НАЦИОНАЛЕН ВОЕНЕН УНИВЕРСИТЕТ „ВАСИЛ ЛЕВСКИ”

5006 гр. Велико Търново, бул. „България” №76

ФАКУЛТЕТ „АРТИЛЕРИЯ, ПВО И КИС“

Катедра „Комуникационни мрежи и системи“

капитан инженер Радостин Стефанов Димов

ИЗСЛЕДВАНЕ НА СИГУРНОСТТА НА КОГНИТИВНИ КОМУНИКАЦИОННИ СИСТЕМИ

А В Т О Р Е Ф Е Р А Т

на дисертационен труд

за присъждане на образователна и научна степен „доктор“

Област на висше образование 5. „Технически науки“
Професионално направление 5.3 „Комуникационна и компютърна техника“
Докторска програма „Киберсигурност“

Научен ръководител: проф. д.н. инж. Жанета Николова Савова

2025 г.

гр. Шумен

Дисертационния труд се състои от 175 листа

Основен текст – 156 листа

Брой на приложенията – 2

Брой на литературните източници – 75

Брой на публикациите по дисертацията – 3

Защитата на дисертационния труд ще се състои на __.__.2025 г. от __.__ ч. в _____ на _____.

Материалите по защитата са на разположение на интересуващите се в _____ на _____, тел. _____.

Дисертационният труд е обсъден на катедрен съвет на катедра „Комуникационни мрежи и системи" при факултет „Артилерия, ПВО и КИС" на Национален военен университет „Васил Левски" – Велико Търново на ___.___.2025 г., и е насочен за защита пред научно жури за придобиване на образователната и научна степен "Доктор".

Докторантът е асистент в катедра „Компютърни системи и технологии" при факултет „Артилерия, ПВО и КИС" на Национален военен университет „Васил Левски" – Велико Търново. Основните изследвания по дисертационния труд са проведени в симулирана среда изградена чрез специализирана платформа "Cyber Range" в рамките на Център за съхранение на данни при факултет „Артилерия, ПВО и КИС".

Автор: капитан инж. Радостин Стефанов Димов

Тема: „Изследване на сигурността на когнитивни комуникационни системи".

Тираж __

Отпечатан на ___.___. 20__ г.

Издателски комплекс на Национален военен университет „Васил Левски

I. ОБЩА ХАРАКТЕРСИТИКА НА ДИСЕРТАЦИОННИЯТ ТРУД

Скоростното развитие на комуникационните технологии, доведе до създаването на когнитивни комуникационни мрежи, които използват изкуствен интелект и алгоритми за машинно обучение, за да оптимизират производителността на мрежата и да подобрят потребителското изживяване. Въпреки това, нарастващата сложност и взаимосвързаност на тези мрежи породиха опасения относно тяхната сигурност, тъй като използват по-сложни и потенциално незащитени технологии, и често са обект на кибератаки, които могат да нарушат тяхната работа или да компрометират чувствителна информация.

В последните десетилетия се забелязва тенденция на увеличаване на научните и практически изследвания в областта на тестване и оценка на сигурността на когнитивни комуникационни системи, в частност на 5G мобилни мрежи. По-голяма част от тях са насочени към изследване на дефанзивната сигурност и малка част засягат прилагането на офанзивна сигурност с цел тестване на сигурността на когнитивни комуникационни системи и в резултат на това отстраняване на техните слабости. Провеждането на реални експерименти в производствени мрежи е ограничено поради риска от нарушаване на работата на услуги и възможни киберинциденти.

За решаването на този проблем се използват симулационни платформи от ново поколение – Cyber Range. Платформите Cyber Range са сложни продукти, изискващи огромни ресурси, които да обезпечат нормалното функциониране на системата при създаването на сложни архитектури на компютърни мрежи и симулирането на множество офанзивни техники, при множество едновременно работещи екипи. От друга страна съществуващите решения с отворен код често са ограничени по функционалност и трудно приложими в специфични мрежови архитектури. В резултат на това възниква необходимост от изграждане на специализиран киберполигон, предназначен да предоставя безопасна и реалистична тестова среда за симулация и анализ на кибератаки и механизми за защита, без да се засяга работата на реалните системи.

Актуалността на темата произтича от широкото разпространение на когнитивни комуникационни системи и тяхното приложение в различни сфери на обществения живот, особено в действащия към 2025 г. стандарт за клетъчни комуникационни мрежи от пето поколение 5G – 3GPP Rel. 20.

Обект на изследване са когнитивните комуникационни системи и тяхната сигурност, разглеждани в контекста на 5G архитектурата, с акцент върху идентифицирането на потенциални тактики, техники и процедури, използвани от злонамерени групи, както и върху практическата им емулация в контролирана среда на специализиран киберполигон.

Основната цел на дисертационният труд е повишаване на киберустойчивостта и защита на когнитивните комуникационни мрежи от дигитални заплахи посредством идентифициране и отстраняване на техните слабости, преди използването им от злонамерени организации и лица

Допълнителни:

1. Подпомагане на възможностите на научни работници, индустрия и държавни институции за провеждане на научно-приложни изследвания и иновации посредством симулиране и емулиране на научни разработки и внедряване на нови технологии за киберсигурност.
2. Създаване на възможност за обучение и повишаване на уменията чрез образователни програми, тренировки и симулации, с цел подготовка на специалисти в областта на киберсигурността.

В съответствие с целта на дисертационният труд са формулирани следните **задачи**:

1. Анализ на сигурността в комуникационни когнитивни системи и в частност на 5G мобилни мрежи.
2. Анализ и синтез на модели за сигурност на когнитивни комуникационни мрежи.
3. Да се изгради специализиран киберполигон, който осигурява възможности за идентифициране на слабости в симулираната инфраструктура, подобряване на координацията между участващите екипи, оценка на реакцията при инциденти, обучение и сертификация на персонала, както и симулация на сигурността на комуникационна мобилна мрежа от 5-то поколение.
4. Провеждане на симулационни експерименти и анализ на резултатите.

Отчитайки обширността на темата в дисертационния труд са приети следните **ограничения**:

1. Изследването да обхваща киберсигурността на клетъчните комуникационни мрежи от пето поколение.
2. Проведените експерименти и симулации да се реализират в емулирана 5G среда, изградена на базата на платформи с отворен код.

Поставените ограничения не омаловажават научно-изследователските задачи в дисертационния труд и могат да бъдат използвани в научни разработки и изследвания за развитие и надграждане на реалната киберсигурност в клетъчни системи от пето поколение.

II. СТРУКТУРА НА ДИСЕРТАЦИОННИЯТ ТРУД

Дисертационният труд е структуриран в увод, четири глави, общи изводи, приноси, библиографска справка и 2 приложения. Трудът съдържа 21 таблици, 132 формули и 57 фигури. Цитирани са 75 литературни източника.

В увода се обосновава актуалността на темата и практико-приложния й характер. Дефинирани са целта и основните задачи, посочени са ограниченията, при които се извършва изследването.

В първа глава е решена първа научноизследователска задача, като са разгледани архитектурните особености на 5G мрежата, включително разделянето на контролна и потребителска равнина, виртуализацията на мрежовите функции, както и концепцията за услуго-базирана архитектура. Изследвани са наличните уязвимости за операционни системи на потребителски терминали в периода 2018 – 2024 г. Извършен е литературен обзор въз основна на проучване в научни бази данни, обхващащо разработки, свързани със сигурността на когнитивните комуникационни мрежи за периода 2015 – 2024 г. и клетъчните мрежи от пето поколение за периода 2019 – 2024 г.

Във втора глава е решена втора научноизследователска задача чрез анализ на съществуващи модели за офанзивна и дефанзивна сигурност и представяне на разработените в дисертацията модели на комуникационна мрежа от пето поколение, модел за разузнаване на киберзаплахи и модел на противника.

В трета глава е решена трета научноизследователска задача чрез реализиране на специализирана платформа за създаване на симулационни среди и отработване на офанзивни и дефанзивни кибероперации. Платформата включва потребителски интерфейс, мениджър на способности и блок за виртуализация и автоматизация. Предложен е модел на сценариите, който формализира процеса по създаване на инфраструктурни среди. Създадена е симулационна среда за тестване и оценка на сигурността на комуникационна мрежа от пето поколение.

В четвърта глава е решена четвърта научноизследователска задача, като са анализирани получените резултати при емулация на противникови техники и процедури, и е предложена система за киберсигурност на комуникационни мрежи от пето поколение. Обоснован е изборът на софтуер и средства за защита на компонентите на изследваната 5G инфраструктурна мрежа.

III. КРАТКО СЪДЪРЖАНИЕ НА ДИСЕРТАЦИОННИЯ ТРУД

Глава 1 – Анализ на сигурността в когнитивни комуникационни системи.

За решаването на първа основна задача от дисертационният труд, в първа глава са дефинирани следните подзадачи:

1. Анализ на когнитивното радио и неговите характеристики, като интелигентна комуникационна система.
2. Анализ на 5G мрежовата инфраструктура и системите за нейната емулация.
3. Анализ на сигурността в клетъчните системи от пето поколение като се изведат основните предизвикателства и направления за научно-приложни разработки в изследваната област.

1.1. Когнитивно радио

За решаването на задача 1.1 в параграфа е извършен анализ на концепцията за когнитивно радио като основа на когнитивните комуникационни системи. Извършен анализ на когнитивното

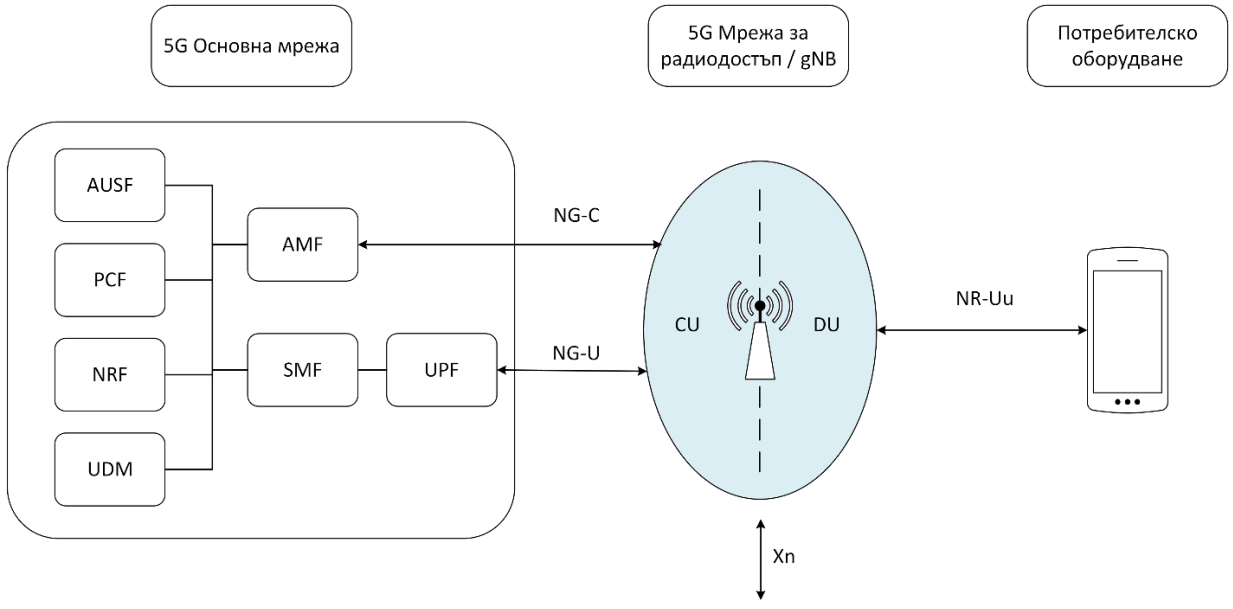
радио и неговото приложение в различни съвременни технологии. Разгледан е проблемът с ограничените спектрални ресурси и е обоснована необходимостта от използване на динамични методи за управление на радиочестотния спектър. Проследена е еволюцията от софтуерно-дефинирани радио системи към когнитивни радио системи, като е подчертано, че последните притежават способност за самонастройване и адаптация спрямо околната среда. Проследени са основните функционални етапи в работата на когнитивното радио, включително наблюдение на спектъра, откриване на свободни честотни ленти, вземане на решения и преконфигурация на параметрите на предаване. Изведена е ролята на когнитивното радио за съвместното използване на честотния спектър между първични и вторични потребители без нарушаване на качеството на комуникация на лицензираните системи.

Разгледана е архитектурата на когнитивните мрежи, състояща се от първична и вторична мрежа, като е посочена отговорността на вторичните потребители за мониторинг на средата и освобождаване на спектъра при повторна активност на първичните потребители. Направено е разграничение между централизирани и ad-hoc когнитивни мрежи и са анализирани механизмите за оперативна съвместимост между хетерогенни комуникационни системи.

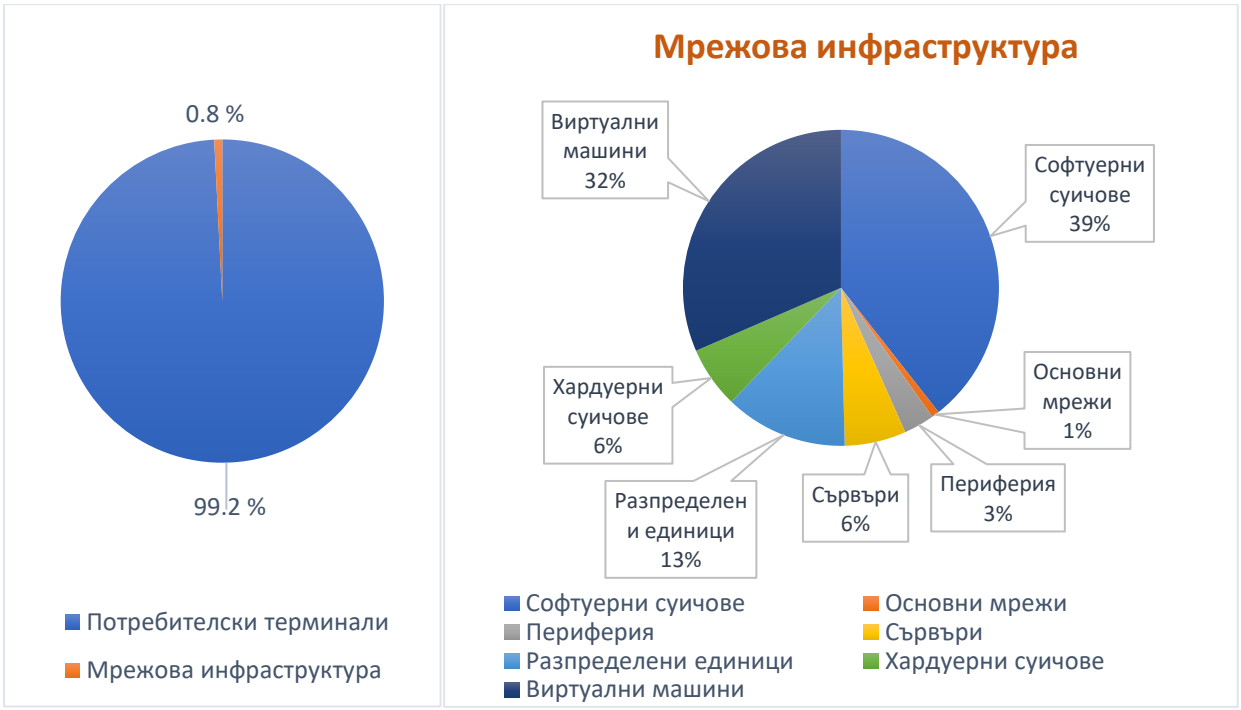
1.2. 5G мрежова архитектура

За решаването на задача 1.2 в параграфа е извършен анализ на архитектурата на клетъчните комуникационни мрежи от пето поколение. Разгледан е принципът на разделение между контролната и потребителската равнина, както и преминаването към услуго-базирана архитектура, при която отделните функции на основната мрежа комуникират помежду си чрез стандартизирани интерфейси. Обособени са основните мрежови функции в 5G Core, като е проследена тяхната роля в управлението на достъпа, мобилността, сесиите на потребителите, сигурността и обработката на трафика. Разгледана е и мрежата за радиодостъп 5G RAN, като е отчетено разделението на базовата станция на централизирана и разпределена част, с цел оптимизиране на обработката на сигналите и намаляване на латентността. Анализирани са ключови радиотехнологии, които подпомагат постигането на висок капацитет и качество на обслужване. Както е показано на фигура 1.5, в контекста на 5G, мрежовата архитектура се разпределя в няколко основни възела: основна мрежа, мрежа за радиодостъп, потребителско оборудване.

Според (Sanchez-Navarro, Bernabe, Alcaraz-Calero, & Wang, 2021), 99.2 % от възлите в 5G представляват потребителско оборудване. Също така, е отбелязано, че само 4% от мрежовата инфраструктура са физически машини в основната мрежа или периферията (фигура 1.6). Съвсем закономерно по-голямата част от кибератаките засягащи такъв тип мрежи са насочени именно към клиентите и техните крайни устройства. Следващата фигура визуализира разпределението на възлите и периферията в мрежовата топология на когнитивните мрежи.

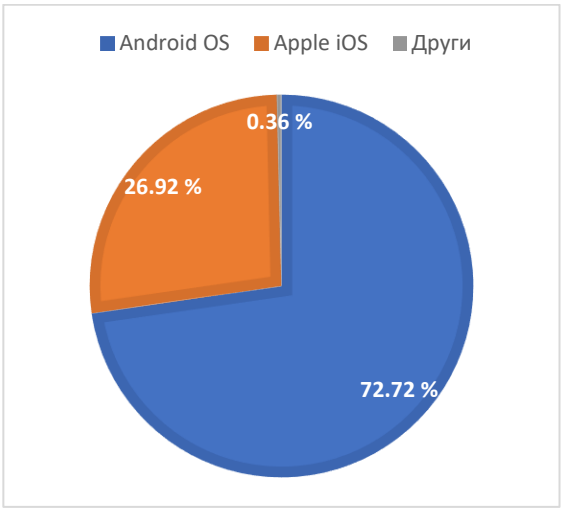


Фиг. 1.5. 5G Мрежова архитектура



Фиг. 1.6. Разпределение на мрежовата топология (Sanchez-Navarro, Bernabe, Alcaraz-Calero, & Wang, 2021)

Анализирана е ролята на потребителското оборудване като крайна точка на взаимодействие с мрежата. Посочено е, че поради доминиращия му дял в мрежовата среда и директното взаимодействие с радиоинтерфейса, именно потребителските устройства представляват значителен вектор на кибератаки и следователно изискват специално внимание при анализа на сигурността. Направено е проучване сред 250 респонденти по метода лице в лице, което потвърждава тези данни, като се забелязва, че Android се ползва на над 7 от всеки 10 смартфона, докато iOS има стабилно присъствие от близо 3 от 10 устройства. Резултатите са представени на фигура 1.7 и свидетелстват за ясно изразено пазарно разделение.



Фиг. 1.7. Пазарен дял на ОС на потребителско оборудване

Извършен сравнителен анализ на системи с отворен код за емулация на 5G инфраструктура. На тази основа е идентифицирано подходящо технологично решение за изграждане на експериментална среда, която да позволи симулиране и оценка на сигурността в условия, близки до реалните. Анализът е представен в таблица 1. Open5GS се откроява като балансирано решение за изграждане на емулирана 5G среда, съчетаваща функционалност, стабилност и удобство при внедряване. Платформата srsRAN осигурява цялостна поддръжка на 5G мрежа за радиодостъп и е подходяща при изграждането на реална хардуерно-базирана опитна постановка на 5G RAN.

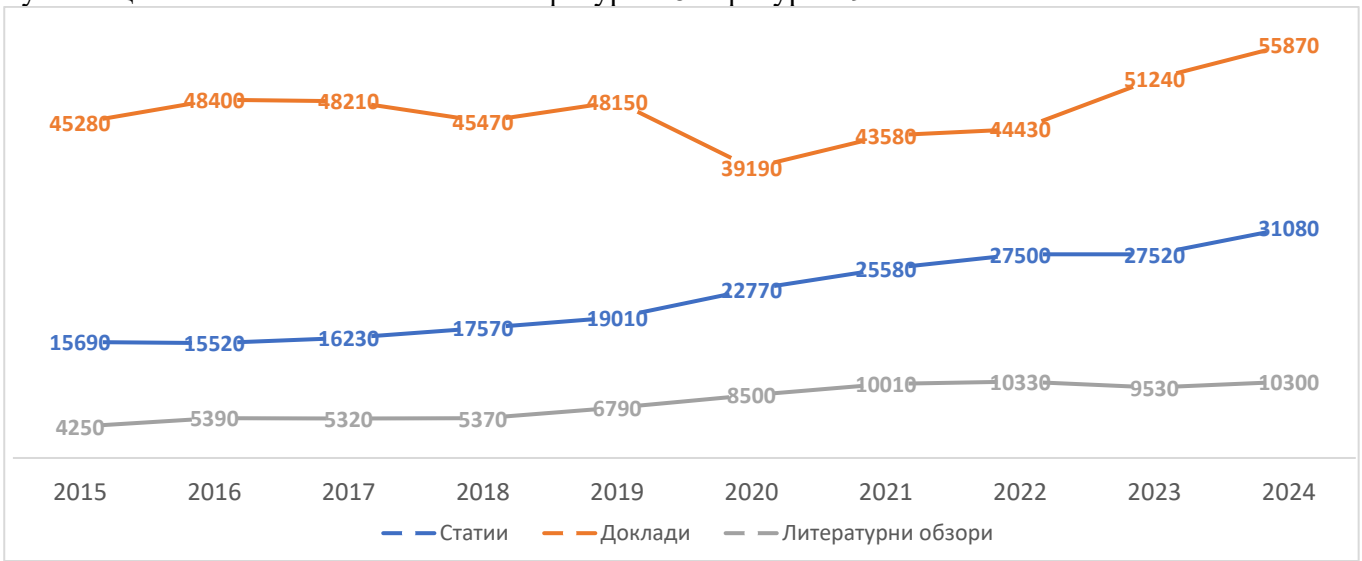
Платформата UERANSIM е подходяща за реализиране на симулационни среди, при липса на хардуерно оборудване. Освен това включва компонент 5G UE за симулация на потребителско оборудване и потребителска активност.

Таблица 1. Сравнителен анализ на системи за емулация на 5G мрежи

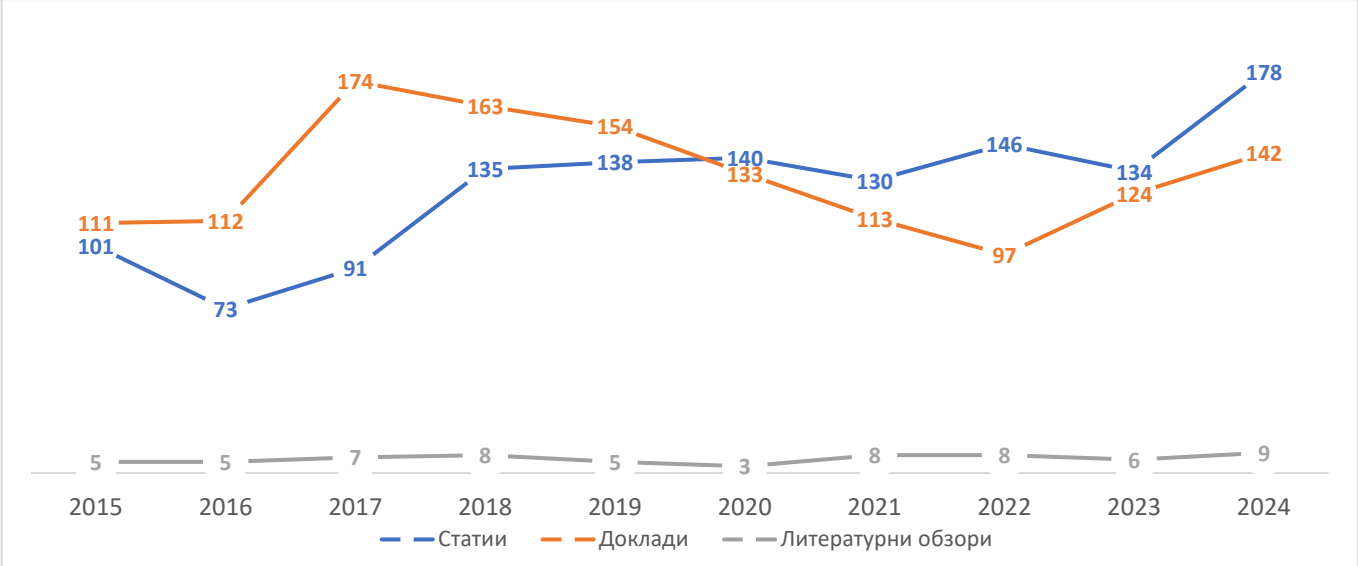
Основна мрежа 5G Core		
Характеристика	**Open5GS**	**Free5GC**
3GPP Rel.	Rel.17	Rel.15
Програмен език	C	Go
Цялостна поддържка на мрежови функции	✓	✓
Интеграция с RAN емулатори	✓	✓
Изграждане върху ВМ/Контейнер	✓	✓
Активна разработка (общност, GitHub активност)	✓	✓
Подробна документация	✓	
Подходящ за образователни и лабораторни среди	✓	✓
Лесен за инсталация и конфигурация	✓	
Приложим за киберполигон (тестова инфраструктура)	✓	✓
Мрежа за радиодостъп		
Характеристика	**srsRAN**	**UERANSIM**
Функция	Реална емулация	Софтуерен симулатор
Пълна поддръжка на 5G RAN	✓	частична
Програмен език	C++	C++
Емулация на gNB	✓	✓
Емулация на UE		✓
Съвместим с Open5GS и Free5GC	✓	✓
Активна разработка (общност, GitHub активност)	✓	✓
Лесен за конфигурация		✓

1.3. Анализ на сигурността в клетъчните мрежи от пето поколение

За решаването на задача 1.3 в параграфа е направен обзор на научни публикации, разглеждащи проблемите на сигурността в когнитивните комуникационни мрежи. За целта са изследвани базите данни от научни публикации на ResearchGate и Scopus, като са проследени разработките в изследваната област от 2015 г. до 2024 г. Резултатите са представени като брой публикации в абсолютни стойности на фигура 1.8 и фигура 1.9.



Фиг. 1.8. Брой научни публикации, разглеждащи сигурността на когнитивни мрежи – ResearchGate



Фиг. 1.9. Брой научни публикации, разглеждащи сигурността на когнитивни мрежи – Scopus

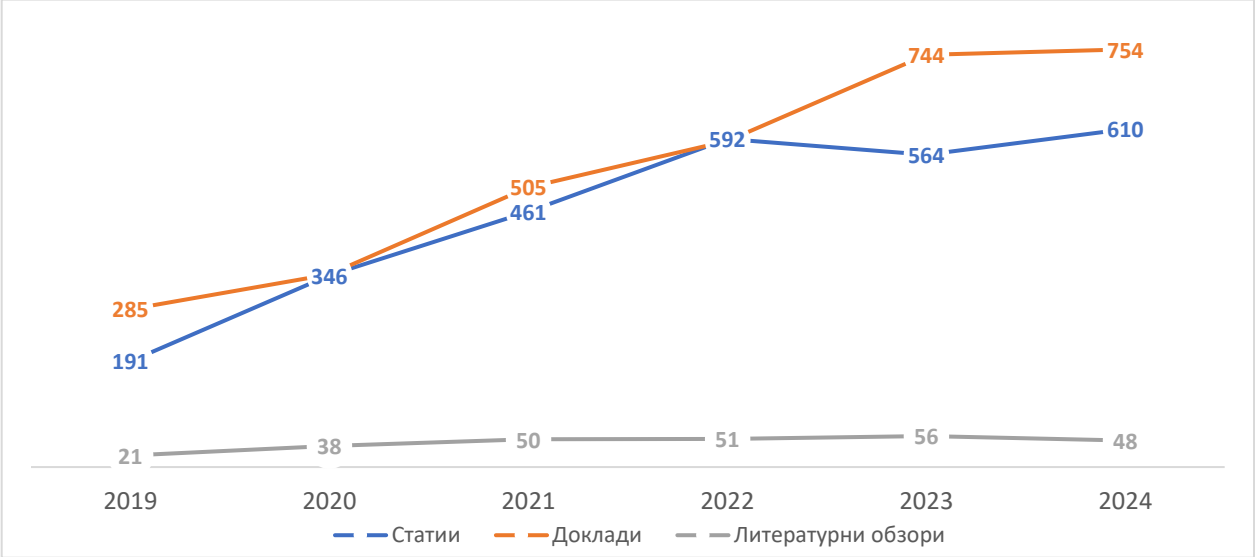
В изследвания период се наблюдава нарастване на броя на публикациите, като най-високите стойности са отчетени в последните 3 години за периода от 2022 до 2024 г. включително. Когнитивните мрежи намират широко приложение в технологиите за 5G мрежи. Към 2025 г. последният практико-приложен стандарт за клетъчни комуникации са клетъчните мрежи от пето поколение 5G. Предвид това е направено друго проучване относно научни публикации, разглеждащи проблема със сигурността в 5G. Резултатите са представени на фигура 1.10 и фигура 1.11.

Фиг. 1.10. Брой научни публикации, разглеждащи сигурността на 5G – ResearchGate

Резултатите показват ясно изразено нарастване на броя на публикациите, разглеждащи проблема със сигурността на клетъчните системи от пето поколение. Литературните източници от базите данни на ResearchGate и Scopus са сортирани по популярност на база брой преглеждания и цитирания, като са извлечени първите 30 източника за периода 2018-2024 г. Същите са класифицирани по две направления за изследване в киберсигурността– дефанзивна сигурност, офанзивна сигурност. Резултатите показват значителен превес в изследването на дефанзивна сигурност на когнитивните комуникационни мрежи и в частност на 5G мрежите (само 1 статия засяга само офанзивна сигурност, а 4 от тях и двете). Следователно съществува необходимост от задълбочаване на научно-приложните разработки в областта на офанзивната сигурност.

В (Angelogianni, Politis, Polvanesi, Pastor, & Xenakis, 2021) е направено проучване сред персонал на телекомуникационни оператори, доставчици и специалисти по киберсигурност, относно необходимостта от използването на симулирани среди за тестване на сигурността в комуникационни мрежи от пето поколение. От общо 60 участвали в проучването 46 (≈77%)

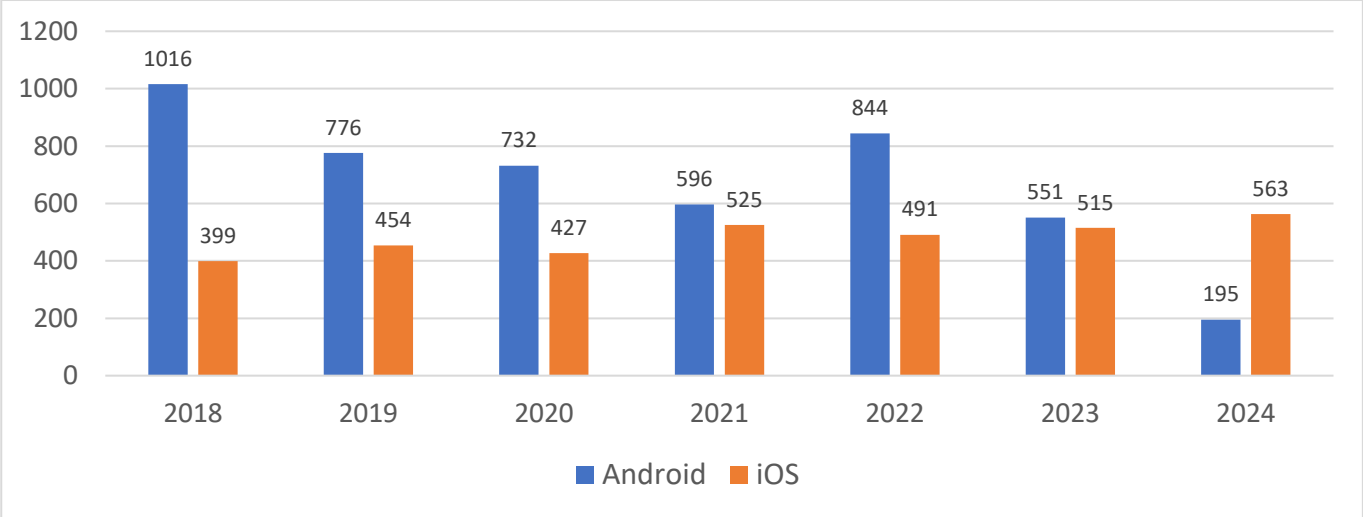
заявяват, че биха използвали специализирани платформи (киберполигони) на симулационни среди, за да тестват сигурността на своята 5G инфраструктура. Овен това 78% заявяват интерес към използването на такъв тип платформи за провеждане на обучения по киберсигурност на 5G.



Фиг. 1.11. Брой научни публикации, разглеждащи сигурността на 5G – Scopus

Въпреки, че изследваните научни публикации разглеждат различни предизвикателства и опасения за сигурността пред когнитивните комуникационни системи, в нито едно от тях не е направено проучване, чрез разузнаване на киберзаплахи, което да идентифицира злонамерените хакерски групировки таргетиращи сектора на телекомуникациите на територията на Европа и НАТО. Това би позволило създаването на обобщена картина на опериращите групировки, таргетиращи този специфичен сектор, географския регион, както и тактиките, техниките и процедурите, които използват.

Потребителското оборудване като смартфоните и таблетите с Android и iOS са най-разпространените крайни устройства и често попадат в центъра на злонамерени атаки. За да се оцени реалният риск, който тези платформи носят от гледна точка на уязвимости, в настоящото изследване е извършен анализ на данни от публичната база данни CVE. Резултатите от анализа са синтезирани и визуализирани на фигура 1.15, като графично представят динамиката на откритите уязвимости по години.



Фиг. 1.15. Разпределение на брой уязвимости за Android и iOS в периода 2018-2024 г.

1.4. Изводи

1. Когнитивното радио е ключова технология за мобилните комуникации 5G, 6G и IoT, където нуждата от гъвкав и ефективен спектрален достъп е критична.

2. Пазарният дял на операционните системи на потребителско оборудване в 5G мрежите показва превес на Android (72,72 %) спрямо Apple iOS (26,92 %) и само 0,36 % за други операционни системи.

3. Платформата Open5GS се откроява като балансирано решение за изграждане на емулирана 5G основна мрежа, като предоставя пълна функционална реализация на ядрото на 5G мрежата и контейнеризация, което я прави подходяща за изграждане на киберполигони и отработване на офанзивни и дефанзивни кибероперации.

4. От направения анализ на сигурността в когнитивни комуникационни мрежи и в частност в клетъчни 5G мрежи могат да се направят следните изводи:

(1)Броят на научно-приложните изследвания и разработки в разглежданите области в базите данни SCOPUS и ResearchGate нараства с всяка година за периода 2019-2024 г., като се откроява максимум през 2024 г: сигурност на когнитивни мрежи (SCOPUS – 329, ResearchGate - 97250) и сигурност на 5G (SCOPUS – 1412, ResearchGate - 111000).

(2)От извлечените първи 30 източника по популярност на база брой преглеждания и цитирания за периода 2018-2024 г. се наблюдава значителен превес в изследването на дефанзивна сигурност на когнитивните комуникационни мрежи и в частност на 5G мрежите (само 1 статия засяга само офанзивна сигурност, а 4 от тях и двете). Следователно съществува необходимост от задълбочаване на научно-приложните разработки в областта на офанзивната сигурност.

(3)Сред основните заплахи в 5G мрежите според доклада на ENISA с най-големи дялове се открояват; „Фишинг и социално инженерство“ – 31%; „Уязвимости в IoT устройства и потребителски терминали“ – 21% и „Неправилна конфигурация на 5G компоненти“ – 17%. Атаките срещу крайните потребители в мобилни мрежи, посредством фишинг, смишинг и вишинг, използват зловредни линкове към файлове за Андроид приложения APK, като метод за първоначален достъп.

(4)За периода 2018-2024 г. Android отчита по-голям брой регистрирани уязвимости с изключение на 2024 г., което се дължи на неговата отворена архитектура и широкото му приложение върху устройства от различни производители.

Глава 2 – Анализ и синтез на модели за сигурност на когнитивни комуникационни мрежи.

За решаването на втора основна задача от дисертационният труд, във втора глава са дефинирани следните подзадачи:

1. Анализ на модели за дефанзивна сигурност;
2. Анализ на модели за офанзивна сигурност;
3. Синтез на математически модели на киберзаплахи в когнитивна комуникационна мрежа, формализиращи компонентите на комуникационна когнитивна мрежа, хетерогенни източници на разузнавателна информация и поведението на противника.

2.1. Модели за дефанзивна сигурност

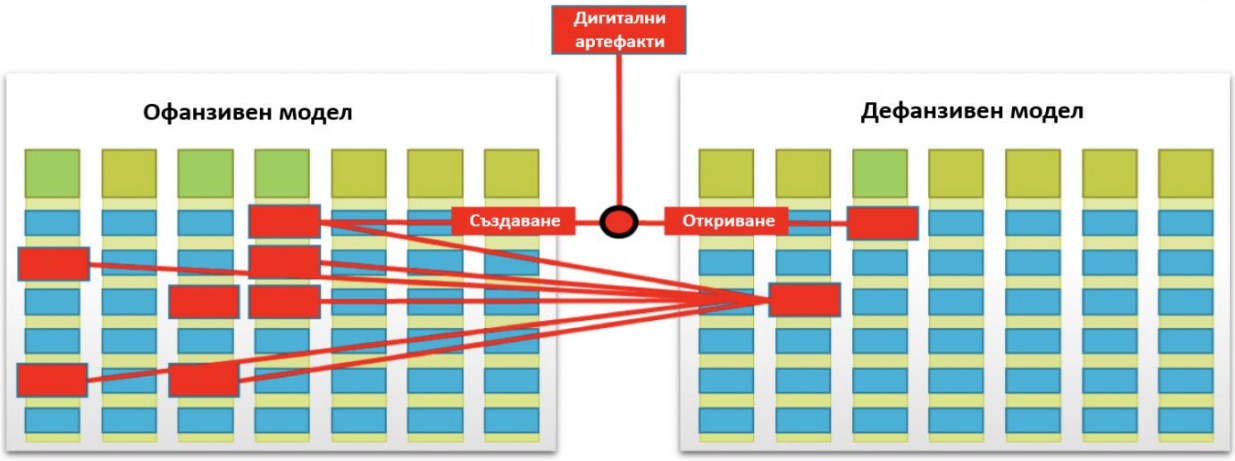
За решаването на задача 2.1 в параграфа за анализирани два модела за дефанзивна сигурност – MITRE D3FEND и NIST рамката за киберсигурност, като са разгледани техните основни компоненти, структура и област на приложение. На тази основа е извършено сравнение между моделите и е оценена тяхната пригодност за използване в среда с висока динамика на трафика и адаптивност, характерна за когнитивните системи.

2.1.1. MITRE D3FEND

Извършен е анализ на структурата и компонентите на рамката D3FEND. Изследвани са основните елементи на модела — графиката на знанието, семантичният модел и потребителският интерфейс, и е описано как те формализират защитните технологии и зависимости между тях. Разгледана е възможността за интеграция на модела с офанзивни таксономии като MITRE ATT&CK.

Цифровите артефакти дефинират концептуалния обхват на D3FEND. Например, политика за силни пароли попада в този обхват, тъй като пряко влияе върху конфигурационната база на една система. За разлика от това, програми за обучение на служители по киберсигурност не се

разглеждат, тъй като не взаимодействат с цифрови артефакти. Когато атакуващ извършва действия – като въвеждане на команди, търсене на информация или създаване на зловреден код – той създава цифрови артефакти. Същото важи за защитните действия: анализ, наблюдение и откриване също взаимодействат с артефакти, макар и от различна страна на конфликта. Фигура 2.4 илюстрира тази взаимовръзка между офанзивни и дефанзивни техники чрез цифрови артефакти.



Фиг. 2.4. Съпоставяне на офанзивни и дефанзивни техники по артефакти

За да се съпоставят ефективно атакуващи и защитни подходи, е необходимо да се дефинират детайлно връзките между тях. D3FEND използва цифровите артефакти като централна ос на моделиране, към която се асоциират и двата типа техники. Тези асоциации могат да бъдат от общ тип („свързан с“) или конкретизирани чрез релации като „генерира“, „изпълнява“, „анализира“, „достъпва“, „инсталира“. D3FEND разширява този модел и с връзки като „наблюдава“, „открива“ или „противодейства“, което позволява дефиниране на логически връзки без необходимост от ръчно съпоставяне на всяка техника. Структурата на модела позволява количествено да се измери покритието на офанзивна техника от дефанзивна. Нека:

- $O = \{o_1, o_2, \dots, o_m\}$ е множеството от офанзивни техники;
- $D = \{d_1, d_2, \dots, d_n\}$ е множеството от дефанзивни техники;
- $A = \{a_1, a_2, \dots, a_k\}$ е множество от артефакти.

