пристрасности одабрано 11735 узорака за укључивање у корпус.

За управљање безопасним фајловима, у смислу преузимања са диска, подношења на анализу и прикупљање и чување резултата, коришћене су сличне Пајтон-скрипте као и за управљање злонамерним фајловима (засноване на позивању АПИ-сервера окружења Куку), са том разликом што су ови фајлови чувани у оригиналном облику (са екстензијом). Као јединствено име фајла, узета је комплетна путања до фајла на диску.

3.5.3 Садржај корпуса

Корпус се састоји од преузетих извештаја које генерише окружење, за поднетих 11735 безопасних и 10465 злонамерних фајлова различитих екстензија. Куку подржава различите формате извештаја, од читљивих извештаја у форми „*HTML*“, до извештаја у специфичној форми намењеној рачунарској размени података „*MAEC Malware Metadata Sharing*“. За генерисање корпуса у овој дисертацији коришћени је Џејсон-формат (енгл. *JSON*) који обједињује све доступне компоненте података. Примери делова извештаја у овом текстуалном формату, за један безопасни и један злонамерни узорак, респективно, дати су на слици 4.



Слика 4: Примери делова Џејсон-извештаја које генерише окружење Куку, насталих подношењем безопасног узорка (лево) и злонамерног узорка (десно).

Иако поједини делови извештаја могу бити изостављени у случајевима кад нису забележене припадајуће активности током извршавања фајла, Џејсон-извештај је, према документацији система Куку (*Cuckoo Sandbox*, 2025) начелно структуриран на следећи начин:

- Део *AnalysisInfo* садржи основне податке о анализи: у којој виртуелној машини је анализа извршена, време трајања анализе, верзију окружења Куку и слично.
- Део *ApkInfo* садржи основне информације о Андроид-анализи у случају да је та анализа спроведена.
- Део *Baseline* садржи сажете резултате из прикупљених информација.
- Део *BehaviorAnalysis* садржи логове понашања фајла, основне трансформације и интерпретације, комплетно праћење процеса и стабло процеса.
- Део *Buffer* садржи резултате анализе бафера насталих покретањем злонамерног софтвера.
- Део *Debug* садржи податке о грешкама и логове везане за сам ток анализе.

- Део *Droidmon* садржи логове из главног модула за анализу спроведену унутар виртуелног окружења.
- Део *Dropped* садржи податке о фајловима насталим током извршавања злонамерног софтвера.
- Део *DumpTls* садржи издвојене криптографске податке и шифре настале комуникацијом злонамерног софтвера.
- Део *GooglePlay* садржи податке добијене комуникацијом злонамерног софтвера са GooglePlay складиштем апликација за Андроид-уређаје.
- Део *IRMA* садржи податке добијене са платформе за анализу инцидената и злонамерног софтвера („*Incident Response and Malware Analysis*“).
- Део *Memory* садржи податке добијене коришћењем алата „*volatillity*“ за анализу меморијских отисака („*memory dump*“).
- Део *MISP* садржи податке добијене са платформе отвореног кода за размену података о злонамерном софтверу („*Malware Information Sharing Platform*“).
- Део *NetworkAnalysis* садржи податке добијене парсирањем фајлова снимљених при мрежном саобраћају који је иницирао злонамерни софтвер, укључујући „*IP*“ адресе, „*DNS*“ саобраћај, „*HTTP*“ и „*HTTPS*“ захтеве и слично.
- Део *ProcMemory* садржи резултате анализе меморијских отисака које су направили процеси, на основу унапред постављених правила.
- Део *ProcMon* садржи податке о издвојеним догађајима које детектује алат *ProcMon*, пакета за надзор рачунара „*SysInternals*“.
- Део *Screenshots* садржи слике екрана приликом извршавања узорка.
- Део *Snort* садржи резултате примене правила „*Snort*“ на податке о мрежном саобраћају.
- Део *StaticAnalysis* садржи резултате основне статичке анализе за препознавање злонамерног софтвера заснованог на техникама распакивања кода, прикривања кода и слично.
- Део *Strings* садржи издвојене низове из бинарног записа кода.
- Део *Suricata* садржи резултате примене правила „*Suricata*“ на податке о мрежном саобраћају.
- Део *TargetInfo* укључује информације о анализираном узорку, његовој хеш-вредности и др.
- Део *VirusTotal* садржи антивирусне потписе анализираног узорка добијене од портала Вирус тотал слањем хеш-вредности фајла.

Осим наведених делова, у Џејсон-извештајима присутан је и део „*Signatures*“, који садржи резултате поклапања појединих делова извештаја са правилима које корисник може дефинисати у формату „*Yara*“, који је намењен за писање правила за безбедносне системе.

3.5.4 Корпус проширених обележја и корпус АПИ-позива

Највећи број истраживања описаних у поглављу 2 заснован је на анализи обележја која се односе на АПИ-позиве. Новоформирани корпус који је описан у овој дисертацији садржи комплетне извештаје из окружења Куку, из којих је накнадно изведен проширени скуп обележја, који по врстама обележја превазилази скуп АПИ-позива (у даљем тексту: корпус проширених обележја).

Раније је наглашено да је један од циљева истраживања представљеног у овој дисертацији да се покаже да је дискриминаторна моћ корпуса проширених обележја већа од дискриминаторне моћи обележја која се односе само на АПИ-позиве. Стога, из корпуса проширених обележја изведен је контролни корпус АПИ-позива. Узорци у овом контролном корпусу садрже само обележја која се односе на АПИ-позиве изведене из узорака у корпусу проширених обележја (в. колоне „корпус АПИ-позива“ у табели 6). Генерисање контролног корпуса извршено је коришћењем посебно подешених Пајтон-скрипти заснованих на регуларним изразима, дефинисаним за издвајање секвенци АПИ-позива. Наведене секвенце се чувају у контролном корпусу који садржи само секвенце АПИ-позива и податке о томе да ли су узорци безопасни или злонамерни (у даљем тексту: корпус АПИ-позива).

Треба приметити да они узорци који током свог извршавања нису индуковали АПИ-позиве, нису укључени у корпус АПИ-позива, што је дискутовано у наредној секцији.

3.6 Расподела узорака

Корпус садржи узорке (тј. извештаје о динамичкој анализи фајлова које генерише окружење Куку) који се односе на 54 типа фајла. За сваки тип фајла, табела 4 даје следеће податке:

- укупан број фајлова који припадају датом типу у корпусу проширених обележја (колона „Укупно узорака - корпус проширених обележја“),
- укупан број фајлова који припадају датом типу у корпусу АПИ-позива (колона „Укупно узорака - корпус АПИ-позива“),

- број безопасних фајлова који припадају датом типу у корпусу проширених обележја (колона „Број безоп. узорака - корпус проширених обележја“),
- број безопасних фајлова који припадају датом типу у корпусу АПИ-позива (колона „Број безоп. узорака - корпус АПИ-позива “),
- број злонамерних фајлова који припадају датом типу у корпусу проширених обележја (колона „Број злон. узорака - корпус проширених обележја“),
- број злонамерних фајлова који припадају датом типу у корпусу АПИ-позива (колона „Број злон. узорака - корпус АПИ-позива“),

при чему су сивом позадином наглашени типови представљени различитим бројевима фајлова у корпусу проширених обележја и корпусу АПИ-позива.

Раније је напоменуто да је из почетног скупа од приближно 430000 злонамерних и 74000 безопасних фајлова, програмски и без субјективне пристрасности одабрано 11735 безопасних и 10465 злонамерних фајлова, који су поднети на анализу у окружење Куку. Тиме је генерисан корпус од 22200 узорака проширених обележја.

Значајан подскуп узорака не индукује АПИ-позиве, а њихова расподела по типу фајла дата је у табели 7. Из табела 6 и 7 може се извести један значајан увид: 23,78% извештаја не укључује АПИ-позиве, односно 39.5% безопасних фајлова и приближно 6.1% злонамерних фајлова не индукује АПИ-позиве. У апсолутним вредностима: 5280 фајлова не индукује АПИ-позиве, од којих је 4638 безопасних и 642 злонамерна.

Уочена расподела узорака који не индукују АПИ-позиве очекивана је за безопасне узорке, тј., очекивано је да значајан удео безопасних узорака не индукује АПИ-позиве, посебно када се ради о документима који су објекти уобичајене пословне кореспонденције. Међутим, пажњу привлачи чињеница да око 6% злонамерних узорака не индукује АПИ-позиве. То се може објаснити намером аутора софтвера да прикрије злонамерне активности током динамичке анализе и остане маскиран ради тежег откривања, и овакви узорци су у том смислу, посебно интересантни. Осим тога, овај увид индикује да је за детектовање злонамерног софтвера значајно посматрати обележја која се не односе на АПИ-позиве.

Табела 6: Расподела типова фајлова у генерисаном корпусу (преузето од Илића и других, 2024а и прилагођено).

Тип фајла	Укупно узорака		Број безоп. узорака		Број злонам. узорака	
	Корпус проширених обележја	Корпус АПИ-позива	Корпус проширених обележја	Корпус АПИ-позива	Корпус проширених обележја	Корпус АПИ-позива
Exe	8492	6941	1395	344	7097	6597
Dll	6812	3591	6574	3354	238	237
doc	2606	2606	1659	1659	947	947
Xls	1108	1106	631	631	477	475
Txt	807	803	4	0	803	803
Fxp	558	558	558	558	0	0
docx	269	269	0	0	269	269
pdf	238	0	118	0	120	0
cdx	223	0	223	0	0	0
Dbf	222	222	222	222	0	0
xlsx	190	190	0	0	190	190
Prg	138	138	138	138	0	0
html	128	128	0	0	128	128
none	60	59	0	0	60	59
ppd	52	52	52	52	0	0
Docm	45	45	0	0	45	45
Fpt	35	35	35	35	0	0
Fpx	33	33	0	0	33	33
Zip	24	0	7	0	17	0
Rar	24	23	0	0	24	23
Htm	19	19	19	19	0	0
Crt	12	12	12	12	0	0
Inf	10	10	10	10	0	0
Bat	9	9	9	9	0	0
dat	8	0	8	0	0	0
Ini	8	8	8	8	0	0
Lnk	7	7	7	7	0	0
Xlsm	5	5	0	0	5	5
Accdb	5	5	5	5	0	0
Cat	5	5	5	5	0	0
Gdl	5	5	5	5	0	0
xml	5	5	5	5	0	0
Db	4	0	4	0	0	0
Pptx	4	4	0	0	4	4
Sch	3	3	3	3	0	0
Ppt	3	3	0	0	3	3
Avi	2	2	2	2	0	0
Bin	2	2	2	2	0	0
bmp	2	0	2	0	0	0
Och	2	2	2	2	0	0
Tbk	2	2	2	2	0	0
Msg	2	2	0	0	2	2
agcoc	1	1	1	1	0	0

Bak	1	1	1	1	0	0
Fmt	1	1	1	1	0	0
Ico	1	0	1	0	0	0
Mem	1	1	1	1	0	0
Ms	1	1	1	1	0	0
New	1	1	1	1	0	0
pr1	1	1	1	1	0	0
Ses	1	1	1	1	0	0
Json	1	1	0	0	1	1
mp3	1	1	0	0	1	1
Rtf	1	1	0	0	1	1
Укупно:	22200	16920	11735	7097	10465	9823

Новоформирани корпус проширених обележја организован је у два фолдера, од којих један садржи извештаје о безопасним фајловима, а други извештаје о злонамерним фајловима. Величина фолдера са злонамерним узорцима износи 935GB, а величина фолдера са безопасним узорцима 34GB. Оба фолдера су организована у потфолдере чији називи одговарају типу фајлова који се у њему налазе (нпр., „*dll*“, „*exe*“, „*doc*“ итд.). Фолдер са узорцима злонамерног софтвера заузима више простора на диску, јер су узорци злонамерног софтвера генерисали више активности у окружењу за динамичку анализу од безопасних узорака, па су резултујући извештаји о њиховој динамичкој анализи већи.

Табела 7: Расподела узорака који не укључују АПИ-позиве у односу на тип фајла.

Тип фајла	Укупан број	Број безопасних узорака	Број злонамерних узорака
Dll	3221	3220	1
Exe	1551	1051	500
Pdf	238	118	120
Cdx	223	223	0
Zip	24	7	17
Dat	8	8	0
Txt	4	4	0
Db	4	4	0
Xls	2	0	2
Bmp	2	2	0
Rar	1	0	1
Ico	1	1	0
Без екстензије	1	0	1
Укупно:	5280	4638	642

Сличан однос између укупних величина злонамерних и безопасних узорака присутан је у корпусу АПИ-позива, у којем злонамерни узорци заузимају 18GB, а безопасни 848MB.

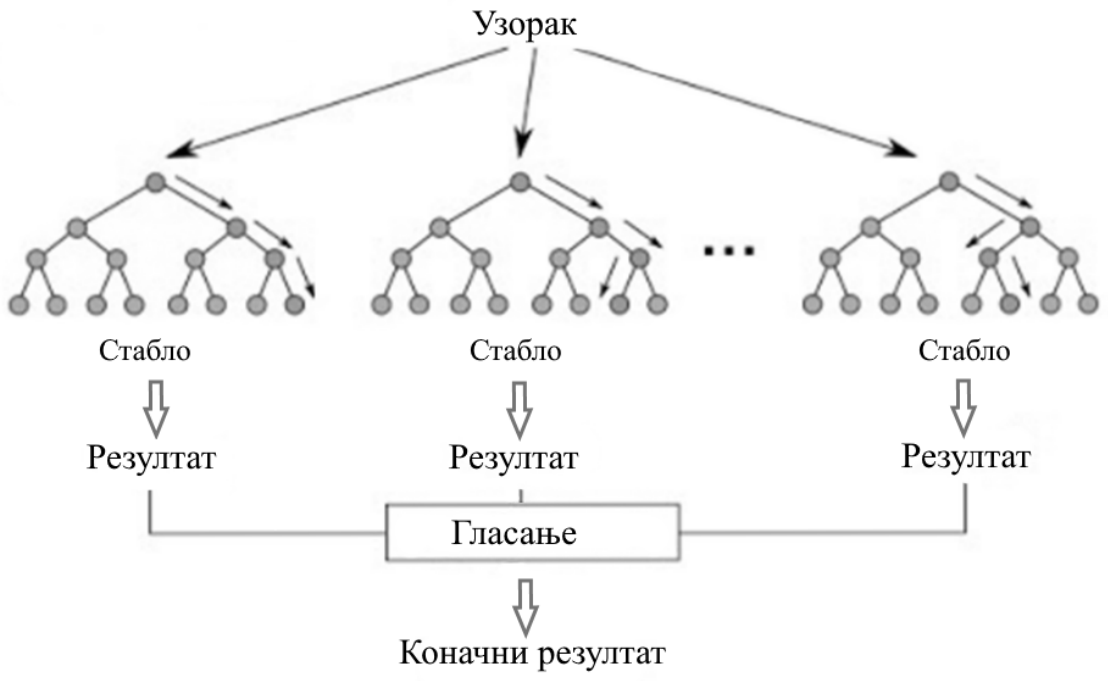
4 Класификовање злонамерног софтвера

4.1 Увод

У знатном броју истраживања наведених у поглављу 2 користи се ансамбл стабала одлучивања (енгл. *Random Forest – RF*). Ансамбл представља скуп различитих стабала одлучивања, која су дизајнирана за исти класификаторни проблем, али су међусобно независна. Заједно чине ансамбл који даје боље резултате од појединачних стабала.

Стабло одлучивања је модел склон прекомерном прилагођавању. Прекомерно прилагођавање (енгл. *overfitting*) јесте проблем који се јавља када се модел превише добро прилагоди узорцима из корпуса за обуку, али не обрађује коректно нове (тј. претходно непознате) узорке. Тада су перформансе модела измерене у фази обуке знатно боље од перформанси измерених у фази тестирања. Да би се превазишао овај проблем, примењује се ансамбл стабала одлучивања. Приликом генерисања ансамбла, користе се два корака за превазилажење проблема прекомерног прилагођавања. Први корак подразумева да се свако стабло тренира на различитом подскупу података, изведеном из почетног скупа података насумичним изабирањем узорака, при чему исти узорак може бити биран више пута („*Bootstrap Sampling*“). Други корак за превазилажење прекомерног прилагођавања односи се на насумичност при избору обележја. Приликом гранања чвора у поступку генерисања стабла, у обзир се не узимају сва доступна обележја, већ само случајно изабрани подскуп обележја (*Ho*, 1995; *Hastie, Tibshirani, Friedman*, 2009; *Cutler, Cutler, Stevens*, 2012).

Ансамбл класификује узорак на основу гласања стабала одлучивања која садржи (в. слику 5). За дати узорак, свако стабло одлучивања генерише засебан резултат класификовања, а коначни резултат одређује се већинским гласањем, тј., узорак се класификује у категорију која добије највише гласова, или се, у случају регресионог проблема, резултат генерише као просечна вредност резултата свих стабала.



Слика 5. Поједностављени приказ ансамбла стабала одлучивања, преузет са веб-сајта Медијум (*Medium*, 2024) и прилагођен.

4.2 Експериментална поставка

У овом истраживању, генерисано је, процењено и накнадно оптимизовано 400 ансамбала стабала одлучивања који се примењују за бинарно класификовање извештаја о динамичкој анализи софтвера, тј., сваки узорак се класификује као безопасан или злонамеран. Ансамбли су имплементирани у програмском језику Пајтон, коришћењем библиотеке „*scikit-learn*“, коју су описали Педрегоса и други (*Pedregosa et al.*, 2011). Израчунавање информационе добити обележја засновано је на информационој ентропији. Информациона добит обележја одражава колика је дискриминаторна моћ датог обележја у датом контексту класификовања и тиме утиче на структуру стабла. Нека је F обележје који може имати вредности $\{f_1, f_2, \ldots, f_n\}$, а C класа која може имати вредности $\{c_1, c_2, \ldots, c_k\}$. Дискриминаторна моћ обележја F у односу на класу C представља информациону добит $\Delta H(F,C)$, која се дефинише на следећи начин (*Hastie, Tibshirani, Friedman*, 2009; *Cutler, Cutler, Stevens*, 2012; Гњатовић, 2025; *Shenon*, 1948).

$$\Delta \mathrm{H}(\mathrm{F}, \mathrm{C}) = H(C) - \sum_{i=1}^{n}(P(F = f_i)H(C|F = f_i)$$

$$= - \sum_{j=1}^{k} P(C = c_j)\, logP\,(C = c_j) \qquad (4.2.1)$$

$$+ \sum_{i=1}^{n}(P(F = f_i) \quad \sum_{j=1}^{k} P(C = c_j \mid F = f_i) \log\, P(C = c_j \mid F = f_i))$$

У експерименту се посматрају следеће независне променљиве:

1. корпус,
2. техника за издвајање обележја,
3. број стабала у ансамблу,
4. максимална дубина стабала,
5. минималан број узорака потребних за гранање чвора,
6. минимални број узорака по листу и
7. број обележја који се разматрају приликом гранања чвора,

које су описане у наредним секцијама. Као зависна променљива посматра се тачност класификовања (уз макропросечну F_1-меру).