Могат да се дефинират функциите:

$$P_O(o_i, a_j) \in \{0,1\}, \tag{2.1}$$

където P_O определя дали офанзивната техника $o_i \in O$ използва артефакта $a_j \in A$.

$$P_D(d_l, a_j) \in \{0,1\}, \tag{2.2}$$

където P_D определя дали дефанзивната техника $d_l \in D$ наблюдава и анализира артефакта $a_j \in A$. Съпоставянето на техники по артефакти се дефинира като:

$$M_{i,l} = \frac{\sum_{j=1}^{k} P_O(o_i, a_j) \,.\, P_D(d_l, a_j)}{\sum_{j=1}^{k} P_O(o_i, a_j)}, \tag{2.3}$$

където:

$$M_{i,l} \in [0,1] \tag{2.4}$$

Числителят измерва броя на общите артефакти, върху които едновременно действа офанзивната техника и наблюдава защитната техника. Знаменателят нормализира резултата спрямо всички артефакти, използвани от атакуващия. Колкото по-близо е стойността на метриката до 1, толкова по-добре техниката d_l защитава срещу o_i. Това става, чрез измерване доколко защитна техника d_l обхваща артефактите, които атакуваща техника o_i използва. С цел количествена оценка на колективната устойчивост на системата спрямо дадена офанзивна техника, може да бъде дефинирана метрика за средно покритие, която измерва степента на съвкупна защитна ефективност от всички налични дефанзивни техники. Тази метрика се базира на вече въведената стойност $M_{i,l}$, която отразява покритието на атакуваща техника o_i от конкретна защитна техника d_l, чрез общите артефакти, които се използват и наблюдават. Средното покритие се дефинира по следния начин:

$$\bar{M}_i = \frac{1}{n}\sum_{l=1}^{n} M_{i,l}\,, \tag{2.6}$$

където $\bar{M}_i \in [0,1]$ е средно покритие на офанзивната техника o_i и n е броят на наличните защитни техники в множеството $D = \{d_1, d_2, \dots, d_n\}$. При това може да се изведе ниво на пропуск в защитата, като се измерва каква част от артефактите, използвани от дадена атакуваща техника o_i, остават ненаблюдавани от защитната техника d_l. Това предоставя директна количествена оценка за дефицита в покритието от страна на защитния механизъм.

$$G_{i,l} = 1 - M_{i,l}, \tag{2.7}$$

където $G_{i,l} \in [0,1]$ представлява нивото на пропуск в защитата. Стойности близки до 0 (напр. $G_{i,l} = 0.00$) означават, че съответната защитна техника напълно покрива атаката. Стойности равни на 1.00 означават пълен пропуск, т.е. защитната техника не наблюдава нито един от артефактите, използвани от съответната атака. Важно е да се оцени доколко системата осигурява видимост върху критичните артефакти, независимо от конкретните техники. За целта може да се изчисли общата защитна видимост C_A (покритието на атакувани артефакти) спрямо реалните заплахи:

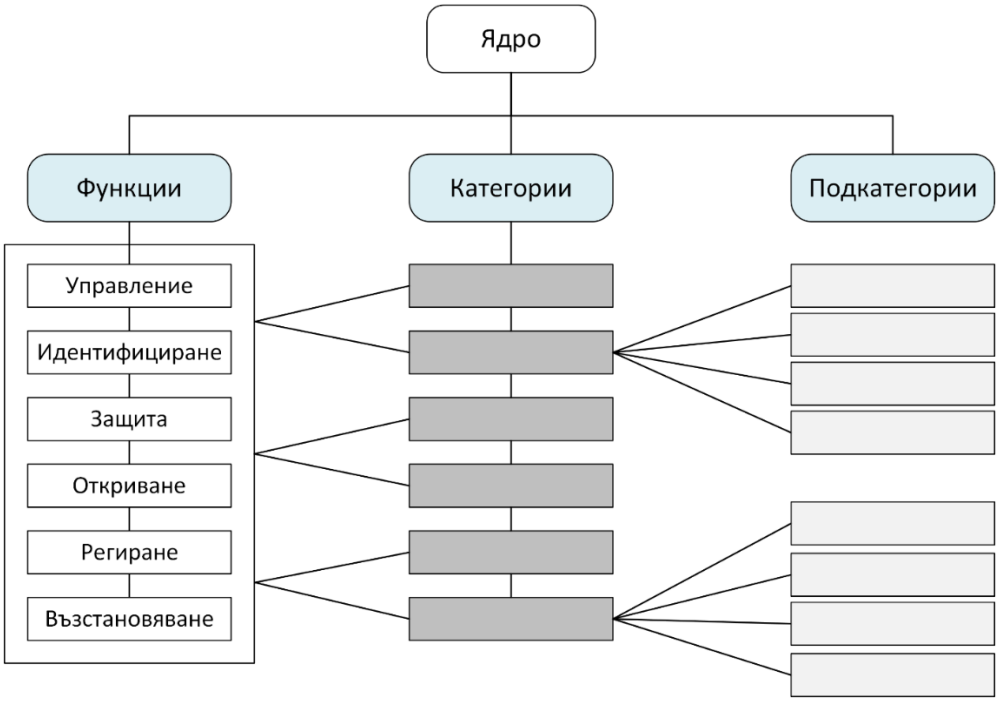
$$C_A = \frac{\sum_{j=1}^{k} 1\left(\sum_{i=1}^{m} P_O(o_i, a_j) > 0 \wedge \sum_{l=1}^{n} P_D(d_l, a_j) > 0\right)}{\sum_{j=1}^{k} 1\left(\sum_{i=1}^{m} P_O(o_i, a_j) > 0\right)}, \tag{2.8}$$

където $C_A \in [0,1]$ - отразява дяла от реално застрашените артефакти, които попадат под обхвата на наблюдение от страна на системата.

2.1.2. NIST модел за киберсигурност

Целта на моделът е организациите да управляват и намаляват рисковете, свързани с киберсигурността. При необходимост начинът, по който организациите прилагат модела, варира. NIST моделът за киберсигурност включва следните компоненти: *Ядро, Организационен профил, Нива*.

Ядрото на модела е набор от резултати, свързани с киберсигурността, организирани по следния начин: *Функция*, *Категория* и *Подкатегория* (фигура 2.6). Основните функции на ядрото – *управление, идентифициране, защита, откриване, реагиране* и *възстановяване*, организират резултатите, свързани с киберсигурността, на най-високо ниво.



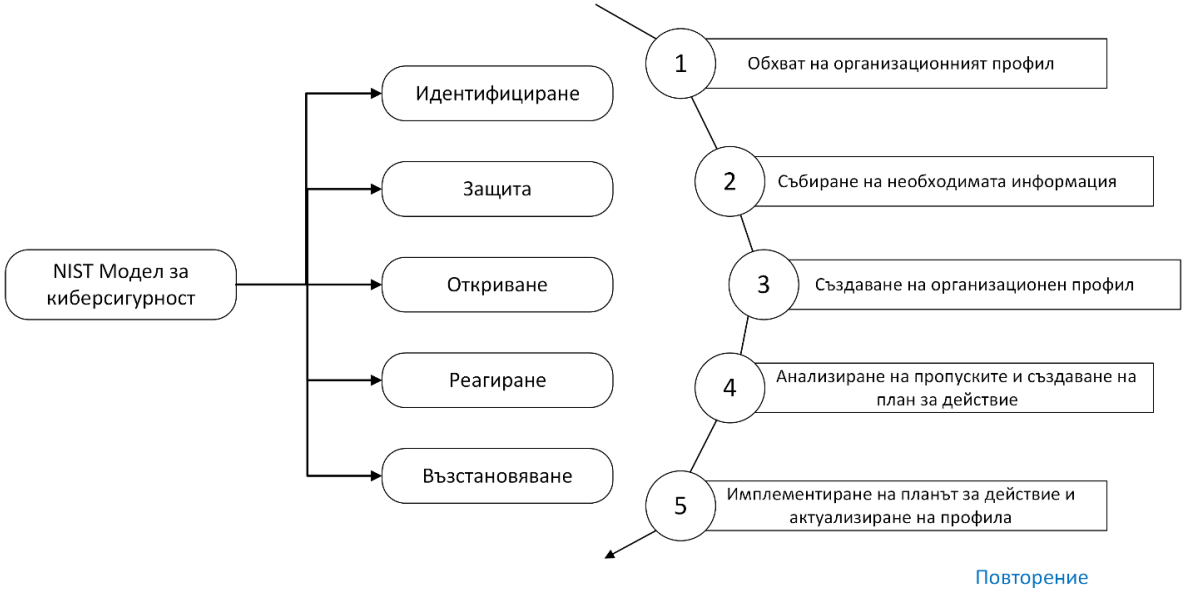
Фиг. 2.6. Ядро на NIST модел за киберсигурност

Организационният профил описва текущото и/или целевото състояние на киберсигурността в една организация, съобразено с резултатите, дефинирани в ядрото. Всеки организационен профил включва едно или и двете от следните:

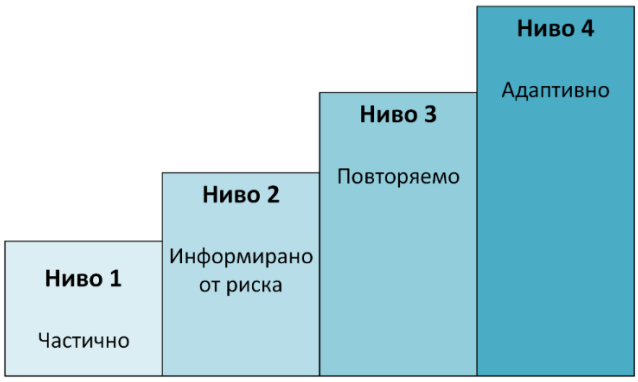
1. *Текущ профил* – указва резултатите, които организацията в момента постига, както и характеристиките или степента на постигане на всеки резултат.
2. *Целеви профил* – указва желаните резултати, които организацията е избрала и приоритизирала, за да постигне своите цели в управлението на киберрисковете.

Стъпките, показани на фигура 2.8 илюстрират един от начините, по които една организация може да използва *Организационния профил*, за да подпомогне процеса на непрекъснато подобряване на своята киберсигурност.

Нивата, както са показани на фигура 2.9, отразяват управленските практики за киберсигурност на организацията в четири степени: Частично (Ниво 1), Информирано от риска (Ниво 2), Повторяемо (Ниво 3) и Адаптивно (Ниво 4). Нивата описват прогресия от неформални, реактивни подходи към подходи, които са адаптивни, основани на анализ на риска и непрекъснато се усъвършенстват. Изборът на *Ниво* помага да се определи общата насока, по която организацията ще управлява своите киберрискове.



Фиг. 2.8. Стъпки за създаване и използване на организационен профил



Фиг. 2.9. Нива в NIST модела за киберсигурност

2.1.3. Сравнителен анализ на модели за дефанзивна сигурност

Сравнителен анализ на двата по-горе разгледани модела за дефанзивна сигурност по седем основни техни характеристики, цел, фокус, структура, ниво на абстракция, приложимост, интеграция с други модели и целеви резултат, е представен в таблица 5. От направения сравнителен анализ могат да се направят следните изводи:

1. NIST е подходящ за организационно ниво, където се вземат решения за управление на риска, създаване на политики и стратегическо планиране. Той е широко използван в регулаторна среда, при одити и за съответствие с изисквания.
2. MITRE D3FEND е подходящ за технически екипи, които се занимават с внедряване на конкретни защитни механизми, анализ на атаки и изграждане на архитектура за сигурност.

Той е полезен при оперативна защита, идентифициране на заплахи и изграждане на защитни стратегии, базирани на реални техники.

3. Комбинирането на двата модела може да доведе до по-ефективна киберсигурност, като NIST CSF моделира рамката и управлението, а MITRE D3FEND – техническото изпълнение и защита.

Таблица 5. Сравнителен анализ на модели за дефанзивна сигурност

Характеристика	**MITRE D3FEND**	**NIST Модел за киберсигурност**
Цел	Описва техники за защита и противодействие на атаки	Предоставя рамка за управление на риска и подобряване на киберсигурността
Фокус	Тактически – насочен към конкретни защитни действия	Стратегически – насочен към процеси и управление
Структура	Онтология от защитни техники, свързани с MITRE ATT&CK	Функции: *Идентифициране*, *Защита*, *Откриване*, *Отговор*, *Възстановяване*
Ниво на абстракция	Ниско – технически детайли за защитни механизми	Високо – организационни и управленски нива
Приложимост	За инженери по сигурността, анализатори, разработчици	За ръководители, мениджъри, консултанти по сигурност
Интеграция с други модели	Свързан с MITRE ATT&CK за съпоставяне на атаки и защити	Може да се интегрира с ISO 27001, COBIT, и други стандарти
Целеви резултат	Подобряване на защитните способности чрез конкретни техники	Подобряване на киберустойчивостта и управление на риска

2.2. Модели за офанзивна сигурност

За решаването на задача 2.2 е извършен анализ на модели за офанзивна сигурност, които описват поведението на противника и начина, по който се реализират кибератаки в реални условия. За целта са разгледани два широко приети тактически модела – MITRE ATT&CK и моделът „Верига за кибератаки“, Cyber Kill Chain. Анализът е насочен към техните структурни характеристики, логика на изграждане, ниво на детайлност и приложимост при моделиране и симулиране на атаки.

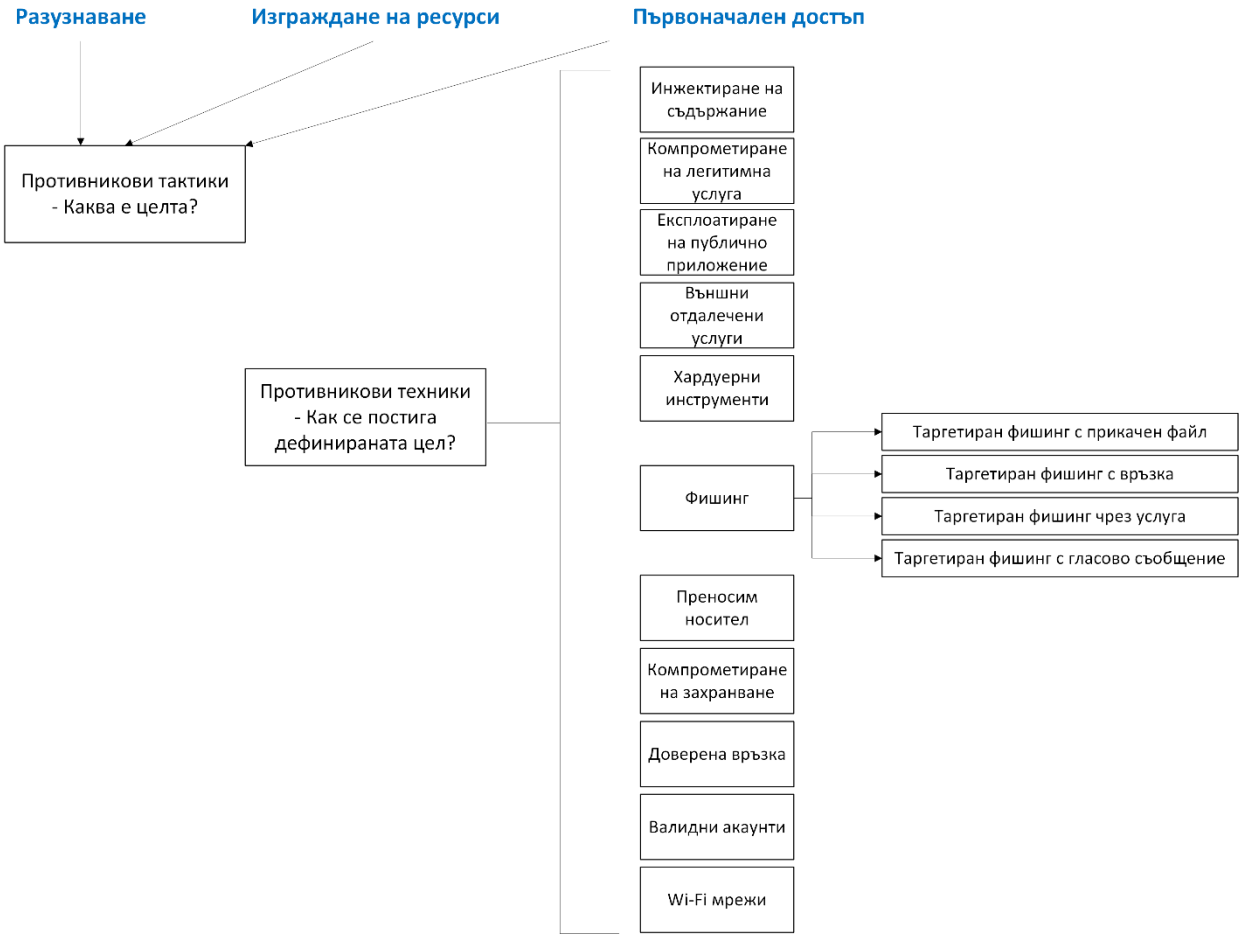
2.2.1. MITRE ATT&CK

MITRE ATT&CK е глобално достъпна база знания, съдържаща тактики, техники и процедури, Tactics, Techniques and Procedures (TTPs), които представят действията, които противниците могат да извършат, за да постигнат своите цели (MITRE Corporation, n.d.). В рамките на ATT&CK, тактиките се третират като „етикети“, с които техниките или подтехниките могат да бъдат свързани, в зависимост от целите, които могат да бъдат постигнати чрез тяхното използване. Всяка тактика съдържа определение, описващо нейната категория и служи като насока за това, какви техники следва да бъдат включени в нея. Така описаната структура е представена на фигура 2.12.

Техниките представляват „как“ противникът постига определена тактическа цел чрез извършване на конкретно действие. Подтехниките допълнително диференцират поведението, описано в техниките, в по-специфични форми.

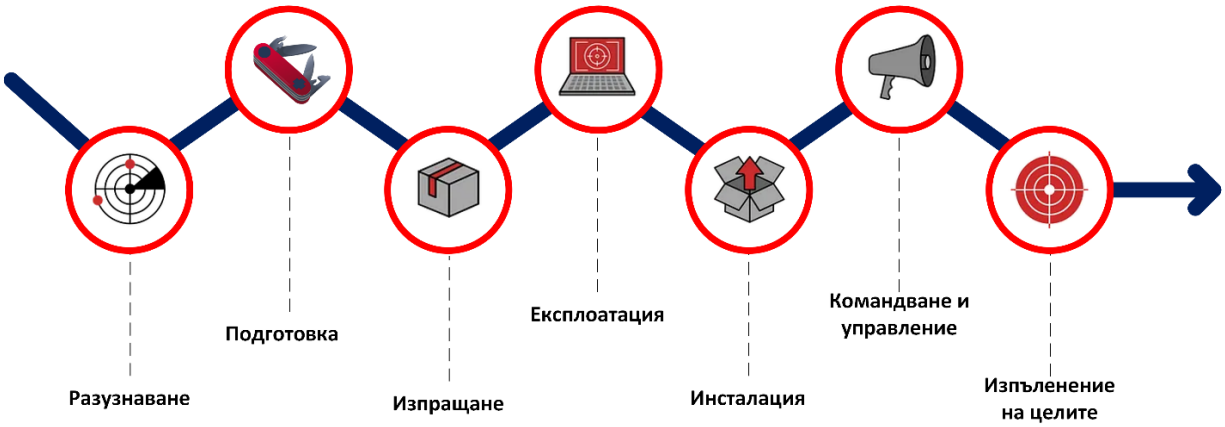
Процедурите са друг съществен компонент от концепцията TTP. Процедурите представляват конкретната реализация, използвана от противници при прилагане на дадена техника или подтехника. Процедурите често включват и конкретни инструменти, чрез които се извършват действията.

Известни противници, които се проследяват от публични и частни организации и за които се докладва в доклади за разузнаване на заплахи, се отразяват в MITRE ATT&CK чрез обекта - *Група*. Групите се дефинират като именувани набори от прониквания, заплахи, документирани злонамерени групировки или кампании, които обикновено представляват целенасочена и продължителна злонамерена активност.



Фиг. 2.12. Структура на тактики и техники

2.2.2. Верига за кибератаки



Фиг. 2.13. Верига за кибератаки

На фигура 2.13 е показана методологията, по която работи веригата (Lockheed Martin, n.d.) Моделът разделя кибератаките в седем етапа (фази).

1. *Разузнаване* - в първия етап нападателят събира информация за целевата система или организация.
2. *Подготовка* - след като е извършено подходящо разузнаване, нападателят създава подходящ зловреден софтуер или модифицира съществуващ, в съответствие с уязвимостите на целевата система.
3. *Изпращане* - когато зловреднят софтуер е готов, той се изпраща към целта – чрез имейл, линк, универсалната серийна шина, Universal Serial Bus (USB) устройство и т.н.
4. *Експлоатация* - при изпълнение, зловреднят код експлоатира уязвимост в системата на жертвата, за получаване на първоначален достъп.

5. *Инсталация* - успешната експлоатация позволява на нападателя да инсталира задна врата (backdoor) или друг тип зловреден софтуер, с цел поддържане на постоянен достъп.
6. *Командване и управление* - чрез инсталираната задна врата, нападателят може да комуникира с компрометираната система и да я управлява дистанционно.
7. *Изпълнение на целите* - в последната фаза нападателят изпълнява крайните си цели – например източване на данни, разпространение към други системи или саботаж.

2.2.3. Сравнителен анализ на модели за офанзивна сигурност

Сравнителен анализ на двата по-горе разгледани модела за офанзивна сигурност, *MITRE ATT&CK* и *Cyber Kill Chain*, е представен в таблица 6.

Таблица 6. Сравнителен анализ на модели за офанзивна сигурност

Характеристика	**MITRE ATT&CK**	**NIST Модел за киберсигурност**
Цел	Каталогизира техники, използвани от атакуващи в реални сценарии	Описва етапите на целенасочена кибератака
Фокус	Тактически – конкретни действия и техники на атакуващите	Стратегически – последователност на атака от началото до целта
Структура	Матрица с тактики и техники, групирани по фази на атака	Линейна последователност от 7 етапа на атака
Ниво на детайлност	Високо – включва техники, под-техники, примери и инструменти	Средно – описва основни фази без технически подробности
Приложимост	За лов на заплахи, анализ на инциденти, симулации и защита	За моделиране на атаки, обучение и стратегическо планиране
Актуализация и развитие	Постоянно обновяван от MITRE с нови техники и групи	По-рядко обновяван, статичен модел
Интеграция с други системи	Широко интегриран в SIEM, SOAR, EDR и платформи за лов на заплахи	Използван основно в обучителни и аналитични контексти

От направения сравнителен анализ могат да се направят следните изводи:

1. Cyber Kill Chain е по-подходящ за стратегическо моделиране на атаки, обучение на персонал и разбиране на логиката зад целенасочени атаки.
2. MITRE ATT&CK е изключително полезен за оперативни екипи по сигурността, които се занимават с идентифициране на заплахи, анализ на инциденти и изграждане на защита.
3. Комбинирането на двата модела може да даде пълна картина за атака като, като Cyber Kill Chain спомага за разбиране на фазите на атака, а MITRE ATT&CK за анализ на конкретните действия и процедури във всяка фаза.

2.3. Синтез на модели на киберзаплахи в когнитивна комуникационна мрежа.

За решаването на задача 2.3, в раздела са синтезирани последователно модел на когнитивна комуникационна мрежа, модел за разузнаване на киберзаплахи и модел на противник.

2.3.1. Модел на когнитивна комуникационна мрежа

Моделът осигурява математико-приложна основа за емулация и анализ на поведението на 5G инфраструктура при тестване, одит на сигурността и провеждане на стрес тестове. Моделът на мрежата е представен на фигура 2.14 и се дефинира като:

$$S = \{C, N, R, U, D\}, \tag{2.9}$$

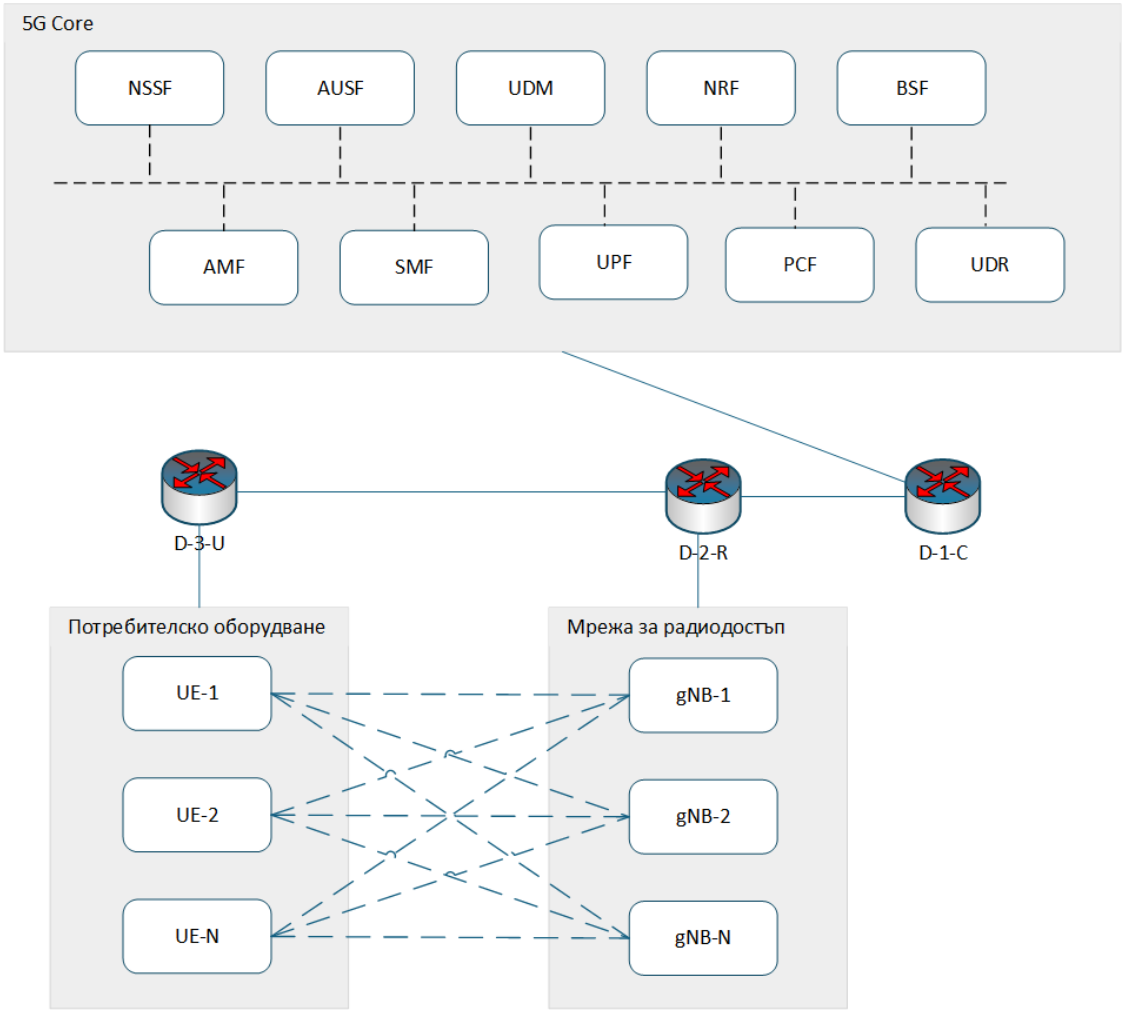
където C – основна мрежата, R – мрежа за радиодостъп, U – потребителско оборудване, N – подмрежи на мрежовата инфраструктура и D – мрежово оборудване. Мрежовата инфраструктура се състои от три компонента:

$$N = \{N_{core}, N_{RAN}, N_{UE}\} \tag{2.10}$$

Всеки възел v_i в мрежата има уникален $\mathrm{IP}(v_i) \in N_j$ за съответния домейн. Компонентът, описващ основната мрежа, се дефинира като:

$$C = \{E_{core}, F, OS_{core}\}, \tag{2.11}$$

където E_{core} е емулатора на основната 5G мрежа, $OS_{core} \in Linux$ дистрибуции - операционната система на виртуалната машина, върху която работи емулатора и F – мрежовите функции в основната мрежа.



Фиг. 2.14. Модел на когнитивна мрежа от пето поколение.

Основна мрежа C се състои от $n = 10$ виртуални мрежови функции (NF):

$$F = \{NF_1, NF_2, \dots, NF_{10}\} \quad (2.12)$$

Всяка мрежова функция се представя като наредена тройка:

$$NF_i = (id_i, IP_i, \phi_i), \quad (2.13)$$

където $id_i \in \mathrm{ID}_{NF}$ – идентификатор (напр. 5gs-amf), $IP_i \in N_{core}$ – асоцииран IPv4 и ϕ_i – функционална роля на възела (напр. маршрутизиране, сигнализация, управление и т.н.). Компонентът R дефинира gNB възлите, които изпълняват функцията на базова станция.

$$R = \{R_1, R_2, \dots, R_q\} \quad (2.14)$$

Всеки gNB възел се дефинира като:

$$R_i = (id_{gNB}, IP_{gNB}, E_{gNB}), \quad (2.15)$$

където id_{gNB} е уникален идентификатор на gNB, $IP_{gNB} \in N_{RAN}$ – IP адрес на базова станция и E_{gNB} – емулатор на UE (напр. UERANSIM). Компонентът U моделира набор от потребителски терминали.

$$U = \{UE_1, UE_2, \dots, UE_m\} \quad (2.16)$$

Всеки потребителски терминал се дефинира като:

$$UE_i = (id_i, IP_i, OS_i, \psi_i), \quad (2.17)$$

където id_i – уникален идентификатор на UE устройство, $IP_i \in N_{UE}$ – асоцииран IP за UE възел, $OS_i \in \{iOS, Android\}$ – операционна система на емулатора на UE и $\psi_i \subseteq \Psi$ – набор от приложения (напр. WhatsApp, Messanger и т.н.). Мрежовото оборудване D включва мрежови устройства като рутери, суичове и др. Всяко мрежово устройство се дефинира като наредена тройка:

$$d_i = (id_i, IF_i, \theta_i), \quad (2.18)$$

където $id_i \in \mathrm{ID}_D$ е уникален идентификатор на устройството, $IF_i = \{(IP_{i1}, N_{i1}, link_{i1}), \dots, (IP_{in}, N_{in}, link_{in})\}$ – множество от интерфейси с IP адреси и мрежи, към които е свързан и $\theta_i \in \{router, switch, bridge, firewall, \dots\}$ – роля или тип на устройството. Предложеният модел включва следното множество от мрежови устройства:

$$D = \left(d_1^{(c)}, d_2^{(r)}, d_3^{(u)}, d_4^{(c)}, d_5^{(r)}, d_6^{(u)}\right), \tag{2.19}$$

където $d_1^{(c)}$- рутер в основна мрежа, $d_2^{(r)}$ – рутер в мрежата за радиодостъп, $d_3^{(u)}$- рутер в мрежата за потребителско оборудване и $d_4^{(c)}, d_5^{(r)}, d_6^{(u)}$ са суичовете за локално свързване на възлите в трите подмрежи.

$$d_1^{(c)} = (id = routercore, IF_1, \theta_1 = router), \tag{2.20}$$

където:

$$IF_1 = \left\{\left(IP_1, N_{core}, d_4^{(c)}\right), \left(IP_2, N_{RAN}, d_2^{(r)}\right)\right\} \tag{2.21}$$

$$d_4^{(c)} = (id = swcore, IF_4, \theta_2 = switch), \tag{2.22}$$

където:

$$IF_4 = \left\{\left(IP_1, N_{core}, d_1^{(c)}\right), (IP_2, N_{core}, NF_1), (IP_3, N_{core}, NF_2) \dots\right\} \tag{2.23}$$

$$d_2^{(r)} = (id = routerran, IF_2, \theta_2 = router), \tag{2.24}$$

където:

$$IF_2 = \left\{\left(IP_1, N_{core}, d_1^{(c)}\right), \left(IP_2, N_{RAN}, d_5^{(r)}\right), \left(IP_3, N_{UE}, d_3^{(u)}\right)\right\} \tag{2.25}$$

Рутерът в мрежата на потребителското оборудване се дефинира като:

$$d_3^{(u)} = (id = routerue, IF_3, \theta_3 = router), \tag{2.27}$$

където:

$$IF_3 = \left\{\left(IP_1, N_{RAN}, d_2^{(r)}\right), \left(IP_2, N_{UE}, d_6^{(u)}\right)\right\} \tag{2.28}$$

Суичът в мрежата на потребителските терминали доставя подмрежа, която се използва като основа за свързване на UE емулатора към gNB.

$$d_6^{(u)} = (id = swue, IF_6, \theta_6 = switch), \tag{2.29}$$

където:

$$IF_6 = \left\{\left(IP_1, N_{UE}, d_3^{(r)}\right), (IP_2, N_{UE}, UE_1), (IP_3, N_{UE}, UE_2), \dots\right\} \tag{2.30}$$

2.3.2. Модел на разузнаване за киберзаплахи в публични бази данни

Предложеният модел е проектиран да идентифицира релевантните противници и заплахи, насочени към конкретни сектори и региони, да картографира свързаните с тях тактики, техники и процедури, както и да включи броя на документираните кампании като теглови коефициент в анализа. На фигура 2.15 е представена обща блок-схема на предложеният модел.

В първия етап се извличат всички APT групи от публичните бази данни, обозначени с уникален идентификатор *G_x*.

Нека $G = \{g_1, g_2, \dots, g_n\}$ да е множеството от злонамерени групи, $T = \{t_1, t_2, \dots, t_m\}$ е множеството от офанзивни техники и $\tau = \{\tau_1, \tau_2, \dots, \tau_p\}$ е множество от офанзивни тактики.

Дефинира се функция за принадлежност, като се приема, че всяка техника принадлежи към дадена тактика:

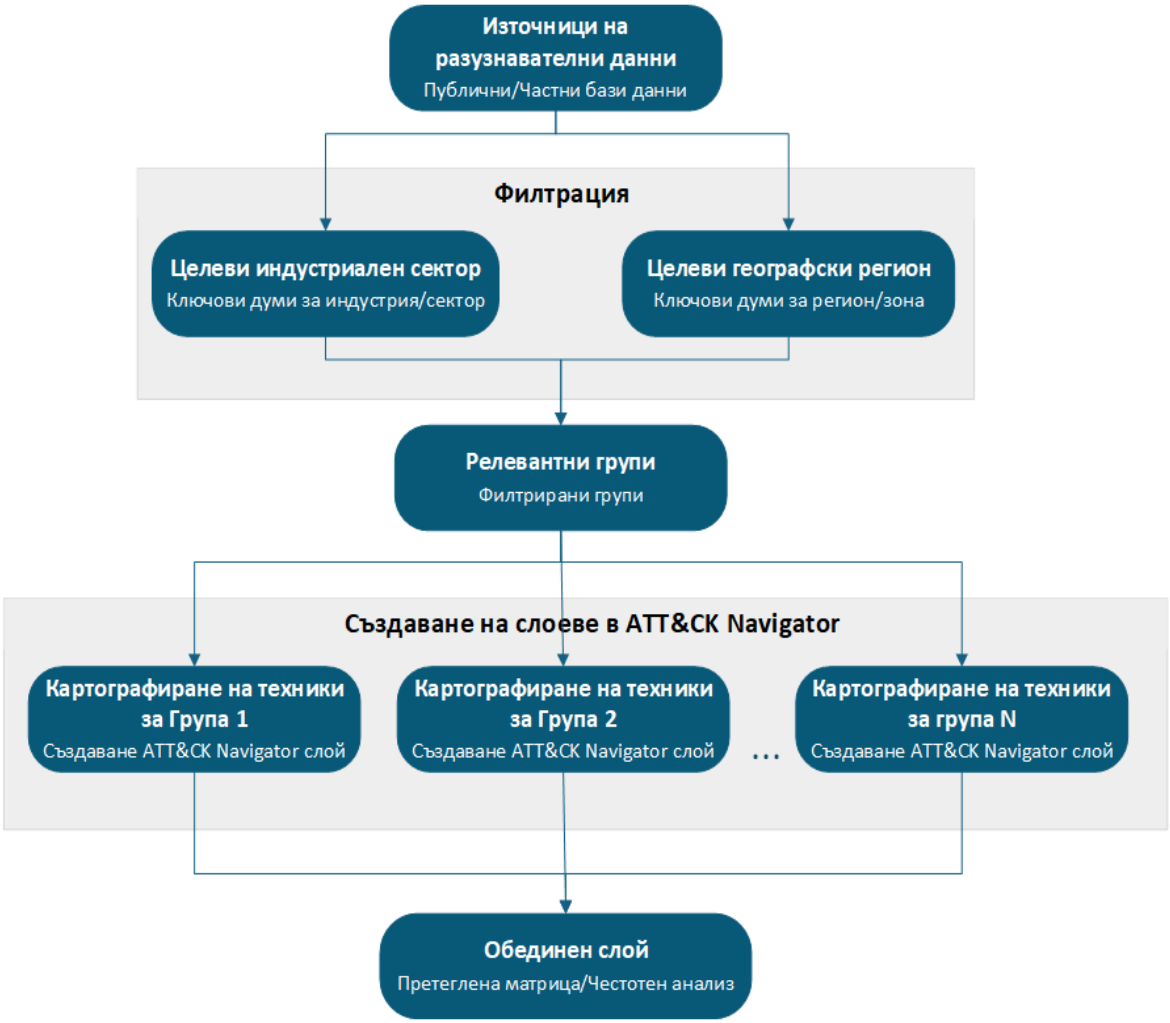
$$\mu: T \to \tau, \tag{2.39}$$

където $\mu(t_j) = \text{тактиката, към която принадлежи техниката } t_j$. Всяка група g_i има свързани два атрибута: сектор на интерес: $S(g_i) \subseteq \Sigma$, където Σ е множеството от всички индустриални сектори и географска активност: $R(g_i) \subseteq R$, където R е множеството от географски региони (напр. Европа, Азия, НАТО).

Определят се множествата на целевите сектори и региони:

$$S_{target} = \{S_1, S_2, \dots, S_k\} \tag{2.40}$$

$$R_{target} = \{R_1, R_2, \dots, R_j\} \tag{2.42}$$



Фиг. 2.15. Блокова схема на модел за разузнаване на киберзаплахи

S_{target} дефинира целевия сектор чрез набор от ключови думи. R_{target} дефинира целевия регион чрез набор от ключови думи.

Множеството от релевантни групи $G' \subseteq G$ се дефинира като:

$$G' = \{g_i \in G \mid S(g_i) \cap S_{target} \neq \emptyset \wedge R(g_i) \cap R_{target} \neq \emptyset\}, \quad (2.44)$$

където $i \in [1, n]$ и $G' \subseteq G$. За всяка група $g_i \in G'$, се дефинира множеството от използвани ATT&CK техники:

$$T(g_i) = \{t_1, t_2, \dots, t_k\} \subseteq T \quad (2.45)$$

Индикаторната функция за използваните техники може да се изчисли като:

$$\delta_{j,a}(g_i) = \begin{cases} 1, & \text{ако } t_j \in T(g_i) \wedge \mu(t_j) = \tau_a \\ 0, & \text{в противен случай} \end{cases} \quad (2.46)$$

Създава се бинарна матрица за всяка релевантна група g_i:

$$\mathbf{M}(\boldsymbol{g_i}) = [\delta_{ja}(g_i)] \in \{0,1\}^{n\,x\,m}, \quad (2.47)$$

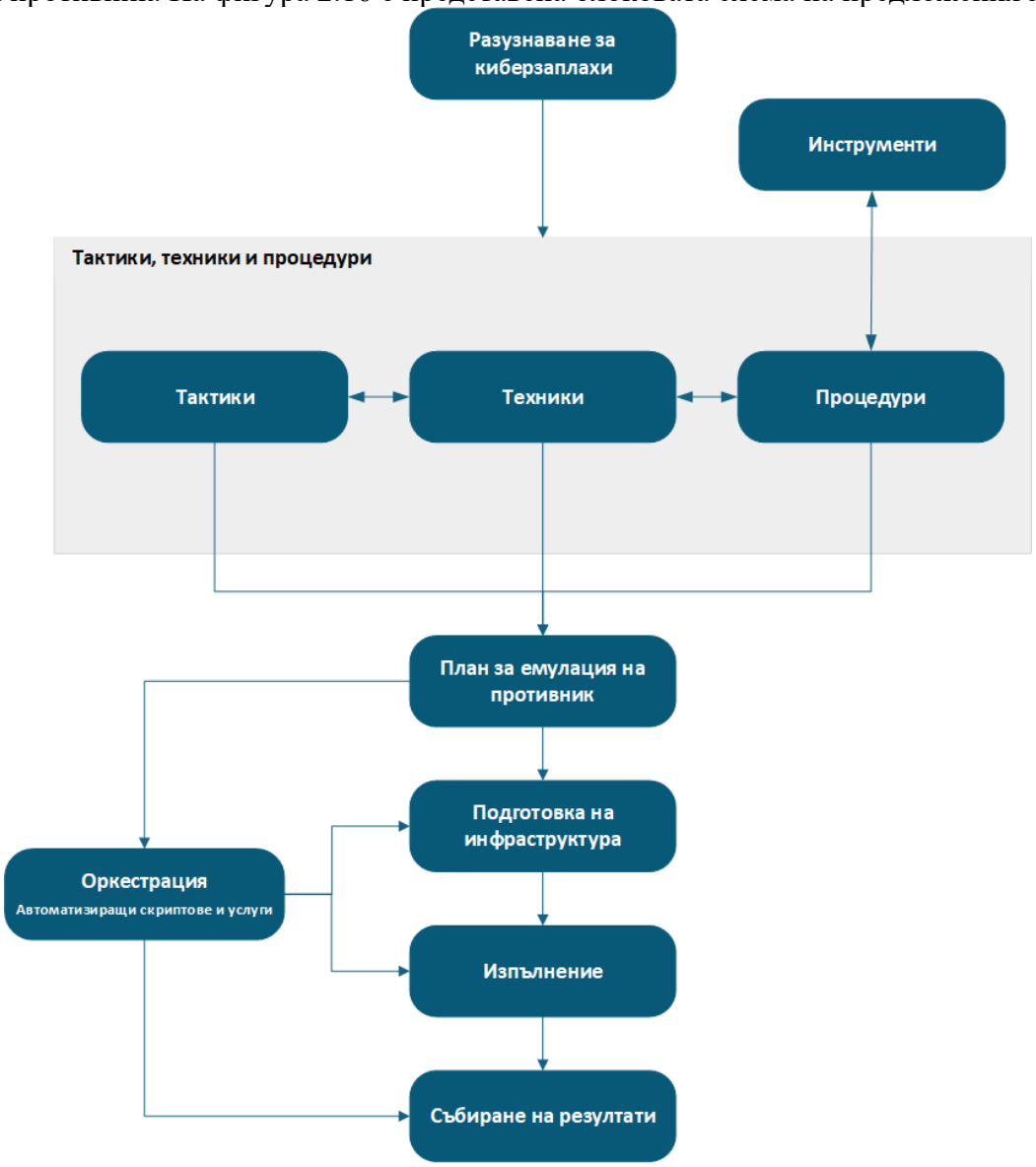
където ред j съответства на техника t_j, колона a съответства на тактика τ_a и $n = |T|,\ m = |\tau|$. Тогава честотата на използване на офанзивни техники от релевантни групи може да се извлече от резултантна матрица, като се сумират матриците на всички извлечени групи.

$$\mathbf{M}^*_{j,a} = \sum_{i=1}^{|G'|} \mathrm{M}(g_i)_{j,a}, \qquad 1 \le j \le n, \qquad 1 \le a \le m \quad (2.49)$$

2.3.3. Модел на противник

Предложеният модел описва ключовите компоненти на противника — уникален идентификатор на групата, набор от използвани техники, стратегия за избор и последователност на техниките, план за емулация с конкретни процедури, както и необходимата инфраструктура. Това

позволява възпроизвеждане и автоматизирана симулация на реални APT кампании и техники на злонамерен противник. На фигура 2.16 е представена блоковата схема на предложения модел.



Фиг. 2.16. Блокова схема на модел на противник

Противникът (група, кампания или актьор) се дефинира като:

$$A = \left(g, {T_g}^w, \Pi_g^P, E, I_g\right), \tag{2.54}$$

където:

- $g \in G$ – уникален идентификатор на групата (APT, хактивисти и т.н.);
- ${T_g}^w \subseteq T$ – множество от използвани офанзивни техники по данни от CTI;
- Π_g^P – стратегия на противника (Ross, 2014);
- E – план за емулация (стратегия на поведение - описва реда на фази, техники и процедури);
- I_g – инфраструктура на противника.

Групата g идентифицираща конкретен противник, принадлежи към множеството:

$$G = \{g_1, g_2, \dots, g_n\}, \tag{2.55}$$

където G е множество от известни групи в публични бази данни. Тогава $g \in G$ ще е уникалния идентификатор на противниковата група, който често се дефинира като APT__ или G_x. Избраната стойност $g \in G$ се използва като референтна точка за всички свързани множества $T_g, C_g, R_g, S_g, \Pi_g$. Нека:

$$T_g = \{t \in T \mid t \text{ е използвана техника от група } g\} \tag{2.56}$$

бъде множеството от техники, използвани от групата g.

С цел количествено отчитане на честотата на използване на техниките, се разширява T_g до претеглено множество:

$$T_g{}^w = \{(t, w_t) \mid t \in T_g, w_t \in N\}, \tag{2.59}$$

където $w_t \in N$ – теглови коефициент на техника t, измерено като броя на кампаниите, в които t участва по данни от CTI. Тогава претегленото множество, може да се представи, чрез декомпозиция по тактики:

$$T_g{}^w = \bigcup_{\tau \in X} T_g{}^{w(\tau)}, \tag{2.60}$$

където:

$$T_g{}^{w(\tau)} = \{(t, w_t) \mid t \in T^{(\tau)} \cap T_g\} \tag{2.61}$$

Дефинира се компонент Π_g, който моделира логиката и реда, по който група g прилага техники $t \in T$, за да постигне цели от различни тактически категории. Планът за емулация на противник се генерира като последователност от стъпки:

$$E = \left[\left(\varphi^{(1)} t^{(1)}, p^{(1)}\right), \left(\varphi^{(1)} t^{(2)}, p^{(2)}\right), \ldots, \left(\varphi^{(L)} t^{(L)}, p^{(L)}\right)\right], \tag{2.72}$$

където за всяка стъпка i от плана се определя избрана техника $t^{(i)}$ и процедура $p^{(i)}$. Условието за завършване на изпълнението е достигането на определения брой стъпки L. Инфраструктурата на противника I_g се дефинира като:

$$I_g = \left(H_g, D_g, N_g, A_g\right), \tag{2.73}$$

където H_g е множество от хостове в противниковата инфраструктура, D_g е множество от домейни, N_g е мрежовата топология за развръщане на инфраструктурата и A_g е множество на креденциите за достъп до реализираните услуги. Множеството от хостове се разглежда като наредена четворка:

$$H_g = (id_h, type_h, ip_h, os_h), \tag{2.74}$$

където id_h е уникален идентификатор на хоста, $type_h$ дефинира услугата реализирана на конкретен хост (роля), ip_h - присвоен IPv4 адрес и os_h дефинира операционната система на хоста. Услугите, които може да поддържа определен хост принадлежат към следното множество:

$$type_h \in \{WEB, C2, Mail, Files, DNS, Attackbox, VPN, Proxy\} \tag{2.75}$$

Наборът от операционни системи, които се присвояват за всеки хост, се определя като:

$$os_h \in \{Ubuntu, Debian, Win22, Win10, Win11, Opnsense, Kali\} \tag{2.76}$$

Множеството домейни D_g се разглежда като множество от елементи, представени като:

$$D_g = (id_d, n_d, r_d, rr_d), \tag{2.77}$$

където id_d е уникален идентификатор на домейна, n_d е име на домейна, дефиниран при регистрация, r_d – метаданни за регистрация (WHOIS информация), rr_d – набор от DNS записи (A, AAAA, CNAME, MX, TXT). Мрежовата топология определя възлите и връзките между тях:

$$N_g = (V, E, S, R), \tag{2.78}$$

където V е множество от върхове (възли). В представения модел $V = H_g$. E е множество от връзки между възлите, S е множество от подмрежи и R е множество от правила на мрежовите устройства. Множеството от подмрежи, с които разполага противниковата инфраструктура, се дефинира като:

$$S = \{s_1, s_2, \ldots, s_k\}, \tag{2.80}$$

където всяка подмрежа s_i се определя с няколко параметъра:

$$s_i = (net_i, mask_i, gw_i), \tag{2.81}$$

където net_i – адрес на мрежата, $mask_i$ – подмрежова маска и gw_i – адрес на гейтуей за включване към подмрежа от по-горно ниво. Правилата за филтрация на мрежов трафик в мрежовите устройства са дефинирани като множество:

$$R = \{r_1, r_2, \ldots, r_p\}, \tag{2.82}$$

Всяко правило r_i включва следните параметри:

$$r_i = (proto_i, srcip_i, srcprt_i, dstip_i, dstprt_i), \tag{2.83}$$

Креденциите за достъп определят идентификационните данни за операционните системи и услугите в противниковата инфраструктура A_g:

$$A_g = \{\mathrm{a}_1, \mathrm{a}_2, \dots, \mathrm{a}_c\}, \quad (2.84)$$

Всеки елемент $\mathrm{a}_i \in A_g$ се представя като наредена четворка:

$$\mathrm{a} = (id_a, type_a, scope_a, sercret_a), \quad (2.85)$$

където id_a – уникален идентификатор на записа с креденциите, $scope_a$ – обхват от ресурси за които креденциите са валидни, $sercret_a$ – референция (указател) към стойностите на креденциите и $type_a \in P_{cred}$ – тип на креденциите.

$$P_{cred} = (user\_pass,\ ssh\_key,\ api\_token,\ oauth,\ cert), \quad (2.86)$$

$$scope_a \subseteq H_g \cup D_g, \quad (2.87)$$

т.е. множеството от хостове, услуги или домейни (или комбинация), към които този креденшъл дава достъп.

$$sercret_a = ptr(id_a) \quad (2.88)$$

Следва изпълнение на планът за емулация. Изпълнението се дефинира като:

$$X = (E, L, Y_x), \quad (2.89)$$

където E – план за емулация дефиниран в 2.72, L – брой на стъпките за изпълнение и Y_x – последователност от състояния на инфраструктурата и целевата среда след всяка стъпка.

$$Y_x = [y_1, y_2, \dots, y_L] \quad (2.90)$$

Всяко състояние y_i включва:

$$y_i = \left(t^{(i)}, p^{(i)}, r^{(i)}\right), \quad (2.91)$$

където $t^{(i)}$ – избрана техника от плана, $p^{(i)}$ – конкретната процедура, която реализира техниката и $r^{(i)}$ – резултат от изпълнението на стъпката (успех/неуспех). Резултатът Z се дефинира като обединение на всички резултати от изпълнението на стъпките $r^{(i)}$:

$$Z = \bigcup_{i=1}^{L} r^{(i)} \quad (2.92)$$

2.4. Изводи

1. Комбинираното използване на стратегически рамки като NIST CSF и Cyber Kill Chain с тактически модели като MITRE ATT&CK и MITRE D3FEND осигурява цялостен подход за провеждане на офанзивни и дефанзивни кибероперации. NIST CSF и Kill Chain предоставят структурирана визия за управление на риска и фазите на атака. ATT&CK и D3FEND допълват с конкретни техники за атака и защита. Това позволява на организациите да изграждат ефективни стратегии, които обхващат както управленски, така и технически аспекти, подобрявайки откриването, реакцията и устойчивостта срещу заплахи.
2. За целите на дисертационния труд са избрани тактическите модели на MITRE ATT&CK и MITRE D3FEND за синтез на модел на заплахите на когнитивна комуникационна мрежа, защото те предоставят богата база данни за реални техники, използвани от атакуващи, и са подходящи за технически екипи, работещи по оперативна защита, идентифициране на заплахи и изграждане на защитни стратегии, базирани на реални техники.
3. Синтезирани са 3 модела, които дават математико-приложна формализация на компонентите на комуникационна когнитивна мрежа и предоставят основа за емулация и анализ на поведението на 5G инфраструктура при тестване, одит на сигурността и провеждане на стрес тестове. Предложените модели са полезни за научни и технически екипи, които се занимават с внедряване на конкретни защитни механизми, анализ на атаки и изграждане на архитектура за сигурност:
   (1) **Модел на когнитивна комуникационна мрежа**, който дефинира логическата и физическата структура на системата и компонентите на мрежата като наредени множества и релации, описващи тяхната идентичност, функционална роля, мрежови адреси и взаимосвързаност.
   (2) **Модел на разузнаване за киберзаплахи в публични бази данни**, който осигурява количествена оценка за анализ на поведението на противника. Моделът идентифицира релевантните противници и заплахи, насочени към конкретни сектори и региони, и картографира свързаните с тях TTPs.

(3) **Модел на противник**, който математически формализира основните характеристики на противника, като уникален идентификатор на групата, множеството от използвани офанзивни техники, стратегията на противника, плана за емулация и инфраструктурата на противника.

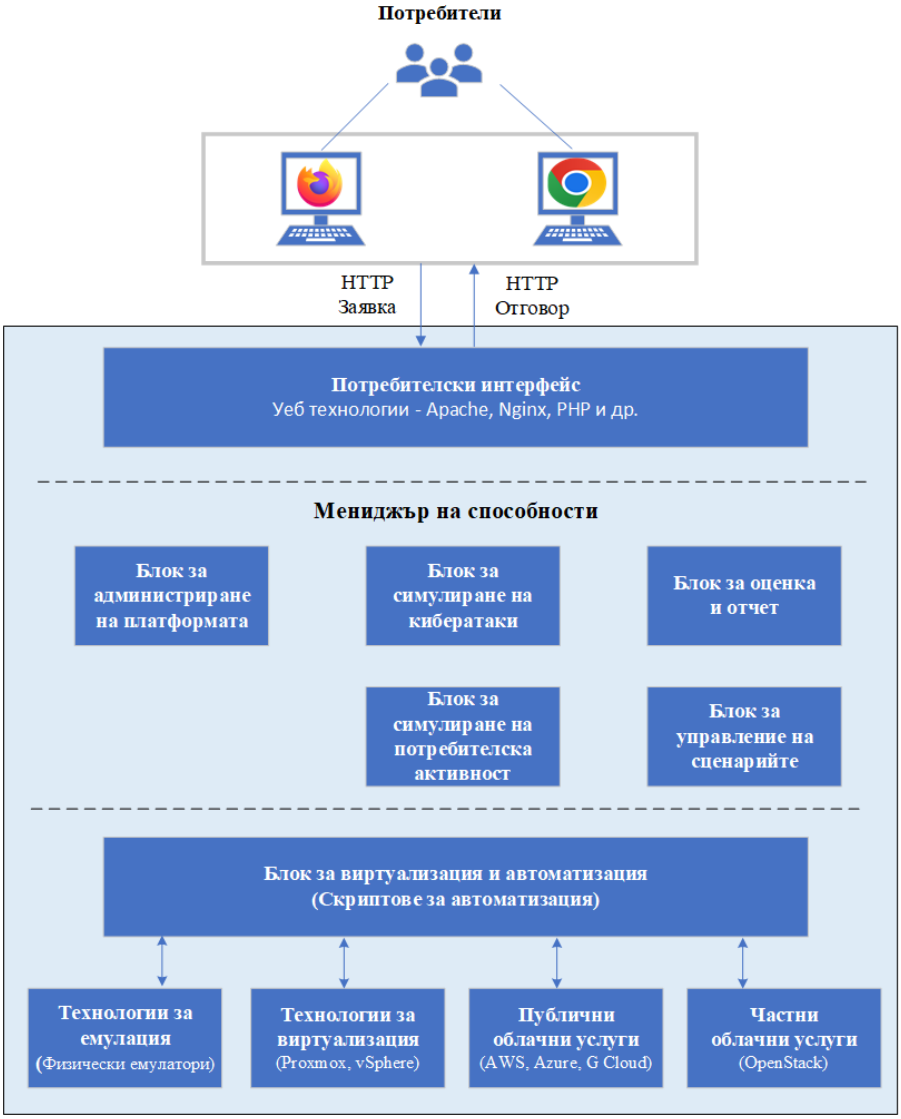
Глава 3 – Специализиран киберполигон за реализиране на симулационни модели.

За решаването на трета основна задача от дисертационният труд, в трета глава са дефинирани следните подзадачи:

1. Да се проектира и изгради концептуалната и мрежова архитектура на специализирания киберполигон.
2. Да се разработи математико-приложен модел, формализиращ сценариите в специализирания киберполигон.
3. Да се разработи симулационен сценарий на експериментална среда за изследване на сигурността на потребителско оборудване и мрежова инфраструктура на 5G мобилна мрежа

3.1. Състав на предложения киберполигон

За решаването на задача 3.1. в този раздел е предложена специализирана платформа за създаване на симулационни среди за провеждане на стрес тестове, както и офанзивни и дефанзивни кибероперации. Концептуалната архитектура на предложеният модел е представена на блок схемата на фигура 3.1, като функционалните елементи са групирани в 3 слоя в зависимост от услугите, които реализират: *Потребителски интерфейс*, *Мениджър на способности* и *Блок за виртуализация и автоматизация*.



Фиг. 3.1. Архитектура на предложената платформа.

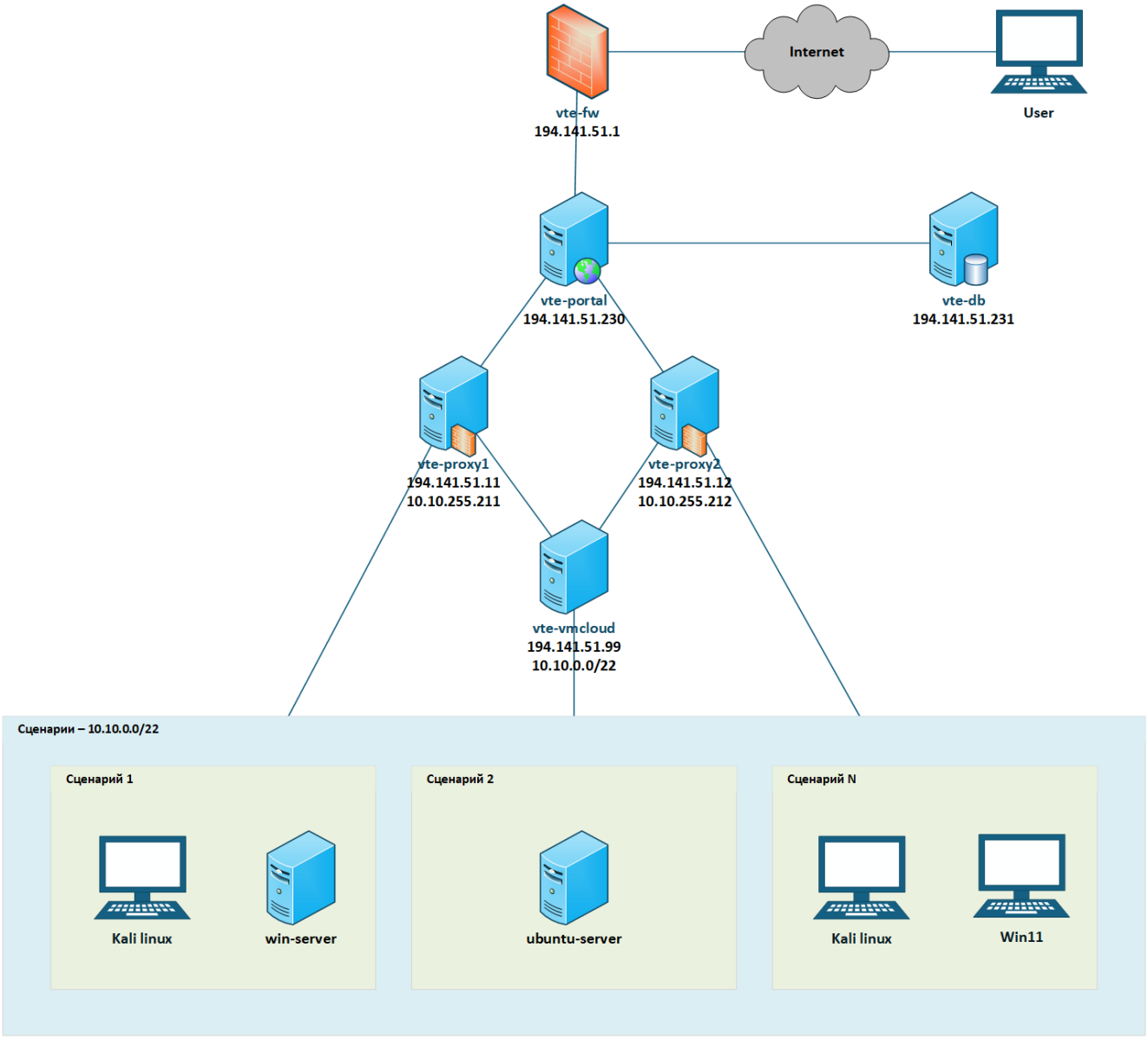
Потребителският интерфейс осигурява взаимодействието между потребителя и услугите на платформата и използва уеб сървърна технология Apache за изграждането на интерактивен портал за потребителите, чрез който да взаимодействат с услугите на платформата, без да се налага инсталиране и конфигуриране на допълнителен софтуер.

Мениджърът на способности осигурява основните функции на платформата и включва *Блок за администрация и управление, Блок за симулиране на кибератаки, Блок за оценка и отчет, Блок за симулация на потребители* и *Блок за управление на сценариите*.

Мениджърът на способности дефинира възможностите на платформата за организиране на симулациите, провеждане на обученията, оценка на екипите, както и взаимодействието с блока за виртуализация при изграждането на средите за емулация на мрежи и операционни системи.

Блокът за виртуализация и автоматизация служи за автоматизирано конфигуриране, координиране и управление на компютърните системи и софтуер. Състои се от платформа за виртуализация Proxmox и множество автоматизиращи скриптове.

Мрежовата архитектура на платформата е представена на фигура 3.2.



Фиг. 3.2. Мрежова архитектура на предложената платформа

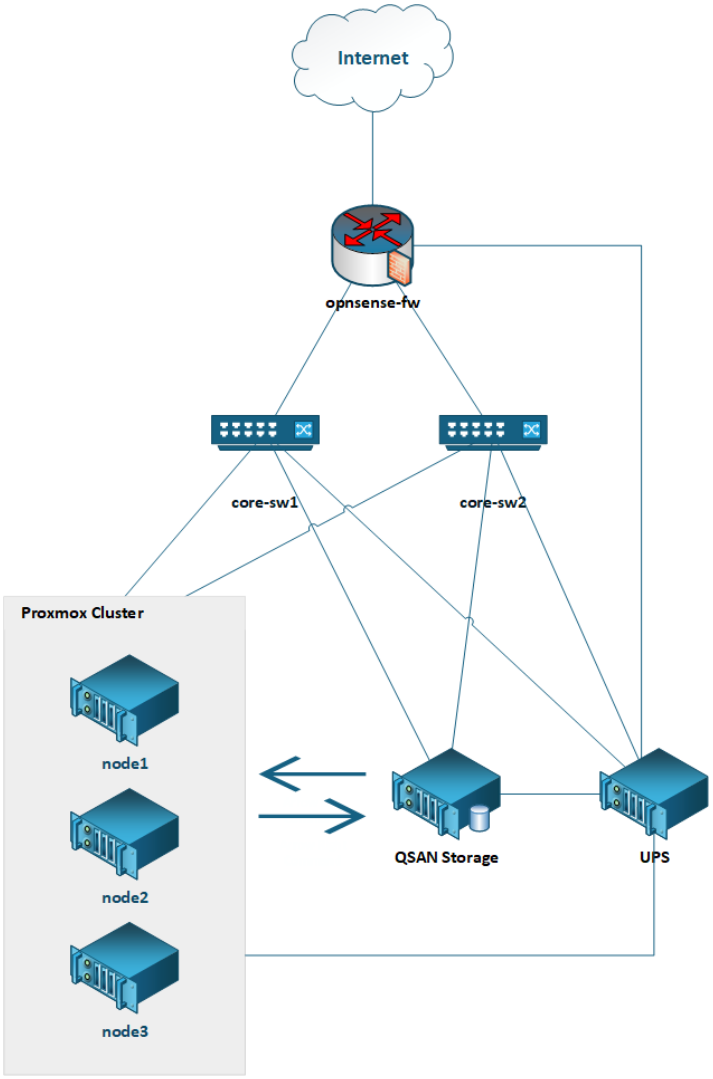
На външния периметър е разположена защитна стена *vte-fw*, която осъществява контрол на трафика между Интернет и вътрешната инфраструктура на платформата. Основната точка за достъп до платформата е възела *vte-portal*, който обслужва уеб интерфейса за потребителите. Чрез него се осъществява автентикация, управление на сценарии и инициализация на виртуални машини. Порталът комуникира както с потребителите през Интернет, така и с останалите подсистеми –

базата данни и прокси сървърите. Вътрешната база данни *vte-db* съхранява информация за потребители, сценарии, виртуални машини и активни сесии. За да се гарантира баланс на натоварването са внедрени два прокси сървъра – *vte-proxy1* и *vte-proxy2* Тяхната роля е да препращат заявките за стартиране на виртуални машини към облачния клъстер и да осигуряват защитен графичен уебдостъп до машините по протоколи VNC, RDP и SSH. Всички сценарии и тестови среди се реализират в рамките на облачен клъстер *vte-vmcloud*. В този сегмент се намират всички виртуални машини, асоциирани със сценариите. В таблица 7 са представени машините, изградени в мрежовия модел.

Таблица 7. Описание на машините в мрежовия модел

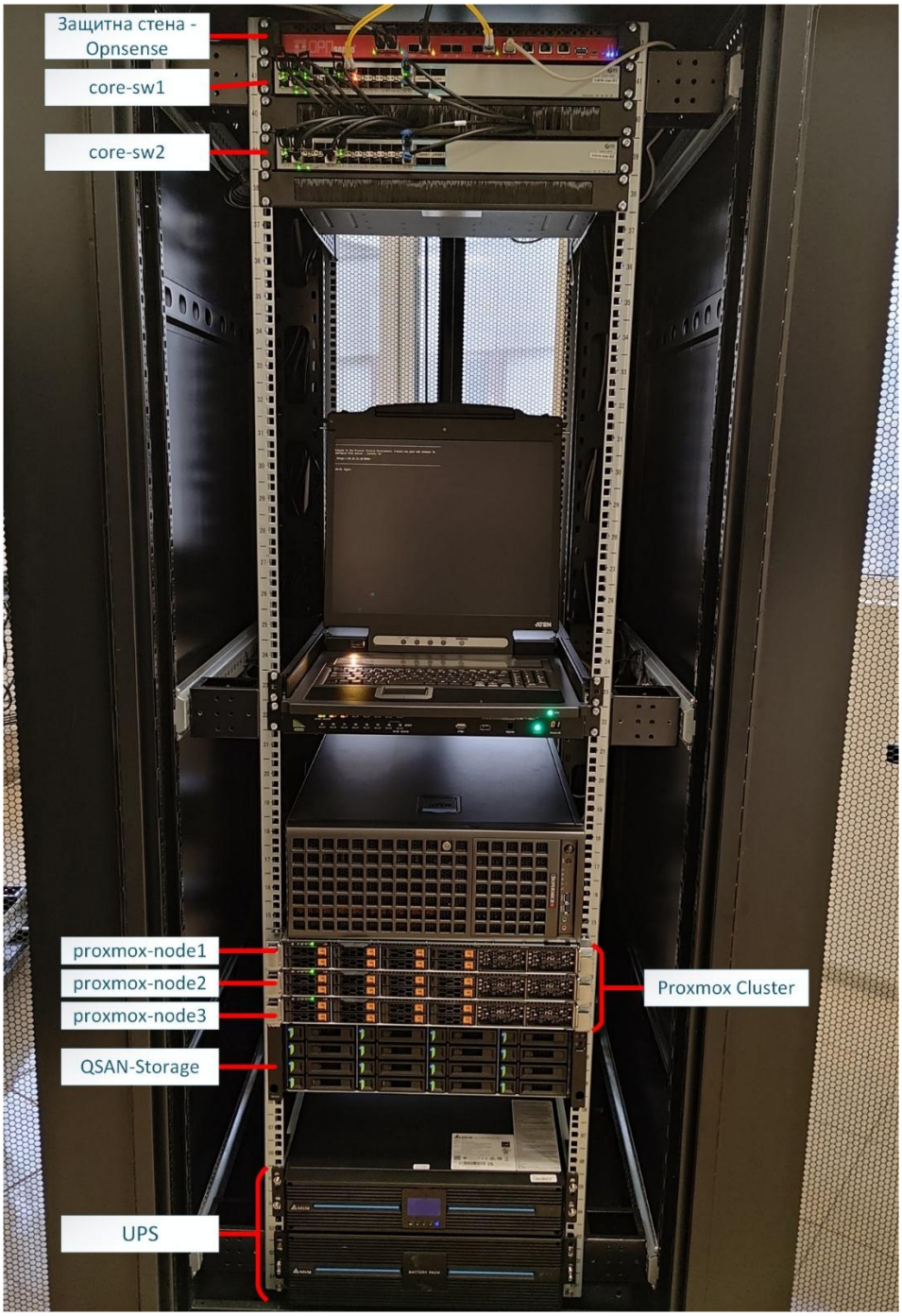
Машина	Роля	IPv4	Порт	Услуга
vte-fw	Гранична защитна стена	194.141.51.1	-	OPNsense
vte-portal	Уеб-портал на платформата	194.141.51.230	443	WEB/HTTPS
vte-db	База данни	194.141.51.231	3306	MySQL
vte-proxy1	Препращане на потребителски трафик	194.141.51.11	443	VM API, VNC, RDP, SSH
vte-proxy2	Препращане на потребителски трафик	194.141.51.12	443	VM API, VNC, RDP, SSH
vte-vmcloud	Клъстер за виртуализация	194.141.51.99	-	Proxmox VE

Физическата архитектура на предложения модел е представена на фигура 3.3. Тя включва граничен рутер (защитна стена), два основни суича, които гарантират резервираност при отказ на комуникационен канал, UPS система за резервно захранване и три идентични сървъра, обединени в проксмокс клъстерна конфигурация за реализиране на облачната инфраструктура, поддържаща виртуалните машини.



Фиг. 3.3. Физическа архитектура на платформа Cyber Range

Описаната физическа архитектура е реализирана в центъра за съхранение на данни на факултет „Артилерия, ПВО и КИС“ (фигура 3.4).



Фиг. 3.4. Реализирана архитектура на специализиран киберполигон

3.2. Модел на сценариите

За решаването на задача 3.2 е разработен формален модел на сценариите в симулационната платформа. Моделът описва структурата и взаимовръзките между основните компоненти, участващи в изграждането и управлението на учебните и тренировъчните сценарии. Освен това е представен алгоритъм за стартиране и управление на виртуални среди чрез приложно-програмен интерфейс, Application Programming Interface (API), който осигурява контрол върху ресурсите, генерирането на сесии и автоматизираната инициализация на машините в рамките на сценариите. Нека:

$$S = \{S_1, S_2, \dots, S_n\} \tag{3.1}$$

е множество от всички сценарии в платформата. Всеки сценарий $S_i \in S$ се дефинира като:

$$S_i = \{t_i, d_i, c_i, l_i, \tau_i, a_i, T_i\}, \tag{3.2}$$

където $t_i \in \Sigma^{\leq 40}$ – заглавие (низ с дължина ≤ 40 символа), $d_i \in \Sigma^{\leq 255}$ – кратко описание (низ с дължина ≤ 255 символа), c_i – уникален идентификатор за всеки сценарий, $l_i \in \{1, 2, 3\}$ – ниво на трудност (1 = лесно, 2 = средно, 3 = трудно), $\tau_i \in N^+$ – време за изпълнение (минути). Целочислена положителна стойност, $a_i \in \{0, 1\}$ – двоична стойност, определяща възможност за включване на атакуваща машина в сценария и $\mathrm{T}_i \in \{t_{i1}, t_{i2}, \dots, t_{im}\}$ – множество от задачи (стъпки) за сценария. Всяка задача $t_{ij} \in \mathrm{T}_i$ се дефинира с няколко ключови елемента:

$$t_{ij} = \{\theta_{ij}, \kappa_{ij}, v_{ij}, Q_{ij}\}, \tag{3.4}$$

където $\theta_{ij} \in \{0, 1, 2\}$ – тип на задачата (0 = без виртуална машина VM, 1 = VM, 2 = статичен сайт), κ_{ij} – съдържанието на задачата, $v_{ij} \in V \cup \{\theta\}$ – виртуална машина или статичен сайт привързани към задачата (ако има дефинирани такива) и $Q_{ij} = \{q_{ij1}, q_{ij2}, \dots, q_{ijm}\}$ – множество от въпроси в конкретната задача. Всеки въпрос $q_{ijk} \in Q_{ij}$ е съставен от следните компоненти:

$$q_{ijk} = \{\varphi_{ijk}, \alpha_{ijk}, \eta_{ijk}\}, \tag{3.5}$$

където φ_{ijk} – съдържание на въпроса и α_{ijk} – правилен отговор, дефиниран от инструктор. В случай, че инструктор не дефинира правилен отговор $\alpha_{ijk} = 0$, въпросът се трансформира автоматично в стъпка, в която потребителя просто потвърждава нейното преминаване. η_{ijk} – подсказка към въпроса в помощ на участниците/екипите (незадължителен елемент). Множеството от интерактивни ресурси, които поддържа платформата, се дефинира като:

$$V = \{v_1, v_2, \dots, v_n\} \tag{3.6}$$

Всеки интерактивен материал $v \in V$ към задача включва следните дефиниции: id_v – уникален идентификатор, $n_v \in \Sigma^{\leq 40}$ – име на материалът (стринг с дължина ≤ 40 символа) и $p_v \in \{0, 1, 2, 3\}$ – протокол за уеб достъп. Платформата поддържа 3 протокола за осигуряване на графична връзка към машините (0 – без GUI, 1 – VNC, 2 – RDP, 3 – SSH).

Приложно програмният интерфейс за внедряване и стартиране на виртуалните машини в сценариите работи по следните входни данни:

- $u \in U$ – потребител, който стартира VM (идентифициран чрез сесия), където U е множеството от потребители в базата данни на платформата
- $s \in S$ – сценарий, в който потребителят е стартирал виртуалната машина. Определя се чрез идентификатора на сценария c_i, който е дефиниран в базата данни при създаването му.
- n_v - името на машината, която трябва да се стартира.
- $t_p \in \{0,1, 2, 3\}$ – тип достъп до машината: 0 – целева система (без GUI), 1 – SSH, 2 – RDP, 3 – VNC).

При това API формализира процеса по стартиране на VM, използвайки алгоритъма, представен на фигура 3.5. Основните стъпки на алгоритъма са:

1. Валидация на входните данни: ако потребителят u не е автентикиран → връщане на грешка 403 и ако липсва v_{id} или s → връщане на грешка 400.
2. Проверка за ограничение на ресурси: изчислява се:

$$\phi(u) = |\{v \in V \mid v \text{ е стартирана от } u\}|, \tag{3.7}$$

където ако $\phi(u) \geq 3$ → връщане на грешка („макс. 3 машини“). По такъв начин се предотвратява възможността от претоварване на ресурсите на платформата в следствие на спам при създаване на неограничен брой машини.

3. Генериране на сесия към VM: извлича се v_{id} по n_vот базата данни. Идентификаторът на VM шаблон $v_{id} \in V$ – указва коя машина да се клонира и стартира. Изчислява се случаен идентификатор $c \in \Sigma^{\geq 15}$с дължина ≥ 15 символа. Генерира се *url* адреса за графичен уеб достъп до машината:

$$url = f(d, c, t_p), \tag{3.8}$$

където f е функция, която съставя адрес по домейн d и тип на машината t_p.

4. Регистрация в базата данни: генерира се случаен идентификатор за новата машина в границите от 10000-20000:

$$v'_{id} \in \{10000 - 20000\} \tag{3.9}$$

Записва се новата VM в таблица със стартирани машини V':

$$V' = (u, s, v'_{id}, url, \tau_{exp}), \tag{3.10}$$

където $\tau_{exp} = now() + 2h$. При това новостартираната VM разполага с 2 часа преди да бъде унищожена от API за терминиране на VM.

5. Отговор към клиента: определя се времето за стартиране на VM по данни от платформата за виртуализация в променлива δ и се връща отговор към клиента в JSON масив: $\{status: success, url: url, expires_{in}: \tau_{exp}, countdown: \delta\}$.
6. Бекенд инициализация:
   a. избира се автоматизиращ скрипт $g(t_p)$ за старт на VM:

$$g(t_p) = \begin{cases} start_{ssh}, & t_p = 1 \\ start_{rdp}, & t_p = 2. \\ start_{vnc}, & t_p = 3 \end{cases} \tag{3.11}$$

   b. Извиква се:

$$exec(g(t_p), c, v_{id}, v'_{id}). \tag{3.12}$$

Algorithm 1: Start VM API

Input: $u \in U$ – user; $s \in S$ – scenario; n_v – VM name; $t_p \in \{0, 1, 2, 3\}$ – access type
Output: JSON response with status, URL and expiration time
Input validation:;
if *user u is not authenticated* **then**
  return error 403
if $n_v = \emptyset$ *or* $s = \emptyset$ **then**
  return error 400
Resource limitation check:;
$\varphi(u) \leftarrow |\{v \in V' \mid vstartedbyu\}|$;
if $\varphi(u) \geq 3$ **then**
  return error 403 (maximum 3 VMs allowed)
Session generation:;
$v_{id} \leftarrow$ SELECT vmid FROM V WHERE name $= n_v$;
if $v_{id} = \emptyset$ **then**
  return error 404 (VM template not found)
$c \leftarrow$ RandomString(≥ 15);
$url \leftarrow f(d, c, t_p)$;
Database registration:;
$v'_{id} \leftarrow$ RandomInteger(10000,20000);
$\tau_{exp} \leftarrow now() + 2h$;
INSERT INTO Started_VMs$(u, s, v'_{id}, url, \tau_{exp}, t_p)$;
Client response:;
$\delta \leftarrow$ StartupDelay(t_p);
return {status: success, url: *url*, expires_in: τ_{exp}, countdown: δ};
Backend initialization:;
switch t_p **do**
  case *0* **do**
    start target
  case *1* **do**
    start ssh
  case *2* **do**
    start rdp
  case *3* **do**
    start vnc
ExecuteRemote$(t_p, c, v_{id}, v'_{id})$;

Фиг. 3.5. Псевдо код на алгоритъм за изграждане на VM

3.3. Симулационна среда за изследване на сигурността на 5G

За изследване на сигурността на потребителско оборудване и мрежова инфраструктура е реализиран симулационен сценарий на експериментална среда в рамките на изградения

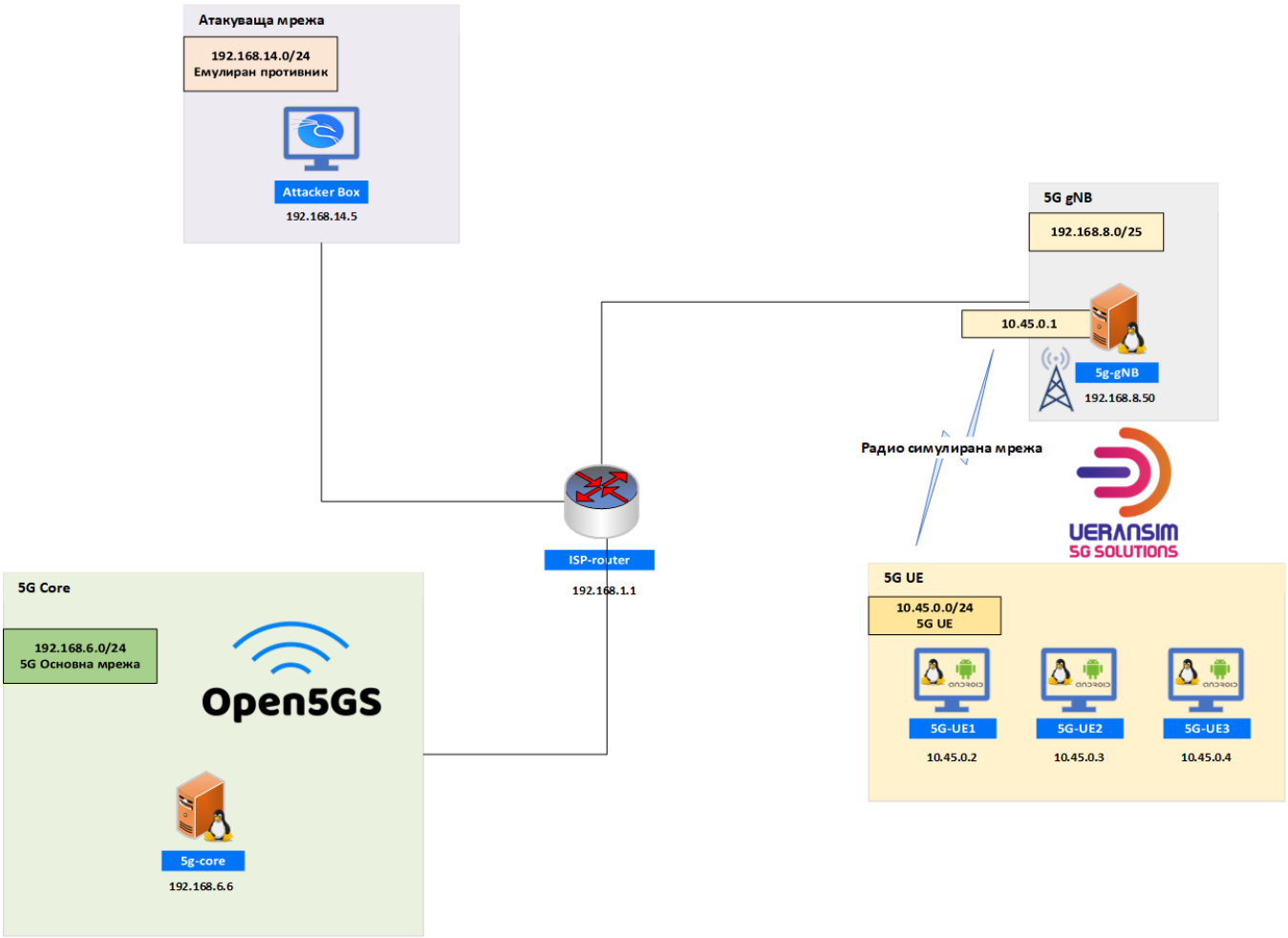
специализиран киберполигон. За целта е изградена самостоятелна 5G частна мрежа (5G SA), която предоставя услуги за комуникация на данни (без глас) на потребители, включени чрез потребителско оборудване (UE). Потребителското оборудване (клиентите), свързани към 5G мрежата, получават достъп до услугите на мрежата и Интернет.

Публичната мобилна мрежа, Public Land Mobile Network (PLMN) на 5G SA използва мобилен код на държавата, Mobile Country Code (MCC) 999 и мобилен мрежов код (MNC) 70. 5G SA мрежата се състои от основна мрежа (CN) и мрежа за радиодостъп (RAN), свързани помежду си с останалите външни мрежи и Интернет, чрез защитна стена/рутер. Виртуалната инфраструктура на изградената 5G мрежа е показана на фигура 3.6. 5G SA инфраструктурата е внедрена в следните мрежи:

- 5G Core – служи за имплементиране на услугите в основна мрежа с адрес IPv4: 192.168.6.0/24.
- 5G gNB – служи за имплементиране на базова станция с адрес на мрежата IPv4: 192.168.8.0/24.
- 5G UE – мрежа с дефиниран IP обхват, от която потребителското оборудване получава своите IP адреси за потребителски трафик.

Функционалността на виртуалните машини в мрежата за радиодостъп (gNB и UE) зависи от правилното функциониране на основните 5G мрежови функции. Виртуалните машини в тази мрежа са дефинирани както следва: 5g-core – емулатор на 5G основна мрежа, 5g-gNB – емулатор на базова станция gNB и 5G-UE1-3 – емулатор на потребителско оборудване. Конфигурацията на мрежовите услуги във внедрената 5G мрежа с дефинираните портове е представена в таблица 8.