4.2.1 Прва независна променљива – корпус

Прва независна променљива односи се на корпус који се користи за обучавање и тестирање ансамбла. Ова независна променљива може да реферише на корпус проширених обележја или корпус АПИ-позива. Оба корпуса су описана у поглављу 3. За потребе развоја ансамбла стабала одлучивања, корпуси су представљени структуром података пандас (енгл. pandas), коју је детаљно описао Мек Кинеи (*McKinnei*, 2010). Ове структуре података смештене су у Пајтон-пикл фајлове (енгл. *pickle files*), у серијализованом бинарном облику, који је описао Росум (*Rossum*, 2020).

Пикл-фајлови се уобичајено користе за чување и накнадну реконструкцију стања програма, укључујући објекте, листе, речнике и друге структуре података, без потребе за поновним извршавањем кода. Овај метод омогућава једноставно чување и учитавање података, али носи одређене безбедносне ризике, јер је, у општем случају, подложан нападу којим се током учитавања података извршава злонамеран код. Овде, пикл-фајлови садрже извештаје о динамичкој анализи који су настали током истраживања, па такав ризик не постоји.

Сваки узорак унутар корпуса састоји се од текста извештаја (колона „*text*“) и ознаке категорије (колона „*label*“), где су вредношћу 0 означени безопасни узорци, а вредношћу 1 злонамерни узорци.

Почетни скуп података, који садржи 22000 извештаја и чија је величина приближно 900GB, био је превелики да стане у расположиву примарну меморију. Због тога што је део корпуса са злонамерним узорцима доминантан по величини, извршено је селектовање узорака, у следећим корацима:

- учитавање корпуса у четири пандас-структуре (три са злонамерним и једна са безопасним узорцима) и размештање узорака на случајан начин (енгл. *data shuffle*),
- подузорковање, тј., насумично бирање 70% узорака из сваке пандас-структуре и
- смештање изабраних узорака у једну пандас-структуру, у којој су узорци опет размештени на случајан начин. Ова структура података је сачувана као пикл-фајл.

Након ове фазе обраде, изабрани део корпуса садржи 15542 насумично одабрана узорка (тј., 8217 безопасних и 7325 злонамерних узорака). Из овог корпуса додатно је изведен корпус АПИ-позива, применом регуларних израза дефинисаним за издвајање секвенци АПИ-позива, на начин описан у поглављу 3.

4.2.2 Друга независна променљива – техника за извођење обележја

Друга независна променљива односи се на технику за извођење динамичких обележја. Ова независна променљива реферише две технике: ЦВ (енгл. *Count Vector, CV*) и ТФ-ИДФ (енгл. *Term Frequency – Inverse Document Frequency, TF-IDF*), које су описали Педрегоса и други (*Pedregosa et al.*, 2011).

ЦВ је техника која дати документ представља вектором нумеричких обележја, чија је дужина једнака величини речника система. Вредност елемента овог вектора дефинише се као број појављивања речи придружене датом елементу у посматраном документу.

ТФ-ИДФ је техника која такође дати документ представља вектором нумеричких обележја, чија је дужина једнака величини речника система. Међутим, вредност елемента овог вектора дефинисана је као производ учесталости речи придружене датом елементу у посматраном документу и инверзне учесталости докумената који садрже дату реч. Учесталост речи у посматраном документу одражава важност речи у датом документу и дефинише се формулом (Гњатовић, 2025):

$$TF(t,d) = \begin{cases} 1 + log_{10}\, n(t,d) & \text{ако је } n(t,d) > 0 \\ 0 & \text{у супротном} \end{cases}, \quad (4.2.2.1)$$

где је *n(t,d)* број појављивања речи *t* у документу *d*.

Инверзна учесталост документа одражава важност дате речи у односу на корпус докумената и дефинише се формулом:

$$IDF(t,D) = log_{10} \frac{|D|}{n(t,D)}, \quad (4.2.2.2)$$

где |*D*| представља број докумената у корпусу *D*, а *n(t,D)* број докумената у корпусу *D* који садрже реч *t*.

Коначна нумеричка вредност која представља дату реч изражена је формулом (Гњатовић, 2025):

$$TF\_IDF = TF(t,d) \cdot IDF(t,D) \quad (4.2.2.3)$$

Након комбиновања прве две независне променљиве (тј., два корпуса са две технике издвајања обележја), изведена су четири нова корпуса података:

- корпус проширених обележја у којем су извештаји представљени ЦВ-векторима обележја,
- корпус проширених обележја у којем су извештаји представљени ТФ-ИДФ-векторима обележја,
- корпус АПИ-позива у којем су извештаји представљени ЦВ-векторима обележја и
- корпус АПИ-позива у којем су извештаји представљени ТФ-ИДФ-векторима.

Ови корпуси служе за обучавање и тестирање модела у истраживању. Сви корпуси су сачувани у пикл-фајловима.

4.2.3 Остале независне променљиве

Остале независне променљиве и њихове вредности јесу:

- број стабала у ансамблу (*n_estimators*) узима вредности из опсега [1, 100].
- максимална дубина стабла (*max_depth*) узима вредности из скупа {*None*, 100, 1000}, где *None* означава да дубина стабла није ограничена,

- минималан број узорака потребних да гранање чвора (*min_samples_split*) узима вредности из скупа {2, 100, 1000},
- минимални број узорака по листу (*min_samples_leaf*) узима вредности из скупа {1, 100, 1000},
- број обележја који се разматрају приликом гранања чвора (*max_features*) узима вредности из скупа{*sqrt*, 0,1, 0,2}, где *sqrt* означава заокружену вредност квадратног корена броја доступних обележја.

Експеримент је спроведен на следећи начин. Комбинујући вредности прве три независне променљиве (корпуса, технике за издвајање обележја и броја стабала) генерисано је 400 (2x2x100) ансамбала одлучивања. Ови ансамбли су додатно оптимизовани комбиновањем вредности преостале четири независне променљиве.

Обука и тестирање модела обављени су на серверској виртуелној машини (*Debian* 12), са ресурсима од 16 виртуелних процесора, приближно 900GB *DDR5* радне меморије и виртуелним диском од 2,5TB.

4.3 Експериментални резултати

Сваки од четири корпуса описаних у секцији 4.2.2 подељен је, на случајан начин, на скуп података за обучавање и скуп за тестирање, у односу 80%:20%. Посматраних 400 модела може се поделити у четири групе, при чему свака група представља по једну комбинацију корпуса и технике за извођење обележја, и садржи по 100 модела (који одговарају различитим бројевима стабала у моделима). Табела 8 даје додатне информације о оптималном моделу из сваке од посматраних група. За сваки од оптималних модела дате су мере квалитета (тачност, прецизност и одзив за сваку класу, макропросечна прецизност, макропросечни одзив и макропросечна F_1-мера) и информације о корпусу (број узорака и њихова расподела по класама).

Табела 8: Експериментални резултати.

	ЦВ проширена обележја	ТФ-ИДФ проширена обележја	ЦВ АПИ-позиви	ТФ-ИДФ АПИ-позиви
Број стабала у оптималном моделу	7	26	32	66
Максимална тачност у фази тестирања (%)	99,74	99,68	95,56	95,4
Прецизност (безопасни)	0,997	0,997	0,935	0,933
Прецизност (злонамерни)	0,998	0,996	0,982	0,982
Макропросечна прецизност	0,997	0,998	0,959	0,957
Одзив (безопасни)	0,998	0,997	0,985	0,985
Одзив (злонамерни)	0,996	0,998	0,921	0,918
Макропросечни одзив	0,997	0,995	0,953	0,952
Макропросечна F1 мера	0,997	0,997	0,954	0,954
Укупан број узорака	15542	15542	15542	15542
Број обележја	25066934	25066934	294	294
Безопасних узорака	8217	8217	4968	4968
Злонамерних узорака	7325	7325	6891	6891

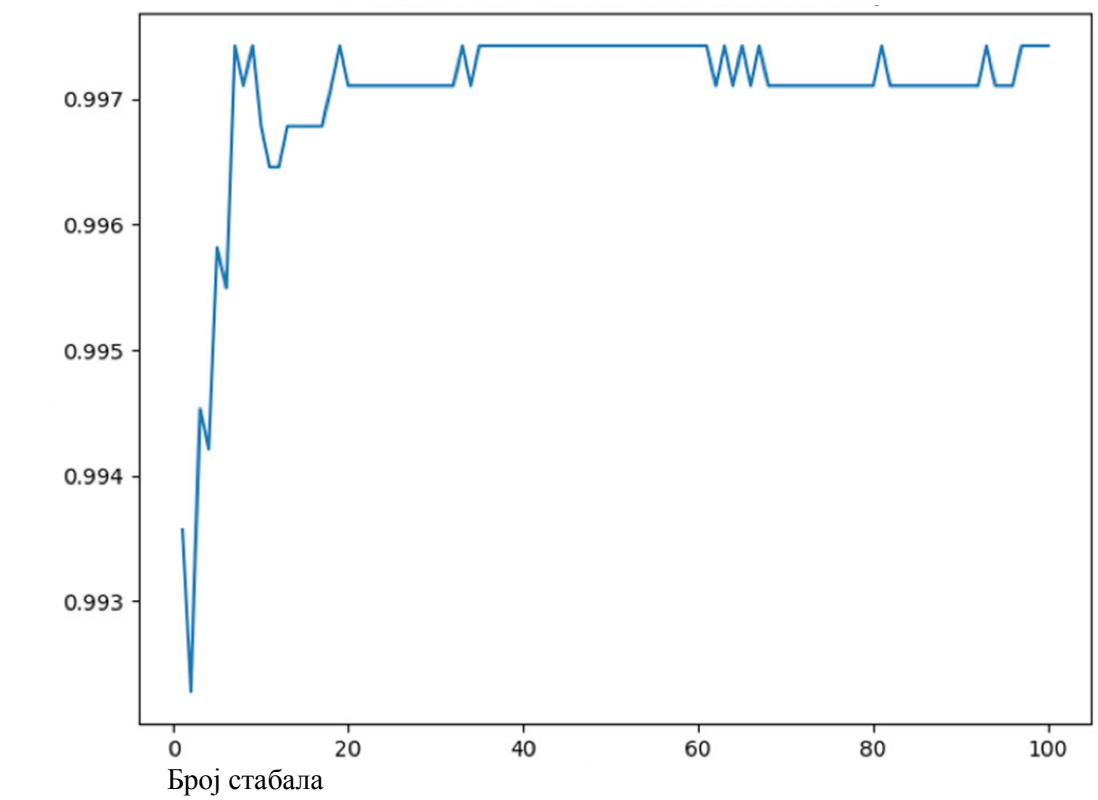
Прва група модела односи се на експерименталне резултате у вези са корпусом проширених обележја представљених ЦВ-векторима. Слика 6 а) даје графички приказ тачности постигнутих у фази тестирања у односу на број стабала одлучивања у моделу. Максимална тачност од 99,74% добијена је за ансамбл са 7 стабала. Матрица конфузије за овај модел дата је на слици 6 б), а изведене мере квалитета дате су у другој колони табеле 8.

Друга група модела односи се на експерименталне резултате у вези са корпусом проширених обележја представљених ТФ-ИДФ-векторима. Графички приказ резултата тестирања дат је на слици 7 а). Максимална тачност од 99,68% добијена је за ансамбл са 26 стабала. Матрица конфузије за овај модел дата је на слици 7 б), а изведене мере квалитета дате су у трећој колони табеле 8.

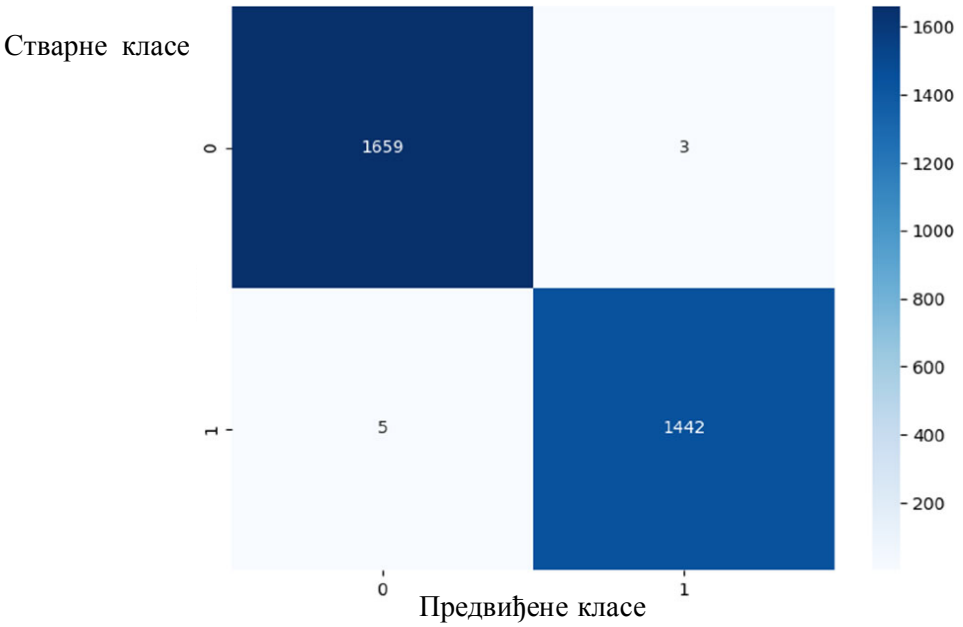
Трећа група модела односи се на експерименталне резултате у вези са корпусом АПИ-позива, представљених ЦВ-векторима обележја. Резултати тестирања су дати на слици 8. Максимална тачност од 95,56% добијена је за ансамбл са 32 стабла (в. четврту колону табеле 8).

Коначно, четврта група модела односи се на експериментални услов који се тиче корпуса АПИ-позива представљених ТФ-ИДФ-векторима обележја. Резултати тестирања су дати на

слици 9. Максимална тачност од 95,4% добијена је за ансамбл са 66 стабала (в. пету колону табеле 8).

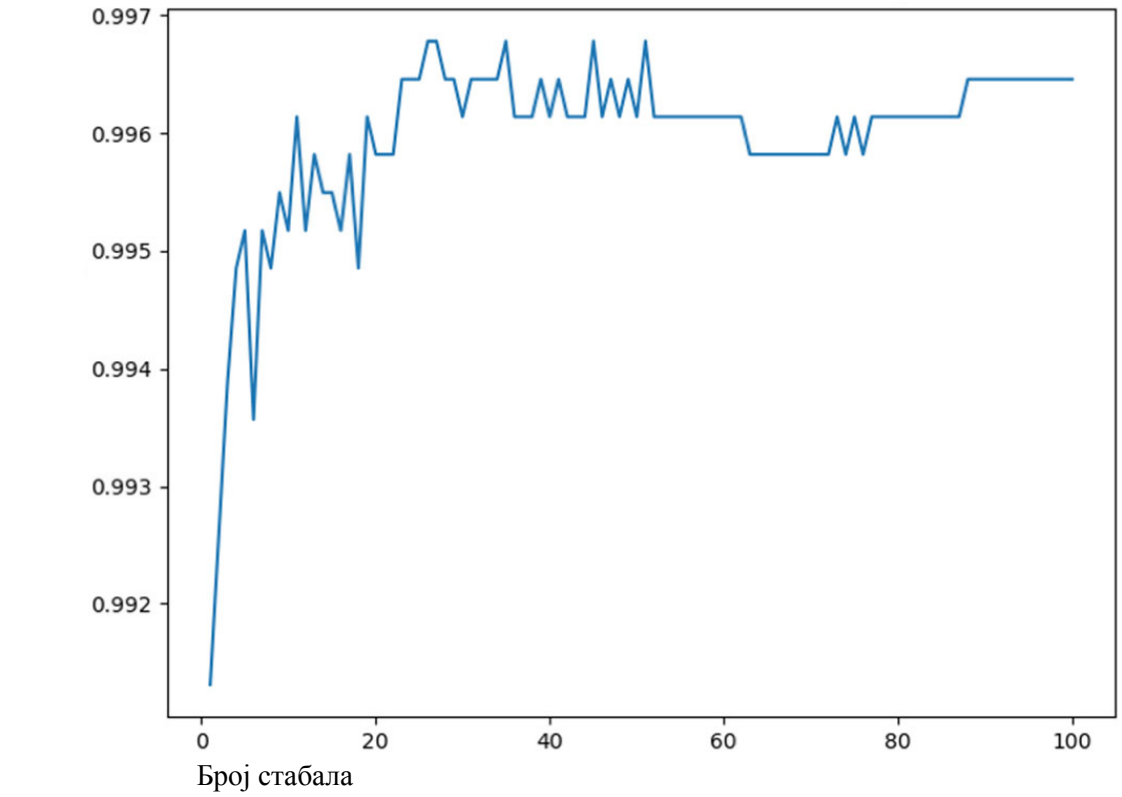


а)

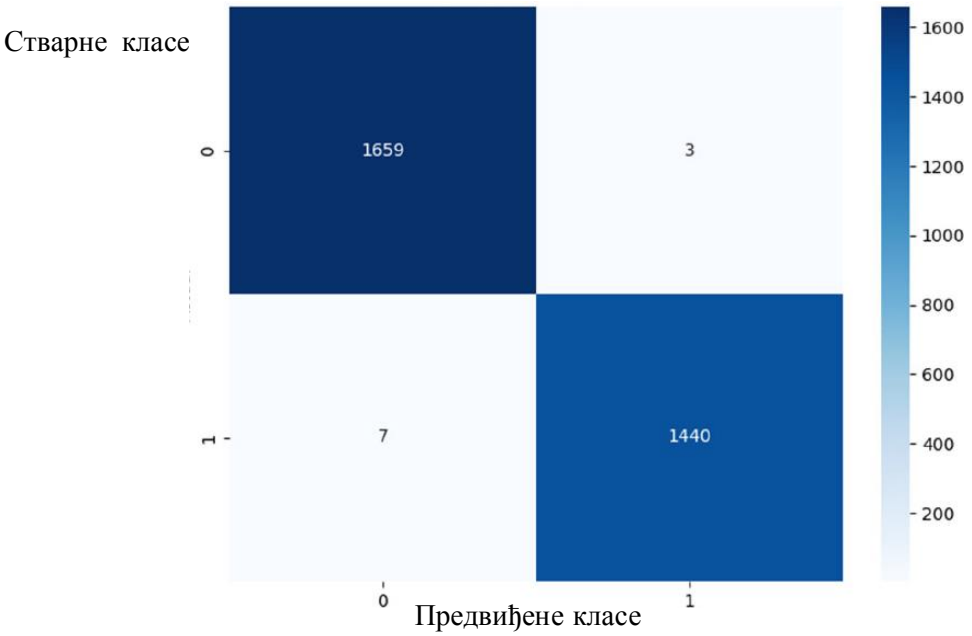


б)

Слика 6: Корпус проширених обележја представљен обележјима издвојеним техником ЦВ: а) графички приказ тачности у фази тестирања, у односу на број стабала одлучивања у ансамблу, б) матрица конфузије добијена за оптимални модел (0 – безопасни, 1 – злонамерни).