Услугите, предоставяни от виртуална машина 5g-core са дефинирани както следва: *5g-db, 5g-webui, 5g-nrf, 5g-udr, 5g-udm, 5g-ausf, 5g-amf, 5g-smf, 5g-pcf, 5g-bsf, 5g-upf* и *5g-nssf*. Мрежовите функции в основна мрежа са софтуерно имплементирани, чрез проекта Open5GS върху виртуална машина с операционна система Ubuntu Server 24.04 LTS. На фигура 3.7 се вижда реализирана мрежова услуга AMF.



Фиг. 3.6. Виртуална инфраструктура на комуникационна мрежа от 5-то поколение.

Таблица 8. Услуги на изградена 5G мрежа

Услуга	Роля	IPv4	Машина	Мрежов сегмент	Порт	Име на услугата
5g-amf	Управление на достъпа и мобилността	127.0.0.5	5g-core	5G Core	SBI – TCP/7777 N2 – SCTP/38412	open5gs-amfd.service
5g-smf	Функция за управление на сесиите	127.0.0.4	5g-core	5G Core	N4 – UDP/8805 SBI – TCP/7777	open5gs-smfd.service
5g-upf	Маршрутизиране на потребителски трафик	127.0.0.7	5g-core	5G Core	N4 – UDP/8805 N3 – UDP/2152	open5gs-upfd.service
5g-ausf	Сървърна функция за удостоверяване	127.0.0.11	5g-core	5G Core	SBI - TCP/7777	open5gs-ausfd.service
5g-nrf	Функция на NF хранилището	127.0.0.10	5g-core	5G Core	SBI - TCP/7777	open5gs-nrfd.service
5g-udm	Унифицирано управление на данни	127.0.0.12	5g-core	5G Core	SBI - TCP/7777	open5gs-udmd.service
5g-udr	Унифицирано хранилище за данни	127.0.0.20	5g-core	5G Core	SBI - TCP/7777	open5gs-udrd.service
5g-pcf	Функция за управление на политиките	127.0.0.13	5g-core	5G Core	SBI - TCP/7777	open5gs-pcfd.service
5g-nssf	Функция за избор на мрежов сегмент	127.0.0.14	5g-core	5G Core	SBI - TCP/7777	open5gs-nssfd.service
5g-bsf	Функция за поддръжка на сесиите	127.0.0.15	5g-core	5G Core	SBI - TCP/7777	open5gs-bsfd.service
5g-db	MongoDB база данни	192.168.6.6	5g-core	5G Core	TCP/3306	mongod.service
5g-webui	Уеб приложение за управление на абонати	192.168.6.6	5g-core	5G Core	HTTP - TCP/80 HTTPS -TCP/443	nginx.service обратно прокси към порт 9999
		127.0.0.1	5g-core	5G Core	HTTP - TCP/9999	open5gs-webui.service
5g-gnb	5G базова станция / gNB емулатор	192.168.8.50	5g-gNB	5G gNB	инициира N2/N3 връзка	nr-gnb.service
5g-ue1-3	5G UE емулатор	10.45.0.2-4	5G-UE1-3	5G UE	Потребителски терминали	nr-ue.service

Фиг. 3.7. Системна услуга на AMF

Освен мрежовите функции, на виртуална машина в основна мрежа се намират базата данни (5g-db) и уеб приложение за управление на абонатите (5g-webui). Базата данни (5g-db) е имплементирана чрез система за управление на бази от данни MongoDB 8.0.10, съхраняваща информация за потребителите, конфигурация на системата и мрежата. Базата данни open5gs на MongoDB сървъра съдържа следните таблици: accounts – използва се за съхраняване на идентификационни данни за потребителите, достъпващи WebUI приложението; profiles - използва се за съхраняване на зададени профили на абонати; sessions – използва се за съхраняване на сесии за webUI приложението; subscribers – информация за абонати, IMSI, Ki, OPc, DNN и др.

Абонатите с IMSI 99970##000000[8-10] са единствените активни абоната в симулационния сценарий. Те са свързани с 5g-UE1-3 и са им присвоени статични IPv4 адреси 10.45.0.2-4.

Уеб приложението за управление на абонати (5g-webui) е базирано на приложението open5gs-webui. Услугата 5g-webui изпълнява nginx уеб сървър като обратно прокси, за да пренасочва портове 80/443 към localhost порт 9999 на уеб приложението.

gNB и UE са реализирани с помощта на проекта с отворен код UERANSIM, работещ на Ubuntu 24.04 LTS. Приложението позволява емулиране на 5G базова станция (gNB) и множество UE, които могат да се интегрират с 5G основната мрежа. Приложението UERANSIM gNB е внедрено и работи на машината 5g-gNB, а приложението UERANSIM UE работи на машини 5G-UE1-3.

3.4. Изводи

1. Разработеният специализиран киберполигон представлява цялостна среда за симулация и анализ на киберсигурността, която обединява виртуализация, автоматизация и управление на сценарии в единна платформа.
2. Архитектурата на полигона е реализирана в три слоя — потребителски интерфейс, мениджър на способности и блок за виртуализация, което осигурява модулност, гъвкавост и възможност за разширяване на функционалността.
3. Разработеният модел на сценарии позволява формализиране на симулационните процеси чрез ясно дефинирани параметри (време, дейност, цел, вид атака и мрежова топология), което улеснява възпроизводимостта на експериментите.
4. Създадената симулационна среда, базирана на Open5GS и UERANSIM, предоставя възможност за реалистично изследване на сигурността в 5G инфраструктурни мрежи.

Глава 4 – Резултати от проведените изследвания за сигурност

4.1. Резултати от специализиран киберполигон

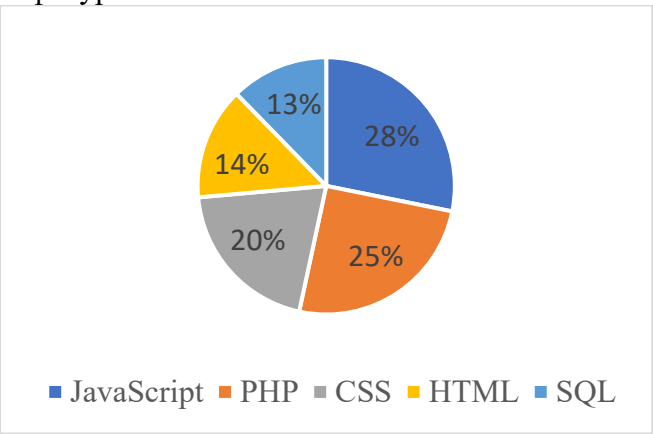
Специализираният киберполигон DigiCode е изграден в центъра за съхранение на данни в рамките на факултет „Артилерия, ПВО и КИС“. Платформата е изградена в 296709 реда код. На фигура 4.1 са представени обобщените данни относно използваните програмни езици при реализацията на платформата.

Извършен е сравнителен анализ на функционалността на киберполигона спрямо други платформи за изграждане на симулационни и тестови среди (Cyber Range). Резултатите от анализа са представени в таблица 9.

Проведен е експеримент относно средното време за създаване на инфраструктурни среди. Тъй като платформата stratocyberlab не поддържа виртуални машини, експериментът е проведен спрямо останалите две платформи. За целта са дефинирани три сценария.

В Сценарий 1 се отчита времето за разгръщане на инфраструктура, съставена от 1 виртуална машина – Ubuntu Server 24.04. В сценарий 2 се отчита времето за разгръщане на мрежа от 5

виртуални машини. В сценарий 3 се отчита времето за разврьщане на инфраструктура съставена от 3 подмрежи и 15 виртуални машини. Платформите са тествани в еднакви условия и хардуерни ресурси. Резултатите относно времето за разврьщане и стартиране на всеки от сценариите са представени в таблица 10 и фигура 4.2.



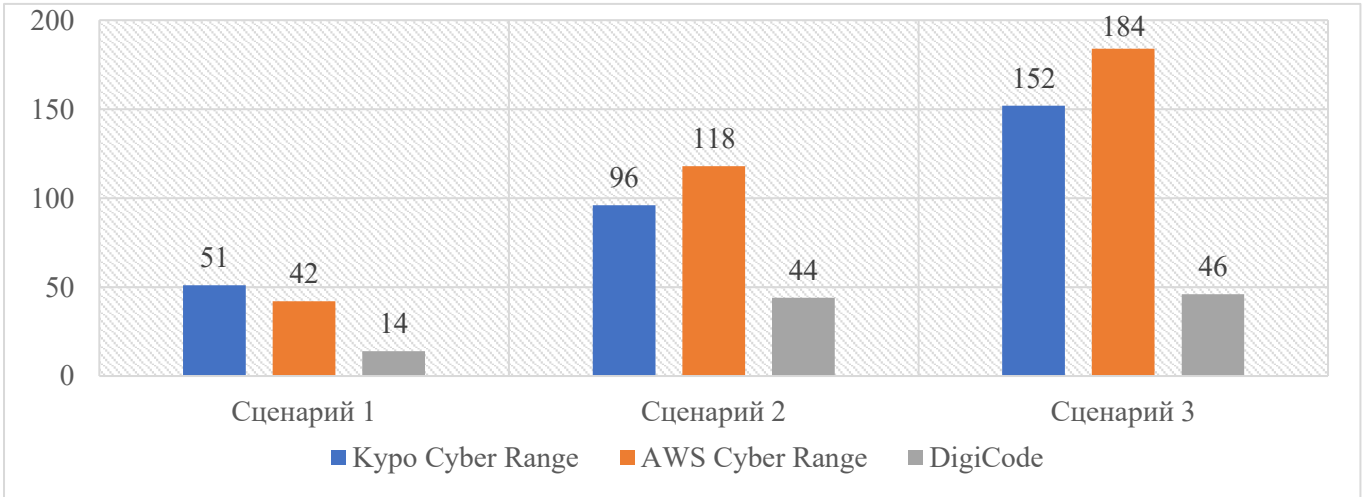
Фиг. 4.1. Утилизация на програмни езици в изградената платформа

Таблица 9. Сравнителен анализ на платформи Cyber Range

Функционалност	Kypo Cyber Range	AWS Cyber Range	Stratocyberlab	DigiCode
Оркестрация	X		X	X
Виртуални машини	X	X		X
Контейнеризация			X	частично
Симулация на атаки				X
Симулация на потребителска активност				X
Създаване на сценарии и съдържание	X		X	X
Събиране на данни и анализ	X	X	X	X
Оценка и докладване				X
Инструменти за инструктори	X		X	X
Интерактивни материали				X
Обучения по киберсигурност	X		X	X
Тестване на сигурност		X		X
Изграждане на учения	X			X

Таблица10. Време за разврьщане на виртуални инфраструктури

Платформа	Сценарий 1 (с)	Сценарий 2 (с)	Сценарий 3 (с)
Kypo Cyber Range	51	96	152
AWS Cyber Range	42	118	204
DigiCode	14	44	46



Фиг. 4.2. Време за разврьщане на инфраструктурни среди

Новоизградената платформа е в състояние да емулира една машина Ubuntu Server за под 15 секунди в сценарий 1. Резултатите от сценарий 3 показват, че увеличаването на броя виртуални машини не оказва съществено влияние върху времето за разврьщане на инфраструктура в рамките на новоизградената платформа. Причината се дължи на това, че платформата DigiCode използва шаблони на виртуалните машини, като времето за клониране и създаване на нова виртуална машина се свежда до под 1 секунда. Останала част от времевия период до осигуряването на достъп в изградената инфраструктура се използва за буут на системите.

4.2. Разузнаване за киберзаплахи

За верификация на предложения модел на разузнаване за киберзаплахи се извличат техниките по матрици от връзки група-техника и се сравняват получените честоти относно утилизация на използвана техника. В началото се дефинират S_{target} и R_{target} според предложения сценарий.

$$S_{target} = \{telecom, communications, mobile\} \quad (4.1)$$

$$R_{target} = \{Europe, NATO, Balkans\} \quad (4.2)$$

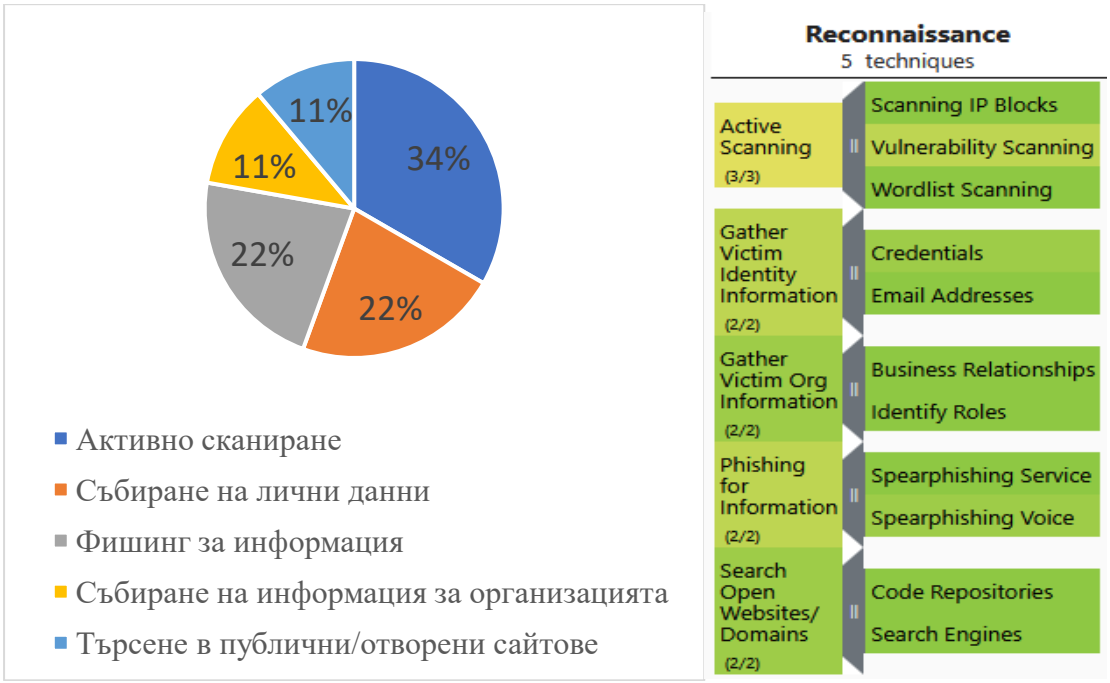
След прилагане на филтрация по целеви сектор и географски регион са идентифицирани 15 противникови групи, представени в таблица 12. Всяка от идентифицираните групи е съпоставена с нейните асоциирани ATT&CK техники в отделен слой на MITRE ATT&CK навигатора. От получената бинарна матрица се изчислява честотата на използваните техники. Това се прилага в MITRE ATT&CK навигатора, чрез задаване на теглови коефициент (оценка) със стойност 1 за използвана техника и 0 в противен случай за всеки слой. След това получените слоеве се обединяват (сумират) в един слой, който подчертава най-често използваните техники. Най-често използваните техники, идентифицирани в този поднабор, са представени на фигури 4.3, 4.4 и 4.5, оцветени от зелено към червено според честотата на тяхната утилизация.

Таблица 12. Групировки атакували сектор телекомуникации в НАТО/Европа

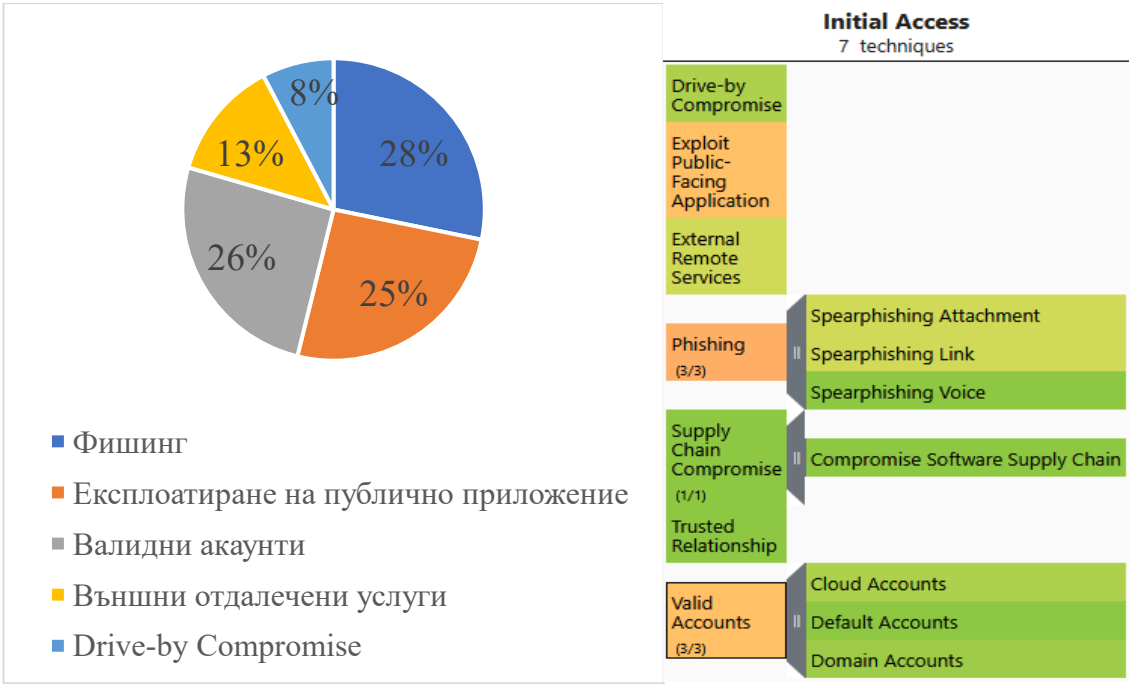
№	Групировка	Източник	Първоначално забелязване
1.	G0073 - APT19	Китай	2017
2.	G0087 - APT39	Иран	2014
3.	G0096 - APT41	Китай	2012
4.	G1023 - APT5	Китай	2007
5.	G0143 - Aquatic Panda	Китай	2020
6.	G0135 - BackdoorDiplomacy	Китай	2017
7.	G1006 - Earth Lusca	Китай	2019
8.	G1003 - Ember Bear	Русия	2020
9.	G0093 - GALLIUM	Китай	2012
10.	G1004 - LAPSUS$	Великобритания	2021
11.	G0095 - Machete	Испания	2010
12.	G0069 - MuddyWater	Иран	2017
13.	G1045 - Salt Typhoon	Китай	2019
14.	G1015 - Scattered Spider	Великобритания/САЩ	2022
15.	G1044 – APT42	Иран	2015

Фигура 4.3 представя резултатите, свързани с разпределението на използваните техники за разузнаване от злонамерени групировки, атакували сектора на мобилните комуникации и телекомуникации, в регионите НАТО, Европа и Балканите. Резултатите показват, че най-често прилаганите техники са активно сканиране, фишинг за информация и събиране на идентификационни данни за жертвата.

Фигура 4.4 илюстрира най-често прилаганите техники за първоначален достъп. Анализът показва силна зависимост от методи, базирани на социално инженерство и компрометирани идентификационни данни, допълнени от експлоатация на уязвими публично достъпни приложения. Наблюдавани са също вектори за отдалечен достъп, като външни услуги и опортюнистични компромати, макар и с по-ниска честота.

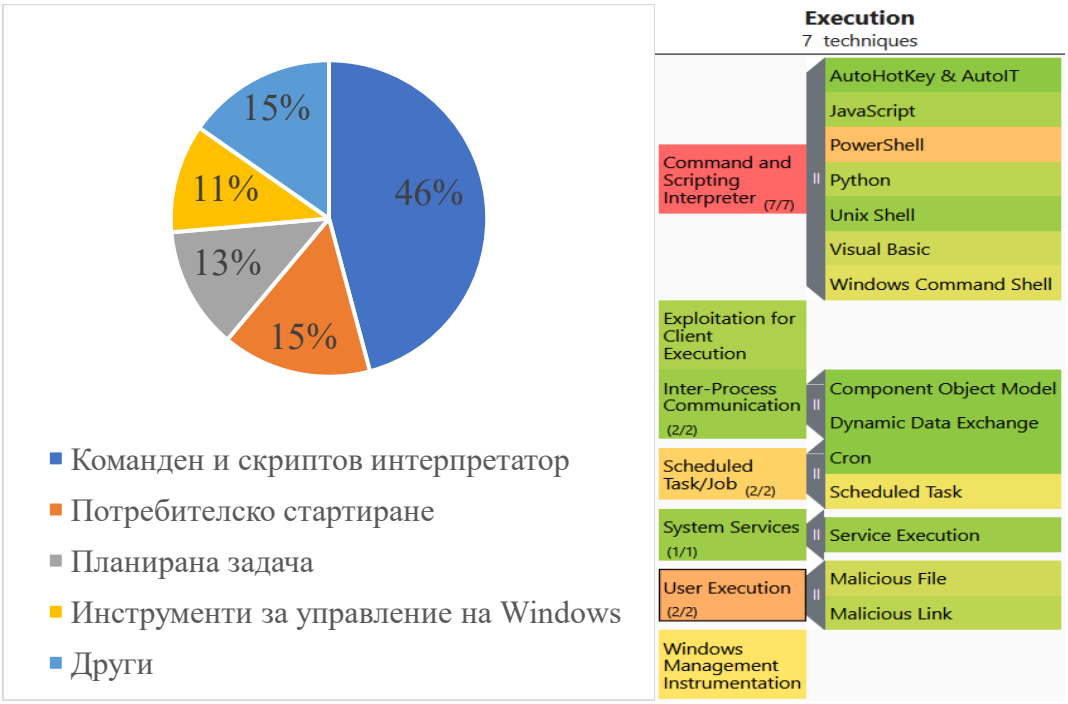


Фиг. 4.3. Използвани техники за разузнаване



Фиг. 4.4. Използвани техники за първоначален достъп

Фигура 4.5. представя техниките с най-висок интензитет в тактика „Изпълнение“. Тактиката обхваща техниките и методите, чрез които потребителят стартира зловредния код при получаването на достъп до целевата система. Резултатите подчертават интензивното използване на командни и скриптови интерпретатори, които осигуряват инструменти като PowerShell и Python (Dimov & Savova, Antivirus Performance Evaluation against PowerShell Obfuscated Malware, 2024).



Фиг. 4.5. Разпределение на техники за стартиране на зловреден код

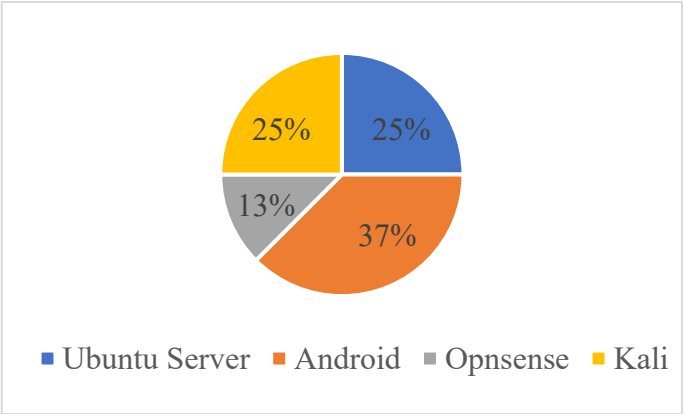
4.3 Резултати при тестване на сигурността, чрез емулация на противник

Информацията за разузнаване на киберзаплахи се моделира като входни данни в противниковия модел, след което се стартира генерирания емулационен план върху изградената комуникационна мрежа от пето поколение в специализирания киберполигон и се събират резултати относно успеваемостта на всяка от стъпките при реализирането на различните техники и процедури. Специализираният киберполигон е въведен в експлоатация на 18.01.2024 г. Данните са събрани за период от 6 месеца – до 18.07.2024 г. По време на тестовете в посочения период са развърнати над 140 инфраструктурни мрежи и внедрени над 1000 виртуални машини, като резултатите относно утилизацията на предвидения инфраструктурен модел са представени в таблица 13.

Таблица 13. Внедрени виртуални машини

№	Операционна система	Роля	Брой
1.	Ubuntu Server 24.04 LTS	5G Core	142
2.	Ubuntu Server 24.04 LTS	5G gNB	142
3.	Android 14	5G UE	426
4.	Opnsense 24.4	Маршрутизатор	142
5.	Kali 2024.4	Атакуваща машина	284

Статистическите данни за брой стартирани виртуални машини при тестване на сигурността са нормализирани в процентно съотношение и представени на фигура 4.6.

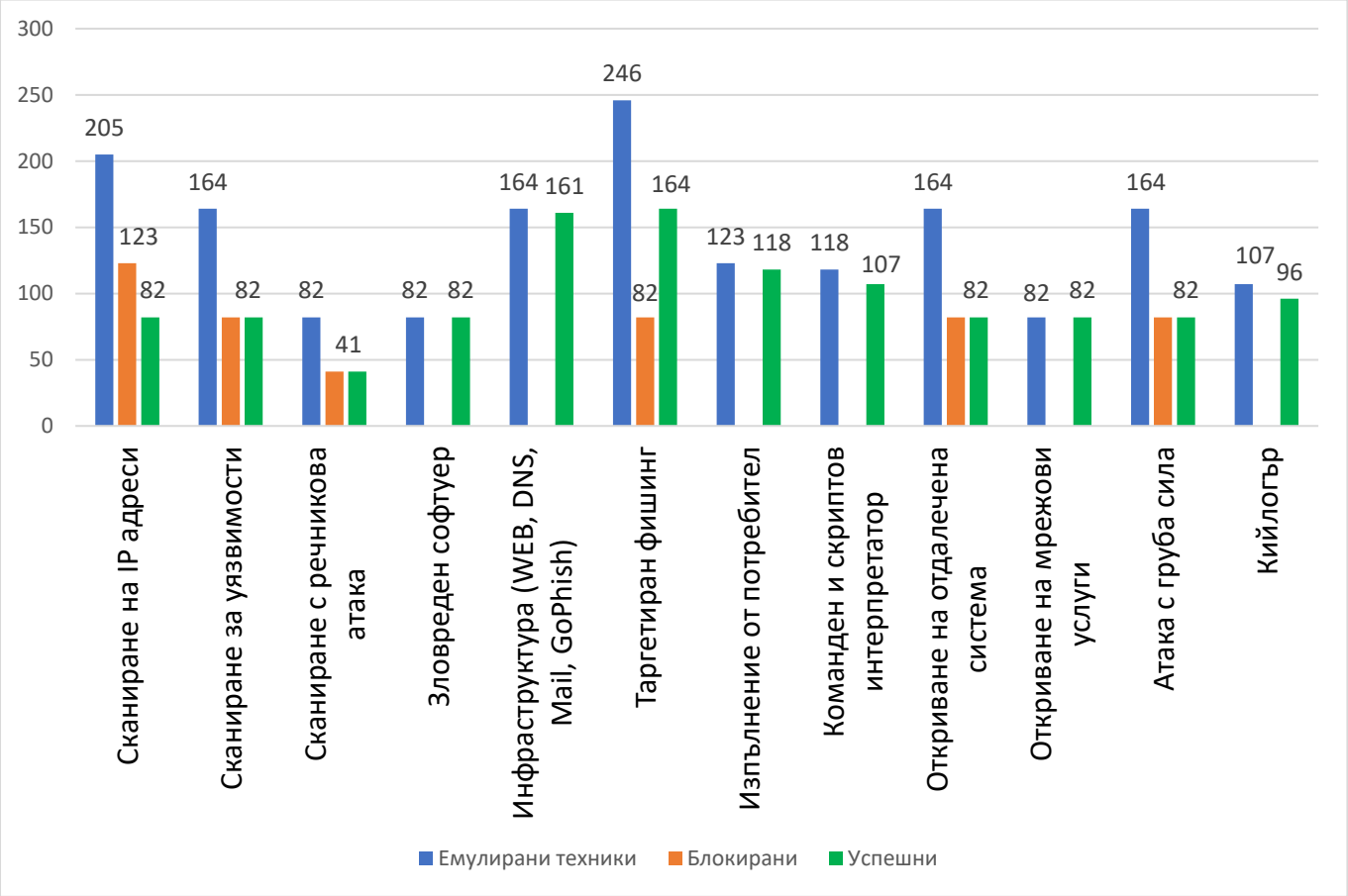


Фиг. 4.6. Внедрени машини в 5G инфраструктурен модел

Симулирани са над 1700 техники, като резултатите са представени в таблица 14 и визуализирани на фигура 4.7.

Таблица 14. Получени резултати при емулация на техники

Тактика	**Техника**	**Брой**	**Успешни**
Разузнаване	Сканиране на IP адреси	205	82
	Сканиране за уязвимости	164	82
	Сканиране с речникова атака	82	41
Изграждане на ресурси	Зловреден софтуер	82	82
	Инфраструктура (WEB, DNS, Mail, GoPhish)	164	161
Първоначален достъп	Таргетиран фишинг	246	164
Изпълнение	Изпълнение от потребител	123	118
	Команден и скриптов интерпретатор	118	107
Откриване	Откриване на отдалечена система	164	82
	Откриване на мрежови услуги	82	82
Достъп до кредънции	Атака с груба сила	164	82
	кийлогър	107	96



Фиг. 4.7. Емулирани техники по модел MITRE ATT&CK

В таблица 15 са посочени засегнатите мрежови сегменти относно успешните техники.

Нека:

- $A =$ брой опити на емулирана техника;
- $S =$ брой усшени опити за емулирана техника;
- $t =$ идентификатор на техника по модел $MITRE\ ATT\&CK$.

Тогава ефективността на защитата, Mitigation Effectiveness (ME) за дадена техника може да се изчисли като:

$$ME_t = \left(1 - \frac{S}{A}\right).100\% \tag{4.3}$$

Таблица 15. Успешни техники в мрежови сегменти

Техника	**Бр. успешни/ емулирани**	**Мрежов сегмент**
Сканиране на IP адреси	82/205	5G Core, 5G gNB
Сканиране за уязвимости	82/164	5G Core, 5G gNB
Сканиране с речникова атака	41/82	5G Core
Зловреден софтуер	82/82	Атакуваща мрежа
Инфраструктура (WEB, DNS, Mail, GoPhish)	161/164	Атакуваща мрежа
Таргетиран фишинг	164/246	5G UE
Изпълнение от потребител	118/123	5G UE
Команден и скриптов интерпретатор	107/118	5G UE
Откриване на отдалечена система	82/164	5G Core, 5G gNB
Откриване на мрежови услуги	82/82	5G Core, 5G gNB
Атака с груба сила	82/164	5G Core, 5G gNB
Кийлогър	96/107	5G UE

Метрика 4.3 показва каква част от опитите за дадена техника са успешно блокирани. Тъй като някои техники са отработвани в противниковата инфраструктура, то метриката е пресметната за техники емулирани в останалите мрежови сегменти.

За техника *Сканиране на IP адреси*, ефективността на защитата е изчислена като:

$$ME_{T1595.001} = \left(1 - \frac{82}{205}\right).100\% = 60.00\% \quad (4.4)$$

За техники *Сканиране за уязвимости* и *Сканиране с речникова атака* $ME_{T1595.002,\ T1595.003} = 50\%$.

За техника *Таргетиран фишинг*:

$$ME_{T1566} = \left(1 - \frac{164}{246}\right).100\% = 33.33\% \quad (4.5)$$

Тъй като техника *Изпълнение от потребител*, се изпълнява от автоматизиращ скрипт за симулация на потребителска активност, то тук вградените защитни механизми не оказват влияние. За техника *Команден и скриптов интерпретатор*:

$$ME_{T1059} = \left(1 - \frac{107}{118}\right).100\% = 9.32\% \quad (4.6)$$

За техники *Откриване на отдалечена система* и *Атака с груба сила* $ME_{T1018,\ T1110} = 50\%$.

След установяване на сесия, атакуващият пренасочва мрежовият трафик на атакуваща система през мрежата на компрометираният потребителски терминал. Експериментирано е със Socks_proxy в meterpreter сесия, както и с инструмент Chisel. Трафикът се пренасочва успешно, т.е. системата по подразбиране не осигурява защита срещу атаки с пренасочване на мрежовият трафик. В този случай $ME_{\mathrm{T1046}} = 0\%$.

За техника *Кийлогър*:

$$ME_{\mathrm{T1056.001}} = \left(1 - \frac{96}{107}\right).100\% = 10.28\% \quad (4.7)$$

Тъй като са използвани множество техники, общата агрегирана ефективност на защитата се изчислява като:

$$ME' = \left(1 - \frac{S'}{A'}\right).100\%, \quad (4.8)$$

където S'- агрегиран брой успешни опити за всички техники и A'- общ брой опити за всички емулирани техники.

$$S' = \sum_{i=1}^{n} S_i \quad (4.9)$$

$$A' = \sum_{i=1}^{n} A_i\,, \quad (4.10)$$

Изчислява се общ брой опити за съответните техники A': Сканиране на IP адреси $A_{\mathrm{T1595.001}} = 205$; Сканиране за уязвимости $A_{\mathrm{T1595.002}} = 164$; Сканиране с речникова атака $A_{\mathrm{T1595.003}} = 82$; Таргетиран фишинг $A_{\mathrm{T1566}} = 246$; Команден и скриптов интерпретатор $A_{\mathrm{T1059}} = 118$; Откриване на отдалечена система $A_{\mathrm{T1018}} = 164$; Откриване на мрежови услуги $A_{\mathrm{T1046}} = 82$; Атака с груба сила $A_{\mathrm{T1110}} = 164$; Кийлогър $A_{\mathrm{T1056.001}} = 107$;

Тогава от уравнение 4.10, $A' = 1332$.

Изчислява се агрегиран брой успешни опити от всички емулирани техники S': Сканиране на IP адреси $S_{\mathrm{T1595.001}} = 82$; Сканиране за уязвимости $S_{\mathrm{T1595.002}} = 82$; Сканиране с речникова атака $S_{\mathrm{T1595.003}} = 41$; Таргетиран фишинг $S_{\mathrm{T1566}} = 164$; Команден и скриптов интерпретатор $S_{\mathrm{T1059}} = 107$; Откриване на отдалечена система $S_{\mathrm{T1018}} = 82$; Откриване на мрежови услуги $S_{\mathrm{T1046}} = 82$; Атака с груба сила $S_{\mathrm{T1110}} = 82$; Кийлогър $S_{\mathrm{T1056.001}} = 96$;

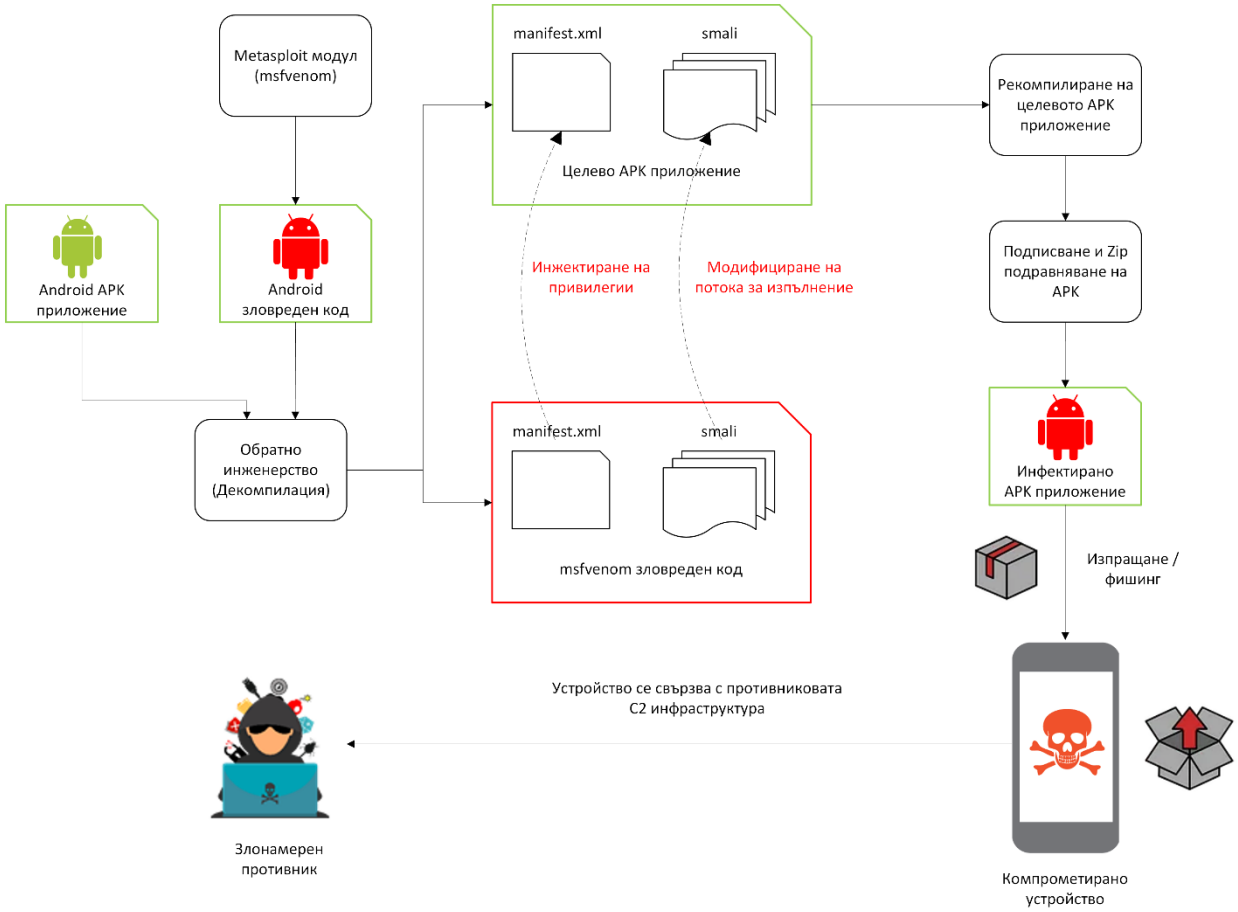
Тогава от уравнение 4.9, $S' = 818$.

От 4.8, изчислената агрегирана ефективност на защитата спрямо емулираните техники ME' е:

$$ME' = \left(1 - \frac{818}{1332}\right).100\% = 38.58\% \qquad (4.11)$$

4.4. Моделиране на система за вмъкване на код в APK приложения

Най-популярният инструмент за получаване на достъп в Android OS е генераторът на зловреден софтуер msfvenom в рамката Metasploit (таблица 16). Въпреки това генерираният софтуер не работи под Android след версия 14, поради надграждане на версията в приложно програмният интерфейс на пакета от софтуери за разработка, Software Development Kit (SDK). На фигура 4.10 е представена методология за инжектиране на код в APK приложение, която коригира поддръжката за msfvenom при версии на Android след 14 и допълнително разширява неговите възможности, като зловредния код се инжектира в легитимно APK приложение.



Фиг. 4.10. Методология за инжектиране на код в легитимно APK приложение

Android приложенията се разпространяват под формата на APK (Android Package) файлове. Въпреки че платформата Android използва механизми за подписване и контрол на разрешенията,

APK файловете могат да бъдат напълно декомпилирани, модифицирани и повторно компилирани с помощта на публично достъпни инструменти като apktool.

Методологията, по-която работи инструмента, е представена в последователни стъпки: *Генериране на зловреден код, Декомпилиране на APK файлове, Инжектиране на зловреден код, Вмъкване на hook, Компилиране на модифицираното приложение, Подписване на APK* и *Подравняване на APK*. Проведен е експеримент за тестване на функционалността на модела спрямо други инструменти за създаване на зловредни APK файлове. Тествани са четири инструмента: ApkInfector, backdoor-apk, msfvenom и apk-mitm. Изборът на инструменти за сравнение е направен въз основа на емпирично проучване в GitHub като са избрани четирите най-популярни инструмента към месец март 2024 г. представени в таблица 16.

Таблица 16. Избор на инструменти

Инструмент	**Популярност в Github (бр. звезди)**
Metasploit msfvenom	35 800
apk-mitm	4 300
backdoor-apk	2 300
apkinfector	292

Метриките за оценка са подбрани по четири ключови количествени показателя, формулирани по-долу. Нека:

- $A = \{a_1, a_2, \dots, a_n\}$: множеството от входни APK файлове;
- $S_i \in \{0,1\}$: индикатор дали инжекцията върху a_i е била успешна;
- $\mathrm{E}_i \in \{0,1\}$: индикатор дали зловреднният софтуер е изпълнен успешно на устройството;
- $\mathrm{T}_i \in R+$: време (в секунди) за изпълнение на инструмента върху a_i;
- $D_i \in [0,1]$: дял на антивирусните софтуери, които са класифицирали полученият APK като зловреден.

$D_i = \frac{d_i}{m}$, където d_i е броят антивирусни засекли софтуера за даден опит и m е общ брой антивирусни системи, използвани в сканирането на VirusTotal.

Следните показатели се изчисляват за всеки инструмент: Успеваемост на инжектиране, Injection Success Rate (ISR):

$$ISR = \frac{1}{n}\sum_{i=1}^{n} S_i \tag{4.12}$$

Успеваемост на изпълнение на модифицирания софтуер, Payload Execution Rate (PER):

$$PER = \frac{1}{n}\sum_{i=1}^{n} E_i \tag{4.13}$$

Средно време за изпълнение ($\bar{T}$) измерва средното време за обработка на един APK файл:

$$\bar{T} = \frac{1}{n}\sum_{i=1}^{n} T_i \tag{4.14}$$

Степен на преодоляване на антивирусна защита, Evasion Rate (ER):

$$ER = 1 - \left(\frac{1}{n}\sum_{i=1}^{n} D_i\right) \tag{4.15}$$

Инструментите се тестват в изградения специализиран киберполигон при контролирана среда и еднакви условия. За всяко приложение се прилагат следните стъпки:

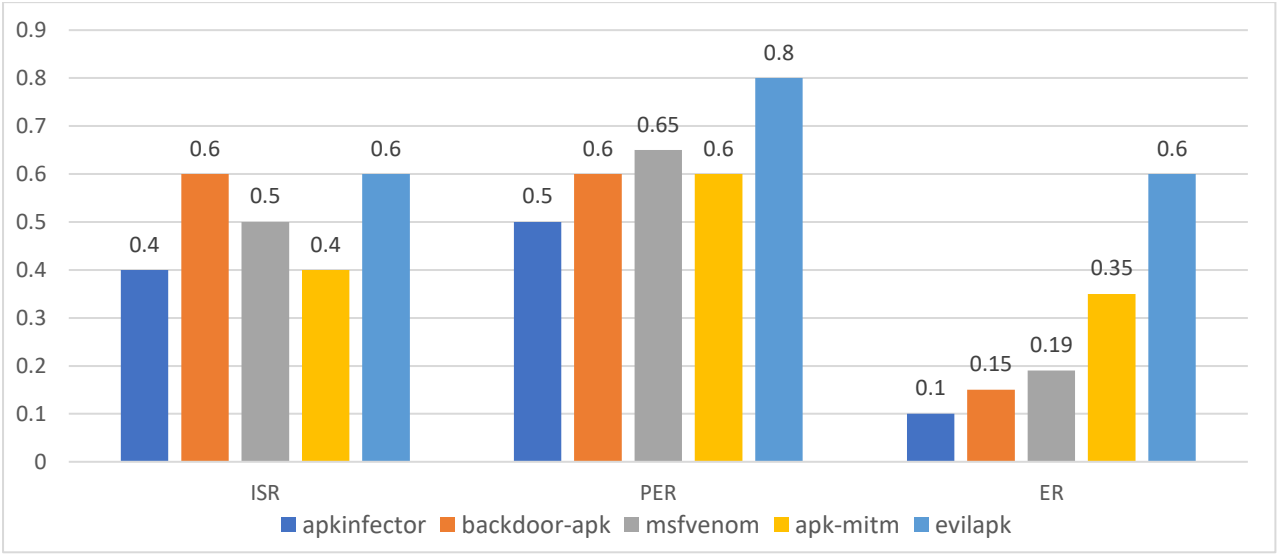
1. Избира се легитимен APK от множеството *A*
2. Генерира се тестов зловреден софтуер чрез msfvenom
3. Инструментът опитва инжектиране на зловредния софтуер
4. Засича се времето за обработка
5. Полученият APK се изпълнява в емулатор за тестване на функционалността и се анализира с VirusTotal за засичане от антивирусен софтуер

Всеки инструмент е тестван с *n* = 50 APK файла. Резултатите са сравнени с тези, получени от новосъздаденият инструмент (evilapk). Данните са представени в таблица 17.

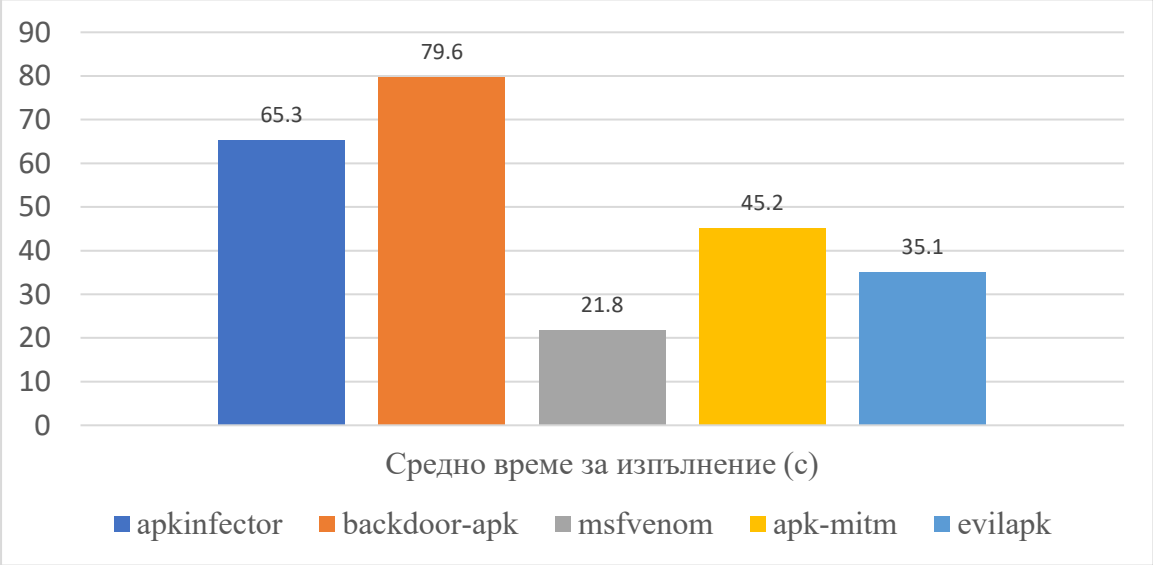
Таблица 17. Получени резултати от избраните инструменти

Инструмент	***ISR***	***PER***	$\overline{T}$ **(с)**	***ER***
apkinfector	0.4	0.5	65.3	0.1
backdoor-apk	0.6	0.6	79.6	0.15
msfvenom	0.5	0.65	21.8	0.19
apk-mitm	0.4	0.6	45.2	0.35
Средно ариметично	0.475	0.5875	52.975	0.1975
evilapk	0.6	0.8	35.1	0.6

На фигура 4.12. е представена диаграма за сравнение на получените стойности. На фигура 4.13 са представени получените резултати относно средното време за изпълнение на всеки от тестваните инструменти.



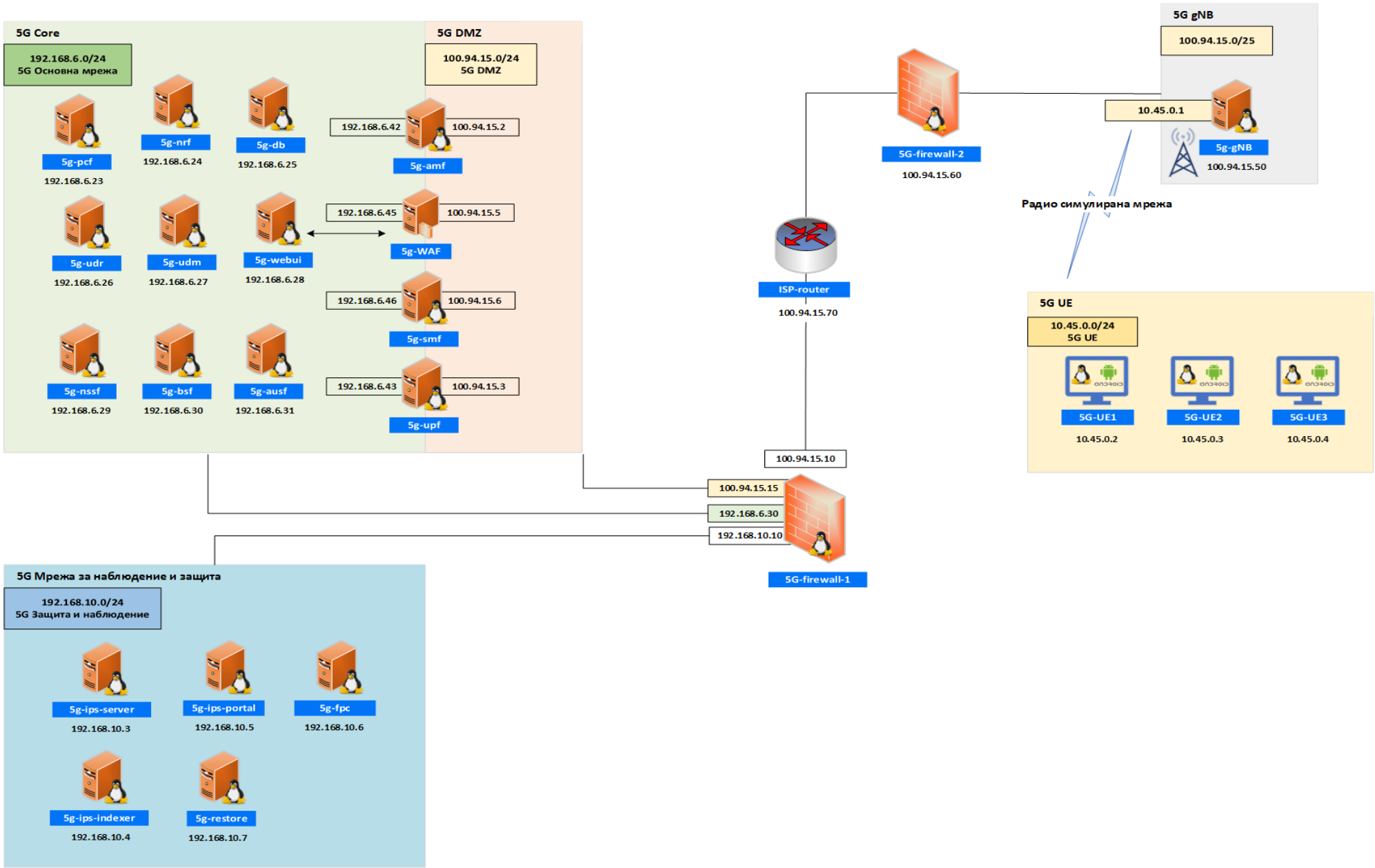
Фиг. 4.12. Сравнение на резултатите от избраните инструменти



Фиг. 4.13. Получени резултати за средно време на изпълнение

4.5 Система за киберсигурност на когнитивни комуникационни мрежи

На база на получените резултати при тестване на сигурността в 4.3, е предложена система за киберсигурност, целяща повишаване на устойчивостта и осигуряване на непрекъснат контрол върху компонентите на изследваната комуникационна инфраструктура. Архитектурата на предложената система за киберсигурност е представена на фигура 4.14.



Фиг. 4.14. Система за киберсигурност на 5G мрежова инфраструктура.

Предложената система за киберсигурност е съставена от следните компоненти:

- 5G-WAF (Web Application Firewall) – защитна стена за уеб приложение;
- 5G-firewall-1 – защитна стена за периметъра на мрежата;
- 5G-firewall-2 – защитна стена за филтрация на мрежов трафик към базовата станция и потребителските терминали;
- 5G-ips-server – сървър за анализ на телеметрични данни за сигурност;
- 5G-ips-indexer – сървър за индексиране и проследяване на събития за сигурност;
- 5G-ips-portal – сървър на уеб приложение за мониторинг и управление на системата за киберсигурност;
- 5G-ips-fpc (Full Packet Capture) – сървър за записване и анализ на мрежов трафик;
- 5g-restore – сървър за архивиране и възстановяване.

Защитната стена 5G-WAF служи за активно блокиране на зловредни заявки на приложен слой (HTTP/S) към уеб приложението на 5G системата за управление на абонатите. Конфигурацията включва операционна система Ubuntu Server 24.04, пакетен филтър ModSecurity и Wazuh агент.

Защитната стена 5G-firewall-1 служи за маршрутизация, анализ и блокиране на зловреден мрежови трафик в периметъра на мрежата и вътрешната мрежа на мрежовите функции. Конфигурацията на защитната стена включва софтуер за филтрация и маршрутизация на мрежов трафик OpnSense 24.10. За повишаване на сигурността, защитната стена разделя 5G мрежовите функции в две подмрежи и дефинира подмрежа за наблюдение и защита:

- 5G Core с IPv4 адрес на мрежата 192.168.6.0/24 - вътрешна подмрежа, ограничена за публичен достъп от други мрежи;
- 5G DMZ (Demilitarized Zone) с IPv4 адрес на мрежата 100.94.15.0/24 – демилитаризирана зона на публична мрежа за услуги, изискващи публичен достъп;
- 5G мрежа за наблюдение и защита с IPv4 адрес на мрежата 192.168.10.0/24 – вътрешна подмрежа с включени услуги за мониторинг, анализ, активен отговор и реакция на събития за сигурност.

Защитната стена осигурява достъп на 5G операторите до вътрешната подмрежа посредством криптиран VPN тунел, реализиран чрез OpenVPN. Защитната стена е конфигурирана да маршрутизира 5G трафикът към мрежовата функция за пренос на потребителски данни UPF и функцията за управление на достъпа и мобилността AMF посредством правилата за входящ трафик, представени в таблица 18.

Таблица 18. Правила за филтрация на входящ мрежов трафик

Източник (IPv4)	**Порт**	**Дестинация (IPv4)**	**Порт**	**Действие**
*	*	*	*	блокиране
gNB – 100.94.15.60	*	amf - 100.94.15.2/24	N2 - SCTP/38412	разрешаване
gNB – 100.94.15.60	*	upf - 100.94.15.3/24	N3 - UDP/2152	разрешаване
gNB – 100.94.15.60	*	ips – 192.168.10.3	TCP/1514,1515	разрешаване
5G Core	*	ips – 192.168.10.3	TCP/1514,1515	разрешаване
5G Core	*	restore - 192.168.10.7	SMB – TCP/445	разрешаване

Сървърът 5G-ips-server служи за анализиране на данните, постъпващи от компонентите на 5G мрежата, включително мрежовите устройства, анализира ги чрез правила, създава аларми и управлява активните отговори. Конфигурацията му включва операционна система Ubuntu Server 24.04 и софтуер за откриване и отговор на кибератаки Wazuh сървър. За повишаване на киберсигурността са конфигурирани правила за разпределен активен отговор, които следят мрежовия трафик към хостовете и услугите на 5G мрежата и при опити за активно сканиране и налучкване на пароли, добавят правило в защитната стена, забраняващо достъпа от IP адреса на атакуващия и го синхронизират с всички хостове в 5G мрежата (Slavyanov & Dimov, 2024). На фигура 4.15 е представена конфигурацията за активен отговор.

- правило 5710 – генерира събитие, когато атакуващ IP адрес опита налучкване на парола по ssh;

- правило 100000 – генерира събитие, когато атакуващ IP адрес извърши активно сканиране на портове;
- правило 100001 – генерира събитие, когато атакуващ IP адрес опита налучкване на парола при автентикация с уеб приложението за управление на абонати 5g-webui;
- правило 100002 – генерира събитие при откриване на зловреден софтуер;
- правило 100003 – генерира събитие, за успешна автентикация на хост по ssh от IP адрес на друга мрежа или в часови диапазон от 17:00 до 08:00 ч.;
- правило 100004 – генерира събитие, за опит на модификация на конфигурационен файл на 5G мрежова услуга;
- правило 100005 – генерира събитие, за опит на модификация на конфигурационен файл на gNB.

```
<active-response>
    <command>firewall-drop</command>
    <location>all</location>
    <rules_id>5710,100000,100001,100002,100003,100004,100005</rules_id>
    <timeout>604800</timeout>
</active-response>
```

Фиг. 4.15. Конфигурация за активен отговор.

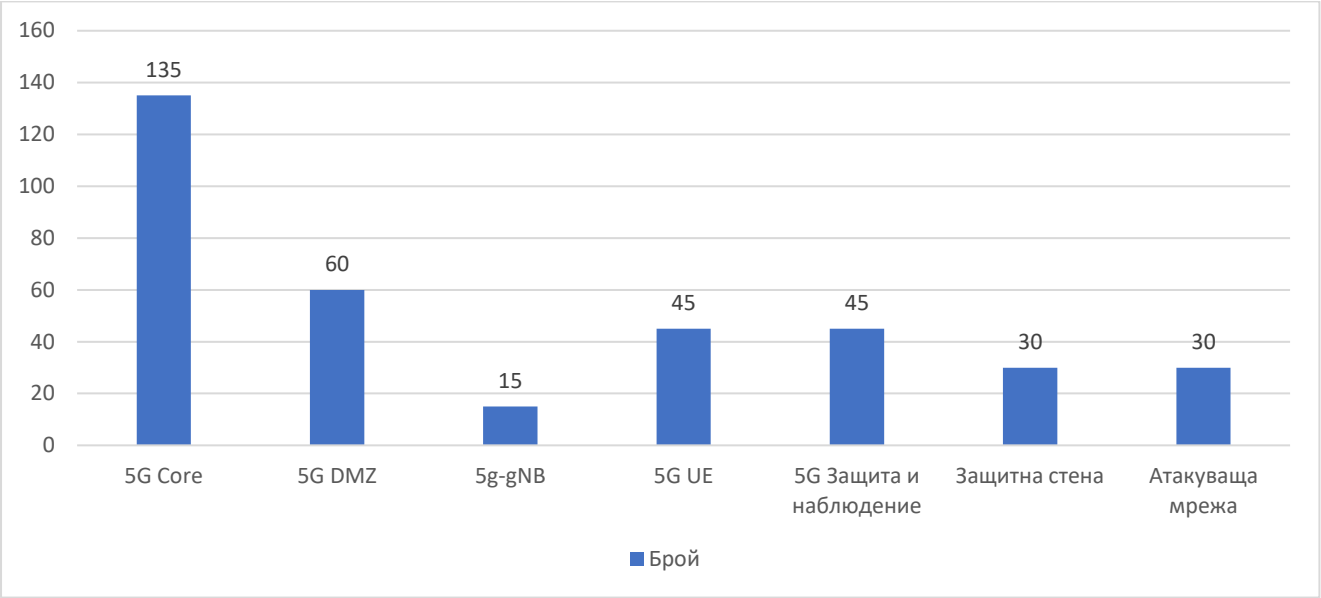
Сървърът 5G-ips-indexer служи за индексиране и съхранение на всички събития и аларми от сървъра. Конфигурацията включва операционна система Ubuntu Server 24.04 и Wazuh Indexer. Сървърът 5G-ips-portal осигурява уеб-базиран потребителски интерфейс за визуализация на данните за сигурност, управление на политиките за сигурност, дефиниране на правила и конфигурация на системата. Конфигурацията включва операционна система Ubuntu Server 24.04 и софтуер на уеб приложение Wazuh-dashboard.

Софтуерно-базираната антивирусна система ClamAV се използвана всички хостове и възли в основната мрежа, мрежата за радиодостъп и сървъра за резервни копия. За защита на потребителските терминали от зловреден софтуер е внедрена антивирусна система LibreAV. За борба срещу активно сканиране на портове е използван софтуерен инструмент с отворен код Suricata, който е конфигуриран на всички сървъри. Освен това, логовите данни относно открити мрежови атаки се препращат от Wazuh агентите към Wazuh сървъра за генериране на събитие, чрез конфигурацията на Wazuh агентите, показана на фигура 4.16. Правилата за засичане на активно сканиране на портове са представени в приложение 2.

```
<ossec_config>
  <localfile>
    <log_format>journald</log_format>
    <location>journald</location>
  </localfile>

  <localfile>
    <log_format>apache</log_format>
    <location>/var/log/apache2/error.log</location>
  </localfile>

  <localfile>
    <log_format>apache</log_format>
    <location>/var/log/apache2/access.log</location>
  </localfile>

  <localfile>
    <log_format>syslog</log_format>
    <location>/var/ossec/logs/active-responses.log</location>
  </localfile>

  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/dpkg.log</location>
  </localfile>

  <localfile>
    <log_format>json</log_format>
    <location>/var/log/suricata/eve.json</location>
  </localfile>

</ossec_config>
```

Фиг. 4.16. Препращане на логови данни от Suricata към Wazuh

Предложената система за киберсигурност е тествана спрямо моделът на противник в периода 13.02.2025 г. до 21.04.2025 г. По време на тестовете в посочения период са развърнати 15 инфраструктурни мрежи и внедрени над 380 виртуални машини, като резултатите относно утилизацията на предвидения инфраструктурен модел са визуализирани на фигура 4.17.



Фиг. 4.17. Внедрени виртуални машини в мрежови сегменти

Резултатите при емулация на противник, след прилагане на системата за киберсигурност, са представени в таблица 20 и визуализирани на фигура 4.18.

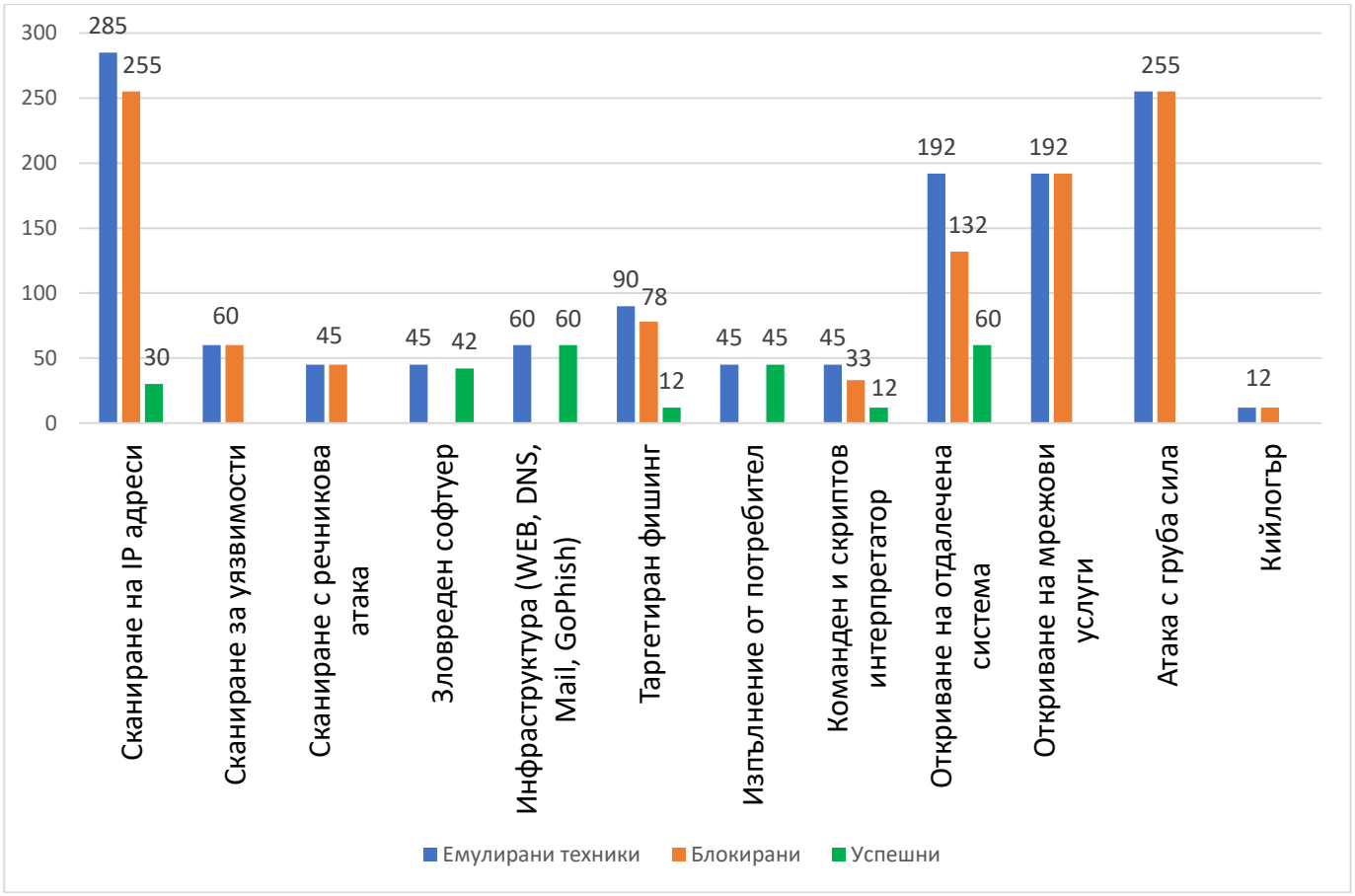
Таблица 20. Получени резултати при емулация на техники

Тактика	**Техника**	**Брой**	**Успешни**
Разузнаване	Сканиране на IP адреси	285	30
	Сканиране за уязвимости	60	-
	Сканиране с речникова атака	45	-
Изграждане на ресурси	Зловреден софтуер	45	42
	Инфраструктура (WEB, DNS, Mail, GoPhish)	60	60
Първоначален достъп	Таргетиран фишинг	90	12
Изпълнение	Изпълнение от потребител	45	45
	Команден и скриптов интерпретатор	45	12
Откриване	Откриване на отдалечена система	192	60
	Откриване на мрежови услуги	192	-
Достъп до кредениции	Атака с груба сила	255	-
	кийлогър	12	-

В таблица 21 са посочени засегнатите мрежови сегменти относно успешните техники.

Таблица 21. Успешни техники в мрежови сегменти

Техника	**Бр. успешни/ емулирани**	**Мрежов сегмент**
Сканиране на IP адреси	30/285	Защитни стени
Зловреден софтуер	41/45	Атакуваща мрежа
Инфраструктура (WEB, DNS, Mail, GoPhish)	60/60	Атакуваща мрежа
Таргетиран фишинг	12/90	5G UE
Изпълнение от потребител	45/45	5G UE
Команден и скриптов интерпретатор	12/45	5G UE
Откриване на отдалечена система	60/192	5G DMZ, 5G gNB



Фиг. 4.18. Резултати при емулация на противник

От уравнение 4.3, отново се изчислява ефективността на защитата ME_t, за емулираните техники. За техника *Сканиране на IP адреси*, ефективността на защитата е изчислена като:

$$ME_{T1595.001} = \left(1 - \frac{30}{285}\right).100\% = 89.47\% \quad (4.24)$$

За техника *Таргетиран фишинг*:

$$ME_{T1566} = \left(1 - \frac{12}{90}\right).100\% = 86.66\% \quad (4.25)$$

За техника *Команден и скриптов интерпретатор*:

$$ME_{T1059} = \left(1 - \frac{12}{45}\right).100\% = 73.33\% \quad (4.26)$$

За техника *Откриване на отдалечена система*:

$$ME_{T1018} = \left(1 - \frac{60}{192}\right).100\% = 68.75\% \quad (4.27)$$

Останалите техники от фиг. 4.18 са напълно блокирани, както следва: Сканиране за уязвимости (T1595.002), Сканиране с речникова атака (T1595.003), Откриване на мрежови услуги (T1046), Атака с груба сила (T1110) и Атака с кийлогър (T1056.001).

Изчислява се общ брой опити за съответните техники A': Сканиране на IP адреси $A_{\mathrm{T1595.001}} = 285$, Сканиране за уязвимости $A_{\mathrm{T1595.002}} = 60$, Сканиране с речникова атака $A_{\mathrm{T1595.003}} = 45$, Таргетиран фишинг $A_{\mathrm{T1566}} = 90$, Команден и скриптов интерпретатор $A_{\mathrm{T1059}} = 45$, Откриване на отдалечена система $A_{\mathrm{T1018}} = 192$, Откриване на мрежови услуги $A_{\mathrm{T1046}} = 192$, Атака с груба сила $A_{\mathrm{T1110}} = 255$, Кийлогър $A_{\mathrm{T1056.001}} = 12$.

Тогава от уравнение 4.10, $A' = 1176$.

Изчислява се агрегиран брой успешни опити от всички емулирани техники S': Сканиране на IP адреси $S_{\mathrm{T1595.001}} = 30$, Таргетиран фишинг $S_{\mathrm{T1566}} = 12$, Команден и скриптов интерпретатор $S_{\mathrm{T1059}} = 12$, Откриване на отдалечена система $S_{\mathrm{T1018}} = 60$.

Тогава от уравнение 4.9, $S' = 114$.

От 4.8, изчислената агрегирана ефективност на предложената система за киберсигурност спрямо емулираните техники ME' е:

$$ME' = \left(1 - \frac{114}{1176}\right).100\% = 90.30\% \quad (4.28)$$

От получените резултатите могат да се направят следните изводи:

1. Защитните стени успяват да блокират техники за активно сканиране и разузнаване на инфраструктурната мрежа. Въпреки това, атакуващите са в състояние да разузнаят IP адресите на защитните стени, но не и вътрешните мрежи, които защитават.
2. Тактиката *Изграждане на ресурси* не се влияе от системата за киберсигурност, тъй като съответните техники се реализират изцяло в средата на противника и са извън обсега на защитните механизми.
3. Системата за киберсигурност успява да блокира изпълнението на зловреден софтуер, генериран от вече известни и публично достъпни инструменти, но не и от модифициран зловреден код.
4. В случай на компрометиране на клиентско устройство, услугите в основна мрежа остават защитени. Въпреки това, атакуващият може да събере разузнавателна информация за услуги, разположени в периферията на мрежата 5G DMZ и 5G gNB.

4.6 Изводи

1. В резултат на проведените изследвания е доказана приложимостта на специализирания киберполигон за изграждане и изпълнение на симулации, тестване и оценка на сигурността, обучения и оценка на киберзаплахи.
2. Анализът на публичните бази данни за киберзаплахи позволява идентифициране на тактики, техники и процедури, характерни за атаките срещу 5G комуникационни мрежи, и тяхното статистическо разпределение по MITRE ATT&CK.
3. Емулацията на противник в контролирана среда доказва възможността за реалистично възпроизвеждане на фази от кибератаки и оценка на ефективността на защитните механизми.
4. Разработеният модел за вмъкване на код в APK приложения демонстрира уязвимост в Android екосистемата и предоставя основа за изследване на защитни техники срещу подобен тип атаки.
5. Предложената система за киберсигурност на когнитивни комуникационни мрежи интегрира резултатите от симулациите и разузнаването, като осигурява основа за централизирано наблюдение, откриване и реакция при инциденти. Тя повишава агрегирана ефективност за киберсигурност от 38.58 % на 90.3 % спрямо емулираните техники. Въпреки това, не успява да елиминира напълно риска от компрометиране на клиентско устройство с модифициран зловреден софтуер.

Научно-приложни приноси

1. Предложени са 3 нови математико-приложни модели: **модел на когнитивна комуникационна мрежа**, **модел на разузнаване за киберзаплахи в публични бази данни и модел на противник,** формализиращи компонентите на комуникационна когнитивна мрежа, основните характеристики на хетерогенни източници на разузнавателна информация и противник, с което предоставят основа за емулация и анализ на поведението на 5G инфраструктура при тестване, одит на сигурността и провеждане на стрес тестове. Предложените модели са полезни за научни и технически екипи, които се занимават с внедряване на конкретни защитни механизми, анализ на атаки и изграждане на архитектура за сигурност.
2. В рамките на изградения специализиран киберполигон, е реализиран симулационен сценарий на експериментална среда, с цел изследване на сигурността на потребителско оборудване и мрежова инфраструктура на 5G частна мрежа 5G-SA, която предоставя услуги за комуникация на данни (без глас) на потребители, включени чрез потребителско оборудване UE. Виртуална инфраструктура на комуникационна мобилна мрежа от 5-то поколение емулира работата на основната мрежа 5G Core, мрежата за радиодостъп 5G RAN и потребителското оборудване 5G UE, свързани помежду си с останалите външни мрежи и Интернет, чрез защитна стена/рутер.

Приложни приноси

1. Изграден е специализиран киберполигон, мрежовата архитектура на който е многослойна инфраструктура, осигуряваща изолация, контрол на достъпа и гъвкаво управление на виртуалните сценарии. Киберполигонът осигурява възможности за идентифициране на слабости в симулираната инфраструктура, подобряване на координацията между участващите екипи, оценка на реакцията при инциденти, както и обучение и сертификация на персонала.
2. Разработен е практико-приложен модел, формализиращ сценариите в симулационната платформа, като описва тяхната структурата и взаимовръзките между основните компоненти, участващи в тях. Моделът дефинира параметрите на всеки сценарий, задачите и въпросите в тях, както и множеството интерактивни ресурси, като виртуални машини и статични сайтове.
3. Предложена е система за киберсигурност на 5G когнитивни клетъчни мрежи, която повишава агрегирана ефективност за киберсигурност на 90.3 %, но не елиминира напълно риска от компрометиране с модифициран зловреден софтуер на потребителското оборудване 5G UE.

Насоки за по-нататъшна работа

1. Разширяване на възможностите на модела за разузнаване на киберзаплахи с дълбока невронна мрежа, която да се обучи с бази данни за противникови тактики, техники и процедури.
2. Разработване на симулационен модел за емулация на 6G инфраструктурни мрежи, като основа за тестване и оценка на нейната сигурност.
3. Разширяване на възможностите на системата за киберсигурност чрез реализиране на софтуерен агент за поддръжка на потребителско оборудване.

Списък на публикациите по дисертацията

1. Dimov, R., & Savova, Z. (2025). Algorithm for Cyber Threat Intelligence utilizing the MITRE ATT&CK. *Defense Technology Forum DTF25*. Shumen.
2. Dimov, R., & Savova, Z. (2024). Antivirus Performance Evaluation against PowerShell Obfuscated Malware. *Environment. Technology. Resources*. *Proceedings of the International Scientific and Practical Conference*, 4, pp. 71-78, https://doi.org/10.17770/etr2024vol4.8201, Scopus, https://www.scopus.com/inward/record.uri?eid=2-s2.0-85200497963&doi=10.17770%2fetr2024vol4.8201&partnerID=40&md5=47e062e5206b83481eb2068b900d3cff.
3. Nedelchev, M., Dimov, R., & Parvanov, S. (2023). Algorithm for application of information for management of users spatial disposition in the process of automated software-defined radio communication network implemented by orthogonal frequency division and multiplexing. *Scientific Research and Education in the Air Force AFASES*, pp. 39-46. Brasov, https://doi.org/10.19062/2247-3173.2023.24.5.

“VASIL LEVSKI” NATIONAL MILITARY UNIVERSITY

5006 g. Veliko Tarnovo, blvd. "Bulgaria" No.76

FACULTY OF ARTILLERY, AIR DEFENSE AND CIS

Department of Communication Networks and Systems

Captain Engineer Radostin Stefanov Dimov

INVESTIGATION OF THE SECURITY OF COGNITIVE COMMUNICATION SYSTEMS

A V T O R E F E R A T

of a dissertation

for awarding the educational and scientific degree "Doctor"

Field of Higher Education 5. "Technical Sciences"
Professional field 5.3 "Communication and Computer Engineering"
PhD Program "Cybersecurity"

Supervisor: Prof. Eng. Zhaneta Nikolova Savova, DSc

2025

city of Shumen

The dissertation consists of 175 sheets

Main text – 156 sheets

Number of applications – 2

Number of literary sources – 75

Number of publications on the dissertation – 3

The defense of the dissertation will take place on 13.02.2025 from __.__ h. in _____ _____.

The materials on the defense are available to those interested in _____ at _____, phone. _____.

The dissertation was discussed at the Departmental Council of the Department of Communication Networks and Systems at the Faculty of Artillery, Air Defense and CIS of the National Military University "Vasil Levski" - Veliko Tarnovo on ___.___.2025, and is sent for defense before a scientific jury for the acquisition of the educational and scientific degree "Doctor".

The PhD student is an assistant professor at the Department of Computer Systems and Technologies at the Faculty of Artillery, Air Defense and CIS at the National Military University "Vasil Levski" - Veliko Tarnovo. The main research on the dissertation was carried out in a simulated environment built through a specialized platform "Cyber Range" within the Data Center at the Faculty of Artillery, Air Defense and CIS.

Author: Captain Eng. Radostin Stefanov Dimov

Topic: "Investigation of the security of cognitive communication systems".

Circulation __

Printed on ___.___. 20__ .

Publishing Complex of the National Military University "Vasil Levski"

I. GENERAL CHARACTERISTICS OF THE DISSERTATION

The rapid development of communication technologies has led to the creation of cognitive communication networks that use artificial intelligence and machine learning algorithms to optimize network performance and improve user experience. However, the increasing complexity and interconnectivity of these networks has raised concerns about their security, as they use more sophisticated and potentially unsecured technologies, and are often subject to cyberattacks that can disrupt their operation or compromise sensitive information.

In recent decades, there has been a trend of increasing scientific and practical research in the field of testing and security assessment of cognitive communication systems, in particular 5G mobile networks. Most of them are aimed at studying defensive security and a small part concerns the application of offensive security in order to test the security of cognitive communication systems and, as a result, eliminate their weaknesses. Conducting real experiments in production networks is limited due to the risk of service disruption and possible cyber incidents.

To solve this problem, simulation platforms of a new generation are used - Cyber Range. Cyber Range platforms are complex products that require huge resources to ensure the normal functioning of the system in the creation of complex computer network architectures and the simulation of multiple offensive techniques, with multiple teams working simultaneously. On the other hand, existing open-source solutions are often limited in functionality and difficult to implement in specific network architectures. As a result, there is a need to build a specialized cyber range designed to provide a safe and realistic test environment for simulation and analysis of cyberattacks and protection mechanisms without affecting the operation of real systems.

The relevance of the topic stems from the widespread use of cognitive communication systems and their application in various spheres of public life, especially in the standard for fifth-generation 5G cellular communication networks – 3GPP Rel. 20, which is in force as of 2025.

The object of research is cognitive communication systems and their security, considered in the context of the 5G architecture, with an emphasis on identifying potential tactics, techniques and procedures used by malicious groups, as well as on their practical emulation in a controlled environment of a specialized cyber training ground.

The main goal of the dissertation is to increase cyber resilience and protect cognitive communication networks from digital threats by identifying and eliminating their weaknesses before they are exploited by malicious organizations and individuals

Additional:

1. Supporting the capabilities of researchers, industry and government institutions to conduct research and innovation through simulation and emulation of scientific developments and implementation of new cybersecurity technologies.
2. Creating opportunities for training and upskilling through educational programs, trainings and simulations in order to prepare specialists in the field of cybersecurity.

In accordance with the purpose of the dissertation, the following tasks are formulated:

1. Security analysis in communication cognitive systems and in particular 5G mobile networks.
2. Analysis and synthesis of security models of cognitive communication networks.
3. To build a specialized cyber training ground that provides opportunities to identify weaknesses in the simulated infrastructure, improve coordination between participating teams, assess incident response, train and certify personnel, as well as simulation of the security of a 5th generation communication mobile network.
4. Conducting simulation experiments and analyzing the results.

Taking into account the breadth of the topic in the dissertation, the following **restrictions** are taken:

1. The study should cover the cybersecurity of fifth-generation cellular communication networks.
2. The conducted experiments and simulations should be implemented in an emulated 5G environment built on the basis of open source platforms.

The restrictions set do not detract from the research tasks in the dissertation and can be used in scientific developments and research for the development and upgrade of real cybersecurity in fifth-generation cellular systems.

II. STRUCTURE OF THE DISSERTATION

The dissertation is structured in an introduction, four chapters, general conclusions, contributions, bibliographic reference and 2 appendices. The work contains 21 tables, 132 formulas and 57 figures. 75 literary sources are cited.

The introduction substantiates the topicality of the topic and its practical and applied nature. The purpose and main tasks are defined, the limitations under which the research is carried out are indicated.

In Chapter One, the first research problem is solved, examining the architectural features of the 5G network, including the separation of control and user planes, the virtualization of network functions, as well as the concept of service-based architecture. A literature review was carried out on the basis of a study in scientific databases, covering developments related to the security of cognitive communication networks for the period 2015 – 2024 and fifth-generation cell networks for the period 2019 – 2024.

In the second chapter, a second research task is solved by analyzing existing models for offensive and defensive security and presenting the models of a fifth-generation communication network, a model for cyber threat intelligence and an adversary model developed in the dissertation.

In Chapter Three, a third research task is solved through the implementation of a specialized platform for creating simulation environments and practicing offensive and defensive cyber operations. The platform includes a user interface, an ability manager, and a virtualization and automation unit. A scenario model is proposed, which formalizes the process of creating infrastructure environments. A simulation environment has been created to test and assess the security of a fifth-generation communication network.

In Chapter Four, a fourth research task is solved, analyzing the results obtained in emulating adversarial techniques and procedures, and proposing a cybersecurity system for fifth-generation communication networks. The choice of software and means for protection of the components of the studied 5G infrastructure network is justified.

III. SUMMARY OF THE DISSERTATION

Chapter 1 – Security Analysis in Cognitive Communication Systems.

For the solution of the first main task of the dissertation, the following subtasks are defined in the first chapter:

1. Analysis of cognitive radio and its characteristics, as an intelligent communication system.
2. Analysis of the 5G network infrastructure and its emulation systems.
3. Analysis of security in fifth-generation cell systems, delineating the main challenges and directions for scientific and applied developments in the area under study.

1.1. Cognitive Radio

To solve problem 1.1 in the paragraph, an analysis of the concept of cognitive radio as the basis of cognitive communication systems is carried out. An analysis of cognitive radio and its application in various modern technologies was carried out. The problem of limited spectral resources is considered and the need to use dynamic methods for radio spectrum management is substantiated. The evolution from software-defined radio systems to cognitive radio systems is traced, emphasizing that the latter have the ability to self-adjust and adapt to the environment. The main functional stages in the operation of cognitive radio are traced, including spectrum monitoring, detection of free frequency bands, decision-making and reconfiguration of transmission parameters. The role of cognitive radio for the joint use of the frequency spectrum between primary and secondary users without disturbing the quality of communication of licensed systems is debunked.

The architecture of cognitive networks, consisting of a primary and secondary network, is examined, indicating the responsibility of secondary users for monitoring the environment and releasing the spectrum during repeated activity of primary users. A distinction is made between centralized and ad-hoc cognitive networks and the mechanisms of interoperability between heterogeneous communication systems are analyzed.

1.2. 5G Network architecture

To solve problem 1.2 in the paragraph, an analysis of the architecture of the fifth generation cellular communication networks was performed. The principle of separation between the control and user planes

is considered, as well as the transition to a service-based architecture, in which the individual functions of the core network communicate with each other through standardized interfaces. The main network functions in 5G Core are distinguished, and their role in access management, mobility, user sessions, security and traffic processing is traced. The 5G RAN radio access network is also considered, taking into account the division of the base station into centralized and distributed parts, in order to optimize signal processing and reduce latency. Key radio technologies that support the achievement of high capacity and quality of service are analyzed. As shown in Figure 1.5, in the context of 5G, the network architecture is distributed into several main nodes: core network, radio access network, user equipment.