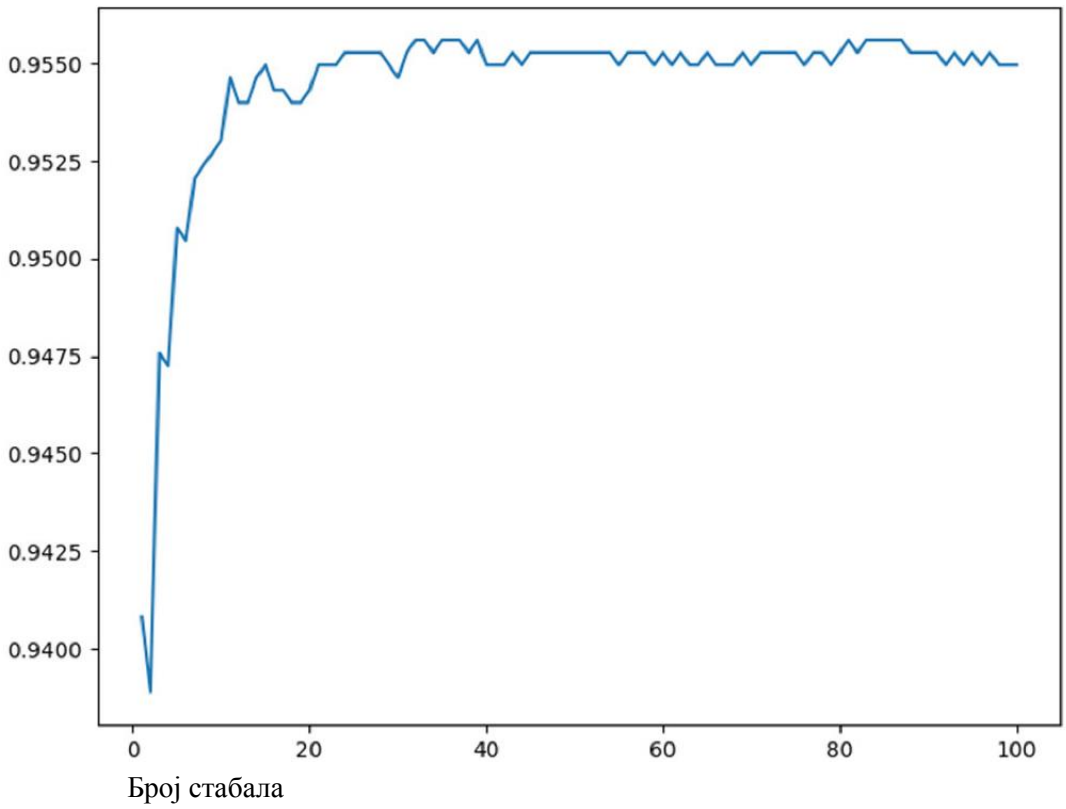


а)

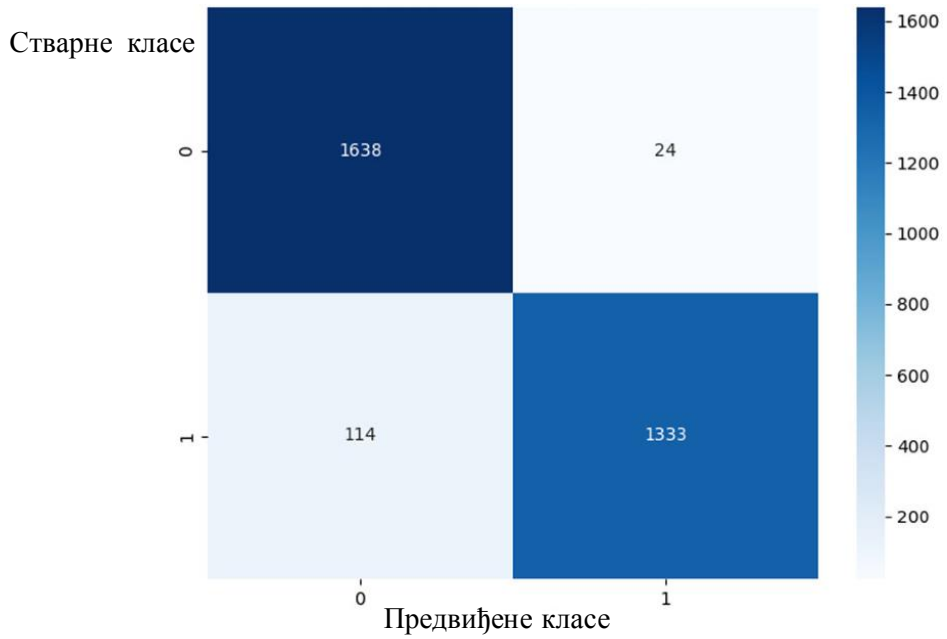


б)

Слика 7: Корпус проширених обележја представљен обележјима издвојеним техником ТФ-ИДФ: а) графички приказ тачности у фази тестирања, у односу на број стабала одлучивања у ансамблу, б) матрица конфузије добијена за оптимални модел (0 – безопасни, 1 – злонамерни).

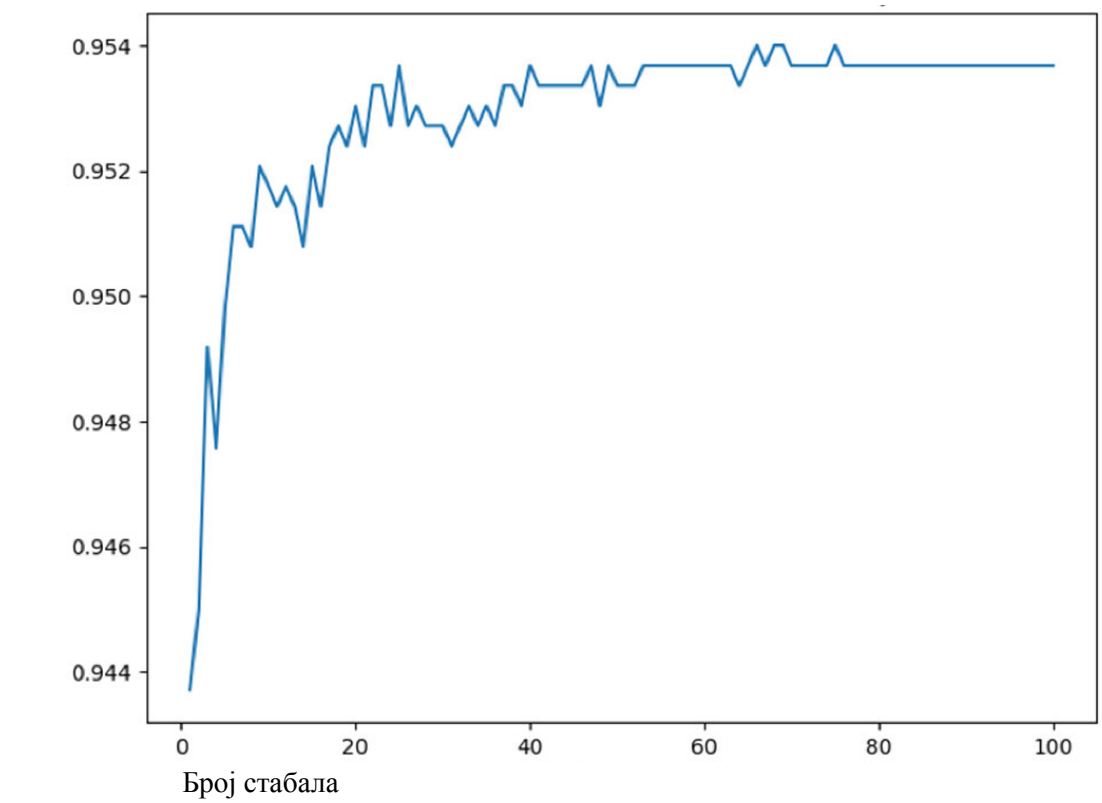


а)

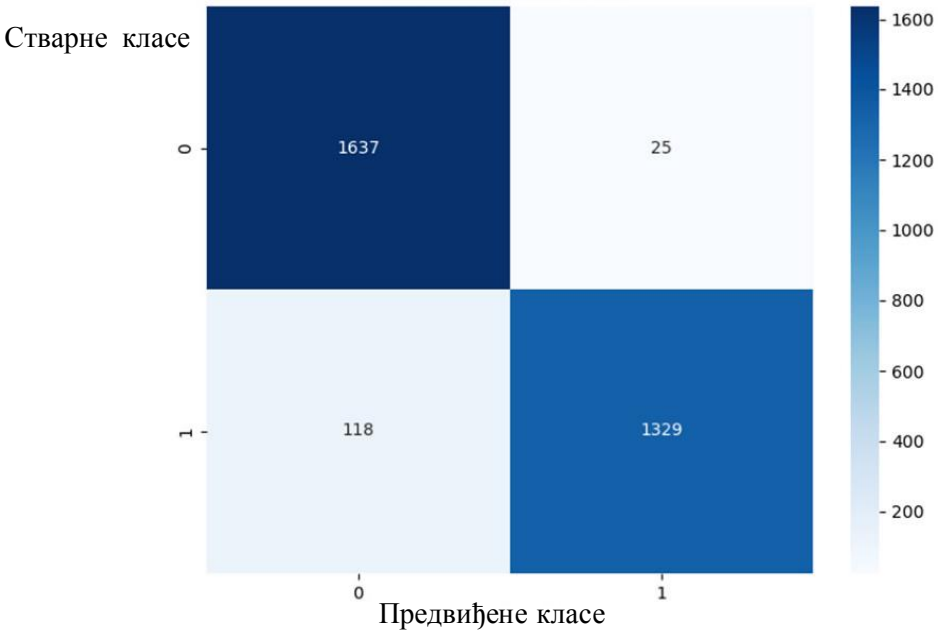


б)

Слика 8: Корпус АПИ-позива представљен обележјима издвојеним техником ЦВ: а) графички приказ тачности у фази тестирања, у односу на број стабала одлучивања у ансамблу, б) матрица конфузије добијена за оптимални модел (0 – безопасни, 1 – злонамерни).



а)



б)

Слика 9: Корпус АПИ-позива представљен обележјима издвојеним техником ТФ-ИДФ: а) графички приказ тачности у фази тестирања, у односу на број стабала одлучивања у ансамблу, б) матрица конфузије добијена за оптимални модел (0 – безопасни, 1 – злонамерни).

4.4 Додатно оптимизовање модела

Ансамбли стабала одлучивања додатно су оптимизовани комбиновањем вредности преостале четири независне променљиве: максимална дубина стабала, минималан број узорака потребних за гранање чвора, минимални број узорака по листу и број обележја који се разматрају приликом гранања чвора.

Резултати процеса оптимизовања приказани су у табели 9, са следећим редоследом колона:

- корпуси на којима су вршени обучавање и тестирање модела,
- вредности четири независне променљиве (*max_depth*, *min_samples_split*, *min_samples_leaf* и *max_features*),
- ангажовани рачунарски ресурси (време извршавања, максимално заузеће меморије и максимално ангажовање процесора) за учитавање података и обучавање и тестирање модела који су процењени као оптимални у својој групи,
- број стабала у оптималном ансамблу и
- тачност остварена приликом тестирања оптималног ансамбла у групи.

Максималне вредности тачности (ћелије обележене плавом бојом у последњој колони) постигнуте су комбинацијом вредности *max_depth*=100 и *max_features*=0,2, али са значајним временом извршавања (више од 70 и 60 сати за корпус проширених обележја са ЦВ-векторима и ТФ-ИДФ-векторима, респективно, в. ћелије обележене светлоцрвеном бојом). Додатна експериментална подешавања, која су дала нешто нижу, али и даље задовољавајућу тачност у фази тестирања, са знатно нижим временима извршавања, обележена су зеленом бојом.

Може се уочити да повећање вредности хиперпараметара *min_samples_split* и *min_samples_leaf* није позитивно утицало на тачност. Међутим, подешавање хиперпараметара *max_depth* на 100 и *max_features* на 0,1 и 0,2 (тј. 10% и 20% обележја, респективно) повећало је тачност, али и време извршавања. Знатно време извршавања наглашено је црвеном бојом, највише вредности тачности остварене у фази тестирања плавом, а прихватљиве вредности тачности са ефикасним извршавањем зеленом.

Табела 9: Резултати оптимизовања ансамбала стабала одлучивања и ангажовани рачунарски ресурси.

		Хиперпараметри				Рачунарски ресурси			Резултати	
	Корпуси	max_depth	min_samples_split	min_samples_leaf	max_features	Време извршавања (h)	Макс. меморије (GB)	Макс. процесора (%)	Бр. стабала при најв. тачности	Највећа тачност тест. (%)
1	Проширена обележја ЦВ	None	2	1	sqrt	7,00	19,3	6	89	99,067
2		100	2	1	sqrt	7,80	19,3	6,7	93	99,099
3		None	100	1	sqrt	6,25	17,8	6,78	25	98,874
4		None	2	100	sqrt	5,90	18	6,47	72	89,836
5		None	2	1	0.1	50,21	18,31	6,47	45	99,742
6		1000	2	1	sqrt	7,01	18,34	6,81	87	99,067
7		None	1000	1	sqrt	5,50	18	6,5	33	98,617
8		None	2	1000	sqrt	7,75	18	6,5	69	76,841
9		100	2	1	0.2	70,84	45	22	7	99,743
10	Проширена обележја ТФ-ИДФ	None	2	1	sqrt	6,11	20	7	86	99,035
11		100	2	1	sqrt	7,50	20	7	86	99,035
12		None	100	1	sqrt	5,34	18,8	7	79	98,617
13		None	2	100	sqrt	5,00	18	6,44	95	92,879
14		None	2	1	0.1	43,33	19	6,38	94	99,614
15		1000	2	1	sqrt	6,00	19,03	6,47	16	98,874
16		None	1000	1	sqrt	5,00	19,99	6,62	51	97,813
17		None	2	1000	sqrt	5,00	18,44	6,7	55	79,607
18		100	2	1	0.2	61,09	20,4	7	26	99,678

		Хиперпараметри				Рачунарски ресурси			Резултати	
	Корпуси	max_depth	min_samples_split	min_samples_leaf	max_features	Време извршавања (h)	Макс. меморије (GB)	Макс. процесора (%)	Бр. стабала при најв. тачности	Највећа тачност тест. (%)
19	АПИ позиви ЦВ	None	2	1	sqrt	0,06	18	6	33	95,529
20		100	2	1	sqrt	0,06	18	6	79	95,529
21		None	100	1	sqrt	0,06	18	6	9	94,854
22		None	2	100	sqrt	0,06	18	6	8	92,536
23		None	2	1	0.1	0,06	18	6	21	95,433
24		1000	2	1	sqrt	0,06	18	6	13	95,497
25		None	1000	1	sqrt	0,06	18	6	38	93,921
26		None	2	1000	sqrt	0,06	18	6	16	78,868
27		100	2	1	0.2	0,09	18	6	32	95,561
28	АПИ позиви ТФ-ИДФ	None	2	1	sqrt	0,07	18	6,5	43	95,24
29		100	2	1	sqrt	0,07	18	6,5	43	95,24
30		None	100	1	sqrt	0,05	18	6,5	22	94,757
31		None	2	100	sqrt	0,03	18	6,5	42	93,149
32		None	2	1	0.1	0,08	18	6,5	92	95,368
33		1000	2	1	sqrt	0,06	18	6,5	43	95,24
34		None	1000	1	sqrt	0,03	18	6,5	93	93,599
35		None	2	1000	sqrt	0,02	18	6,5	14	89,643
36		100	2	1	0.2	0,13	18	6,5	66	95,4

Са повећањем вредности хиперпараметара *max_features*, број стабала са којима је постигнута максимална тачност смањио се на 7 за ЦВ-векторе, односе на 26 за ТФ-ИДФ-векторе. Ово указује на то да се максималан број стабала у ансамблу може смањити на 30, што би значајно смањило време извршавања потребно за развој модела (тј., са приближно 70 сати на 20 сати за корпус са ЦВ-векторима, односно са приближно 61 сат на 18 сати за корпус са ТФ-ИДФ-векторима).

Време обучавања и тестирања модела на корпусима АПИ-позива је краће, али је и постигнута тачност модела нижа. Такви резултати су последица знатно нижег броја обележја разматраних у корпусу АПИ-позива, тј., постоји 294 АПИ-обележја и више од 25 милиона осталих обележја.

Резултати овог експеримента потврђују општу и посебну хипотезу ове дисертације:

- Постоје динамичка обележја софтвера која су релевантна за аутоматско детектовање злонамерних софтвера, а не припадају скупу АПИ-позива.
- Проширени скуп динамичких обележја софтвера унапређује тачност система за аутоматско детектовање злонамерних софтвера у односу на систем који би био заснован само на скупу обележја која се односе на АПИ-позиве.

5 Анализа обележја

5.1 Увод

Резултати експеримента саопштени у претходном поглављу потврђују тачност почетне претпоставке да је дискриминаторна моћ пуних извештаја већа од дискриминаторне моћи АПИ-позива. У овом поглављу, посматра се питање: која обележја највише доприносе овој дискриминаторној моћи?