According to the report, 99.2% of nodes in 5G are user equipment. It is also noted that only 4% of network infrastructure is physical machines on the core network or edge (Figure 1.6). Quite naturally, the majority of cyberattacks affecting this type of network are aimed precisely at customers and their end devices. The following figure visualizes the distribution of nodes and peripherals in the network topology of cognitive networks.(Sanchez-Navarro, Bernabe, Alcaraz-Calero, & Wang, 2021)

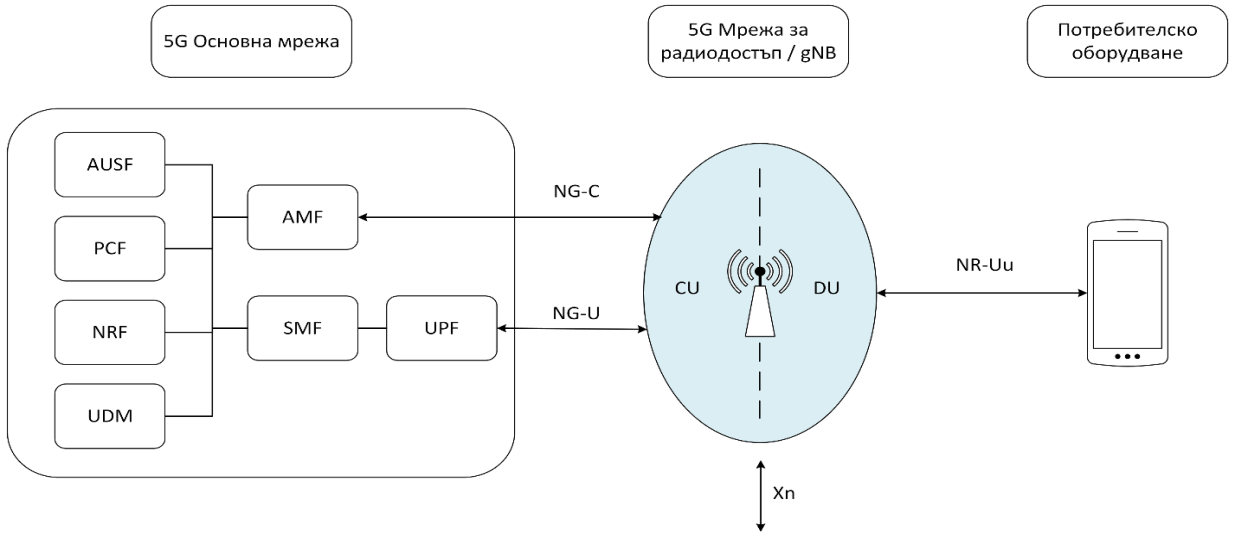


Fig. 1.5. 5G Network architecture

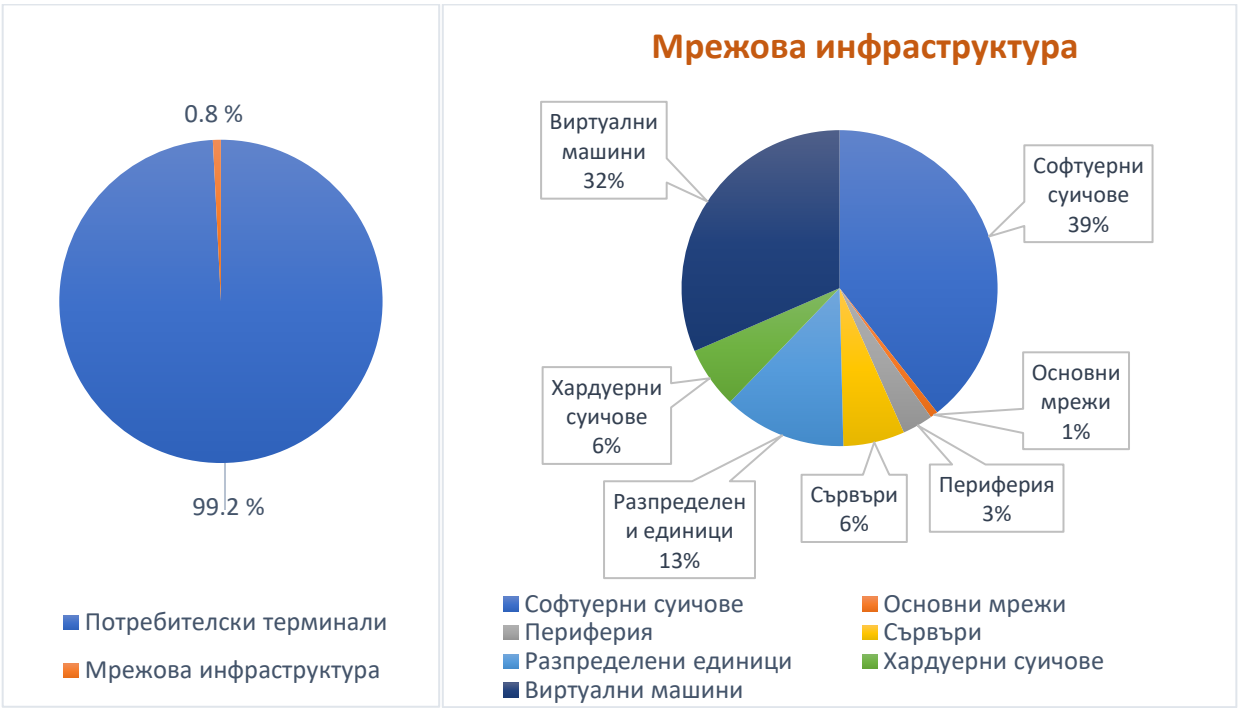


Fig. 1.6. Network topology distribution (Sanchez-Navarro, Bernabe, Alcaraz-Calero, & Wang, 2021)

The role of user equipment as an end point of interaction with the network is analyzed. It is pointed out that, due to its dominant share in the network environment and direct interaction with the radio interface, it is user devices that represent a significant vector of cyberattacks and therefore require special attention

in security analysis. A survey of 250 respondents using the face-to-face method confirmed this data, noting that Android is used on more than 7 out of every 10 smartphones, while iOS has a stable presence from nearly 3 out of 10 devices. The results are presented in Figure 1.7 and indicate a pronounced market division.

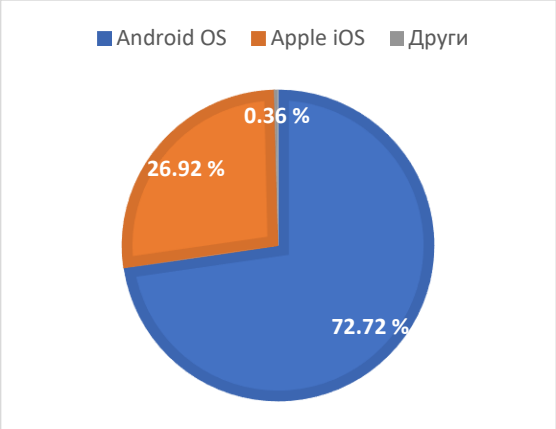


Fig. 1.7. Market share of user equipment fixed assets

Comparative analysis of open source systems for emulation of 5G infrastructure was performed. On this basis, a suitable technological solution has been identified to build an experimental environment that will allow simulation and assessment of security in conditions close to real. The analysis is presented in Table 1. Open5GS stands out as a balanced solution for building an emulated 5G environment combining functionality, stability, and deployability. The srsRAN platform provides comprehensive support for a 5G radio access network and is suitable for building a real hardware-based experiential 5G RAN setup. The UERANSIM platform is suitable for the implementation of simulation environments in the absence of hardware equipment. It also includes a 5G UE component for simulation of user equipment and user activity.

Table 1. Benchmarking of 5G network emulation systems

5G Core Core Core Network		
Characteristics	**Open5GS**	**Free5GC**
3GPP Rel.	Rel.17	Rel.15
Programming language	C	Go
Comprehensive support for network functions	✓	✓
Integration with RAN emulators	✓	✓
Construction on a VM/Container	✓	✓
Active development (community, GitHub activity)	✓	✓
Detailed documentation	✓	
Suitable for educational and laboratory environments	✓	✓
Easy to install and configure	✓	
Applicable to Cyber Polygon (Test Infrastructure)	✓	✓
Radio access network		
Characteristics	**srsRAN**	**UERANSIM**
Function	Real Emulation	Software Simulator
Full 5G RAN support	✓	partial
Programming language	C++	C++
gNB emulation	✓	✓
UE Emulation		✓
Compatible with Open5GS and Free5GC	✓	✓
Active development (community, GitHub activity)	✓	✓
Easy to configure		✓

1.3. Security analysis in fifth-generation cellular networks

To solve problem 1.3 in the paragraph, an overview of scientific publications addressing security problems in cognitive communication networks is made. For this purpose, the databases of scientific publications of ResearchGate and Scopus were examined, tracking developments in the studied field from 2015 to 2024.

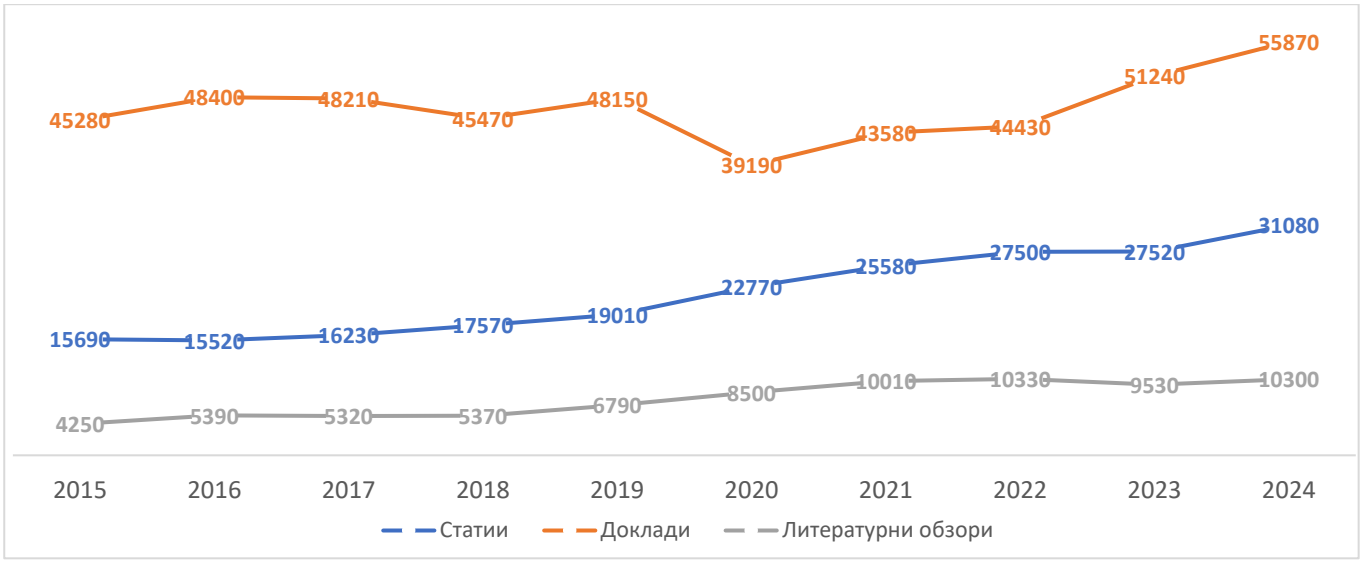


Fig. 1.8. Number of Scientific Publications Addressing Cognitive Network Security – ResearchGate

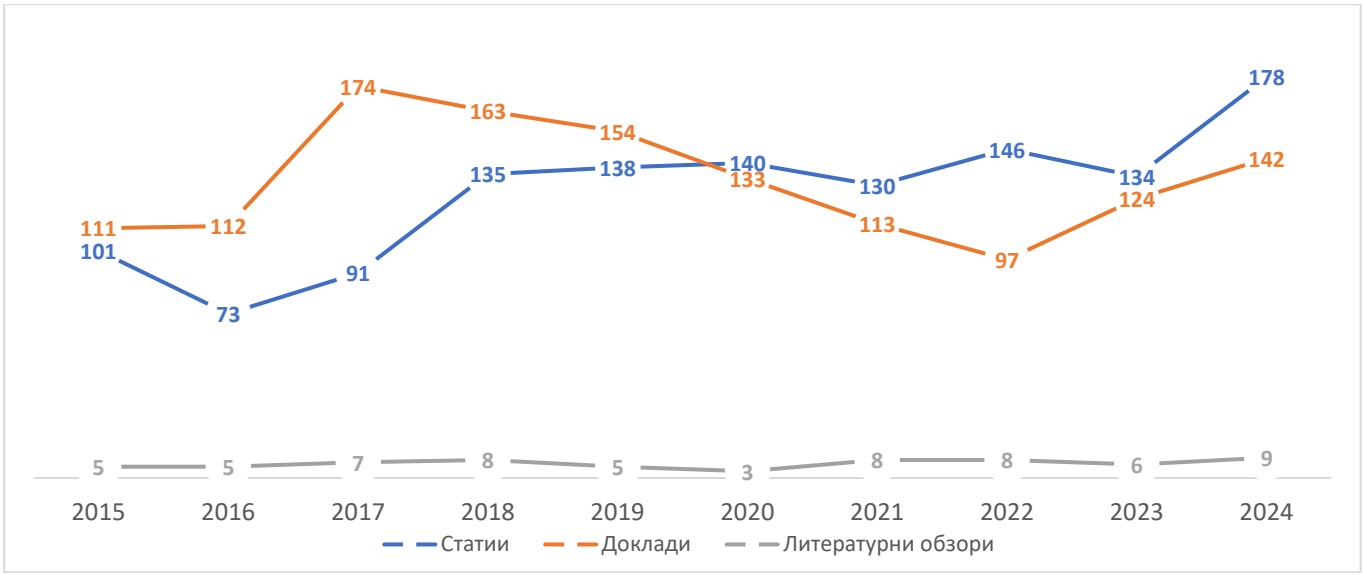


Fig. 1.9. Number of scientific publications addressing the security of cognitive networks – Scopus

In the period under study, there was an increase in the number of publications, with the highest values reported in the last 3 years for the period from 2022 to 2024 inclusive. Cognitive networks are widely used in 5G network technologies. By 2025, the last practical standard for cellular communications is fifth-generation 5G cellular networks.

Fig. 1.10. Number of scientific publications addressing 5G security – ResearchGate

The results show a pronounced increase in the number of publications addressing the security issue of fifth-generation cellular systems. The literature sources from the ResearchGate and Scopus databases are sorted by popularity based on the number of views and citations, and the first 30 sources for the period 2018-2024 are extracted. The results show a significant predominance in the study of defensive security of cognitive communication networks, and in particular of 5G networks (only 1 article concerns only offensive security, and 4 of them both). Therefore, there is a need to deepen scientific and applied developments in the field of offensive security.

In (Angelogianni, Politis, Polvanesi, Pastor, & Xenakis, 2021) a survey was done among personnel of telecommunications operators, suppliers, and cybersecurity professionals on the need to use simulated environments for security testing in fifth-generation communication networks. Of the 60 participants in the survey, 46 (≈77%) said they would use specialized platforms (cyber polygons) of simulation environments to test the security of their 5G infrastructure. This 78% express interest in using this type of platform to conduct cybersecurity trainings on 5G.

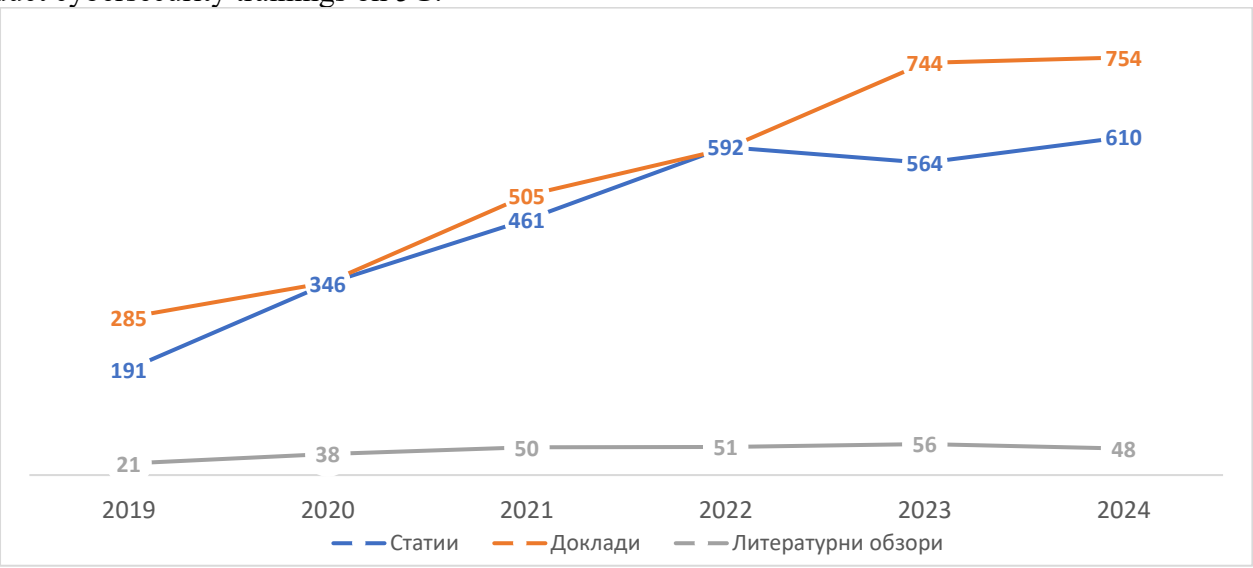


Fig. 1.11. Number of scientific publications dealing with 5G security – Scopus

Although the scientific publications examined look at various security challenges and concerns facing cognitive communication systems, none of them conducted a cyber threat intelligence study to identify malicious hacker groups targeting the telecommunications sector in Europe and NATO. This would allow for the creation of a summary picture of the operating groups targeting this specific sector, the geographical region, as well as the tactics, techniques and procedures they use.

User equipment such as Android and iOS smartphones and tablets are the most common end devices and often fall at the center of malicious attacks. In order to assess the real risk that these platforms carry in

terms of vulnerabilities, data from the public CVE database was analyzed in this study. The results of the analysis are synthesized and visualized in Figure 1.15, graphically representing the dynamics of the detected vulnerabilities by year.

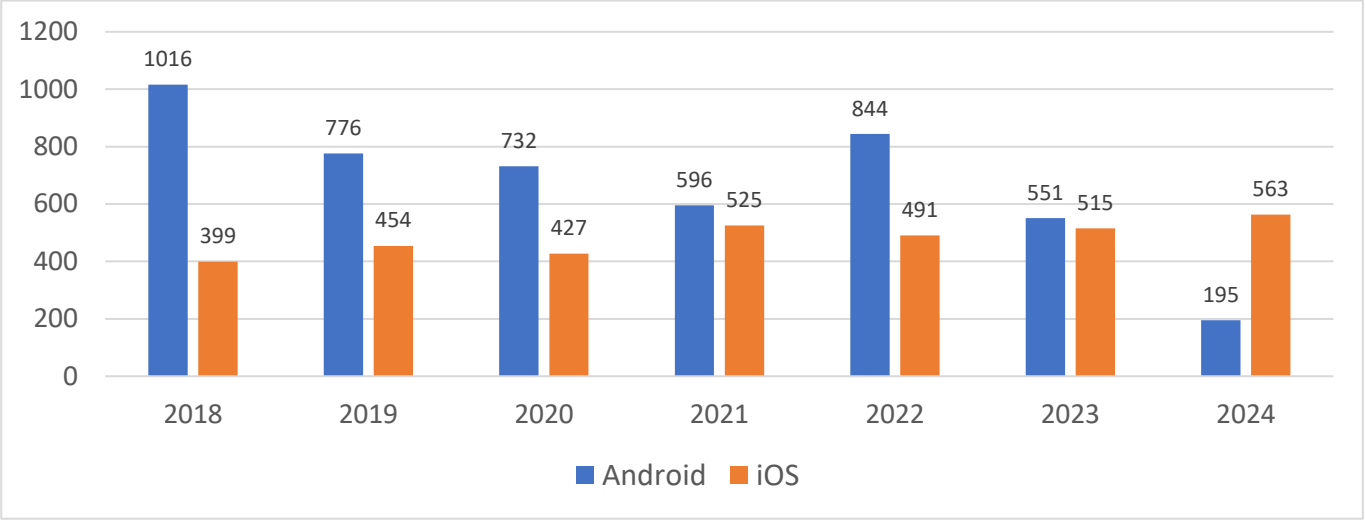


Fig. 1.15. Distribution of the number of vulnerabilities for Android and iOS in the period 2018-2024

1.4. Conclusions

1. Cognitive radio is a key technology for 5G, 6G, and IoT mobile communications, where the need for flexible and efficient spectral access is critical.

2. The market share of user equipment operating systems in 5G networks shows a predominance of Android (72.72%) over Apple iOS (26.92%) and only 0.36% for other operating systems.

3. The Open5GS platform stands out as a balanced solution for building an emulated 5G core network, providing full functional implementation of the 5G network core and containerization, making it suitable for building cyber testing grounds and practicing offensive and defensive cyber operations.

4. The following conclusions can be drawn from the security analysis in cognitive communication networks and in particular in cellular 5G networks:

(1)The number of research and development in the areas covered in the SCOPUS and ResearchGate databases is growing every year for the period 2019-2024, highlighting a maximum in 2024: cognitive network security (SCOPUS – 329, ResearchGate – 97250) and 5G security (SCOPUS – 1412, ResearchGate – 111000).

(2)Of the first 30 sources extracted by popularity based on the number of views and citations for the period 2018-2024, there is a significant predominance in the study of defensive security of cognitive communication networks and in particular of 5G networks (only 1 article concerns only offensive security, and 4 of them both). Therefore, there is a need to deepen scientific and applied developments in the field of offensive security.

(3)Among the main threats in 5G networks, according to the ENISA report, the largest shares stand out; "Phishing and social engineering" – 31%; "Vulnerabilities in IoT devices and user terminals" – 21% and "Incorrect configuration of 5G components" – 17%. Attacks on end users on mobile networks, through phishing, smishing and vishing, use malicious links to files for Android applications APK as an initial access method.

(4)For the period 2018-2024 Android reports a higher number of reported vulnerabilities except for 2024, which is due to its open architecture and its widespread application on devices from different manufacturers.

Chapter 2 – Analysis and synthesis of security models of cognitive communication networks.

For the solution of the second main task of the dissertation, the following subtasks are defined in the second chapter:

1. Analysis of defensive security models;
2. Analysis of offensive security models;

3. Synthesis of mathematical models of cyber threats in a cognitive communication network, formalizing the components of a communication cognitive network, heterogeneous sources of intelligence information, and adversary behavior.

2.1. Defensive security models

To solve problem 2.1 in the paragraph about two defensive security models – MITRE D3FEND and the NIST cybersecurity framework, their main components, structure and scope of application are considered. On this basis, a comparison was made between the models and their suitability for use in an environment with high traffic dynamics and adaptability characteristic of cognitive systems was assessed.

2.1.1. MITRE D3FEND

An analysis of the structure and components of the D3FEND framework has been performed. The main elements of the model – the knowledge graph, the semantic model and the user interface – are examined and how they formalise security technologies and dependencies between them. The possibility of integrating the model with offensive taxonomies such as MITRE ATT&CK is considered.

Digital artifacts define the conceptual scope of D3FEND. For example, a strong password policy falls within this scope because it directly affects the configuration base of a system. In contrast, cybersecurity officer training programs are not considered because they do not interact with digital artifacts. When an attacker performs actions – such as typing commands, searching for information, or creating malicious code – they create digital artifacts. The same goes for defensive actions: analysis, observation and detection also interact with artifacts, albeit from a different side of the conflict. Figure 2.4 illustrates this relationship between offensive and defensive techniques through digital artifacts.

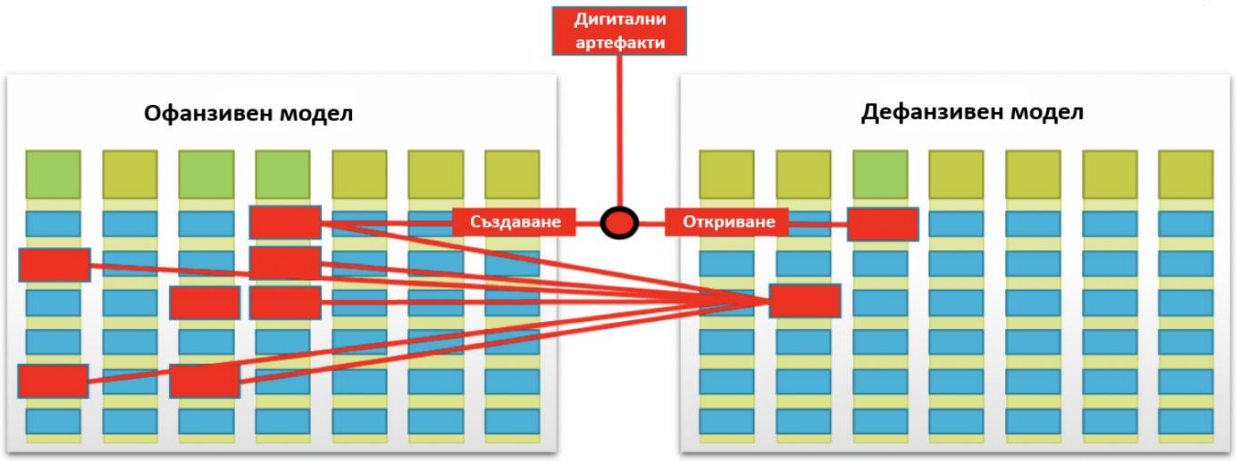


Fig. 2.4. Comparison of offensive and defensive techniques by artifacts

In order to effectively compare attacking and defensive approaches, it is necessary to define in detail the relationships between them. D3FEND uses digital artifacts as a central modeling axis, to which both types of techniques are associated. These associations can be of a general type ("related to") or specified by relationships such as "generate", "execute", "analyze", "access", "install". D3FEND also extends this model with relationships such as "observe", "detect" or "counteract", which allows defining logical relationships without the need to manually match each technique. The structure of the model makes it possible to quantify the coverage of offensive equipment from defensive equipment. Let's say:

- $O = \{o_1, o_2, \dots, o_m\}$ is the multitude of offensive techniques;
- $D = \{d_1, d_2, \dots, d_n\}$ is the multitude of defensive techniques;
- $A = \{a_1, a_2, \dots, a_k\}$ is a multitude of artifacts.

Functions can be defined:

$$P_O(o_i, a_j) \in \{0,1\}, \tag{2.1}$$

where P_O determines whether the offensive technique $o_i \in O$ uses the artifact $a_j \in A$.

$$P_D(d_l, a_j) \in \{0,1\}, \tag{2.2}$$

where P_D determines whether the defensive technique $d_l \in D$ monitors and analyzes the artifact $a_j \in A$. Matching techniques by artifacts is defined as:

$$M_{i,l} = \frac{\sum_{j=1}^{k} P_O(o_i, a_j) \,.\, P_D(d_l, a_j)}{\sum_{j=1}^{k} P_O(o_i, a_j)}, \tag{2.3}$$

where:

$$M_{i,l} \in [0,1] \tag{2.4}$$

The numerator measures the number of common artifacts on which the offensive technique acts and monitors the defensive technique at the same time. The denominator normalizes the score to all artifacts used by the attacker. The closer the metric value is to 1, the better the technique d_l defends against o_i. This is done by measuring the extent to which a defensive technique d_l encompasses the artifacts that an attacking technique o_i uses. In order to quantify the collective resilience of a system to an offensive technique, an average coverage metric can be defined that measures the degree of aggregate defensive effectiveness of all available defensive techniques. This metric is based on the value already entered $M_{i,l}$, which reflects the coverage of an attacking technique o_i by a specific defensive technique d_l, through the common artifacts that are used and monitored. The average coverage is defined as follows:

$$\bar{M}_i = \frac{1}{n} \sum_{l=1}^{n} M_{i,l}\,, \tag{2.6}$$

where $\bar{M}_i \in [0,1]$ is the average coverage of the offensive technique o_i and n is the number of available defensive techniques in the set $D = \{d_1, d_2, \dots, d_n\} o_i d_l$. In doing so, a level of security gap can be derived by measuring how much of the artifacts used by an attacking technique remain unobserved by the defensive technique. This provides a direct quantitative estimate of the coverage deficit by the safeguard.

$$G_{i,l} = 1 - M_{i,l}, \tag{2.7}$$

where $G_{i,l} \in [0,1]$ represents the level of protection gap. Values close to 0 (e.g. $G_{i,l} = 0.00$) mean that the respective defensive technique completely covers the attack. Values equal to 1.00 mean a complete omission, i.e. the defensive technique does not monitor any of the artifacts used by the respective attack. It is important to assess the extent to which the system provides visibility over critical artifacts C_A:

$$C_A = \frac{\sum_{j=1}^{k} 1\left(\sum_{i=1}^{m} P_O(o_i, a_j) > 0 \wedge \sum_{l=1}^{n} P_D(d_l, a_j) > 0\right)}{\sum_{j=1}^{k} 1\left(\sum_{i=1}^{m} P_O(o_i, a_j) > 0\right)}, \tag{2.8}$$

where $C_A \in [0,1]$ - reflects the proportion of actually endangered artifacts that fall under the scope of surveillance by the system.

2.1.2. NIST Cybersecurity Model

The goal of the model is for organizations to manage and mitigate cybersecurity risks. If necessary, the way organizations implement the model varies. The NIST cybersecurity model includes the following components: *Core, Organizational Profile, Levels*.

The core of the model is a set of cybersecurity-related outcomes, organized as follows: *Function*, *Category,* and *Subcategory* (Figure 2.6). The core core functions – *management, identification, protection, detection, response,* and *recovery*, organize cybersecurity-related outcomes at the highest level.

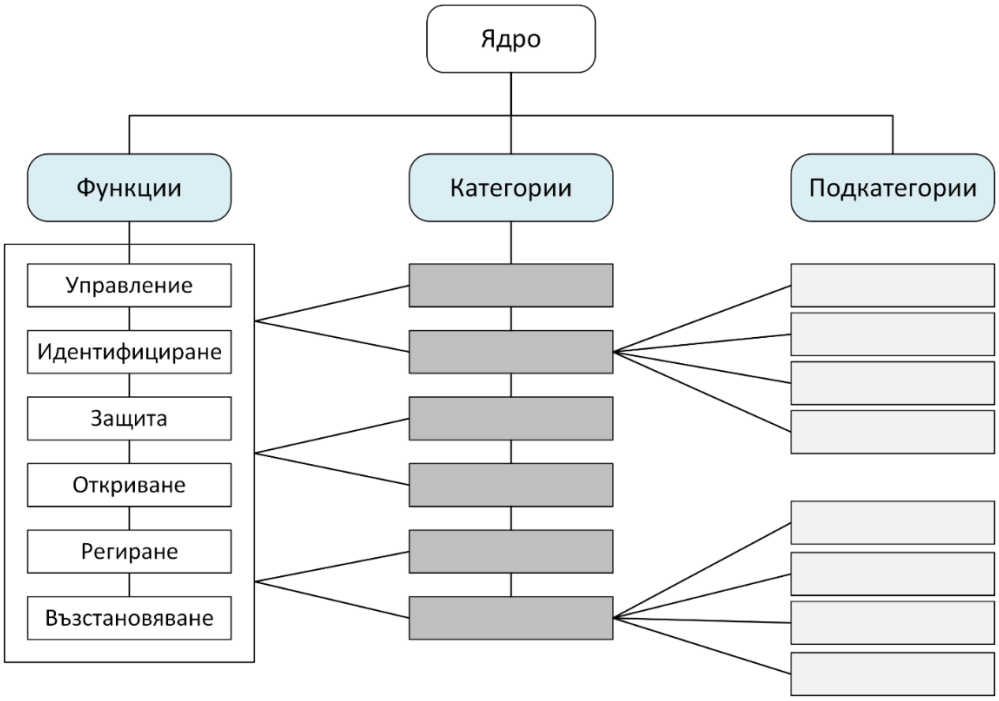


Fig. 2.6. Core NIST Cybersecurity Model

An organizational profile describes the current and/or target cybersecurity state in an organization, consistent with the outcomes defined in the core. Each organizational profile includes one or both of the following:

1. *Current profile* – indicates the results that the organization is currently achieving, as well as the characteristics or degree of achievement of each result.
2. *Target Profile* – indicates the desired outcomes that the organization has selected and prioritized to achieve its cyber risk management goals.

The steps shown in Figure 2.8 illustrate one of the ways an organization can use *the Organizational Profile* to support the process of continuously improving its cybersecurity.

The levels, as shown in Figure 2.9, reflect an organization's cybersecurity management practices in four levels: Partial (Level 1), Risk-Informed (Level 2), Repetitive (Level 3), and Adaptive (Level 4). Levels describe a progression from informal, reactive approaches to approaches that are adaptive, based on risk analysis, and are constantly being improved. helps determine the general direction by which the organization will manage its cyber risks.

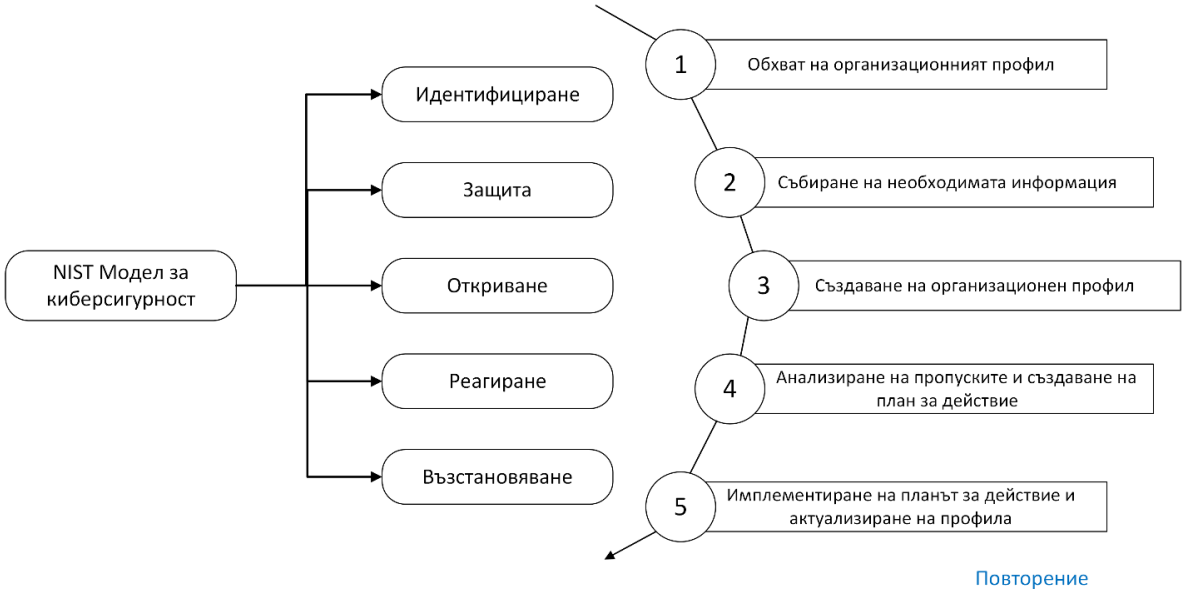


Fig. 2.8. Steps to create and use an organizational profile

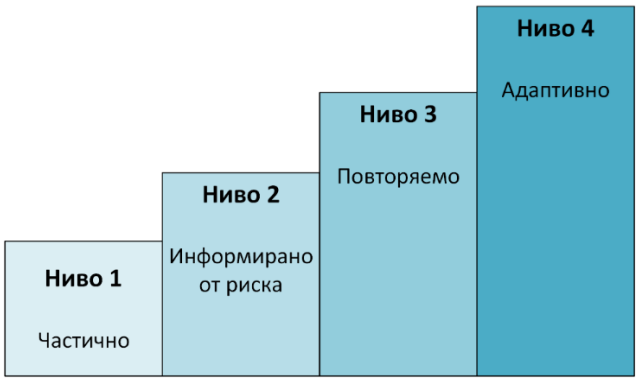


Fig. 2.9. Levels in the NIST Cybersecurity Model

2.1.3. Comparative analysis of defensive security models

A comparative analysis of the two models of defensive security discussed above by seven of their main characteristics, purpose, focus, structure, level of abstraction, applicability, integration with other models and target result, is presented in Table 5. The following conclusions can be drawn from the comparative analysis:

1. NIST is suitable for the organizational level where decisions are made on risk management, policy making, and strategic planning. It is widely used in regulatory environments, audits and compliance.
2. MITRE D3FEND is suitable for technical teams that are engaged in the implementation of specific defense mechanisms, attack analysis, and building a security architecture. It is useful in operational defense, threat identification, and building defense strategies based on real-world techniques.
3. Combining the two models can lead to more effective cybersecurity, with NIST CSF modeling the framework and governance, and MITRE D3FEND modeling technical implementation and protection.

Table 5. Comparative analysis of defensive security models

Characteristics	**MITRE D3FEND**	**NIST Cybersecurity Model**
Purpose	Describes techniques for defending and countering attacks	Provides a framework for risk management and cybersecurity improvement
Focus	Tactical – aimed at specific defensive actions	Strategic – focused on processes and management
Structure	Ontology of defense techniques related to MITRE ATT&CK	Functions: *Identify*, *Protect*, *Detect*, *Respond*, *Restore*
Level of abstraction	Low – technical details for protective mechanisms	High – organizational and managerial levels
Applicability	For Security Engineers, Analysts, Developers	For executives, managers, security consultants
Integration with other models	Connected to MITRE ATT&CK for attack mapping and defenses	Can be integrated with ISO 27001, COBIT, and other standards
Target result	Improving defensive abilities through specific techniques	Improving cyber resilience and risk management

2.2. Offensive security models

To solve problem 2.2, an analysis of offensive security models was carried out, which describe the behavior of the enemy and the way cyberattacks are carried out in real conditions. For this purpose, two widely accepted tactical models are considered – MITRE ATT&CK and the "Cyber Attack Chain" model, Cyber Kill Chain. The analysis focuses on their structural features, construction logic, level of detail and applicability in modeling and simulating attacks.

2.2.1. MITRE ATT&CK

MITRE ATT&CK is a globally accessible knowledge base containing Tactics, Techniques and Procedures (TTPs), which represent the actions that adversaries can perform to achieve their goals. Within ATT&CK, tactics are treated as "labels" with which techniques or sub-techniques can be associated, depending on the goals that can be achieved through their use. Each tactic contains a definition describing its category and serves as a guideline for what techniques should be included in it. The structure described in this way is presented in Figure 2.12.(MITRE Corporation, n.d.)

Techniques represent "how" the enemy achieves a certain tactical goal by performing a specific action. Sub-techniques further differentiate the behaviors described in the techniques into more specific forms.

Procedures are another essential component of the TTP concept. The procedures represent the specific implementation used by opponents in the application of a technique or subtechnique. Procedures often also include specific tools through which actions are carried out.

Known adversaries that are tracked by public and private organizations and reported in threat intelligence reports are reflected in MITRE ATT&CK through the entity - *Group*. Groups are defined as named sets of intrusions, threats, documented malicious groups, or campaigns that typically constitute targeted and sustained malicious activity.

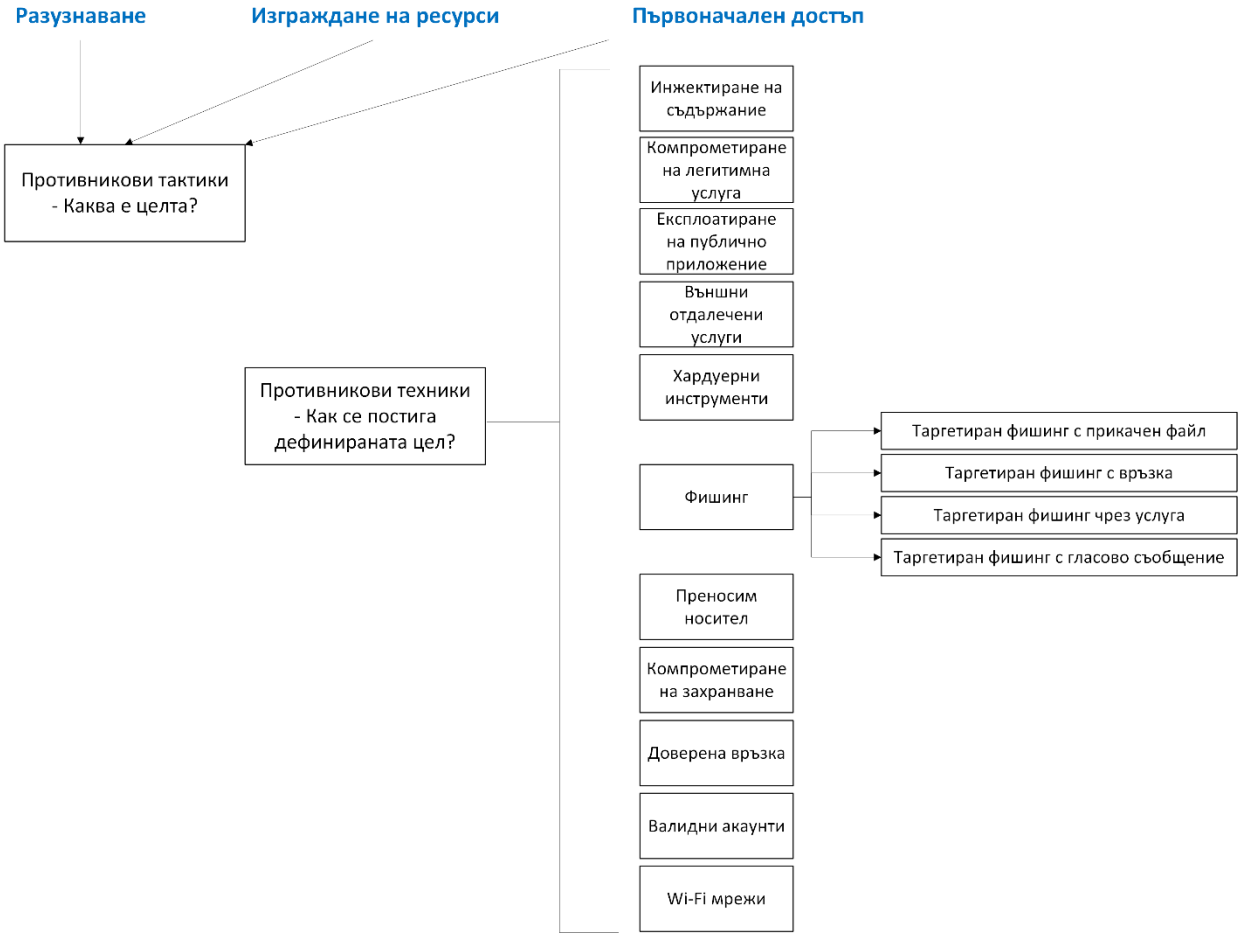


Fig. 2.12. Structure on tactics and techniques

2.2.2. Cyber-attack chain

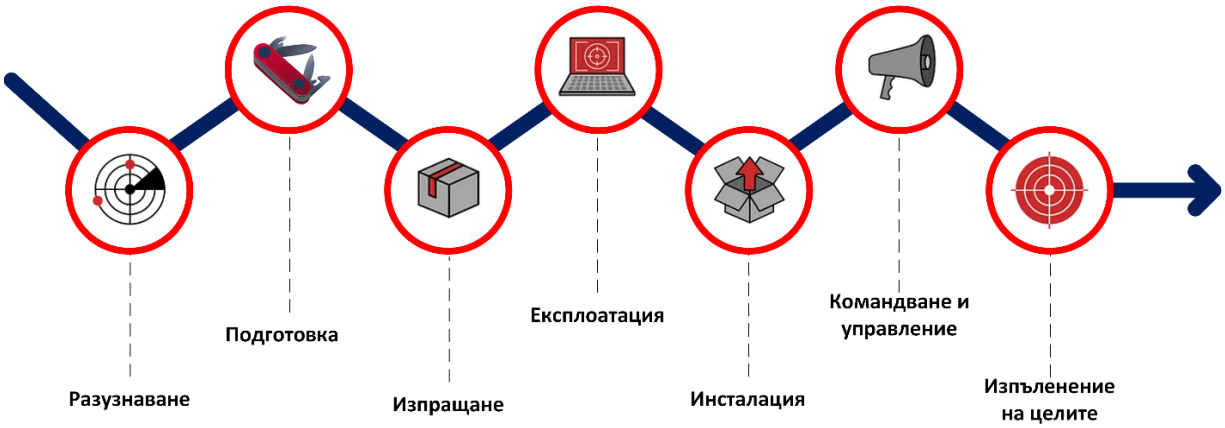


Fig. 2.13. Cyberattack chain

Figure 2.13 shows the methodology by which the chain works. The model divides cyberattacks into seven stages (phases). (Lockheed Martin, n.d.)