Да би се добио одговор на ово питање спроведено је додатно истраживање, које се може поделити у следеће фазе:

1. Издвајање обележја са највећом информационом добити (в. секцију 5.2).
2. Анализа издвојених обележја (в. секцију 5.3).
3. Расподела обележја по типовима (в. секцију 5.4).

5.2 Издвајање обележја са највећом информационом добити

За свако од преко 25 милиона (25066933) обележја из корпуса израчуната је информациона добит, а потом је листа обележја уређена као монотоно нерастући низ. За ово је коришћен следећи код (дат је пример за обележја издвојена техником ТФ-ИДФ, а исти принцип примењен је и за случај кад су обележја издвојена техником ЦВ):

```
feature_names_tfidf = tfidf_vectorizer.get_feature_names_out()
feature_importances_tfidf = rf_classifier_tfidf.feature_importances_
sorted_indices_tfidf = np.argsort(feature_importances)[::-1]
```

Наведени код израчунава, између осталог, информациону добит обележја изведених применом технике ТФ-ИДФ над текстуалним подацима из корпуса. Функција *get_feature_names_out*() издваја сва обележја из тежинске инвертоване матрице карактеристичне за технику ТФ-ИДФ. У атрибут *feature_importances_* смештена је секвенца вредности информационе добити за свако појединачно обележје. На крају, применом функције *np.argsort(),* обележја се уређују по важности, од најважнијих ка мање важнима, чиме се омогућава увид у њихов утицај на класификовање. Аналоган код је употребљен и за обележја изведена техником ЦВ, чиме су добијене две листе обележја поређаних по важности.

Интересантно је да вредности информационе важности обележја издвојених техникама ТФ-ИДФ припадају опсегу (0, 0,00636), при чему првих 32134 обележја имају ненулте вредности. Слично, вредности информационе добити за обележја издвојена техником ЦВ припадају опсегу (0, 0,00585), при чему првих 37285 обележја имају ненулте вредности.

Ако узмемо у обзир да је број обележја релативно велики (око 25 милиона) и да су вредности информационе добити релативно мале (за пример, в. табеле 10 и 11), нулте вредности информационих добити могу се објаснити ефектом аритметичког поткорачења, који је инхерентно присутан у организацији рачунара (Гњатовић, 2023). Због ограниченог броја битова доступних за представљање реалних бројева, вредности које су блиске нули аутоматски се заокружују на нулу. Овде је ефекат аритметичког поткорачења искоришћен као први критеријум за одстрањивање оних обележја чија је информациона добит релативно мала.

За преостала обележја, којима су придружене ненулте вредности информационе добити, било је потребно одредити вредности прага који би раздвојио најважнија обележја од осталих. Иако постоје различите методе за одређивање прага, овде је изабрана техника која се изворно примењује у области обради дигиталних слика, а односи се на аутоматско одређивање глобалног прага, које је описано у књигама: Ши (*Shih*, 2010, в. поглавље „*Basic Global Thresholding*“, стр. 120-121), Гонзалес и Вудс (*Gonzalez, Woods*, 2018, в. секцију „*5.1 Thresholding*“, стр. 746-747) и Гњатовић (2024, в. секцију 5.6.1, стр. 101-103). Наведена метода је коришћена у радовима Гњатовић и други (2018) и (2022), уз претходно прилагођење, које је на сличан начин изведено и у овој дисертацији.

Основна идеја ове методе може се описати у неколико корака:

I. Почетна вредност прага израчунава се као аритметичка средина информационих добити обележја:

$$\mathrm{A}_0 = \frac{1}{n} \sum_{i=0}^{n} a_i \qquad (5.2.1)$$

Ова вредност постаје текућа вредност прага.

II. За текућу вредност прага A_i, скуп обележја се дели на два подскупа. Први подскуп M_1 садржи обележја чија је информациона вредност већа од текуће вредности прага, а други подксуп скуп M_2 обележја чија је информациона вредност мања од текуће вредности прага. Следећа вредност прага израчунава се по формули:

$$\mathrm{A}_{i+1} = \frac{1}{2}\left(\frac{1}{|\mathrm{M}_1|}\sum_{i=1}^{|\mathrm{M}_1|} a_i + \frac{1}{|\mathrm{M}_2|}\sum_{i=1}^{|\mathrm{M}_2|} a_i\right) \quad (5.2.2)$$

III. Ако промена у вредности прага није значајна, тј., ако је апсолутна вредност разлике два сукцесивна прага није већа од предефинисане вредности *K*:

$$|\mathrm{A_i} - \mathrm{A_{i+1}}| \leq \mathrm{K} \quad (5.2.3)$$

алгоритам се завршава, а коначна вредност прага једнака је а A_{i+1}. У супротном, прелази се на корак (II).

Овај алгоритам примењује се на две листе обележја, изведених, респективно, применом техника TF-IDF и ЦВ, из којих су уклоњена сва обележја којима су додељене нулте вредности. За сваку листу засебно се израчунава вредност параметра *K* у трећем кораку алгоритма, као најмања вредност разлике две суседне вредности у листи. Тј., за уређену листу вредности

$$a_1, a_2, a_3, \ldots, a_n, \quad (5.2.4)$$

вредност параметра *K* израчунава се на следећи начин:

$$\mathrm{K} = \min_{(1 \leq i < n) \wedge (a_i \neq a_{i+1})}\{a_i - a_{i+1}\} \quad (5.2.5)$$

Коначно, скуп релевантних обележја из дате листе обележја укључује само она обележја чија је информациона добит већа од израчунате вредности глобалног прага.

На тај начин издвојено је 175 најважнијих обележја изведених применом технике ЦВ, и 195 обележја изведених применом технике ТФ-ИДФ. Део издвојених обележја приказан је у табелама 10 и 11, а потпуни скупови најважнијих обележја дати су у секцији 5.5.

Табела 10: Двадесет најважнијих обележја издвојених применом технике ЦВ.

Р. Бр.	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
1	0x7fefd650000	0,005858	1016	0	1016
2	0x7fefc770000	0,005592	2070	1644	426
3	0x7fefd2c0000	0,005132	1016	0	1016
4	0x7fefce20000	0,004864	1016	0	1016
5	0x7fefce50000	0,00481	2070	1644	426
6	u00b2e	0,004423	663	107	556
7	sha512	0,003817	3108	1645	1463
8	u0002	0,003772	1537	456	1081
9	0x7fefaad0000	0,003426	2070	1644	426
10	96	0,003301	3108	1645	1463
11	0x7fefe270000	0,003253	2070	1644	426
12	u00c7	0,003146	1200	187	1013
13	0x7fefd0a0000	0,003009	2070	1644	426
14	u00f6e	0,002964	489	3	486
15	Ttp	0,00294	3108	1645	1463
16	0x75730000	0,002935	1023	3	1020
17	0x7fefc540000	0,002884	2070	1644	426
18	Logs	0,002884	823	23	800
19	l2nocm9tzv9lehrlbnnpb24vymxvynmvnzi0qufxnv9zt2rvduwymeresezgvmjnqq	0,002741	0	0	0
20	0x7fefd590000	0,002728	2070	1644	426

Табела 11: Двадесет најважнијих обележја издвојених применом технике ТФ-ИДФ.

Р. Бр.	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
1	0x7fefd590000	0,006361	2070	1644	426
2	0x7fefce50000	0,006325	2070	1644	426
3	0x7fefc770000	0,005822	2070	1644	426
4	0x7fefc9a0000	0,004682	2070	1644	426
5	0x7fefc4a0000	0,00411	2070	1644	426
6	legitimate	0,003855	2231	1353	878
7	0x7fefcba0000	0,003583	1016	0	1016
8	0x7fefcfc0000	0,003498	2070	1644	426
9	0x7fefce20000	0,003469	1016	0	1016
10	u00c7	0,003347	1200	187	1013
11	u008ck	0,003119	545	2	543
12	Msctf	0,003067	725	301	424
13	0x73010000	0,002969	1748	1433	315
14	u00f6e	0,002962	489	3	486
15	0x7fefece0000	0,002959	1016	0	1016
16	0x7fefd550000	0,002913	1016	0	1016
17	u0002	0,002906	1537	456	1081
18	0x76eb0000	0,002898	2070	1644	426
19	Mswsock	0,00284	3086	1644	1442
20	897,024	0,002812	3086	1644	1442

У наставку су ова обележја детаљније анализирана.

5.3 Анализа издвојених обележја

Да би се издвојена обележја интерпретирала, извршена је следећа анализа:

1. Из корпуса проширених обележја преузети су сви извештаји који садрже издвојена обележја.
2. За свако издвојено обележје у датом извештају, евидентирана је путања (тј., низ чворова) од корена извештаја до чвора извештаја који представља секцију у којој се дато обележје налази. Тиме је формирана листа свих подсекција са бројем појављивања обележја у њима.
3. Подсекције истих или сличних значења груписане су у надсекције.

4. Формирана је листа свих надсекција, са укупним бројем појављивања издвојених обележја у свакој надсекцији.

Претрага корпуса проширених обележја извршена је тако да је за свако обележје издвојен скуп извештаја који садрже дато обележје. Подскуп извештаја за једно партикуларно обележје (0x7fefd590000) приказан је на слици 10. На овој слици, осим података у каквим извештајима се обележје налази (тј., злонамерним или безопасним), може се видети да претрага враћа и путање у стаблу извештаја, из којих се изводи податак о подсекцији, као и број појављивања сваког обележја у том контексту, тј. на истој путањи.

```
Rec za koju radimo: 0x7fefd590000
Broj uzoraka u kojima se rec nalazi:  2070
Broj benignih:  1644
Broj malicioznih:  426
['behavior', 'processes', 'modules', 'baseaddr']
1
['post\\', 'behavior', 'processes', 'modules', 'baseaddr']
708
['behavior', 'processes', 'modules', 'baseaddr']
708
['\\', 'behavior', 'processes', 'modules', 'baseaddr']
941
['behavior', 'processes', 'modules', 'baseaddr']
941
['messages\\', 'behavior', 'processes', 'modules', 'baseaddr']
1166
['behavior', 'processes', 'modules', 'baseaddr']
1166
['vi_js_posttypes\\', 'behavior', 'processes', 'modules', 'baseaddr']
1716
['behavior', 'processes', 'modules', 'baseaddr']
1716
['strings\\', 'behavior', 'processes', 'modules', 'baseaddr']
1777
['behavior', 'processes', 'modules', 'baseaddr']
1777
```

Слика 10: Приказ резултата претраге корпуса проширених обележја по датом издвојеном обележју.

На тај начин добијени су подаци у форми путање од директног потомка корена извештаја до чвора непосредно надређеног обележју. Нпр., путања „*behavior processes calls arguments buffer*“ означава да се унутар секције *behavior* налази секција *processes*, унутар које се налази секција *calls* итд., док се не дође до подсекције *buffer*, у којој се налази дато обележје. Неколико често заступљених путања и начин груписања подсекција у надсекције описани су у наставку.

5.3.1 Пример груписања у категорију која не представља АПИ-позиве

Како би се стекао утисак која врста секција је најинформативнија у датом контексту, било је потребно груписати секције истих или сличних значења у надсекције и сабрати сва појављивања обележја за дату надсекцију. Наведено груписање је урађено тумачењем значења појединих назива чворова и њиховог међусобног односа на начин који ће бити објашњен за неколико случајева приказаних на слици 11.

На датој слици, у првој врсти је приказан низ чворова "*strings ttp T1059 long*", смештен у мрежну категорију (*Network*). Да бисмо разумели зашто је то урађено, направићемо кратку анализу путање:

1. *Strings* – представља низове текстуалних података који су откривени током анализе фајла у сендбокс-окружењу. Може садржати текстуалне вредности које су идентификоване као део понашања злонамерног фајла, као што су путање до фајлова, домени, ИП-адресе и други текстуални идентификатори.
2. *Ttp T1059* (*Tactics, Techniques, and Procedures*) – Ова вредност се односи на оквир Митре (енгл. *MITRE ATT&CK*, в. *Mitre*, 2025) и вредност кључа *T1059*. *MITRE ATT&CK* је емпиријски заснована база знања о тактикама и техникама напада. Кључ *T1059* указује на технику *Command and Scripting Interpreter*, која се често користи за задавање команди на систему жртве, посредством софтвера *PowerShell*, *Command Prompt* и других алата за извршавање скрипти и команди.
3. *Long* – Ова вредност се може интерпретирати на различите, контекстуално зависне начине, али обично подразумева бројчане вредности широких опсега или временске параметре, попут времена трајања напада или времена повезивања са командним сервером. У неким случајевима, може се односити и на количину података или капацитет параметра.

strings ttp T1059 long	Network	129,116	
strings ttp T1071 long	Network	204,632	
vi_js_posttypes network dns answers	Network	155,477	
vi_js_posttypes network dns request	Network	52,393	
vi_js_posttypes network tcp dst	Network	30,839	
vi_js_posttypes target file sha256	Network	131,939	
vi_js_posttypes ttp T1059 long	Network	129,085	
vi_js_posttypes ttp T1129 long	Network	156,725	568,619,664
behavior processes modules basename	Process	429,046	
conflictingProcess3 arguments module	Process	255,595	
conflictingProcess3 modules basename	Process	185,193	
messages behavior processes modules baseaddr	Process	1,246,982	
messages behavior processes modules basename	Process	430,420	
signatures marks martian_process	Process	486,251	
strings behavior processes modules basename	Process	434,032	
vi_js_posttypes behavior processes modules baseaddr	Process	1,256,604	
vi_js_posttypes behavior processes modules basename	Process	433,672	5,157,795
extracted raw	RAW	2,102,402	
icons arguments buffer	RAW	2,126,963	
icons arguments value	RAW	994,558	
images	RAW	222,407	
info machine	RAW	82,678	
info machine label	RAW	57,109	
info machine name	RAW	108,607	
info package	RAW	53,752	
software_reporter arguments buffer	RAW	2,394,424	
software_reporter arguments value	RAW	165,290	
software_reporter debug dbgview	RAW	293,214	
software_reporter debug errors	RAW	279,384	9,162,344
signatures marks process	Registry	398,674	
signatures marks reg_key	Registry	950,238	
signatures marks reg_value	Registry	19,802,738	21,151,650
buffer yara meta description	Signatures	34,008,932	
buffer yara name	Signatures	14,657,314	
buffer yara strings	Signatures	10,614,370	
endpoints signatures description	Signatures	1,503,196	
endpoints signatures marks category	Signatures	246,766	

Слика 11: Груписање подсекција (колона 1) у надсекције (колона 2) и сабирање бројева појављивања обележја у свим надсекцијама истог типа (колона 4). На слици је приказан само један део генерисане листе секција.

Сврха наведеног напада, категорисаног као *T1059*, јесте удаљена контрола система, а команде које се шаљу у модификовани интерпретатор команди могу се детектовати у мрежном саобраћају, па је дата путања смештена у надсекцију која се односи на мрежу – *Network*.

Још упечатљивији пример приказан је у другој врсти на слици 11, у којој се кључ *T1071* односи на коришћење познатих протокола, као што су *HTTP*, *HTTPS*, *DNS*, *SMB* итд., за комуникацију са командним и контролним сервером. Зато је и ова путања сврстана у надсекцију *Network*.

5.3.2 Пример груписања у категорију која представља АПИ-позиве

Добар пример за груписање у категорију АПИ-позива је низ родитељских чворова „*behavior processes calls arguments buffer*“. Ако рашчланимо ову секвенцу добићемо следеће:

1. *Behavior* - садржи детаљне информације о понашању анализираног узорка. Овде се описују активности као што су покретање процеса, АПИ-позиви, промена регистра, мрежна активност и други значајни догађаји који се дешавају током извршавања узорка. Секција *behavior* повезује све активности које су уочене током анализе.
2. *Processes* – садржи описе свих процеса који су покренути током извршавања узорка. Ова секција укључује информације као што су *ID* процеса (*PID*), име извршног фајла и други важни подаци о процесима. За сваки процес овде су наведене и информације о родитељском процесу, времену када је процес покренут и другим релевантним детаљима, што се може видети на слици 12.

Слика 12: Приказ подсекције *Processes* у генерисаном извештају.

3. *Calls* – садржи позиве које је процес иницирао током свог извршавања. Ови позиви представљају функције позване из оперативног система или других библиотека, које омогућавају интеракцију између програма и система (нпр., читање података из фајлова, упис података у фајлове, приступ примарној меморији, креирање нових процеса). У овој секцији наводе се аргументи који су прослеђени у појединачним

позивима, а понекад и име функције која је позвана. У зависности од имена функције, оваква путања може индиковати АПИ-позив.

4. *Arguments* – садржи податке о аргументима који су прослеђени позиву. Аргументи могу представљати путање до фајлова, кључеве регистра или друге параметре који су потребни за извршавање одређене функције унутар оперативног система.

5. *Buffer* – односи се на податке који су коришћени у позиву. Ови подаци могу представљати садржај фајлова, вредности у регистрима или друге информације које процес размењује са системом током свог извршавања. Подаци у овој секцији представљени су обично као низови вредности (нпр., хексадецималних) података у примарној меморији или фајловима.

5.3.3 Проблеми у препознавању обележја

У аутоматском класификовању злонамерних и безопасних програма доминантно се разматрају АПИ-позиви (в. поглавље 2). У овој дисертацији (в. поглавље 3) показано је да проширени скуп обележја који превазилази АПИ-позиве има већу дискриминаторну моћ. Због тога, потребно је додатно интерпретирати обележја ван скупа АПИ-позива која показују значајну дискриминаторну моћ. Због великог броја обележја, примењена је аутоматска обрада извештаја о динамичкој анализи софтвера.

Извештаји о динамичкој анализи софтвера представљају прикладан извор обележја која превазилазе скуп АПИ-позива. Међутим, аутоматска анализа ових извештаја у циљу интерпретирања изведених обележја повезана је са одређеним изазовима, који се могу груписати на следећи начин:

1. Неисправно структурирани извештаји – Структура извештаја генерисаних од стране сендбокс-система не одговара увек стандардној структури Џејсон-фајлова.
2. Недоследно означавање обележја – У извештајима је присутан релативно велики број различитих појавних облика истих обележја, на различитим путањама.
3. Некомплетни извештаји – Подаци у извештају који описују обележја нису увек довољни за њихово потпуно интерпретирање. На пример, за дато обележје постоји податак о путањи, из којег се може закључити да се обележје односи на одређени процес и позивања одређеног модула. Међутим, осим путање, обележје је описано само меморијском адресом, па се не може закључити да ли се ради о учитавању фајла, учитавању регистарског кључа, АПИ-позиву итд. Овај проблем посебно долази до

изражаја приликом одређивања да ли обележје представља АПИ-позив, нпр., када у секцији *calls* не постоје специфични подаци о извршавању АПИ-позива на оперативном систему Виндоуз 7.

4. Појављивање истог обележја у различитим подсекцијама – Нпр., дато обележје може бити присутно у секцијама *signatures*, *network* и *files*. Један пример оваквог обележја односи се на злонамерну скрипту у облику фајла (због чега је присутна у секцији *files*), која се транспортује мрежним путем (због чега је присутна у секцији *network*), а препозната је на нивоу потписа, којима су дефинисани обрасци понашања софтвера (због чега је присутна у секцији *signatures*).

Основни циљ ове анализе јесте да се покаже да постоји знатан удео обележја која нису АПИ-позиви, а показују значајну дискриминаторну моћ. Због наведених проблема, за одређена обележја се посредством аутоматске анализа извештаја не може поуздано потврдити да припадају скупу АПИ-позива, нити поуздано искључити да припадају овом скупу. Да би се избегла пристрасност у закључивању у односу на важност обележја која нису АПИ-позиви, сва обележја за која се није могло потврдити нити искључити да припадају АПИ-позивима класификована су као АПИ-позиви.

У табели 12 приказане су подсекције које су категорисане као АПИ-позиви. За сваку секцију наведене су најчешће заступљене путање у корпусу.

Табела 12: Подсекције које су категорисане као АПИ-позиви.

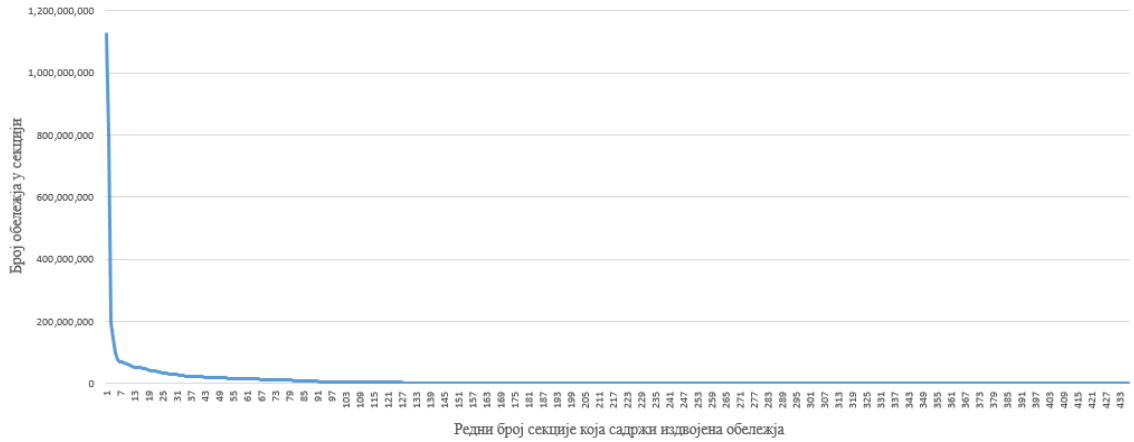
Најчешће путање до подсекције	Подсекција која садржи обележје
behavior processes calls arguments calls arguments buffer calls arguments endpoints behavior processes calls arguments endpoints signatures marks call arguments messages behavior processes calls arguments signatures marks call arguments signatures marks call category strings behavior processes calls arguments vi_js_posttypes behavior processes calls arguments was calls arguments	9Kbuffer
	access
	access
	arguments
	base_address
	basename
	buffer
	callback_function
	callsflagsiid
	category
	clsid
	cmd
	command_line
	context_handle
	crypto_handle
	desired_access
	endpointsarguments
	exceptioninstruction
	exceptioninstruction_r
	exceptionmodule
	exceptionsymbol
	file_handle
	filepath
	filepath_r
	function_address
	function_name
	handle
	headers
	iid
	input_buffer
	key
	key_handle
	key_name
	message
	module
	module_address

Најчешће путање до подсекције	Подсекција која садржи обележје
behavior processes calls arguments calls arguments buffer calls arguments endpoints behavior processes calls arguments endpoints signatures marks call arguments messages behavior processes calls arguments signatures marks call arguments signatures marks call category strings behavior processes calls arguments vi_js_posttypes behavior processes calls arguments was calls arguments	module_handle
	module_name
	mutant_name
	newfilepath_r
	object_handle
	oldfilepath
	output_buffer
	parameters
	path
	pointer
	post_data
	process_handle
	process_name
	provider_handle
	regkey
	regkey_r
	resource_handle
	Rhandle
	section_handle
	section_name
	service_handle
	snapshot_handle
	stacktrace
	status_code
	string
	target_handle
	text
	thread_handle
	type
	uuid
	value
	volume_mount_point
	wasbuffer
	wascmd
	window_name

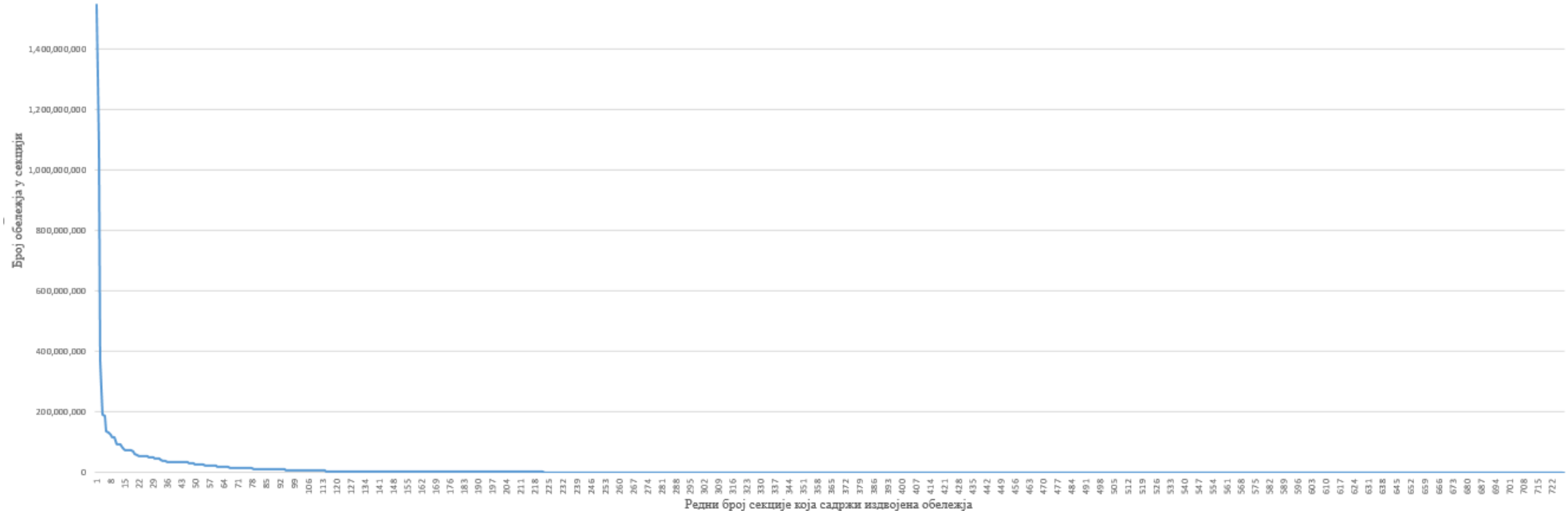
5.4 Расподела обележја по типовима

На начин описан у претходној секцији, свака путања до секције која садржи обележје описује тип обележја. Интерпретирање путања извршено је на основу доменског искуства. Потом је извршено груписање путања по сличности у тзв. надсекције.

За обележја издвојена техником ЦВ откривено је 437 различитих путања, чији бројеви појављивања припадају интервалу [12031, 1125578042], што је илустровано на слици 13а). За обележја издвојена техником ТФ-ИДФ откривено је 728 различитих путања, чији бројеви појављивања припадају интервалу [16274, 1545021470], што је илустровано на слици 13б). Увид ограниченог обима у скуп издвојених путања дат је на слици 11.



а)



б)

Слика 13: Расподела појављивања обележја а) ЦВ и б) ТФ-ИДФ.

За додатно илустровање груписања појединачних секција, у табели 15 је приказано десет секција, поређаних опадајуће по укупном броју обележја које садрже. Ова табела се односи само на обележја издвојена техником ЦВ. За обележја издвојена техником ТФ-ИДФ добија се сличан увид.

Табела 15: Објашњење значења десет секција које садрже највећи број обележја, на основу ког се може видети зашто је одређена секција груписана у одговарајућу надсекцију.

Р. Бр.	Ознака секције у извештају	Објашњење	Надсекција	Укупан број обележја
1	[]	Обележје се налази у корену или резултат анализе није индикативан (в. проблеме описане у сек. 5.3.3).	[]	1545021470
2	behavior processes calls arguments buffer	Представља секцију у којој се прикупљају подаци о понашању злонамерног софтвера – *behavior*. Унутар наведене секције наводе се подаци о понашању везаном за позиве који се извршавају у оквиру процеса, нпр., позив ка другом процесу, ка одређеном модулу или функцији. Такође, овај случај може да се односи и на АПИ-позиве. Заједничка основа ових секција јесте та да укључују податке о баферу који садржи аргументе који се прослеђују у позиву, нпр., вредност улазног аргумента, вредност регистарског кључа, путања до фајла или вредност у примарној меморији. Податак у секцији *buffer* обично је приказан као низ хексадецималних вредности.	API	1084839707
3	signatures description	Листа потписа (*signatures*) који су идентификовани током анализе. Потписи (нпр., *YARA rules*) су дефинисани обрасци који се користе за откривање специфичних активности или понашања софтвера (попут системских позива, мрежних приступа, техника избегавања	Signatures	372174079

Р. Бр.	Ознака секције у извештају	Објашњење	Надсекција	Укупан број обележја
		итд.). Секција *description* садржи опис потписа, тј. опис детектованог понашања софтвера.		
4	behavior processes calls arguments value	Објашњење ових секција слично је објашњењу датом у врсти ове табеле са индексом 2. Једина разлика је што су у овим секцијама садржани подаци који се прослеђују (*values*).	API	190276974
5	behavior processes modules baseaddr	Секције садрже податке о модулима који су учитани у одређени процес, попут динамичких библиотека (*DLL*), извршних фајлова или других делова кода. Обележје представља меморијску адресу (*baseaddr*).	Memory	185880964
6	behavior processes calls arguments module_address	Овај кључ даје адресу у меморији на којој се налази модул који се извршава. Пошто садржи чвор *calls*, категорисан је као АПИ-позив.	API	134425819
7	signatures marks ioc	Индикатори компромитовања (*IOC, Indicators of Compromise*) представљају податке које се користе за идентификовање угрожених система. У сендбокс-извештајима, *IOC* обухвата различите индикаторе, као што су ИП адресе, домени, хеш-вредности фајлова, *URL*-адресе, који указују да је систем био нападнут или заражен. Ова врста индикатора, осим одређених листа (*blacklist*) на којима се могу налазити конкретни субјекти, може се препознати према идентификованим потписима (*signatures*), који су индикативни за обрасце понашања.	Signatures	130237424

Р. Бр.	Ознака секције у извештају	Објашњење	Надсекција	Укупан број обележја
8	target file sha512	Односи се на хеш-вредност *SHA-512* специфичног фајла који је био циљ напада злонамерног софтвера.	File	125451346
9	dropped ssdeep	Неки злонамерни софтвери генеришу, преузимају или смештају фајлове у систему. *SSDeep* је алат који користи технику *fuzzy hashing* за упоређивање фајлова. За разлику од традиционалних приступа заснованих на хеш-вредностима (нпр. *SHA-512*), који једнозначно пресликавају фајл у хеш вредност, *SSDeep* омогућава упоређивање фајлова који су слични, али не нужно и идентични, што је корисно у откривању варијанти злонамерног софтвера. Пошто се ради о фајловима, секција је категорисана је у надсекцију *File*.	File	117801509
10	signatures ttp T1045 short	Образац понашања препознат на основу идентификованог потписа (signatures). У конкретном случају, образац се односи на технику TTP (Tactics, Techniques, and Procedures) вредност кључа T1045, тј., описује нападаче који користе комуникацију преко легитимних канала.	Signatures	59573171

Резултати груписања подсекција у надсекције приказани су у Табелама 16 и 17, које приказују типове обележја и бројеве њиховог појављивања у корпусу. Ове табеле односе се на обележја издвојена применом техника ЦВ и ТФ-ИДФ, респективно. Типови обележја описани су на следећи начин:

- *API* – обележја која се односе на АПИ-позиве,
- [] – резултат анализе типа обележја није индикативан,

- *File* – обележја која се односе на операције над фајловима, искључујући АПИ-позиве.
- *Signatures* – обележја која се односе на детектовање потписа, тј., образаца понашања софтвера, обухватајући правила *YARA* и друге унапред дефинисане начине препознавања злонамерног понашања, али искључујући операције над фајловима, мрежни саобраћај и остале активности карактеристичне за друге типове обележја.
- *Network* – обележја која се односе на мрежни саобраћај, укључујући фајлове који се транспортују мрежом или команде које се извршавају на локалном систему, у сврху даљинског приступа.
- *Memory* – обележја која се односе на рад са меморијом (*buffer, base address* итд.), искључујући АПИ-позиве.
- *IOC* – обележја која се односе на индикаторе компромитовања система, тј., на шаблоне штетног понашања софтвера или евидентиране субјекте, попут ИП-адреса, домена, врста комуникације и друго.
- *Static* – обележја која се односе на статичку анализу узорка.
- *Registry* – обележја која се односе на промене у регистрима, искључујући оне које су настале извршавањем позива произвољног типа.
- *Process* – обележја која се односе на извршавање процеса, искључујући АПИ-позиве.
- *Raw* – обележја која се односе на изворне податке који нису обрађени од стране неког модула.
- *Browser* – обележја која се односе на понашање интернет прегледача. За разлику од обележја типа *Network*, ова обележја се односе на активности корисника приликом коришћења прегледача, које су примарно повезане са техникама избегавања извршавања уграђеним у одређене злонамерне софтвере.
- *Debuger* – обележја која се односе на откривање инструкција које софтвер користи, и њиховог утицаја на системске ресурсе, процесе, меморију итд., као и на откривање тешко доступних или скривених компоненти софтвера које не манифестују јасно своје активности током стандардне анализе.
- *Metadata output* – обележја која се односе на метаподатке о спроведеној анализи, укључујући податке о виртуелној машини, трајању анализе итд.
- *Other* – обележја која не придају ни једном од претходно наведених типова.

Табела 16: Расподела обележја издвојених техником ЦВ.

	Назив надсекције	Број обележја са великом инф. вредношћу у надсекцији	%
1	API	1501324301	32,39
2	[]	1125578042	24,29
3	File	622124090	13,42
4	Signatures	477768187	10,31
5	Network	414239559	8,94
6	Memory	240971020	5,20
7	IOC	144485410	3,12
8	Static	63356723	1,37
9	Registry	24485506	0,53
10	Process	13025688	0,28
11	Raw	5611861	0,12
12	Browser	1881050	0,04
	Укупно:	4634851437	100,00

Табела 17: Расподела обележја издвојених техником ТФ-ИДФ.

	Назив надсекције	Број обележја са великом инф. вредношћу у надсекцији	%
1	API	2371952592	33,37
2	[]	1545021470	21,74
3	File	1012990353	14,25
4	Signatures	785811277	11,06
5	Network	568619664	8,00
6	Memory	456314015	6,42
7	IOC	131045470	1,84
8	Debuger	96501492	1,36
9	Static	88053151	1,24
10	Registry	21151650	0,30
11	Raw	9162344	0,13
12	Metadata output	7063072	0,10
13	Browser	6090517	0,09
14	Process	5157795	0,07
15	Other	2932785	0,04
	Укупно:	7107867647	100

Из табела 16 и 17 може се видети да заступљеност обележја која се односе на АПИ-позиве не прелази 32,39% свих обележја издвојених техником ЦВ, односно 33.37% процената свих обележја издвојених техником ТФ-ИДФ. Ово упућује на закључак да је проширење скупа динамичких обележја није занемарљиво. Осим тога, предност примене технике ТФ-ИДФ у односу на технику ЦВ јесте што она резултује не само већим бројем изведених обележја, већ и већом разноврсношћу њихових типова. Овај увид је у складу с резултатима изложеним у поглављу 4.

5.5 Обележја са највећом информационом добити

Табеле 13 и 14 приказују издвојена обележја чија је информациона добит већа од адаптивно одређеног прага, за технике издвајања обележја ЦВ и ТФ-ИДФ, респективно.

Табела 13: Обележја чија је информациона добит изнад глобалног прага вредности (најважнија обележја) издвојена техником ЦВ.