1. *Intelligence* - in the first stage, the attacker collects information about the target system or organization.
2. *Preparation* - after proper reconnaissance has been carried out, the attacker creates appropriate malware or modifies an existing one, according to the vulnerabilities of the target system.
3. *Sending* – when the malware is ready, it is sent to the target – via email, link, Universal Serial Bus, Universal Serial Bus (USB) drive, etc.
4. *Exploitation* - when executed, the malicious code exploits a vulnerability in the victim's system to gain initial access.
5. *Installation* - successful operation allows an attacker to install a backdoor or other type of malware in order to maintain constant access.
6. *Command and control* - through the installed back door, the attacker can communicate with the compromised system and control it remotely.
7. *Fulfillment of objectives* - in the final phase, the attacker fulfills his ultimate goals - for example, draining data, spreading to other systems, or sabotage.

2.2.3. Benchmarking of offensive security models

A comparative analysis of the two offensive security models discussed above, *MITRE ATT&CK* and *Cyber Kill Chain*, is presented in Table 6.

Table 6. Comparative analysis of offensive security models

Characteristics	**MITRE ATT&CK**	**NIST Cybersecurity Model**
Purpose	Catalogs techniques used by attackers in real-world scenarios	Describes the stages of a targeted cyberattack
Focus	Tactical – specific actions and techniques of the attackers	Strategic – attack sequence from start to goal
Structure	Matrix with tactics and techniques grouped by attack phases	Linear sequence of 7 stages of attack
Level of detail	High – includes techniques, sub-techniques, examples and tools	Medium – describes major phases without technical details
Applicability	For threat hunting, incident analysis, simulation and protection	For attack modeling, training, and strategic planning
Actualization and development	Constantly updated by MITRE with new techniques and groups	Less frequently updated, static model
Integration with other systems	Widely integrated into SIEM, SOAR, EDR, and threat hunting platforms	Mainly used in training and analytical contexts

The following conclusions can be drawn from the comparative analysis:

1. Cyber Kill Chain is better suited for strategic attack modeling, staff training, and understanding the logic behind targeted attacks.
2. MITRE ATT&CK is extremely useful for security operations teams that deal with threat identification, incident analysis and protection building.
3. Combining the two models can give a complete picture of an attack such as Cyber Kill Chain helping to understand the phases of an attack and MITRE ATT&CK for analyzing the specific actions and procedures in each phase.

2.3. Synthesis of cyber threat models in a cognitive communication network.

To solve problem 2.3, the section synthesizes a cognitive communication network model, a cyber threat intelligence model, and an adversary model.

2.3.1. Cognitive Communication Network Model

The model provides a mathematical and applied basis for emulation and analysis of the behavior of 5G infrastructure during testing, security auditing and conducting stress tests. The network model is presented in Figure 2.14 and is defined as:

$$S = \{C, N, R, U, D\}, \tag{2.9}$$

where C – core network, R– radio access network, U – user equipment, N – network infrastructure subnets and D – network equipment. Network infrastructure consists of three components:

$$N = \{N_{core}, N_{RAN}, N_{UE}\} \tag{2.10}$$

Each node on the network v_i has a unique one for the respective domain. $\text{IP}(v_i) \in N_j$. The component describing the core network is defined as:

$$C = \{E_{core}, F, OS_{core}\}, \tag{2.11}$$

where E_{core} is the emulator of the main 5G network, $OS_{core} \in Linux\ distributions$ is the operating system of the virtual machine on which the emulator is running and F - the network functions on the main network.

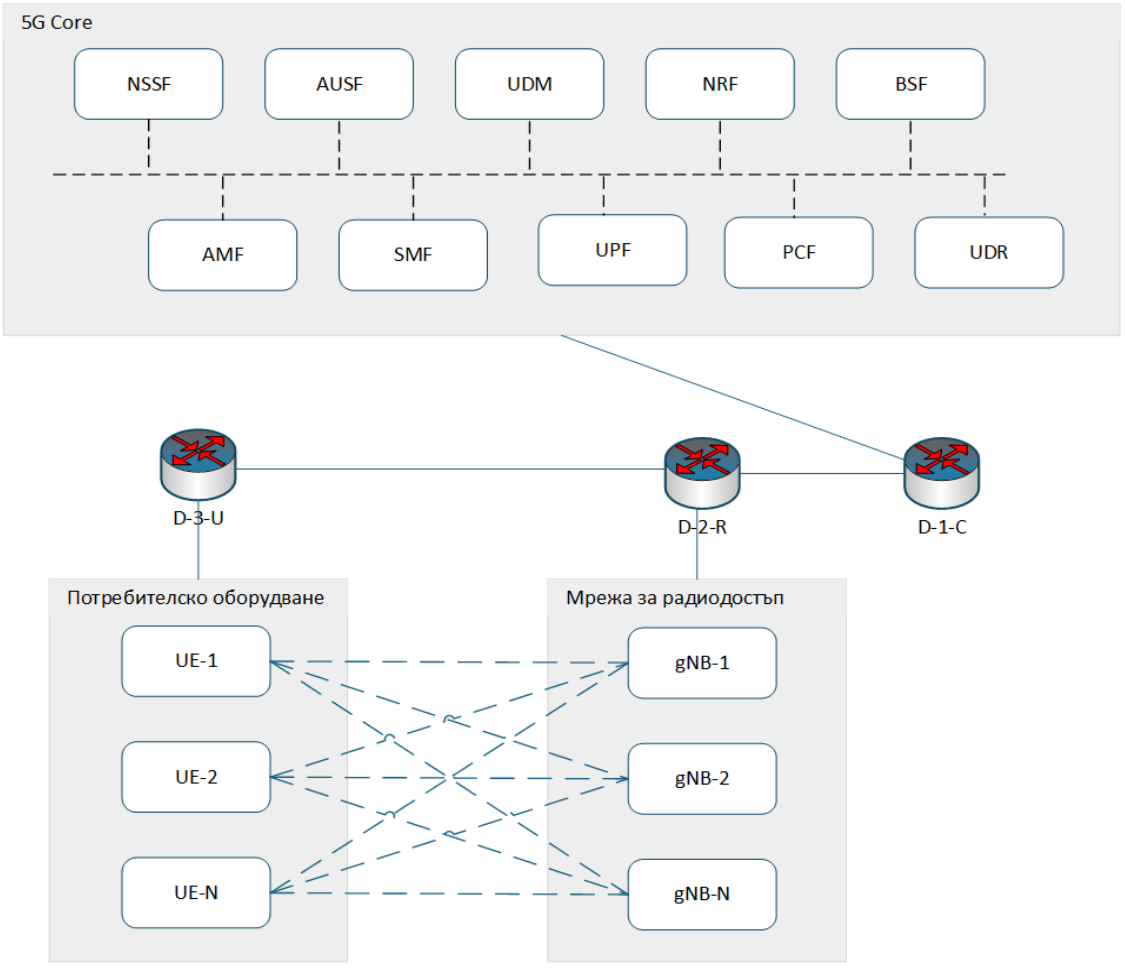


Fig. 2.14. Fifth generation cognitive network model.

A core network C consists of $n = 10$ virtual network functions (NF):

$$F = \{NF_1, NF_2, \dots, NF_{10}\} \quad (2.12)$$

Each network function is represented as an ordered triple:

$$NF_i = (id_i, IP_i, \phi_i), \quad (2.13)$$

where $id_i \in \mathrm{ID}_{NF}$ – identifier (e.g., 5gs-amf), $IP_i \in N_{core}$– associated IPv4, and ϕ_i– functional role of the node (e.g., routing, signaling, management, etc.). The component *R* defines the gNB nodes that perform the function of a base station.

$$R = \{R_1, R_2, \dots, R_q\} \quad (2.14)$$

Each gNB node is defined as:

$$R_i = (id_{gNB}, IP_{gNB}, E_{gNB}), \quad (2.15)$$

where id_{gNB} is a unique gNB identifier, $IP_{gNB} \in N_{RAN}$ – a base station IP address, and E_{gNB}– a UE emulator (e.g. UERANSIM). The component U models a set of user terminals.

$$U = \{UE_1, UE_2, \dots, UE_m\} \quad (2.16)$$

Each user terminal is defined as:

$$UE_i = (id_i, IP_i, OS_i, \psi_i), \quad (2.17)$$

where id_i– a unique identifier of a UE device, $IP_i \in N_{UE}$– an associated IP for a UE node, $OS_i \in \{iOS, Android\}$– the operating system of the UE emulator, and $\psi_i \subseteq \Psi$ – a set of applications (e.g. WhatsApp, Messanger, etc.). Network equipment *D* includes network devices such as routers, switches, etc. Each network device is defined as an ordered triplet:

$$d_i = (id_i, IF_i, \theta_i), \quad (2.18)$$

where $id_i \in \mathrm{ID}_D$ is a unique device identifier, $IF_i = \{(IP_{i1}, N_{i1}, link_{i1}), \dots, (IP_{in}, N_{in}, link_{in})\}$ – a set of interfaces with IP addresses and networks to which it is connected and $\theta_i \in \{router, switch, bridge, firewall, \dots\}$– a role or type of the device. The proposed model includes the following set of network devices:

$$D = \left(d_1^{(c)}, d_2^{(r)}, d_3^{(u)}, d_4^{(c)}, d_5^{(r)}, d_6^{(u)}\right), \quad (2.19)$$

where $d_1^{(c)}$- router on the main network, $d_2^{(r)}$- router on the radio access network, $d_3^{(u)}$- router on the network for user equipment and $d_4^{(c)}, d_5^{(r)}, d_6^{(u)}$are the switches for local connection of the nodes in the three subnets.

$$d_1^{(c)} = (id = routercore, IF_1, \theta_1 = router), \quad (2.20)$$

where:

$$IF_1 = \left\{\left(IP_1, N_{core}, d_4^{(c)}\right), \left(IP_2, N_{RAN}, d_2^{(r)}\right)\right\} \quad (2.21)$$

$$d_4^{(c)} = (id = swcore, IF_4, \theta_2 = switch), \quad (2.22)$$

where:

$$IF_4 = \left\{\left(IP_1, N_{core}, d_1^{(c)}\right), (IP_2, N_{core}, NF_1), (IP_3, N_{core}, NF_2) \dots\right\} \quad (2.23)$$

$$d_2^{(r)} = (id = routerran, IF_2, \theta_2 = router), \quad (2.24)$$

where:

$$IF_2 = \left\{\left(IP_1, N_{core}, d_1^{(c)}\right), \left(IP_2, N_{RAN}, d_5^{(r)}\right), \left(IP_3, N_{UE}, d_3^{(u)}\right)\right\} \quad (2.25)$$

A router in the network of user equipment is defined as:

$$d_3^{(u)} = (id = routerue, IF_3, \theta_3 = router), \quad (2.27)$$

where:

$$IF_3 = \left\{\left(IP_1, N_{RAN}, d_2^{(r)}\right), \left(IP_2, N_{UE}, d_6^{(u)}\right)\right\} \quad (2.28)$$

The switch in the user terminal network delivers a subnet that is used as the basis for connecting the UE emulator to the gNB.

$$d_6^{(u)} = (id = swue, IF_6, \theta_6 = switch), \quad (2.29)$$

where:

$$IF_6 = \left\{\left(IP_1, N_{UE}, d_3^{(r)}\right), (IP_2, N_{UE}, UE_1), (IP_3, N_{UE}, UE_2), \dots\right\} \quad (2.30)$$

2.3.2. Cyber Threat Intelligence Model in Public Databases

The proposed model is designed to identify relevant adversaries and threats targeting specific sectors and regions, map the tactics, techniques and procedures associated with them, and include the number of documented campaigns as a weighting factor in the analysis. Figure 2.15 presents a general flowchart of the proposed model.

In the first stage, all APT groups are extracted from the public databases designated with a unique identifier *G_x*.

Let it be the multitude of malicious groups, $G = \{g_1, g_2, \dots, g_n\} T = \{t_1, t_2, \dots, t_m\}$ the multitude of offensive techniques, and $\tau = \{\tau_1, \tau_2, \dots, \tau_p\}$ the multitude of offensive tactics.

A function of belonging is defined, assuming that each technique belongs to a given tactic:

$$\mu: T \to \tau, \tag{2.39}$$

where $\mu(t_j) = tactic, to\ which\ belongs\ tecnique\ t_j$. Each group g_i has two related attributes: sector of interest: $S(g_i) \subseteq \Sigma$, where Σ is the set of all industrial sectors, and geographic activity: $R(g_i) \subseteq R$, where R is the set of geographical regions (e.g., Europe, Asia, NATO).

The sets of targeted sectors and regions shall be defined:

$$S_{target} = \{S_1, S_2, \dots, S_k\} \tag{2.40}$$

$$R_{target} = \{R_1, R_2, \dots, R_j\} \tag{2.42}$$

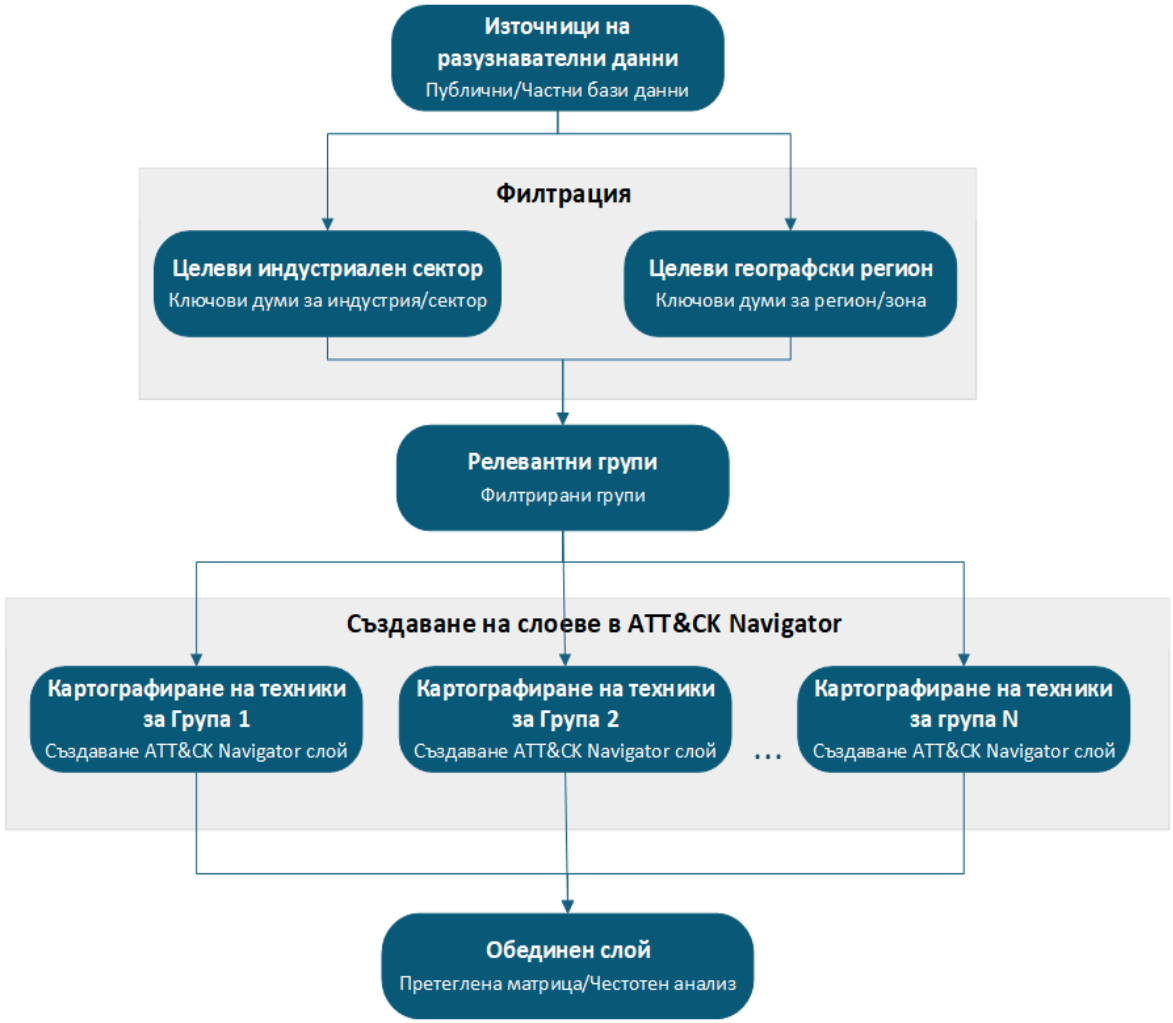


Fig. 2.15. Cyber Threat Intelligence Model Flowchart

S_{target} defines the target sector through a set of keywords. R_{target} Define the target region through a set of keywords.

The set of relevant groups is defined as:

$$G' = \{g_i \in G \mid S(g_i) \cap S_{target} \neq \emptyset \ \wedge R(g_i) \cap R_{target} \neq \emptyset\}, \tag{2.44}$$

where $G' \subseteq G$ and $g_i \in G'$, $i \in [1, n]$. For each group, the set of ATT&CK techniques used is defined:

$$T(g_i) = \{t_1, t_2, \dots, t_k\} \subseteq T \tag{2.45}$$

The indicator function of the techniques used can be calculated as:

$$\delta_{j,a}(g_i) = \begin{cases} 1, & \text{ако } t_j \in T(g_i) \wedge \mu(t_j) = \tau_a \\ 0, & else \end{cases} \tag{2.46}$$

A binary matrix is created for each relevant group:g_i

$$\mathbf{M}(\boldsymbol{g_i}) = [\delta_{ja}(g_i)] \in \{0,1\}^{n\,x\,m}, \tag{2.47}$$

where row j corresponds to technique t_j, column a corresponds to tactics τ_a and $n = |T|$, $m = |\tau|$. The frequency of use of offensive techniques by relevant groups can then be derived from a resultant matrix by summing up the matrices of all extracted groups.

$$\mathbf{M}^*_{\boldsymbol{j,a}} = \sum_{i=1}^{|G'|} \mathrm{M}(g_i)_{j,a}\,, \qquad 1 \le j \le n, \qquad 1 \le a \le m \tag{2.49}$$

2.3.3. Adversary model

The proposed model describes the key components of the adversary, such as a unique group identifier, a set of techniques used, a strategy for selecting and sequencing techniques, an emulation plan with specific procedures, and the necessary infrastructure. This allows for the playback and automated simulation of real-world APT campaigns and malicious adversary techniques. Figure 2.16 shows the flowchart of the proposed model.

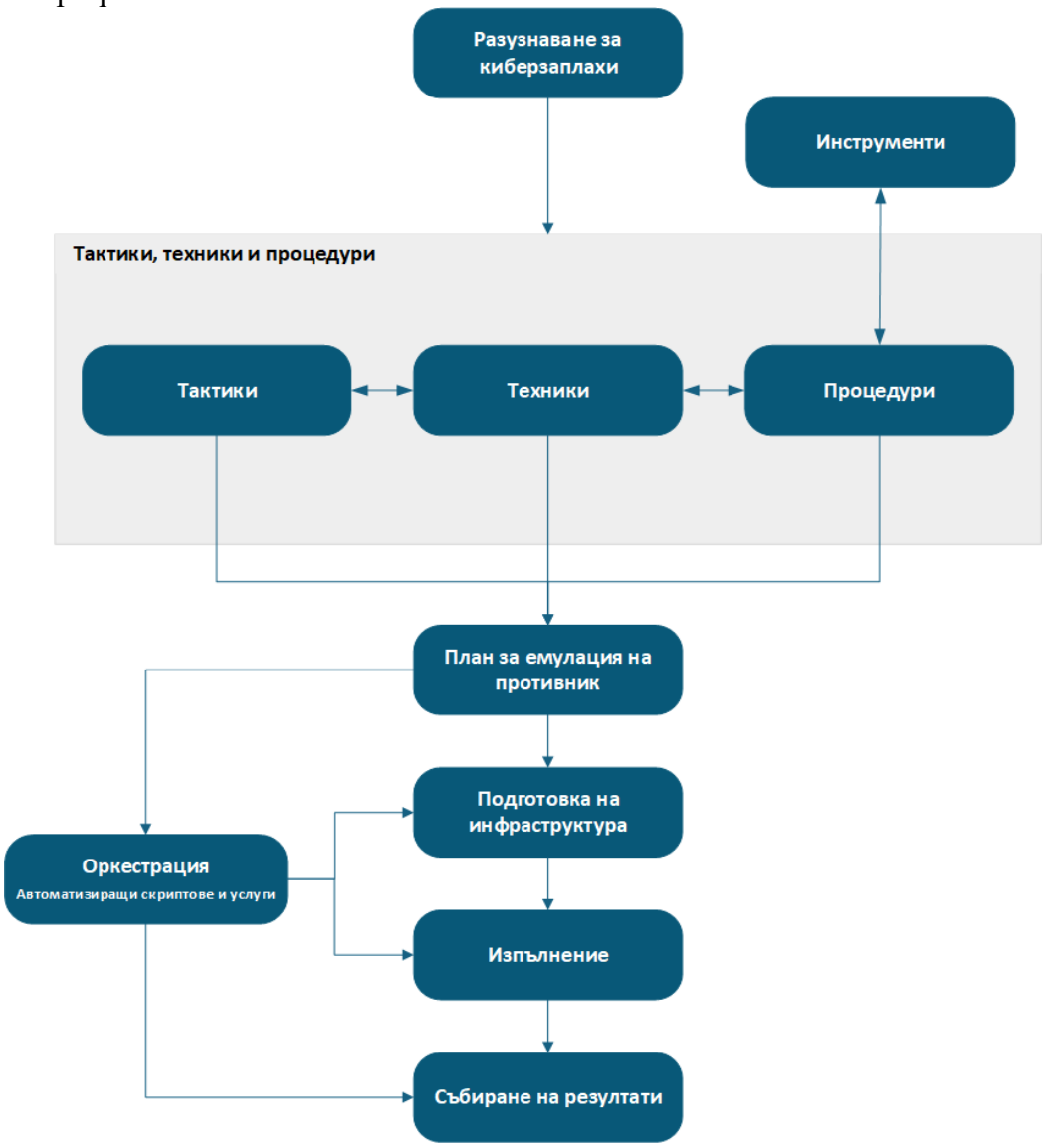


Fig. 2.16. Block diagram of an adversary model

An opponent (group, campaign, or actor) is defined as:

$$A = \left(g, {T_g}^w, \Pi_g^P, E, I_g\right), \tag{2.54}$$

where:

- $g \in G$ – unique identifier of the group (APT, hacktivists, etc.);
- ${T_g}^w \subseteq T$ – a number of offensive techniques used according to CTI data;
- Π_g^P – strategy for the enemy; (Ross, 2014)
- E – emulation plan (behavior strategy – describes the order of phases, techniques and procedures);
- I_g - infrastructure for the enemy.

The group g identifying a specific adversary belongs to the set:

$$G = \{g_1, g_2, \dots, g_n\}, \tag{2.55}$$

where G is a set of known groups in public databases. Then $g \in G$ it will be the unique identifier of the opposing group, which is often defined as APT__ or G_x. It will be a reference point to all associated sets - $T_g, C_g, R_g, S_g, \Pi_g$ Let's say:

$$T_g = \{t \in T |\ t\ is\ used\ tecnique\ by\ group\ g\} \tag{2.56}$$

be the set of techniques used by the group g .

In order to quantify the frequency of use of techniques, T_g will be expanded to a weighted set:

$${T_g}^w = \left\{(t, w_t) \mid t \in T_g, w_t \in N\right\}, \tag{2.59}$$

where $w_t \in N$ – technique weight, measured as the number of campaigns in which the goup participates based on CTI data. Then the weighted set can be represented by decomposition by tactics:

$${T_g}^w = \bigcup_{\tau \in X} {T_g}^{w(\tau)}, \tag{2.60}$$

where:

$${T_g}^{w(\tau)} = \left\{(t, w_t) \mid t \in T^{(\tau)} \cap\ T_g\right\} \tag{2.61}$$

A component is defined $\Pi_g$that models the logic and order in which a group g applies techniques $t \in T$to achieve goals of different tactical categories. An adversary's emulation plan is generated as a sequence of steps:

$$E = \left[\left(\varphi^{(1)} t^{(1)}, p^{(1)}\right), \left(\varphi^{(1)} t^{(2)}, p^{(2)}\right), \dots, \left(\varphi^{(L)} t^{(L)}, p^{(L)}\right)\right], \tag{2.72}$$

where, for each step *I* of the plan, a selected technique $t^{(i)}$ and procedure $p^{(i)}$ is defined. The condition for the completion of the implementation is the achievement of the specified number of steps L. The enemy's infrastructure is defined as I_g.

$$I_g = \left(H_g, D_g, N_g, A_g\right), \tag{2.73}$$

where H_g is a set of hosts in the adversarial infrastructure, D_g is a set of domains, N_g is the network topology for deploying the infrastructure and A_g is a set of credits to access the implemented services. The set of hosts is considered as an ordered four:

$$H_g = (id_h, type_h, ip_h, os_h), \tag{2.74}$$

where id_h is a unique host identifier, $type_h$ defines the service implemented on a specific host (role), ip_h - assigned IPv4 address and os_h defines the host's operating system. The services that a particular host can support belong to the following set:

$$type_h \in \{WEB, C2, Mail, Files, DNS, Attackbox, VPN, Proxy\} \tag{2.75}$$

The set of operating systems that are assigned to each host is defined as:

$$os_h \in \{Ubuntu, Debian, Win22, Win10, Win11, Opnsense, Kali\} \tag{2.76}$$

A set of domains D_g is considered as a set of elements, represented as:

$$D_g = (id_d, n_d, r_d, rr_d), \tag{2.77}$$

where id_d –a unique domain identifier, n_d is the domain name defined during registration, r_d – registration metadata (WHOIS information), rr_d– a set of DNS records (A, AAAA, CNAME, MX, TXT). Network topology defines nodes and the connections between them:

$$N_g = (V, E, S, R), \tag{2.78}$$

where V is a set of vertices (nodes). In the presented model $V = H_g$. E is a set of connections between nodes, S is a set of subnets, and R is a set of rules on network devices. The set of subnets that the adversarial infrastructure has is defined as:

$$S = \{s_1, s_2, \dots, s_k\}, \tag{2.80}$$

where each subnet is defined by several parameters:s_i

$$s_i = (net_i, mask_i, gw_i), \tag{2.81}$$

where net_i– network address, $mask_i$ – subnet mask, and gw_i – gateway address to connect to a higher-level subnet. The rules for filtering network traffic in network devices are defined as a set:

$$R = \{r_1, r_2, \dots, r_p\}, \tag{2.82}$$

Each rule r_i includes the following parameters:

$$r_i = (proto_i, srcip_i, srcprt_i, dstip_i, dstprt_i), \tag{2.83}$$

Access credentials A_g determine the credentials of operating systems and services in the adversarial infrastructure:

$$A_g = \{\mathrm{a}_1, \mathrm{a}_2, \dots, \mathrm{a}_c\}, \tag{2.84}$$

Each element is represented as an ordered four:$\mathrm{a}_i \in A_g$

$$\mathrm{a} = (id_a, type_a, scope_a, sercret_a), \tag{2.85}$$

where id_a– unique identifier of the record with the credentials, $scope_a$ – range of resources for which the credits are valid, $secret_a$ – reference (pointer) to the values of the credits, and $type_a \in P_{cred}$ – type of credits.

$$P_{cred} = (user\_pass,\ ssh\_key,\ api\_token,\ oauth,\ cert), \tag{2.86}$$

$$scope_a \subseteq H_g \cup D_g, \tag{2.87}$$

i.e., the set of hosts, services, or domains (or combination) that this credence gives access to.

$$secret_a = ptr(id_a) \tag{2.88}$$

Next comes the implementation of the emulation plan. Execution is defined as:

$$X = (E, L, Y_x), \tag{2.89}$$

where E– emulation plan defined in 2.72, L – number of steps to implement, and Y_x– sequence of infrastructure and target environment states after each step.

$$Y_x = [y_1, y_2, \dots, y_L] \tag{2.90}$$

Each condition y_i includes:

$$y_i = \left(t^{(i)}, p^{(i)}, r^{(i)}\right), \tag{2.91}$$

where $t^{(i)}$ – selected technique from the plan, $p^{(i)}$ – the specific procedure that implements the technique and $r^{(i)}$– the result of the implementation of the step (success/failure). The result Z is defined as the union of all the results $r^{(i)}$ of the implementation of the steps:

$$Z = \bigcup_{i=1}^{L} r^{(i)} \tag{2.92}$$

2.4. Conclusions

1. The combined use of strategic frameworks such as NIST CSF and Cyber Kill Chain with tactical models such as MITRE ATT&CK and MITRE D3FEND provides a holistic approach to conducting offensive and defensive cyber operations. NIST CSF and Kill Chain provide a structured vision for risk management and attack phases. ATT&CK and D3FEND complement with specific attack and defense techniques. This allows organizations to build effective strategies that cover both management and technical aspects, improving detection, response, and resilience against threats.
2. For the purposes of the dissertation, the tactical models of MITRE ATT&CK and MITRE D3FEND were chosen for the synthesis of a threat model of a cognitive communication network, because they provide a rich database of real techniques used by attackers, and are suitable for technical teams working on operational protection, threat identification and building defense strategies. based on real techniques.
3. 3 models have been synthesized that provide a mathematical and applied formalization of the components of a communication cognitive network and provide a basis for emulation and analysis of the behavior of 5G infrastructure during testing, security auditing and stress testing. The proposed

models are useful for scientific and technical teams that are engaged in the implementation of specific defense mechanisms, attack analysis, and building a security architecture:

(1) **A cognitive communication network model** that defines the logical and physical structure of a system and network components as ordered sets and relationships describing their identity, functional role, network addresses, and interconnectedness.
(2) **Cyber Threat Intelligence Model** that provides a quantitative assessment to analyze adversary behavior. The model identifies relevant adversaries and threats targeting specific sectors and regions and maps their associated TTPs.
(3) **An adversary model** that mathematically formalizes the main characteristics of the opponent, such as a unique group identifier, the set of offensive techniques used, the opponent's strategy, emulation plan, and the enemy's infrastructure.

Chapter 3 – Specialized Cyber Range for the implementation of simulation models.

For the solution of the third main task of the dissertation, the following subtasks are defined in the third chapter:

1. To design and build the conceptual and network architecture of the specialized cyber polygon.
2. To develop a mathematically applied model formalizing the scenarios in the specialized cyber polygon.
3. To develop a simulation scenario of an experimental environment to investigate the security of user equipment and network infrastructure of a 5G mobile network

3.1. Composition of the proposed cyber range

To solve problem 3.1. In this section, a specialized platform is proposed for creating simulation environments for conducting stress tests, as well as offensive and defensive cyber operations. The conceptual architecture of the proposed model is presented in the flowchart in Figure 3.1, and the functional elements are grouped in 3 layers depending on the services they implement: *User interface*, *Capability Manager* and *Virtualization and Automation Unit*.

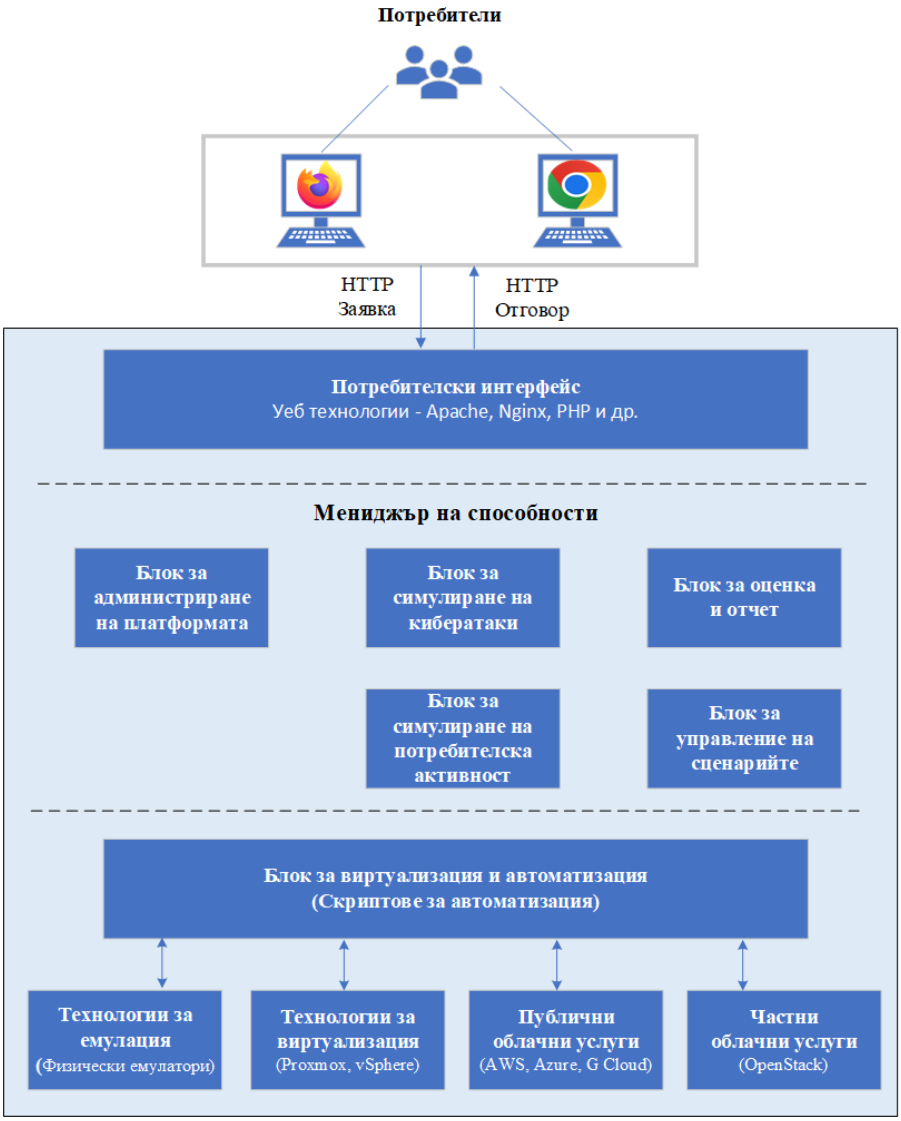


Fig. 3.1. Architecture of the proposed platform.

The user interface ensures the interaction between the user and the platform's services and uses Apache web server technology to build an interactive portal for users to interact with the platform's services without having to install and configure additional software.

The Capability Manager provides the basic functions of the platform and includes *the Administration and Management Unit, the Cyberattack Simulation Unit, the Evaluation and Report Unit, the User Simulation Unit,* and *the Scenario Management Unit*.

The capability manager defines the platform's capabilities for organizing simulations, conducting trainings, evaluating teams, as well as interacting with the virtualization block in building the emulation environments of networks and operating systems.

The virtualization and automation unit serves for automated configuration, coordination and management of computer systems and software. It consists of a Proxmox virtualization platform and multiple automated scripts.

The network architecture of the platform is presented in Figure 3.2.

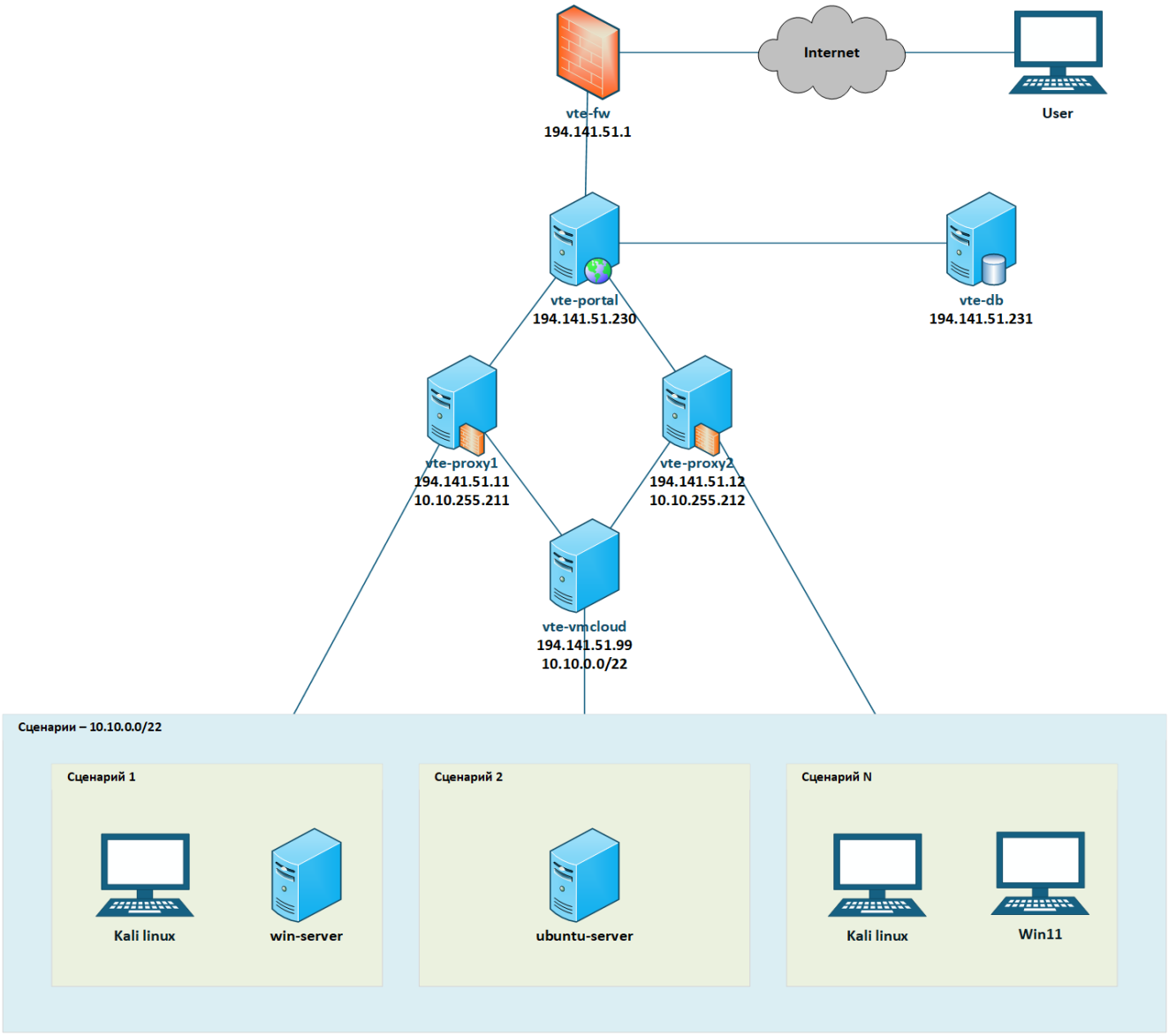


Fig. 3.2. Network architecture of the proposed platform

On the outer perimeter there is a firewall *vte-fw*, which controls the traffic between the Internet and the internal infrastructure of the platform. The main access point to the platform is the *vte-portal* node, which serves the web interface for users. It provides authentication, scenario management and initialization of virtual machines. The portal communicates with users via the Internet as well as with the other subsystems – the database data and proxies. The internal *vte-db* database stores information about users, scenarios, virtual machines and active sessions. To ensure load balance, two proxies have been implemented – *vte-proxy1* and *vte-proxy2*. Their role is to forward requests for running virtual machines to the cloud cluster and to provide secure graphical web access to the machines via VNC, RDP and SSH protocols. *VTE-vmcloud* cloud cluster. This segment contains all the virtual machines associated with the scenarios. Table 7 presents the machines built in the network model.