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
1	2553377	0x7fefd650000	0,005858	1016	0	1016
2	2553115	0x7fefc770000	0,005592	2070	1644	426
3	2553314	0x7fefd2c0000	0,005132	1016	0	1016
4	2553200	0x7fefce20000	0,004864	1016	0	1016
5	2553202	0x7fefce50000	0,00481	2070	1644	426
6	19755040	u00b2e	0,004423	663	107	556
7	14161742	sha512	0,003817	3108	1645	1463
8	14969099	u0002	0,003772	1537	456	1081
9	2552612	0x7fefaad0000	0,003426	2070	1644	426
10	8912063	96	0,003301	3108	1645	1463
11	2553568	0x7fefe270000	0,003253	2070	1644	426
12	21014549	u00c7	0,003146	1200	187	1013
13	2553264	0x7fefd0a0000	0,003009	2070	1644	426
14	23856191	u00f6e	0,002964	489	3	486
15	14423990	ttp	0,00294	3108	1645	1463
16	2536393	0x75730000	0,002935	1023	3	1020
17	2553082	0x7fefc540000	0,002884	2070	1644	426
18	13109106	logs	0,002884	823	23	800
19	13020069	l2nocm9tzv9lehrlbnnpb24vymxvynmvnzi0qufxnv9zt2rv	0,002741	0	0	0

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
		duwymeresezgvmjnqq				
20	2553373	0x7fefd590000	0,002728	2070	1644	426
21	21520330	u00cff	0,002724	699	107	592
22	14272221	t1053	0,002635	0	0	0
23	24103821	u00fah	0,002591	518	1	517
24	2544479	0x77c00000	0,002537	922	0	922
25	23022331	u00e8j	0,002512	598	2	596
26	9767726	arguments	0,002497	3093	1645	1448
27	15393903	u000be	0,002463	557	0	557
28	15151453	u0004q	0,002444	725	117	608
29	15056280	u0003c	0,002425	530	1	529
30	23261644	u00ecj	0,002408	622	4	618
31	23045997	u00e8w	0,00239	632	5	627
32	14034889	ro	0,002354	3108	1645	1463
33	10282129	bq	0,00234	1211	258	953
34	20652509	u00c0z	0,002334	538	1	537
35	23199901	u00ebi	0,002312	588	1	587
36	2553091	0x7fefc590000	0,00231	2070	1644	426
37	20432103	u00bdg	0,002301	520	3	517
38	2553079	0x7fefc4b0000	0,002266	2070	1644	426
39	23556090	u00f1f	0,002261	635	1	634
40	22712706	u00e3b	0,002219	517	4	513
41	15583174	u00108	0,002186	626	4	622
42	10836969	cy	0,002182	1941	678	1263
43	20077257	u00b7p	0,002177	528	3	525
44	22776442	u00e4e	0,002177	540	1	539
45	17552389	u008dx	0,002165	546	1	545
46	15182724	u00058	0,002144	609	6	603
47	14219649	stacktrace	0,002137	2364	985	1379
48	3345117	168	0,002132	3108	1645	1463
49	14926639	u0001g	0,00213	678	126	552
50	2553427	0x7fefd930000	0,002117	1016	0	1016
51	13378320	ng	0,002112	3108	1645	1463
52	21459613	u00cef	0,002101	547	2	545
53	2552266	0x7fef8110000	0,0021	1016	0	1016
54	13522934	nz	0,002087	1200	252	948

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
55	13512736	nx	0,00208	1367	204	1163
56	22336484	u00dcy	0,00208	503	0	503
57	20108249	u00b89	0,002069	597	8	589
58	18604163	u009fd	0,002062	553	3	550
59	11877353	f2	0,002048	2940	1489	1451
60	22005650	u00d7j	0,002045	552	3	549
61	15922104	u0015b	0,002038	539	4	535
62	16183956	u0019c	0,002034	521	0	521
63	21511252	u00cfa	0,002022	524	4	520
64	20500337	u00bek	0,002013	520	3	517
65	2544423	0x77940000	0,002008	1016	0	1016
66	24515585	uc	0,002005	3108	1645	1463
67	19347897	u00abp	0,001997	559	8	551
68	19206117	u00a9d	0,001997	522	4	518
69	20259057	u00bam	0,001994	515	1	514
70	20992770	u00c6o	0,001993	529	3	526
71	21307664	u00cbv	0,001983	528	0	528
72	19271785	u00aag	0,001981	515	4	511
73	16571557	u001fa	0,001967	648	45	603
74	22491182	u00dfk	0,001961	495	4	491
75	21638025	u00d1d	0,001949	630	106	524
76	20056735	u00b7e	0,001945	510	1	509
77	17974608	u0094t	0,001943	522	3	519
78	20359759	u00bca	0,001941	528	0	528
79	23805073	u00f5j	0,001939	554	2	552
80	20736419	u00c2f	0,001939	529	4	525
81	17640438	u008fe	0,001935	520	1	519
82	21656244	u00d1n	0,001934	491	1	490
83	21658054	u00d1o	0,001932	513	1	512
84	15722960	u0012a	0,001921	509	2	507
85	13164699	ma	0,001907	3108	1645	1463
86	2553152	0x7fefcb10000	0,001893	2070	1644	426
87	22476887	u00dfc	0,001885	514	3	511
88	20381950	u00bcm	0,001884	515	1	514
89	2553134	0x7fefca10000	0,001882	2070	1644	426
90	19194819	u00a96	0,001875	521	1	520
91	4295551	251	0,001867	3092	1644	1448

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
92	9950297	b6	0,001861	2929	1476	1453
93	22525812	u00e05	0,001861	804	217	587
94	17252025	u00899	0,001861	559	7	552
95	23263418	u00eck	0,001859	652	107	545
96	20780331	u00c35	0,001852	734	211	523
97	13551219	oc	0,001838	3108	1645	1463
98	2625397	104	0,001838	3067	1624	1443
99	21026569	u00c7_	0,001823	692	107	585
100	20663911	u00c18	0,001814	586	5	581
101	24729603	wb	0,001804	1247	309	938
102	19250264	u00aa2	0,001799	531	1	530
103	19143381	u00a8b	0,001792	547	6	541
104	2750018	123	0,001784	3083	1645	1438
105	10234147	bind	0,001766	584	69	515
106	24579783	urls	0,001743	3108	1645	1463
107	11610601	eax	0,001743	771	4	767
108	2543429	0x77290000	0,001738	1929	1525	404
109	18057710	u0096_	0,001731	519	0	519
110	24330984	u00fe_	0,00173	523	1	522
111	23300702	u00ed7	0,001727	765	213	552
112	17546442	u008du	0,001706	623	1	622
113	21081359	u00c85	0,001689	797	212	585
114	9787626	auxiliary	0,001673	2981	1645	1336
115	2553243	0x7fefcfc0000	0,001656	2070	1644	426
116	11822418	explorer	0,00164	631	228	403
117	16976938	u0084p	0,001635	537	5	532
118	14133720	server_random	0,001633	779	9	770
119	22795848	u00e4p	0,001619	507	1	506
120	2553194	0x7fefcd90000	0,001612	1016	0	1016
121	2553123	0x7fefc860000	0,001605	2070	1644	426
122	20196328	u00b9m	0,001586	518	1	517
123	14305007	task	0,001582	3057	1645	1412
124	2553080	0x7fefc4d0000	0,001575	2070	1644	426
125	14825095	u0000r	0,001564	1454	442	1012
126	15800368	u0013g	0,001548	498	0	498
127	2553222	0x7fefcef0000	0,001544	2075	1644	431
128	2538557	0x75de0000	0,001539	768	0	768

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
129	13076372	likely	0,001503	831	42	789
130	10812474	crx	0,0015	726	2	724
131	10769500	cname	0,001469	23	5	18
132	13672630	page_readwrite	0,001461	0	0	0
133	15599313	u0010g	0,001447	550	4	546
134	19247874	u00aa	0,001423	1250	244	1006
135	24990403	z5	0,001421	965	114	851
136	20311628	u00bbh	0,001417	501	0	501
137	16424045	u001d	0,001386	1270	250	1020
138	12252855	file_non_directory_file	0,001382	0	0	0
139	2553232	0x7fefcf40000	0,00138	1016	0	1016
140	14348624	threadingmodel	0,001378	0	0	0
141	5901950	49170	0,001342	544	7	537
142	24557906	universal	0,001329	58	11	47
143	12498132	guest	0,001323	2961	1645	1316
144	10075745	based	0,001297	1348	243	1105
145	24632037	vb_globalnamespace	0,001294	0	0	0
146	2553611	0x7fefedb0000	0,001287	1016	0	1016
147	12253848	files	0,001267	1951	640	1311
148	2553175	0x7fefccc0000	0,001253	1016	0	1016
149	20774335	u00c3	0,001233	1248	234	1014
150	19975386	u00b6	0,001217	1210	201	1009
151	5901998	49172	0,001208	547	7	540
152	11309863	dport	0,001197	3088	1645	1443
153	7483400	6fe88bb523e529ff004fb78c851d606a665adf5fd5107b80e0cc1d4d5112aa78	0,001191	641	0	641
154	11204015	deflate	0,001172	704	16	688
155	13986525	request	0,001165	3095	1645	1450
156	24864929	xml	0,001164	1490	691	799
157	2655183	1073741809	0,001162	845	321	524
158	24632047	vb_name	0,001159	0	0	0
159	13617965	openxmlformats	0,001157	142	5	137
160	16553779	u001f	0,001151	2305	981	1324
161	9743455	analysis	0,001148	2962	1645	1317

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
162	10075672	baseaddr	0,001145	3086	1644	1442
163	13176884	mb	0,001122	2247	949	1298
164	11319732	dscb	0,001119	1793	1371	422
165	11321844	dst	0,001118	3089	1645	1444
166	16358440	u001c	0,001094	1312	279	1033
167	595773	0x0000000009a2f6d0	0,001069	120	115	5
168	13361841	nend	0,001067	87	0	87
169	10269906	body	0,001058	998	136	862
170	18067730	u0096f	0,001052	570	1	569
171	4053741	2097152	0,001042	910	427	483
172	5902051	49174	0,001041	514	0	514
173	24769464	wmd	0,001039	410	216	194
174	2426015	0x73b06463	0,001032	216	210	6
175	17707308	u0090g	0,001014	0	0	0
Укупно:				203668	78961	124707

Табела 14: Обележја чија је информациона добит изнад глобалног прага вредности (најважнија обележја) издвојена техником ТФ-ИДФ.

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
1	2553373	0x7fefd590000	0,006361	2070	1644	426
2	2553202	0x7fefce50000	0,006325	2070	1644	426
3	2553115	0x7fefc770000	0,005822	2070	1644	426
4	2553132	0x7fefc9a0000	0,004682	2070	1644	426
5	2553074	0x7fefc4a0000	0,00411	2070	1644	426
6	13056369	legitimate	0,003855	2231	1353	878
7	2553160	0x7fefcba0000	0,003583	1016	0	1016
8	2553243	0x7fefcfc0000	0,003498	2070	1644	426
9	2553200	0x7fefce20000	0,003469	1016	0	1016
10	21014549	u00c7	0,003347	1200	187	1013
11	17463862	u008ck	0,003119	545	2	543
12	13261123	msctf	0,003067	725	301	424
13	2422136	0x73010000	0,002969	1748	1433	315
14	23856191	u00f6e	0,002962	489	3	486
15	2553604	0x7fefece0000	0,002959	1016	0	1016
16	2553364	0x7fefd550000	0,002913	1016	0	1016
17	14969099	u0002	0,002906	1537	456	1081
18	2543336	0x76eb0000	0,002898	2070	1644	426
19	13267192	mswsock	0,00284	3086	1644	1442
20	8489427	897024	0,002812	3086	1644	1442
21	13980610	released	0,00274	32	5	27
22	2888397	142	0,002535	2546	1142	1404
23	13020069	l2nocm9tzv9lehrlbnnpb24vymxvynmvnzi0qufxnv9zt2rvduwymeresezgvmjnqq	0,002515	0	0	0
24	23022331	u00e8j	0,002512	598	2	596
25	13638605	output	0,0025	2982	1645	1337
26	2553442	0x7fefda10000	0,002497	1016	0	1016
27	2553116	0x7fefc7d0000	0,002488	2070	1644	426
28	14423990	ttp	0,002465	3108	1645	1463
29	15393903	u000be	0,002462	557	0	557
30	23261644	u00ecj	0,002427	622	4	618
31	15056280	u0003c	0,002427	530	1	529

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
32	23045997	u00e8w	0,00239	632	5	627
33	20652509	u00c0z	0,002334	538	1	537
34	24448832	u00ffy	0,002322	557	4	553
35	2563736	0xffd90000	0,002307	2070	1644	426
36	23199901	u00ebi	0,002305	588	1	587
37	20432103	u00bdg	0,002299	520	3	517
38	24103821	u00fah	0,002296	518	1	517
39	17298035	u0089x	0,002285	601	7	594
40	6409239	55232	0,002262	1942	1440	502
41	23556090	u00f1f	0,002261	635	1	634
42	23183169	u00eb9	0,002244	617	1	616
43	2681002	110592	0,002241	1372	221	1151
44	14343886	that	0,002239	2505	1417	1088
45	10282129	bq	0,002236	1211	258	953
46	21520330	u00cff	0,002224	699	107	592
47	10764371	cmd	0,002219	1631	865	766
48	15583174	u00108	0,002186	626	4	622
49	20077257	u00b7p	0,002177	528	3	525
50	22776442	u00e4e	0,00217	540	1	539
51	22712706	u00e3b	0,002166	517	4	513
52	17552389	u008dx	0,002165	546	1	545
53	17682763	u00901	0,002162	581	5	576
54	24757602	windows	0,002145	2652	1645	1007
55	15182724	u00058	0,002144	609	6	603
56	17119366	u0087	0,002138	1194	194	1000
57	9585267	ac	0,002133	3108	1645	1463
58	22336484	u00dcy	0,00208	503	0	503
59	13029239	la	0,002068	3108	1645	1463
60	18604163	u009fd	0,002062	553	3	550
61	22983928	u00e7v	0,002061	530	1	529
62	24632047	vb_name	0,002058	0	0	0
63	22005650	u00d7j	0,002045	552	3	549
64	20108249	u00b89	0,002044	597	8	589
65	15922104	u0015b	0,002038	539	4	535
66	16183956	u0019c	0,00203	521	0	521
67	16927385	u0083v	0,002024	527	4	523
68	20500337	u00bek	0,002013	520	3	517

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
69	19760669	u00b2h	0,002007	500	0	500
70	10812751	cryptbase	0,002002	3086	1644	1442
71	19206117	u00a9d	0,001997	522	4	518
72	14236355	summary	0,001988	3087	1644	1443
73	21307664	u00cbv	0,001983	528	0	528
74	19347897	u00abp	0,001982	559	8	551
75	21240327	u00car	0,00198	544	5	539
76	20992770	u00c6o	0,001965	529	3	526
77	22491182	u00dfk	0,00196	495	4	491
78	24515585	uc	0,001951	3108	1645	1463
79	16571557	u001fa	0,00195	648	45	603
80	22453010	u00dew	0,00195	513	0	513
81	20056735	u00b7e	0,001945	510	1	509
82	17974608	u0094t	0,001943	522	3	519
83	20359759	u00bca	0,001941	528	0	528
84	20736419	u00c2f	0,00194	529	4	525
85	23429261	u00efc	0,001937	510	2	508
86	17640438	u008fe	0,001935	520	1	519
87	21656244	u00d1n	0,001934	491	1	490
88	21658054	u00d1o	0,00193	513	1	512
89	15722960	u0012a	0,001921	509	2	507
90	12712766	image	0,001897	1331	676	655
91	22476887	u00dfc	0,001885	514	3	511
92	2553314	0x7fefd2c0000	0,001876	1016	0	1016
93	19194819	u00a96	0,001875	521	1	520
94	19293598	u00aas	0,001866	514	3	511
95	18887223	u00a4	0,001855	1260	252	1008
96	12252752	file_created	0,001841	1419	305	1114
97	13781870	processing	0,001832	2961	1645	1316
98	6401167	55079	0,001829	2044	1558	486
99	21346295	u00ccj	0,001823	650	106	544
100	13164699	ma	0,001809	3108	1645	1463
101	8149224	823296	0,001808	3086	1644	1442
102	23678988	u00f3g	0,001807	523	4	519
103	6319096	537	0,001805	2027	749	1278
104	13551219	oc	0,001799	3108	1645	1463
105	19250264	u00aa2	0,001799	531	1	530

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
106	13378320	ng	0,001781	3108	1645	1463
107	24211038	u00fc_	0,00177	588	1	587
108	22187057	u00daj	0,00177	539	1	538
109	24729603	wb	0,001754	1247	309	938
110	11331138	dump	0,001734	2983	1645	1338
111	24330984	u00fe_	0,00173	523	1	522
112	14378693	tmpriph3r	0,00173	1039	1	1038
113	20663911	u00c18	0,001716	586	5	581
114	13986525	request	0,001715	3095	1645	1450
115	18057710	u0096_	0,001713	519	0	519
116	11610601	eax	0,001685	771	4	767
117	9767726	arguments	0,00167	3093	1645	1448
118	2553285	0x7fefd150000	0,00165	1016	0	1016
119	12943510	kf	0,001627	1116	243	873
120	16053693	u0017c	0,001618	514	0	514
121	2534908	0x75410000	0,001611	2023	1525	498
122	17072979	u0086b	0,001605	503	3	500
123	20196328	u00b9m	0,001587	518	1	517
124	24757116	win64	0,00158	9	4	5
125	854266	0x000000a8	0,001553	1020	539	481
126	14131475	sending	0,001546	2960	1644	1316
127	12498132	guest	0,001546	2961	1645	1316
128	13190435	me	0,00154	3108	1645	1463
129	24632016	vb_base	0,001536	0	0	0
130	10233994	binary	0,001529	965	71	894
131	13215663	milliseconds	0,001522	1114	431	683
132	13115996	lpk	0,00152	312	1	311
133	21763171	u00d3g	0,001514	506	3	503
134	19247874	u00aa	0,001471	1250	244	1006
135	14414142	trying	0,001467	2964	1645	1319
136	2553091	0x7fefc590000	0,001448	2070	1644	426
137	14069789	running	0,00141	3036	1645	1391
138	2553553	0x7fefe130000	0,001401	2070	1644	426
139	5171726	3884763	0,001394	2961	1645	1316
140	13906462	qy	0,001385	1192	327	865
141	6918171	61797	0,001375	1041	9	1032
142	2538426	0x75d70000	0,001374	910	0	910

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
143	24791121	ws2_32	0,001368	187	12	175
144	24990403	z5	0,001359	965	114	851
145	10789583	context	0,001353	1722	620	1102
146	15666166	u0011f	0,001324	560	3	557
147	13176884	mb	0,001323	2247	949	1298
148	13512736	nx	0,001319	1367	204	1163
149	21638025	u00d1d	0,001315	630	106	524
150	2540450	0x76650000	0,001281	884	0	884
151	13461213	nprivate	0,001278	0	0	0
152	10788865	contains	0,001255	1505	368	1137
153	6011895	4d	0,001254	3099	1645	1454
154	14133656	server	0,001221	3108	1645	1463
155	2553263	0x7fefd030000	0,001212	1016	0	1016
156	11877353	f2	0,001207	2940	1489	1451
157	2542976	0x76d80000	0,001207	1920	1523	397
158	16424045	u001d	0,001198	1270	250	1020
159	22647896	u00e28	0,001197	525	1	524
160	11798071	errors	0,001196	2982	1645	1337
161	2553351	0x7fefd4f0000	0,001191	2070	1644	426
162	2553135	0x7fefca60000	0,001178	2070	1644	426
163	11121693	dbgview	0,001166	2982	1645	1337
164	2553117	0x7fefc7e0000	0,001155	2070	1644	426
165	18253780	u0099j	0,001155	530	5	525
166	2553540	0x7fefe020000	0,001154	2070	1644	426
167	23539734	u00f15	0,001142	518	3	515
168	845280	0x00000002	0,001135	440	217	223
169	13134130	lv	0,001129	1694	625	1069
170	15304446	u0007	0,001117	1295	277	1018
171	9748202	answers	0,001114	3107	1645	1462
172	18879987	u00a3w	0,001114	547	2	545
173	12330871	function	0,001109	1738	733	1005
174	17694568	u0090a	0,001106	533	3	530
175	10784174	compressed	0,00109	860	43	817
176	24632195	vbaproject	0,001089	0	0	0
177	24632069	vba	0,001069	693	138	555
178	11143313	dccfc164	0,001065	134	6	128
179	2553353	0x7fefd520000	0,001064	1016	0	1016

Р. Бр.	Индекс	Обележје	Важност	Укупно узорака	Безопасних узорака	Злонамерних узорака
180	12616486	html	0,001064	1413	281	1132
181	2748952	122880	0,001053	3086	1644	1442
182	2553079	0x7fefc4b0000	0,00104	2070	1644	426
183	10758232	clients2	0,001034	775	0	775
184	23750232	u00f4m	0,001034	626	107	519
185	12436303	getversionexa	0,001022	0	0	0
186	13265991	msprivs	0,001021	3086	1644	1442
187	21764896	u00d3h	0,001019	538	6	532
188	17700900	u0090d	0,001008	597	3	594
189	2553195	0x7fefce10000	0,001	1016	0	1016
190	21026569	u00c7_	0,001	692	107	585
191	18217167	u0098w	0,000995	583	6	577
192	13125440	lsasrv	0,000994	3086	1644	1442
193	5498046	4103	0,00099	903	259	644
194	10789105	content_types	0,000978	0	0	0
195	12252894	file_share_read	0,000977	0	0	0
Укупно:				250108	104325	145783

6 Дискусија

Три главна доприноса спроведеног истраживања описана су у наставку.

6.1 Нови корпус података

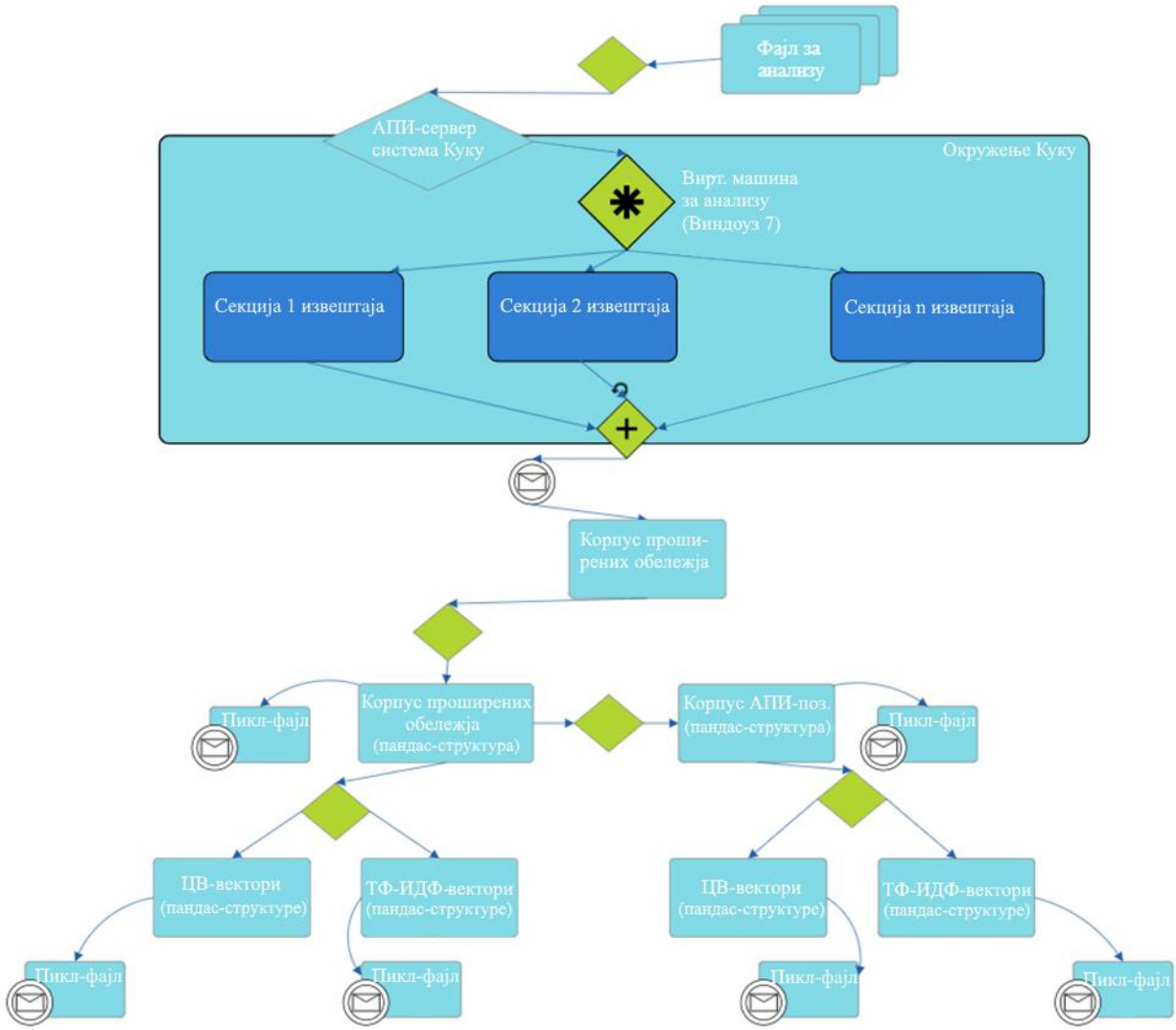
Први допринос овог истраживања представља генерисање новог корпуса за динамичку анализу злонамерног софтвера, који садржи извештаје о резултатима динамичке анализе узорака и задовољава следеће карактеристике:

- сви изворни фајлови који су предмет динамичке анализе представљају примере из реалног живота,
- изворни скуп фајлова садржи педесет четири врсте фајлова (тј., различите екстензије),
- корпус садржи 22200 узорака генерисаних спровођењем аутоматске динамичке анализе изворних фајлова у окружењу Куку, са следећом расподелом:
    - 10465 узорака представља резултате динамичке анализе злонамерних софтвера,
    - 11735 узорака представља резултате динамичке анализе безопасних софтвера,
- величина корпуса износи 969GB, при чему злонамерни узорци заузимају 935GB, а безопасни 34GB,
- из корпуса је изведено више од 25 милиона динамичких обележја софтвера и
- корпус је јавно доступан за потребе истраживања.

Иако су злонамерни и безопасни узорци уравнотежено заступљени, величина злонамерних узорака је знатно већа од величине безопасних узорака. Злонамерни узорци су генерисали више активности у окружењу за динамичку анализу од безопасних узорака, па су резултујући извештаји о њиховој динамичкој анализи већи (чак 228 злонамерних софтвера индукује извештаје веће од 1GB).

Број узорака у корпусу, његова величина и број изведених обележја знатно превазилазе корпусе сличне намене сачињене само од секвенци АПИ-позива. Треба приметити да тренутно не постоје други јавно доступни корпуси са оваквим карактеристикама који би били прикладни за анализу бинарног класификовања злонамерног софтвера засновану на скупу динамичких обележја која, по свом типу, превазилазе АПИ-позиве.

На слици 14 приказан је процес генерисања корпуса, од подношења изворних фајлова на анализу, преко њихове анализе у окружењу Куку, до генерисања корпуса проширених обележја и корпуса АПИ-позива. Такође, на слици су назначени поступци издвајања обележја и смештање корпуса у пикл-фајлове због лакше накнадне примене. Зеленим ромбом на слици 15 означена је примена Пајтон-скрипти за обраду.



Слика 14: Процес генерисања корпуса проширених обележја и АПИ позива, издвајања обележја техником ЦВ и ТФ-ИДФ и чувања у серијализоване пикл фајлове. Слика се најбоље интерпретира у боји.

6.2 Демонстрирање веће дискриминаторне моћи проширеног скупа обележја

Други допринос овог истраживања односи се на:

- развој прототипског система за аутоматску динамичку анализу софтвера у контексту бинарног класификовања софтвера на злонамерни и безопасни, заснованог на динамичким обележјима из датог корпуса и ансамблу стабала одлучивања,
- експериментално демонстрирање да проширени скуп динамичких обележја унапређује перформансе оваквог система у односу на систем који би био заснован само на скупу обележја која се односе на АПИ-позиве.

Из корпуса проширених обележја изведен је означени корпус АПИ-позива, који садржи само узорке који индукују АПИ-позиве. За експерименталну потврду веће дискриминаторне моћи проширеног скупа обележја у односу на обележја која се односе на АПИ-позиве, примењен је ансамбл стабала одлучивања.

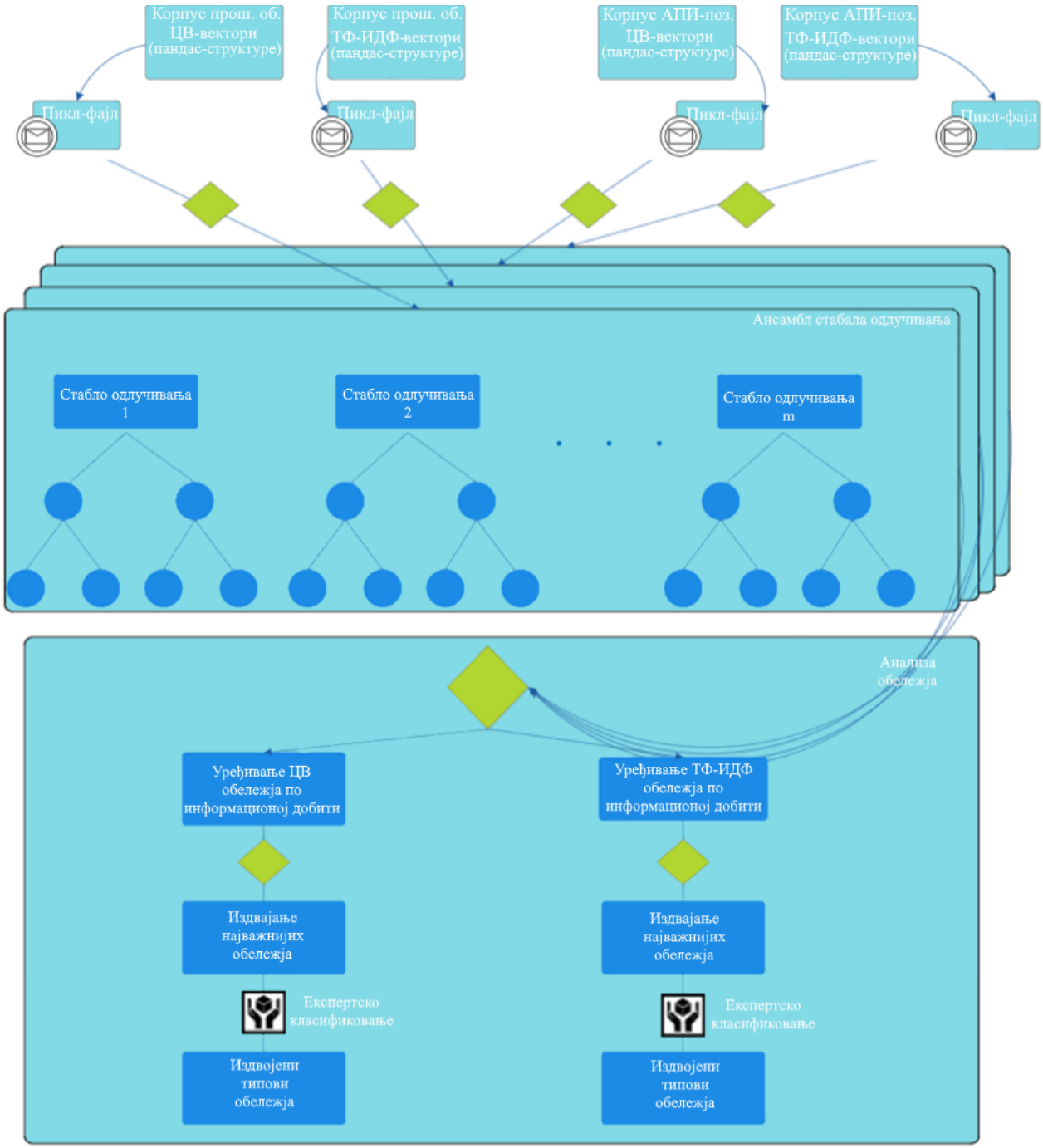
Експериментални резултати, приказани у табелама 8 и 9, потврђују да модели развијени на проширеном скупу динамичких обележја остварују већу тачност (и макропросечну F_1-меру) од модела развијених на скупу обележја која се односе на АПИ-позиве.

Тачност остварена на проширеном скупу обележја износи 99,74%, а на скупу обележја која се односе на АПИ-позиве 95,56%. Треба нагласити да су у корпус АПИ-позива укључени само узорци који су изведени из фајлова који индукују АПИ-позиве. Другим речима, 23,78% узорака који не укључују АПИ-позиве (тј., 39,5% безопасних фајлова и приближно 6,1% злонамерних фајлова) није укључено у овај корпус. Да су и ови узорци били укључени у анализу, тачност анализе засноване на АПИ-позивима била би смањена, а предност примене проширеног скупа обележја још наглашенија.

Иако су сви узорци бирани на случајан начин да би се избегла субјективна пристрасност приликом избора, узорци који не индукују АПИ-позиве нису укључени у изведени корпус АПИ-позива, да би се елиминисала могућност да је тачност модела који се заснивају само на анализи АПИ-позива вештачки смањена повећањем удела узорака који не индукују АПИ-позиве. Другим речима, за развој модела заснованих на АПИ-позивима примењен је корпус са узорцима који индукују АПИ-позиве, чиме је тачност ових модела максимизована. Међутим, чак и у овим условима, тачност модела обучених на проширеном скупу обележја превазилази тачност модела заснованих само на АПИ-позивима – у складу са почетним хипотезама.

На слици 15 илустрован је експериментални поступак спроведен у овом истраживању, који се односи на демонстрирање веће дискриминаторне моћи проширеног скупа обележја и

анализу овог скупа, дискутовану у наредној секцији. Зеленим ромбом на слици 15 означена је примена Пајтон-скрипти за обраду.



Слика 15: Приказ експерименталног поступка у овом истраживању. Слика се најбоље интерпретира у боји.

6.3 Анализа обележја

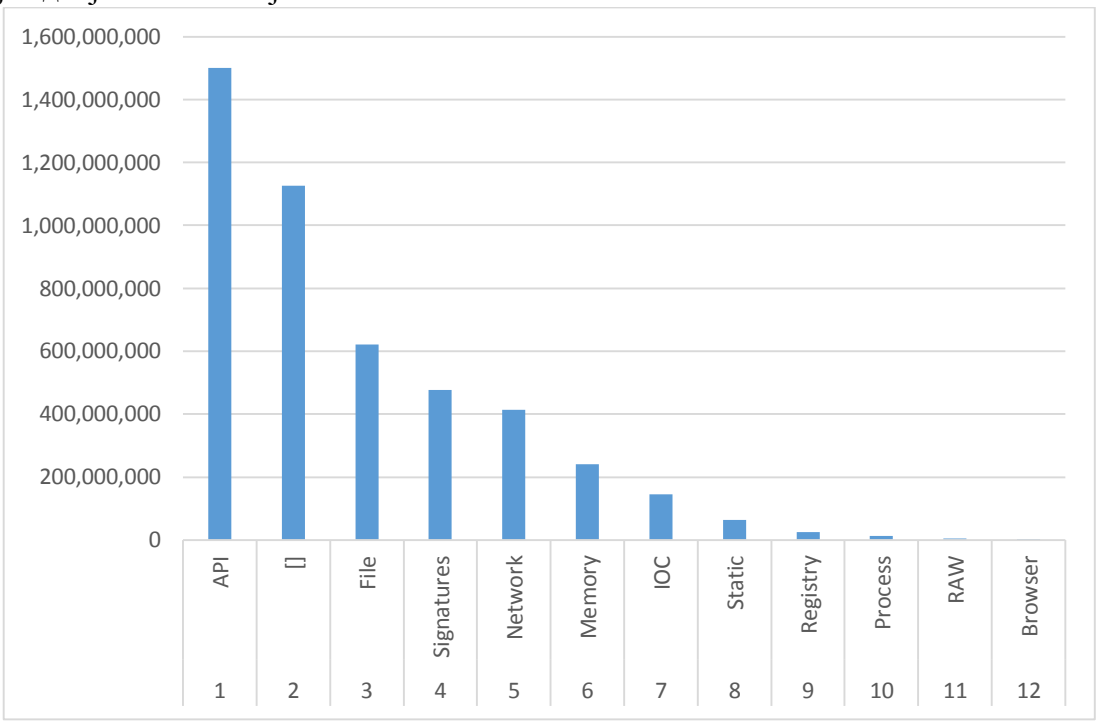
Трећи допринос овог истраживања односи се на анализу изведених обележја и одређивање типова обележја који су најзначајнији за откривање злонамерног софтвера. Слика 16 илуструје расподеле обележја по типу, изведене из табела 16 и 17.

Резултати ове анализе указују да су, поред АПИ-позива, следећи типови обележја важни за детектовање злонамерног софтвера:

- *File* – обележја која се односе на операције над фајловима, искључујући АПИ-позиве.
- *Signatures* – обележја која се односе на детектовање шаблона понашања софтвера, искључујући операције над фајловима, мрежни саобраћај и остале активности карактеристичне за друге типове динамичких обележја.
- *Network* – обележја која се односе на мрежни саобраћај.
- *Memory* – обележја која се односе на рад са меморијом, искључујући АПИ-позиве.
- *IOC* – обележја која се односе на индикаторе компромитовања система.

Ови резултати су у складу с увидима Ђиндела и других (*Jindal et al*., 2019), који следеће типове обележја наводе као важне: *Files, API-calls, Network, DLL-libraries* итд.

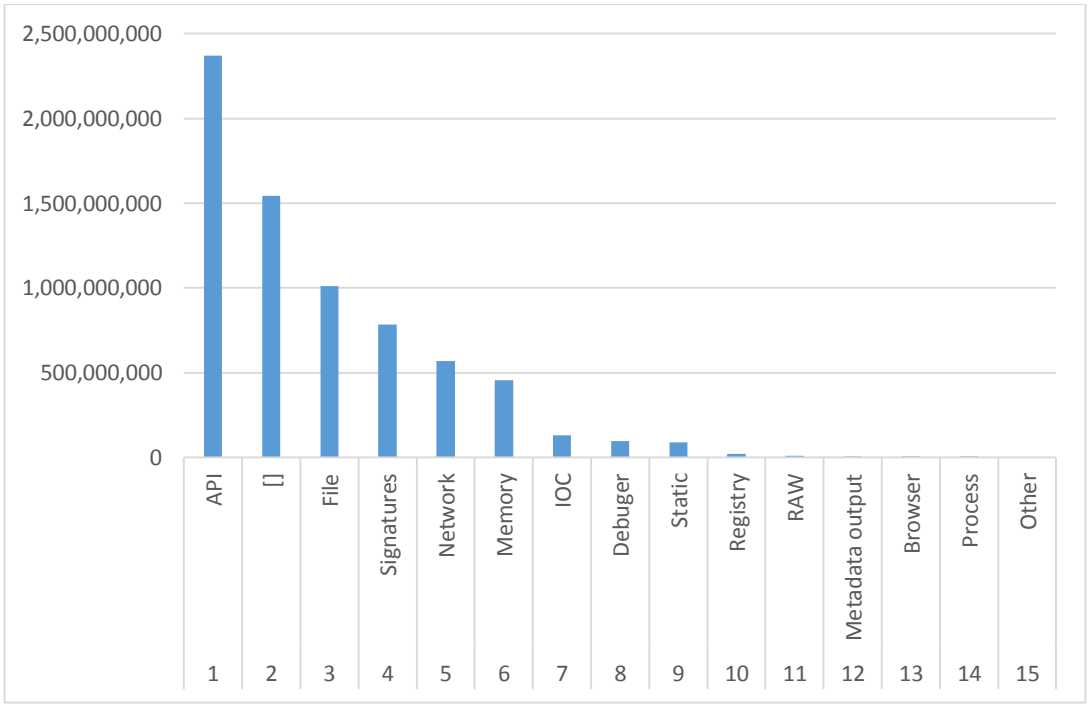
Број издвојених обележја



Тип обележја

а)

Број издвојених обележја



Тип обележја

б)

Слика 16: Расподела обележја по типовима, а) обележја издвојена применом технике ЦВ, б) обележја издвојена применом технике ТФ-ИДФ.

7 Закључак

У овој дисертацији предложено је унапређење динамичке анализе злонамерног софтвера засновано на проширеном скупу обележја и машинском учењу. Полазна тачка за ово истраживање била је та да анализа додатних динамичких обележја софтвера, која не припадају скупу уобичајено посматраних обележја заснованих на АПИ-позивима, може унапредити перформансе система за детектовање злонамерних софтвера.

Доприноси ове дисертације јесу следећи:

- Генерисан је нов корпус за динамичку анализу злонамерног софтвера, који садржи извештаје о резултатима динамичке анализе узорака. Број узорака у корпусу, његове величина и број изведених обележја знатно превазилазе уобичајене корпусе сличне намене сачињене само од секвенци АПИ-позива. Осим тога, тренутно не постоје други јавно доступни корпуси са оваквим карактеристикама који би били прикладни за анализу бинарног класификовања злонамерног софтвера, засновану на скупу динамичких обележја која, по свом типу, превазилазе АПИ-позиве.
- Развијен је прототипски систем за аутоматску динамичку анализу софтвера у контексту бинарног класификовања софтвера на злонамерни и безопасни, заснован на динамичким обележјима из датог корпуса и ансамблу стабала одлучивања. Експериментално је демонстрирано да проширени скуп динамичких обележја унапређује перформансе оваквог система у односу на систем који би био засновани само на скупу обележја која се односе на АПИ-позиве.
- Извршено је у утврђивање типова динамичких обележја софтвера која су, поред типа који се односи на АПИ-позиве, значајна за откривање злонамерног софтвера.

Овим доприносима успешно су потврђене хипотезе постављене у уводном поглављу.

Попис литературе

Alhashmi, A., Darem, A., Alanazi, M., Alashjaee, M., Aldughayfiq, B., Ghaleb, A., Ebad, A., & Alanazi, A. (2023). Hybrid malware variant detection model with extreme gradient boosting and artificial neural network classifiers. Computers, Materials & Continua, 76(3), 3483–3498. https://doi.org/10.32604/cmc.2023.041038

Alshmarni, A., & Alliheedi, M. A. (2023). Enhancing malware detection by integrating machine learning with Cuckoo Sandbox. arXiv e-prints. https://doi.org/10.48550/ARXIV.2311.04372

Anderson, H., & Rothl, P. (2018). EMBER: An open dataset for training static PE malware machine learning models. https://doi.org/10.48550/arXiv.1804.04637

AV-ATLAS. (2025). Malware. AV-ATLAS. https://portal.av-atlas.org/malware. Retrieved February 3, 2025.

Bosansky, B., Kouba, D., Manhal, O., Sick, T., Lisy, V., Kroustek, J., & Somol, P. (2022). Avast-CTU public CAPE dataset. arXiv:2209.03188. https://arxiv.org/abs/2209.03188

Chen, L., Yagemann, C., & Downing, E. (2019). To believe or not to believe: Validating explanation fidelity for dynamic malware analysis. arXiv:1905.00122 [cs.CR]. https://doi.org/10.48550/arXiv.1905.00122

Chen, T., Zeng, H., Lv, M., & Zhu, T. (2024). CTIMD: Cyber threat intelligence enhanced malware detection using API call sequences with parameters. Computers & Security, 136, 103518. https://doi.org/10.1016/j.cose.2023.103518

Cuckoo Sandbox CERT Estonia. (n.d.). About Cuckoo Sandbox. Cuckoo Sandbox. https://cuckoo-hatch.cert.ee/static/docs/about/cuckoo/. Retrieved February 3, 2025.

Cuckoo Sandbox. (n.d.). Customization and processing. Cuckoo Sandbox. https://cuckoo.readthedocs.io/en/latest/customization/processing/. Retrieved February 3, 2025.

Cutler, A., Cutler, D. R., & Stevens, J. R. (2012). Random forests. In C. Zhang & Y. Ma (Eds.), Ensemble machine learning (pp. 157–172). Springer. https://doi.org/10.1007/978-1-4419-9326-7_5

Deore, M., Tarambale, M., Ratna Raja Kumar, J., & Sakhare, S. (2023). GRASE: Granulometry analysis with semi eager classifier to detect malware. International Journal of Interactive Multimedia and Artificial Intelligence, 8(6), 120–134. https://doi.org/10.9781/ijimai.2023.12.002

Düzgün, B., Çayır, A., Demirkıran, F., Kahya, C. N., Gençaydın, B., & Dağ, H. (2022). Benchmark static API call datasets for malware family classification. arXiv e-prints. https://doi.org/10.48550/ARXIV.2111.15205

Ficco, M. (2022). Malware analysis by combining multiple detectors and observation windows. IEEE Transactions on Computers, 71(6), 1276–1290. https://doi.org/10.1109/TC.2021.3082002

Gibert, D., Mateu, C., & Planes, J. (2020). The rise of machine learning for detection and classification of malware: Research developments, trends and challenges. Journal of Network and Computer Applications, 102526. https://doi.org/10.1016/j.jnca.2019.102526

Gnjatović, M. (2025). Lecture notes in Machine Learning. Gnjatovic.info. http://gnjatovic.info/machinelearning/. Retrieved February 26, 2025.

Гњатовић, М. (2023). Организација рачунара из перспективе програмера. Криминалистичко полицијски универзитет. ISBN 978-86-7020-501-7.

Гњатовић, М. (2024). Увод у Процесинг. Криминалистичко полицијски универзитет. ISBN 978-86-7020-517-8.

Gnjatović, M., Tasevski, J., Borovac, B., & Maček, N. (August 22–24 2018). An entropy-based approach to automatic detection of critical changes in human-machine interaction. In Proceedings of the 9th IEEE International Conference on Cognitive Infocommunications (CogInfoCom) (pp. 175–178). Budapest, Hungary.

Gonzalez, R. C., & Woods, R. E. (2018). Digital image processing (4th ed.). Pearson Education. New York, 1022 p.

Greamo, C., & Ghosh, A. (2011). Sandboxing and virtualization: Modern tools for combating malware. IEEE Security & Privacy, 9(1), 79–82. https://doi.org/10.1109/MSP.2011.36

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). Springer. https://doi.org/10.1007/978-0-387-84858-7

Herrera-Silva, J., & Hernández-Álvarez, M. (2023). Dynamic feature dataset for ransomware detection using machine learning algorithms. Sensors, 23(3), 1053. https://doi.org/10.3390/s23031053

Ho, T. K. (1995). Random decision forests. In Proceedings of 3rd International Conference on Document Analysis and Recognition (Vol. 1, pp. 278–282). Montreal, QC, Canada. https://doi.org/10.1109/ICDAR.1995.598994

Hsiao, S. W., Sun, Y. S., & Chen, M. C. (2020). Hardware-assisted MMU redirection for in-guest monitoring and API profiling. IEEE Transactions on Information Forensics and Security, 15, 2402–2416. https://doi.org/10.1109/TIFS.2020.2969514

Huang, Y., Chen, T., Hsiao, S., & Sun, Y. (2022). Learning dynamic malware representation from common behavior. Journal of Information Science and Engineering, 38(6), 1317–1334. https://doi.org/10.6688/JISE.202211 38(6).0012

Huang, Y., Sun, Y., & Chen, M. (2022). TagSeq: Malicious behavior discovery using dynamic analysis. PLOS ONE, 17(5), e0263644. https://doi.org/10.1371/journal.pone.0263644

Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., & Leskovec, J. (2020). Strategies for pre-training graph neural networks. arXiv. https://arxiv.org/abs/2006.09296

Ilić, S., Gnjatović, M., Popović, B., & Maček, N. (2022). A pilot comparative analysis of the Cuckoo and Drakvuf sandboxes: An end-user perspective. Military Technical Courier, 70(2), 372–392. https://doi.org/10.5937/vojtehg70-36196

Ilić, S., Gnjatović, M., Tot, I., Jovanović, B., Maček, N., & Gavrilović Božović, M. (2024). Going beyond API calls in dynamic malware analysis: A novel dataset. Electronics, 13(17), 3553. https://doi.org/10.3390/electronics13173553

Ilić, S., Kuk, K., Stojanović, V., & Petrović, I. (2024). A clustering approach to malware dataset analysis. Journal of Computer and Forensic Sciences, 3(2), 43–56. https://doi.org/10.5937/jcfs3-55513

Irshad, A., & Dutta, M. (2021). Identification of Windows-based malware by dynamic analysis using machine learning algorithm. In X.-Z. Gao, S. Tiwari, M. C. Trivedi, & K. K. Mishra (Eds.), Advances in Computational Intelligence and Communication Technology (Vol. 1086, pp. 207–218). Springer Singapore. https://doi.org/10.1007/978-981-15-1275-9_18

Jain, M., & Singh, S. (2019). A hybrid approach for malware detection using machine learning. Journal of Information Security and Applications, 48, 102353. https://doi.org/10.1016/j.jisa.2019.102353

Jindal, C., Salls, C., Aghakhani, H., Long, K., Kruegel, C., & Vigna, G. (2019). Neurlux: Dynamic malware analysis without feature engineering. arXiv:1910.11376 [cs.CR]. https://doi.org/10.48550/ARXIV.1910.11376

Lee, D., Jeon, G., Lee, S., & Cho, H. (2022). Deobfuscating mobile malware for identifying concealed behaviors. Computers, Materials & Continua, 72(3), 5909–5923. https://doi.org/10.32604/cmc.2022.026395

Lengyel, T. K., Maresca, S., Payne, B. D., Webster, G. D., Vogl, S., & Kiayias, A. (2014). Scalability, fidelity and stealth in the DRAKVUF dynamic malware analysis system. In Proceedings of the 30th Annual Computer Security Applications Conference (pp. 386–395). ACM. https://doi.org/10.1145/2664243.2664252

Li, C., Cheng, C., Zhu, H., Wang, L., Lv, Q., Wang, Y., Li, N., & Sun, D. (2022). DMalNet: Dynamic malware analysis based on API feature engineering and graph learning. Computers & Security, 122, 102872. https://doi.org/10.1016/j.cose.2022.102872

Li, N., Lu, Z., Ma, Y., Chen, Y., & Dong, J. (2024). A malicious program behavior detection model based on API call sequences. Electronics, 13(6), 1092. https://doi.org/10.3390/electronics13061092

Li, S., Wen, H., Deng, L., Zhou, Y., Zhang, W., Li, Z., & Sun, L. (2023). Denoising network of dynamic features for enhanced malware classification. 2023 IEEE International Performance, Computing, and Communications Conference (IPCCC), 32–39. https://doi.org/10.1109/IPCCC59175.2023.10253838

Mandiant. (2024). M-Trends 2024: The threat report. https://www.mandiant.com/resources/reports/m-trends-2024. Retrieved February 3, 2025.

McKinney, W., & others. (2010). Data structures for statistical computing in python. In Proceedings of the 9th Python in Science Conference (Vol. 445, pp. 51–56). PANDAS Conf Paper.

Melvin, A. A., & Kathrine, G. J. (2020). A quest for best: A detailed comparison between Drakvuf-VMI-based and Cuckoo sandbox-based technique for dynamic malware analysis.

Milan, Gnjatović, Ivan Košanin, Nemanja Maček, & Dušan Joksimović. (2022). Clustering of road traffic accidents as a gestalt problem. Applied Sciences, 12(9), 4543. https://doi.org/10.3390/app12094543

Mira, F. (2019). A review paper of malware detection using API call sequences. 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS), 1–6. https://doi.org/10.1109/CAIS.2019.8769564

Mitre. (n.d.). MITRE ATT&CK. MITRE. https://attack.mitre.org/. Retrieved February 3, 2025.

Nunes, M., Burnap, P., Rana, O., Reinecke, P., & Lloyd, K. (2019). Getting to the root of the problem: A detailed comparison of kernel and user level data for dynamic malware analysis. Journal of Information Security and Applications, 48, 102365. https://doi.org/10.1016/j.jisa.2019.102365

Pedregosa, F., & others. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830. http://jmlr.org/papers/v12/pedregosa11a.html

Rafael, C. Gonzalez, & Richard, E. Woods. (2018). Digital image processing (4th ed.). Pearson.

Rossum, V. (2020). The Python library reference, release 3.8.2. Python Software Foundation.

Sethi, K., Kumar, R., Sethi, L., Bera, P., & Patra, P. (2019). A novel machine learning-based malware detection and classification framework. 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), 1–4. https://doi.org/10.1109/CyberSecPODS.2019.8885196

Shannon, C. E. (1948). A mathematical theory of communication. The Bell System Technical Journal, 27(3), 379–423. https://doi.org/10.1002/j.1538-7305.1948.tb00917.x

Shih, F. Y. (2010). Image Processing and Pattern Recognition: Fundamentals and Techniques. John Wiley & Sons: Hoboken, NJ, USA.

Sraw, J., & Kumar, K. (2022). Using static and dynamic malware features to perform malware ascription. ECS Transactions, 107(1), 3187–3198. https://doi.org/10.1149/10701.3187ecst

Station X. (2025). Malware statistics. StationX. https://www.stationx.net/malware-statistics/. Retrieved February 3, 2025.

Syeda, D., & Asghar, M. (2024). Dynamic malware classification and API categorization of Windows Portable Executable files using machine learning. Applied Sciences, 14(3), 1015. https://doi.org/10.3390/app14031015

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2018). Graph attention networks. arXiv. https://arxiv.org/abs/1710.10903

Virus Total. (2025). Virustotal-free online virus, malware, and URL scanner. Virustotal. https://www.virustotal.com/en. Retrieved February 3, 2025.

Xu, Y., & Chen, Z. (2023). Family classification based on tree representations for malware. Proceedings of the 14th ACM SIGOPS Asia-Pacific Workshop on Systems, 65–71. https://doi.org/10.1145/3609510.3609818

Yau, L., Lam, Y., Lokesh, A., Gupta, P., Lim, J., Singh, I., Loo, J., Ngo, M., Teo, S., & Truong-Huu, T. (2023). A novel feature vector for AI-assisted Windows malware detection. 2023 IEEE International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference on Cloud and Big Data Computing, International Conference on Cyber Science and Technology Congress

(DASC/PiCom/CBDCom/CyberSciTech), 355–361. https://doi.org/10.1109/DASC/PiCom/CBDCom/Cy59711.2023.10361451

Zhang, S., Wu, J., Zhang, M., & Yang, W. (2023). Dynamic malware analysis based on API sequence semantic fusion. Applied Sciences, 13(11), 6526. https://doi.org/10.3390/app13116526

Биографија аутора

Славиша Ж. Илић рођен је 1982. године у селу Чукојевцу, у околини Краљева. Завршио је Војну гимназију 2001. године са одличним успехом, и петогодишње студије на Војној академији, смер Информатика, 2006. године са просеком 8,38.

Радио је на различитим позицијама у Војсци Србије и Министарству одбране као дипломирани инжењер информатике и асистент на Катедри информатике на Војној академији.

Поносни је супруг и отац.

Изјава о ауторству

Име и презиме студента докторских студија: Славиша Илић

Број индекса: 2P1/3/20

Изјављујем

да је докторска дисертација под насловом:

Унапређење динамичке анализе злонамерног софтвера засновано на проширеном скупу обележја и машинском учењу

- 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 и
- да нисам кршио ауторска права и користио интелектуалну својину других лица.

У Београду, 05. 03. 2025. год. **Потпис студента докторских студија**

Изјава о истоветности штампане и електронске верзије докторске дисертације

Име и презиме студента докторских студија: Славиша Илић

Број индекса 2Р1/3/20

Студијски програм Докторске студије информатике

Наслов дисертације Унапређење динамичке анализе злонамерног софтвера засновано на проширеном скупу обележја и машинском учењу

Ментор професор доктор Милан Гњатовић

Изјављујем да је штампана верзија мог докторског рада истоветна електронској верзији 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 и 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, 05. 03. 2025. год. **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:

Унапређење динамичке анализе злонамерног софтвера засновано на проширеном скупу обележја и машинском учењу

, која је моје ауторско дело.

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

Моју докторску дисертацију похрањену у Дигиталном 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, 05. 03. 2025. год. **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 одређен 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.