Table 7. Description of the machines in the network model

Machine	Role	IPv4	Port	Service
VTE-FW	Border Firewall	194.141.51.1	-	OPNsense
vte-portal	Web portal of the platform	194.141.51.230	443	WEB/HTTPS
vte-db	Danny Base	194.141.51.231	3306	MySQL
vte-proxy1	User Traffic Forwarding	194.141.51.11	443	VM API, VNC, RDP, SSH
vte-proxy2	User Traffic Forwarding	194.141.51.12	443	VM API, VNC, RDP, SSH
vte-vmcloud	Virtualization cluster	194.141.51.99	-	Proxmox VE

The physical architecture of the proposed model is presented in Figure 3.3. It includes an edge router (firewall), two main switches that guarantee redundancy in the event of a communication channel failure, a UPS backup power supply system and three identical servers united in a proxy cluster configuration to implement the cloud infrastructure supporting the virtual machines.

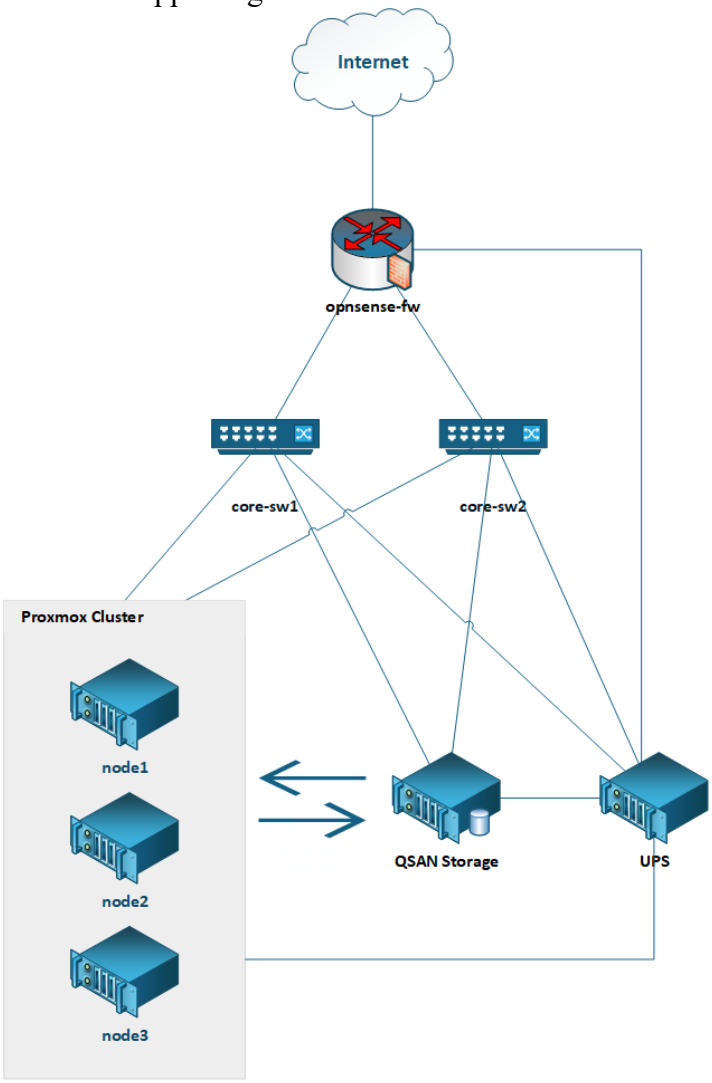


Fig. 3.3. Cyber Range Platform Physical Architecture

The described physical architecture was implemented in the data storage center of the Faculty of Artillery, Air Defense and CIS (Figure 3.4).

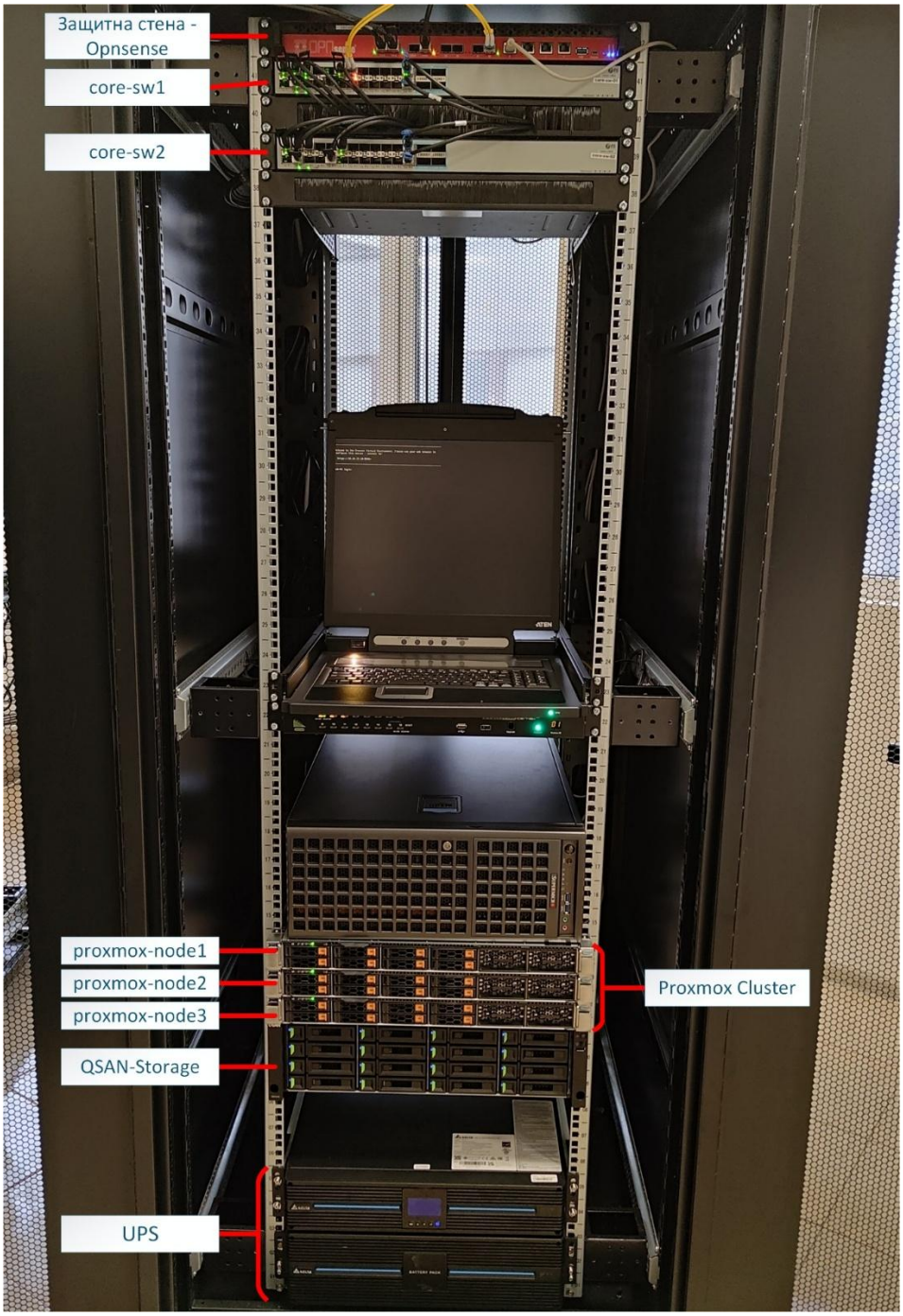


Fig. 3.4. Implemented architecture of a specialized cyber polygon

3.2. Scenario model

To solve problem 3.2, a formal model of the scenarios in the simulation platform has been developed. The model describes the structure and interrelationships between the main components involved in the construction and management of training and training scenarios. In addition, an algorithm for starting and managing virtual environments through an application programming interface, the Application Programming Interface (API), is presented, which provides control over resources, session generation and automated initialization of machines within scenarios. Let's say:

$$S = \{S_1, S_2, \dots, S_n\} \quad (3.1)$$

is a set of all scenarios on the platform. Each scenario $S_i \in S$ is defined as:

$$S_i = \{t_i, d_i, c_i, l_i, \tau_i, a_i, T_i\}, \quad (3.2)$$

Where $t_i \in \Sigma^{\leq 40}$– title (string with symbol length), $d_i \in \Sigma^{\leq 255}$– short description (string with character length), c_i – unique identifier for each scenario, $l_i \in \{1, 2, 3\}$– difficulty level (1 = easy, 2 = medium, 3

= difficult), $\tau_i \in N^+$ – execution time (minutes). Integer positive value, $a_i \in \{0,1\}$ – binary value, determining the possibility of including an attacking machine in the scenario and $\mathrm{T}_i \in \{t_{i1}, t_{i2}, \dots, t_{im}\}$ – set of tasks (steps) for the scenario. Each task $t_{ij} \in \mathrm{T}_i$ is defined by several key elements:

$$t_{ij} = \{\theta_{ij}, \kappa_{ij}, v_{ij}, Q_{ij}\}, \tag{3.4}$$

where $\theta_{ij} \in \{0,1,2\}$ – the type of task (0 = no VM VM, 1 = VM, 2 = static site), κ_{ij} – the content of the task, $v_{ij} \in V$ – a virtual machine or a static site tied to the task (if defined) and $Q_{ij} = \{q_{ij1}, q_{ij2}, \dots, q_{ijm}\}$– a set of questions in the specific task. Each question $q_{ijk} \in Q_{ij}$is made up of the following components:

$$q_{ijk} = \{\varphi_{ijk}, \alpha_{ijk}, \eta_{ijk}\}, \tag{3.5}$$

where φ_{ijk} – content of the question and α_{ijk} – correct answer, defined by an instructor. In the event that an instructor does not define a correct answer, the question is automatically transformed into a step in which the user simply confirms its passage. $\alpha_{ijk} = 0\eta_{ijk}$ – a prompt to help the participants/teams (optional). The set of interactive resources that the platform supports is defined as:

$$V = \{v_1, v_2, \dots, v_n\} \tag{3.6}$$

Each interactive material to a task $v \in V$ includes the following definitions: id_v – unique identifier, $n_v \in \Sigma^{\leq 40}$– name of the material (string with the length of the symbol) and $p_v \in \{0,1,2,3\}$– web access protocol. The platform supports 3 protocols for providing graphical connection to machines (0 – no GUI, 1 – VNC, 2 – RDP, 3 – SSH).

The programming interface for deploying and running virtual machines in scenarios works on the following inputs:

- $u \in U$ – a user who starts a VM (identified by a session), where the set of users in the platform's database is$U$
- $s \in S$ – a scenario in which the user started the virtual machine. It is defined by the scenario identifier that is defined in the database when it is created. c_i
- n_v - the name of the machine to be started.
- $t_p \in \{0,1,2,3\}$ – type of access to the machine: 0 – target system (without GUI), 1 – SSH, 2 – RDP, 3 – VNC).

In doing so, the API formalizes the VM startup process using the algorithm presented in Figure 3.5. The main steps of the algorithm are:

1. Validation of input data: if the user u is not authenticated → return error 403 and if is missing v_{id} or s → return error 400.
2. Resource limit check: calculating:

$$\phi(u) = |\{v \in V \mid v \text{ started from } u\}|, \tag{3.7}$$

where if → $\phi(u) \geq 3$ return an error ("max. 3 machines"). In this way, the possibility of overloading the resources of the platform as a result of spam when creating an unlimited number of machines is prevented.

3. Generate a session to VM: Retrieves v_{id} by n_v from the database. VM template ID $v_{id} \in V$ – specifies which machine to clone and start, and calculates a random identifier $c \in \Sigma^{\geq 15}$with character length. The URL *for graphical web access* to the machine is generated:

$$url = f(d, c, t_p), \tag{3.8}$$

where f is a function that compiles a domain d address and machine type t_p.

4. Registration in the database: a random identifier for the new machine is generated in the range of 10000-20000:

$$v'_{id} \in \{10000 - 20000\} \tag{3.9}$$

Save the new VM to a table with running machines V':

$$V' = (u, s, v'_{id}, url, \tau_{exp}), \tag{3.10}$$

where $\tau_{exp} = now() + 2h$. In this case, the newly launched VM has 2 hours before it is destroyed by the VM termination API.

5. Response to the client: determine the time to start the VM from data from the virtualization platform in a variable δ and return a response to the client in a JSON array: $\{status: success, url: url, expires_{in}: \tau_{exp}, countdown: \delta\}$.
6. Backend initialization:

a. An automated script is selected to start the VM: $g(t_p)$

$$g(t_p) = \begin{cases} start_{ssh}, & t_p = 1 \\ start_{rdp}, & t_p = 2. \\ start_{vnc}, & t_p = 3 \end{cases} \quad (3.11)$$

b. It is called:

$$exec(g(t_p), c, v_{id}, v'_{id}). \quad (3.12)$$

Algorithm 1: Start VM API

Input: $u \in U$ – user; $s \in S$ – scenario; n_v – VM name; $t_p \in \{0, 1, 2, 3\}$ – access type

Output: JSON response with status, URL and expiration time

Input validation:;

if *user u is not authenticated* **then**
  return error 403

if $n_v = \emptyset$ *or* $s = \emptyset$ **then**
  return error 400

Resource limitation check:;

$\varphi(u) \leftarrow |\{v \in V' \mid vstartedbyu\}|$;

if $\varphi(u) \geq 3$ **then**
  return error 403 (maximum 3 VMs allowed)

Session generation:;

$v_{id} \leftarrow$ SELECT vmid FROM V WHERE name $= n_v$;

if $v_{id} = \emptyset$ **then**
  return error 404 (VM template not found)

$c \leftarrow$ RandomString(≥ 15);

$url \leftarrow f(d, c, t_p)$;

Database registration:;

$v'_{id} \leftarrow$ RandomInteger(10000,20000);

$\tau_{exp} \leftarrow now() + 2h$;

INSERT INTO Started_VMs$(u, s, v'_{id}, url, \tau_{exp}, t_p)$;

Client response:;

$\delta \leftarrow$ StartupDelay(t_p);

return {status: success, url: *url*, expires_in: τ_{exp}, countdown: δ};

Backend initialization:;

switch t_p **do**
  case *0* **do**
    start target
  case *1* **do**
    start ssh
  case *2* **do**
    start rdp
  case *3* **do**
    start vnc

ExecuteRemote$(t_p, c, v_{id}, v'_{id})$;

Fig. 3.5. Pseudo code of an algorithm for building a VM

3.3. 5G security research simulation environment

To study the security of user equipment and network infrastructure, a simulation scenario of an experimental environment was implemented within the built specialized cyber range. For this purpose, a standalone 5G private network (5G SA) has been built, which provides data communication services (without voice) to users connected via user equipment (UE). User equipment (customers) connected to the 5G network gain access to network and Internet services.

The public mobile network, the Public Land Mobile Network (PLMN) of 5G SA uses a mobile country code, Mobile Country Code (MCC) 999 and a mobile network code (MNC) 70. A 5G SA network consists of a core network (CN) and a radio access network (RAN) connected to each other with other external networks and the Internet, via a firewall/router. The virtual infrastructure of the built 5G network is shown in Figure 3.6. The 5G SA infrastructure is deployed on the following networks:

- 5G Core – is used to implement services on a core network with an IPv4 address: 192.168.6.0/24.
- 5G gNB – serves to implement a base station with an IPv4 network address: 192.168.8.0/24.
- 5G UE – a network with a defined IP range, from which user equipment receives its IP addresses for user traffic.

The functionality of virtual machines in the radio access network (gNB and UE) depends on the proper functioning of the main 5G network functions. The virtual machines on this network are defined as follows: 5g-core – 5G main network emulator, 5g-gNB – gNB base station emulator and 5G-UE1-3 – user equipment emulator. The configuration of network services in the deployed 5G network with the defined ports is presented in Table 8.

The services provided by a 5g-core virtual machine are defined as follows: *5g-db, 5g-webui, 5g-nrf, 5g-udr, 5g-udm, 5g-ausf, 5g-amf, 5g-smf, 5g-pcf, 5g-bsf, 5g-upf,* and *5g-nssf*.

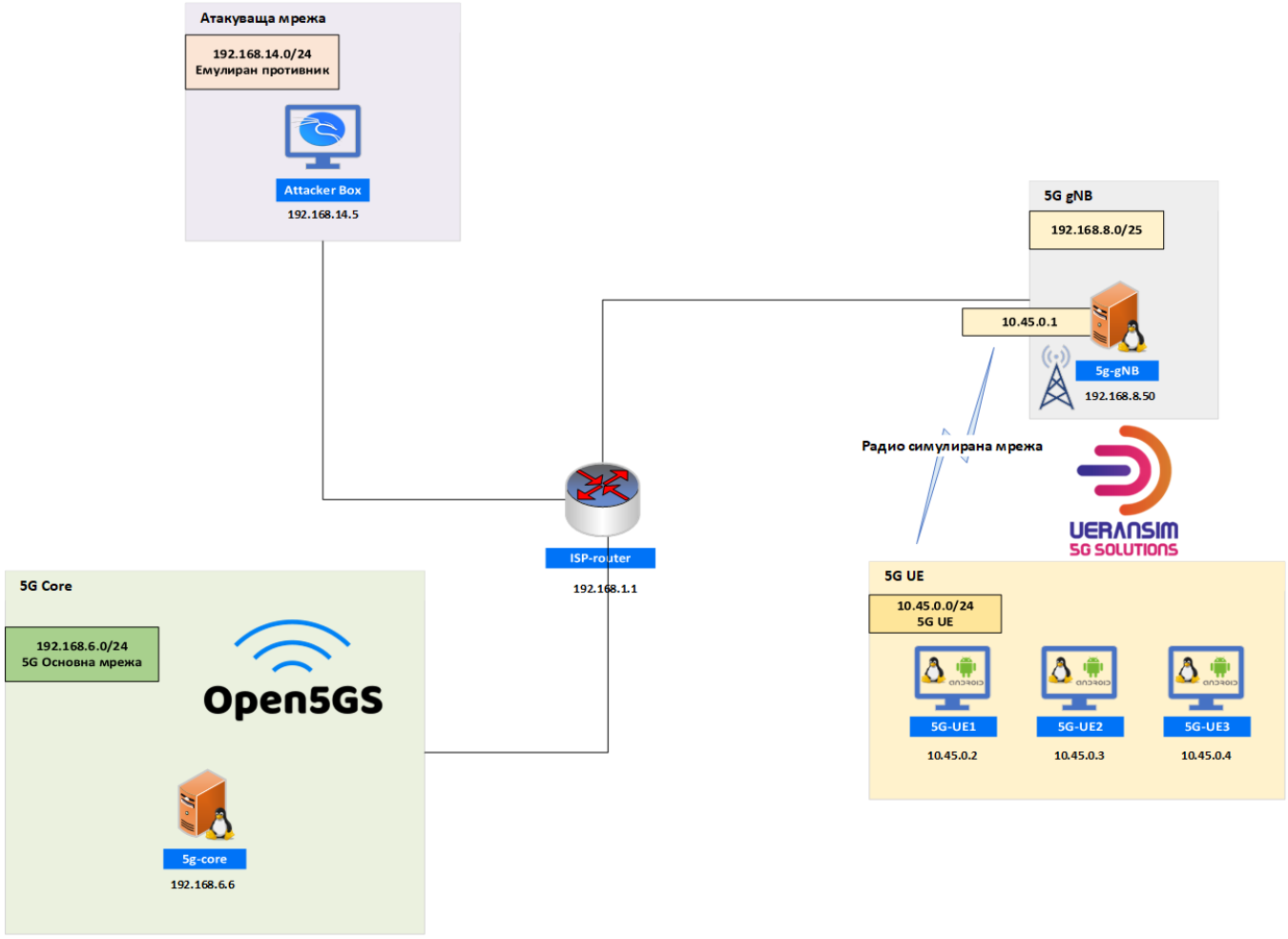


Fig. 3.6. Virtual infrastructure of a 5th generation communication network.

Table 8. Services of an established 5G network

Service	Role	IPv4	Machine	Network segment	Port	Service Name
5g-amf	Access and mobility management	127.0.0.5	5g-core	5G Core	SBI – TCP/7777 N2 – SCTP/38412	open5gs-amfd.service
5g-smf	Session management function	127.0.0.4	5g-core	5G Core	N4 – UDP/8805 SBI – TCP/7777	open5gs-smfd.service
5g-upf	User Traffic Routing	127.0.0.7	5g-core	5G Core	N4 – UDP/8805 N3 – UDP/2152	open5gs-upfd.service
5g-ausf	Server Authentication Feature	127.0.0.11	5g-core	5G Core	SBI - TCP/7777	open5gs-ausfd.service
5g-nrf	NF Storage Function	127.0.0.10	5g-core	5G Core	SBI - TCP/7777	open5gs-nrfd.service
5g-udm	Unified data management	127.0.0.12	5g-core	5G Core	SBI - TCP/7777	open5gs-udmd.service
5G-UDR	Unified Data Warehouse	127.0.0.20	5g-core	5G Core	SBI - TCP/7777	open5gs-udrd.service
5g-pcf	Policy management function	127.0.0.13	5g-core	5G Core	SBI - TCP/7777	open5gs-pcfd.service
5g-nssf	Network segment selection function	127.0.0.14	5g-core	5G Core	SBI - TCP/7777	open5gs-nssfd.service
5g-bsf	Session support feature	127.0.0.15	5g-core	5G Core	SBI - TCP/7777	open5gs-bsfd.service
5g-db	MongoDB database	192.168.6.6	5g-core	5G Core	TCP/3306	mongod.service
5g-webui	Subscriber Management Web App	192.168.6.6	5g-core	5G Core	HTTP - TCP/80 HTTPS -TCP/443	nginx.service reverse proxy to port 9999
		127.0.0.1	5g-core	5G Core	HTTP - TCP/9999	open5gs-webui.service
5g-gnb	5G Base Station / gNB Emulator	192.168.8.50	5g-gNB	5G gNB	initiates an N2/N3 connection	nr-gnb.service
5G-UE1-3	5G UE Emulator	10.45.0.2-4	5G-EU1-3	EU 5G	User equipment	nr-ue.service

Fig. 3.7. AMF System Service

In addition to network functions, virtual machines on the main network are the database (5g-db) and a web application for managing subscribers (5g-webui). The database (5g-db) is implemented through the MongoDB 8.0.10 database management system, storing information about users, system and network configuration. The open5gs database on the MongoDB server contains the following tables: accounts – used to store credentials for users accessing the WebUI application; profiles - used to store set subscriber profiles; sessions – used to store sessions for the webUI application; subscribers – subscriber information, IMSI, Ki, OPc, DNN, etc.

Subscribers with IMSI 99970##000000[8-10] are the only active subscribers in the simulation scenario. They are connected to 5g-UE1-3 and have been assigned static IPv4 addresses 10.45.0.2-4.

The Subscriber Management Web Application (5g-webui) is based on the open5gs-webui application. The 5g-webui service runs nginx web server as a reverse proxy to redirect ports 80/443 to localhost port 9999 of the web application.

gNB and UE were implemented using the open source project UERANSIM running on Ubuntu 24.04 LTS. The application allows emulation of a 5G base station (gNB) and multiple UES that can be integrated with the 5G core network. The UERANSIM gNB application is implemented and works on the 5g-gNB machine, and the UERANSIM UE application works on 5G-UE1-3 machines.

3.4. Conclusions

1. The developed specialized cyber range is a complete environment for simulation and analysis of cybersecurity, which combines virtualization, automation and scenario management in a single platform.
2. The range architecture is implemented in three layers — a user interface, a capability manager, and a virtualization block, which provides modularity, flexibility, and the ability to expand functionality.
3. The developed scenario model allows for the formalization of simulation processes through clearly defined parameters (time, activity, target, type of attack and network topology), which facilitates the reproducibility of experiments.
4. The created simulation environment, based on Open5GS and UERANSIM, provides an opportunity for realistic security research in 5G infrastructure networks.

Chapter 4 – Results of the security studies carried out

4.1. Results of a specialized cyber range

The specialized cyber range DigiCode is built in the data center within the Faculty of Artillery, Air Defense and CIS. The platform is built in 296709 lines of code. Figure 4.1 presents the aggregated data on the programming languages used in the implementation of the platform.

A comparative analysis of the functionality of the Cyber Range compared to other platforms for building simulation and test environments (Cyber Range) has been carried out. The results of the analysis are presented in Table 9.

An experiment was conducted on the average time for creating infrastructure environments. Since the stratocyberlab platform does not support virtual machines, the experiment was conducted against the other two platforms. For this purpose, three scenarios are defined.

Scenario 1 takes into account the deployment time of an infrastructure composed of 1 virtual machine – Ubuntu Server 24.04. Scenario 2 reports the deployment time of a network of 5 virtual machines. Scenario 3 takes into account the deployment time of an infrastructure composed of 3 subnets and 15 virtual machines. The platforms are tested in the same conditions and hardware resources. The results on the deployment and launch times of each of the scenarios are presented in Table 10 and Figure 4.2.

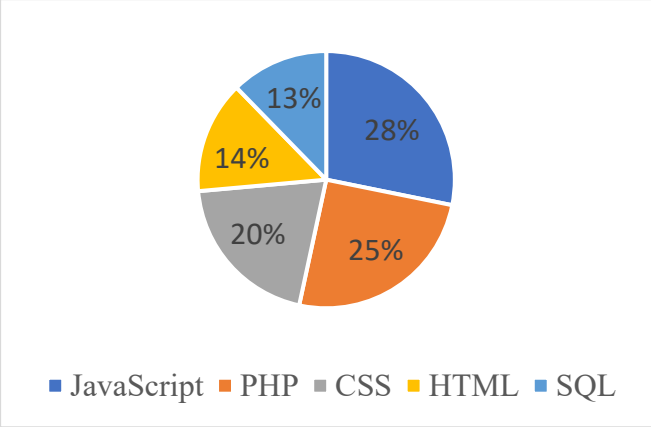


Fig. 4.1. Utilization of programming languages in the built platform

Table 9. Comparative analysis of Cyber Range platforms

Functionality	Kypo Cyber Range	AWS Cyber Range	Stratocyberlab	DigiCode
Orchestration	X		X	X
Virtual machines	X	X		X
Containerization			X	Partially
Simulation of attacks				X
User Activity Simulation				X
Creating Storyboards and Content	X		X	X
Data collection and analysis	X	X	X	X
Evaluation and reporting				X
Tools for instructors	X		X	X
Interactive materials				X
Cybersecurity Trainings	X		X	X
Security testing		X		X
Building Teachings	X			X

Table 10. Time to deploy virtual infrastructures

Platform	Scenario 1 (c)	Scenario 2 (c)	Scenario 3 (c)
Kypo Cyber Range	51	96	152
AWS Cyber Range	42	118	204
DigiCode	14	44	46

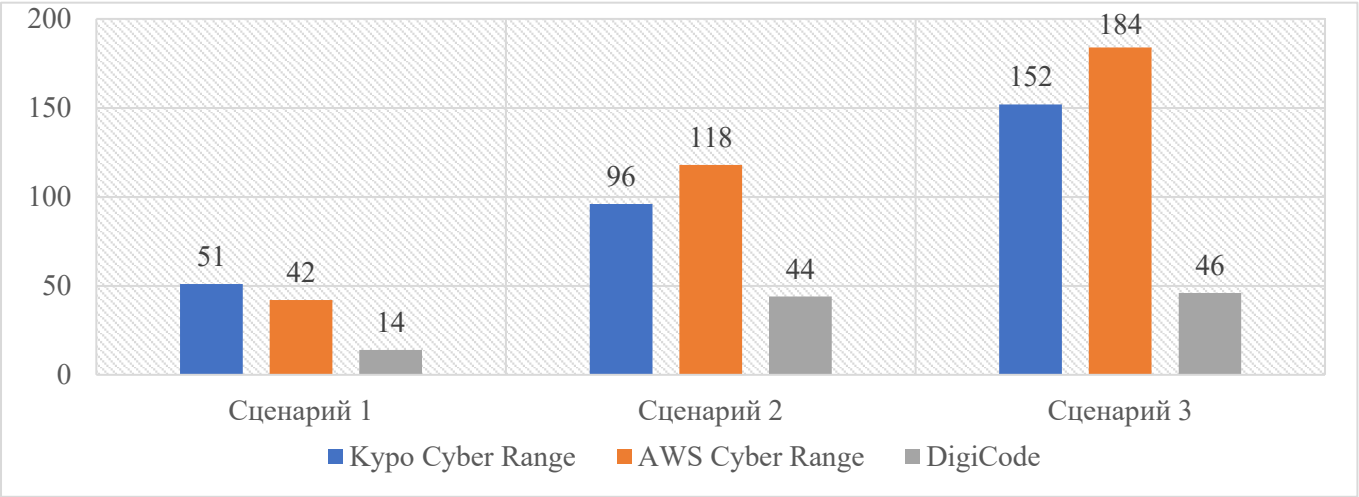


Fig. 4.2. Deployment time of infrastructure environments

The newly built platform is capable of emulating a single Ubuntu Server machine in under 15 seconds in Scenario 1. Scenario 3 results show that the increase in the number of virtual machines does not have a significant impact on the deployment time of infrastructure within the newly built platform. The reason is that the DigiCode platform uses virtual machine templates, and the time to clone and create a new

virtual machine is reduced to less than 1 second. The rest of the time period until the provision of access to the built infrastructure is used for the boot of the systems.

4.2. Cyber Threat Intelligence

To verify the proposed cyber threat intelligence model, techniques are extracted from group–technique relationship matrices, and the resulting frequencies of technique utilization are compared. At the beginning S_{target} and R_{target} are defined according to the proposed scenario.

$$S_{target} = \{telecom, communications, mobile\} \quad (4.1)$$

$$R_{target} = \{Europe, NATO, Balkans\} \quad (4.2)$$

After applying filtration by target sector and geographical region, 15 adversary groups were identified, presented in Table 12. Each of the identified groups is matched with its associated ATT&CK techniques in a separate layer of the MITRE ATT&CK navigator. From the resulting binary matrix, the frequency of the techniques used is calculated. This is applied in the MITRE ATT&CK navigator by setting a weight factor (estimate) with a value of 1 for the equipment used and 0 otherwise for each layer. The resulting layers are then combined (summed up) into a single layer that highlights the most commonly used techniques. The most commonly used techniques identified in this subset are represented in Figures 4.3, 4.4, and 4.5, colored from green to red according to the frequency of their disposal.

Table 12. Factions attacked the telecommunications sector in NATO/Europe

№	Grouping	Source	Initial notice
1.	G0073 - APT19	China	2017
2.	G0087 - APT39	Iran	2014
3.	G0096 - APT41	China	2012
4.	G1023 - APT5	China	2007
5.	G0143 - Aquatic Panda	China	2020
6.	G0135 - BackdoorDiplomacy	China	2017
7.	G1006 - Earth Lusca	China	2019
8.	G1003 - Ember Bear	Russia	2020
9.	G0093 - GALLIUM	China	2012
10.	G1004 - LAPSUS$	United Kingdom	2021
11.	G0095 - Machete	Spain	2010
12.	G0069 - MuddyWater	Iran	2017
13.	G1045 - Salt Typhoon	China	2019
14.	G1015 - Scattered Spider	UK/USA	2022
15.	G1044 – APT42	Iran	2015

Figure 4.3 presents the results related to the distribution of the discovered techniques used by malicious groups attacking the mobile communications and telecommunications sector in the NATO, Europe and Balkans regions in reconnaissance stage. The results show that the most commonly applied techniques are active scanning, phishing for information and collecting victim identification data.

Figure 4.4 illustrates the most commonly applied initial access techniques. The analysis shows a strong dependence on methods based on social engineering and compromised credentials, complemented by the exploitation of vulnerable publicly available applications. Remote access vectors, such as external services and drive-by compromise, were also observed, albeit with a lower frequency.

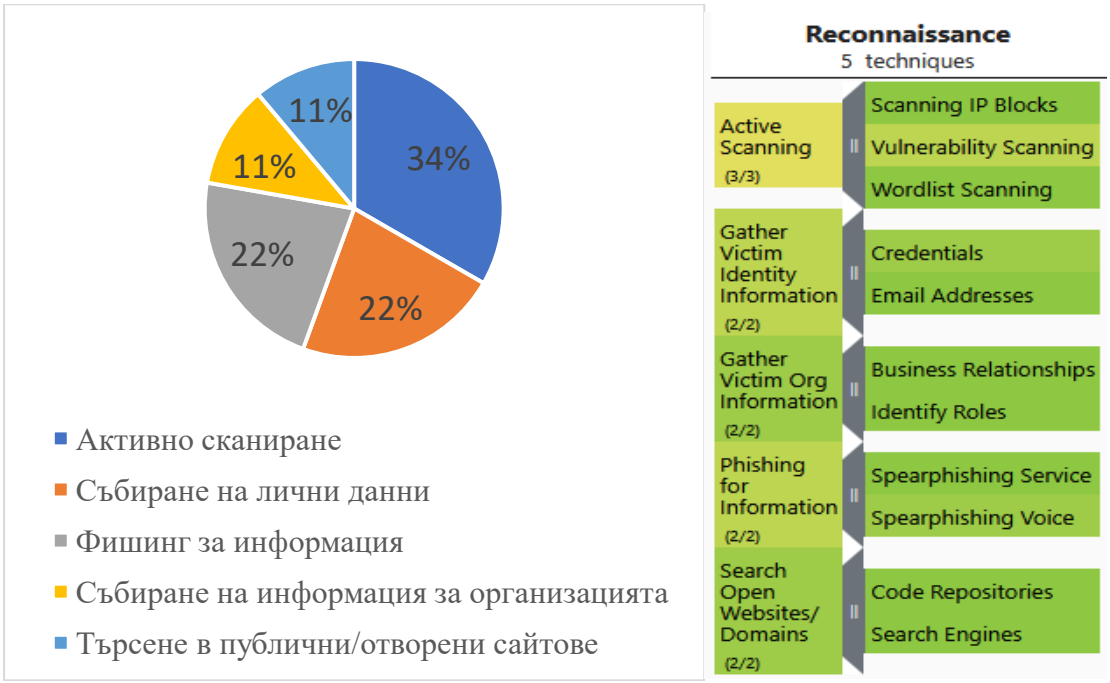


Fig. 4.3. Discovered reconnaissance techniques

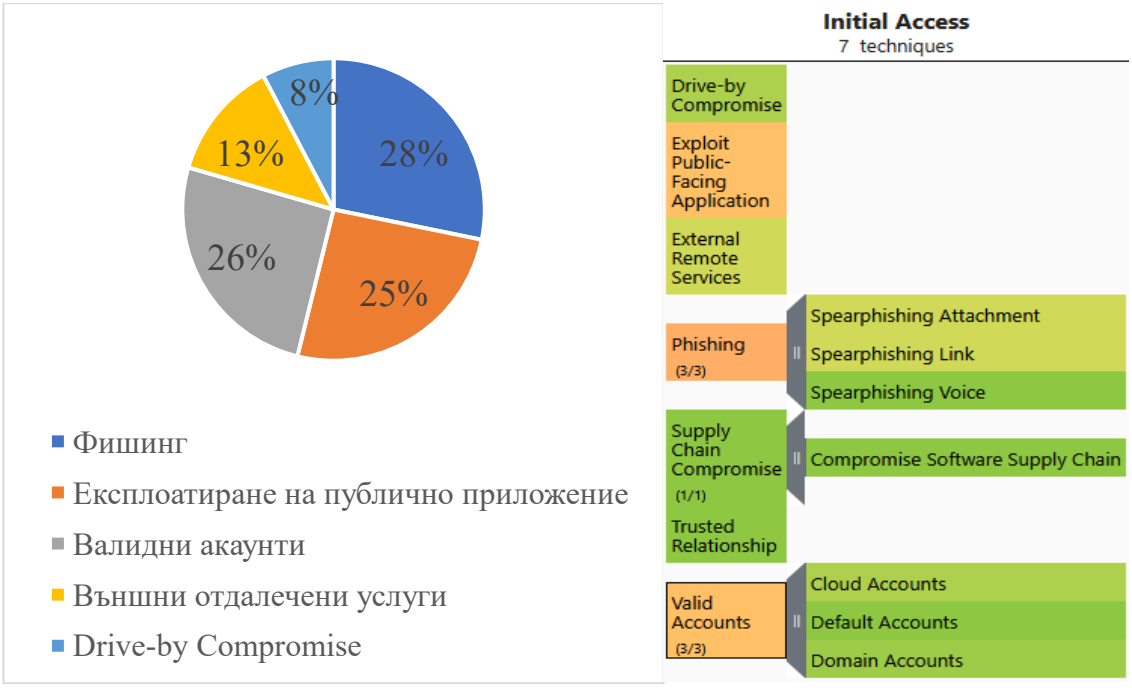


Fig. 4.4. Initial access techniques used

Figure 4.5. presents the techniques with the highest intensity in the "Execution" tactics. The tactic encompasses the techniques and methods by which the user launches the malicious code upon gaining access to the target system. The results highlight the intensive use of command and scripting interpreters that provide tools such as PowerShell and Python.(Dimov & Savova, Antivirus Performance Evaluation against PowerShell Obfuscated Malware, 2024)

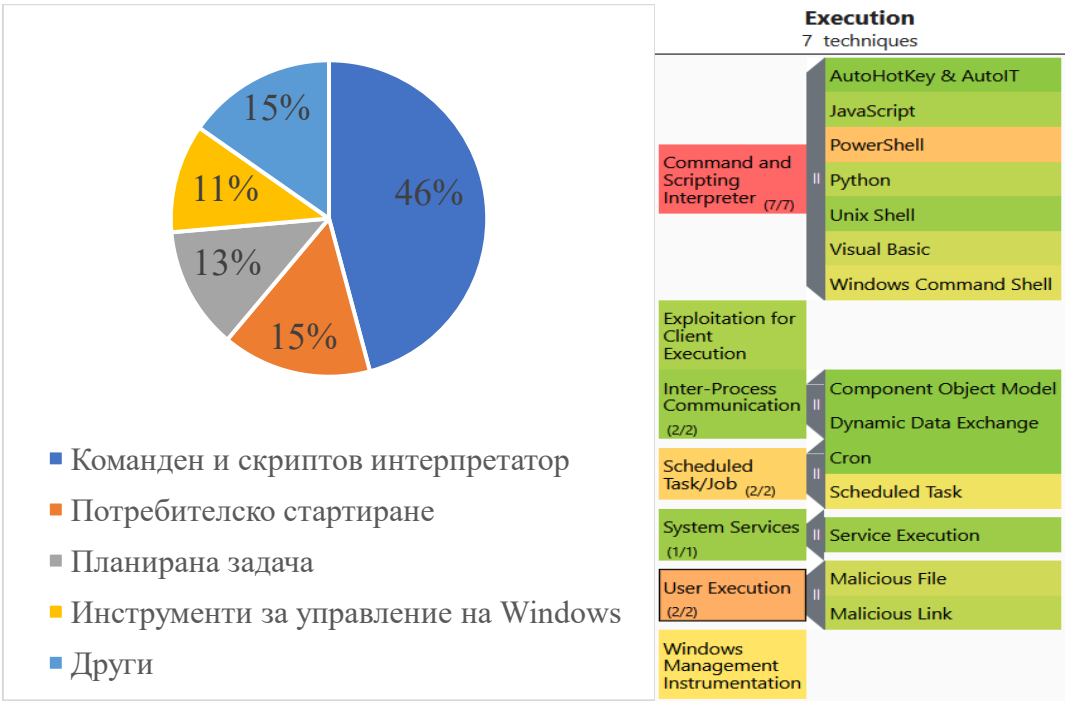


Fig. 4.5. Distribution of techniques for launching malicious code

4.3 Security Testing Results by Adversary Emulation

Cyber threat intelligence information is modeled as input data in the adversarial model, after which the generated emulation plan is launched on the fifth-generation communication network in the specialized cyber range and results are collected on the success of each of the steps in the implementation of the various techniques and procedures. The data was collected for a period of 6 months – until 18.07.2024. During the tests in the specified period, more than 140 infrastructure networks were deployed and more than 1000 virtual machines were deployed, and the results on the utilization of the tested infrastructure model are presented in Table 13.

Table 13. Implemented virtual machines

№	Operating System	Role	Number
1.	Ubuntu Server 24.04 LTS	5G Core	142
2.	Ubuntu Server 24.04 LTS	5G gNB	142
3.	Android 14	EU 5G	426
4.	Opnsense 24.4	Router	142
5.	Term 2024.4	Attacking Vehicle	284

Statistics on the number of virtual machines running in security testing are normalized in percentage terms and presented in Figure 4.6.

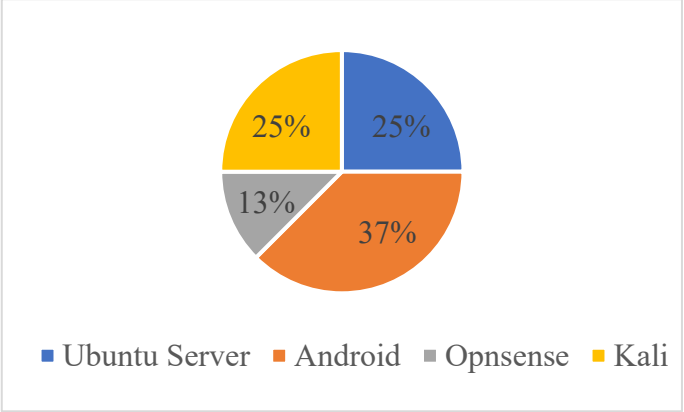


Fig. 4.6. Deployed machines in the 5G infrastructure model

More than 1700 techniques were simulated, with the results presented in Table 14 and visualized in Figure 4.7.

Table 14. Results obtained in emulation techniques

Tactics	Technique	Number	Successful
Reconnaissance	IP address scanning	205	82
	Vulnerability scanning	164	82
	Scan with Dictionary Attack	82	41
Resource Development	Malware	82	82
	Infrastructure (WEB, DNS, Mail, GoPhish)	164	161
Initial access	Spear phishing	246	164
Execution	User execution	123	118
	Command and scripting interpreter	118	107
Discovery	Remote System Discovery	164	82
	Network Service Discovery	82	82
Credential Access	Brute force attack	164	82
	Keylogger	107	96

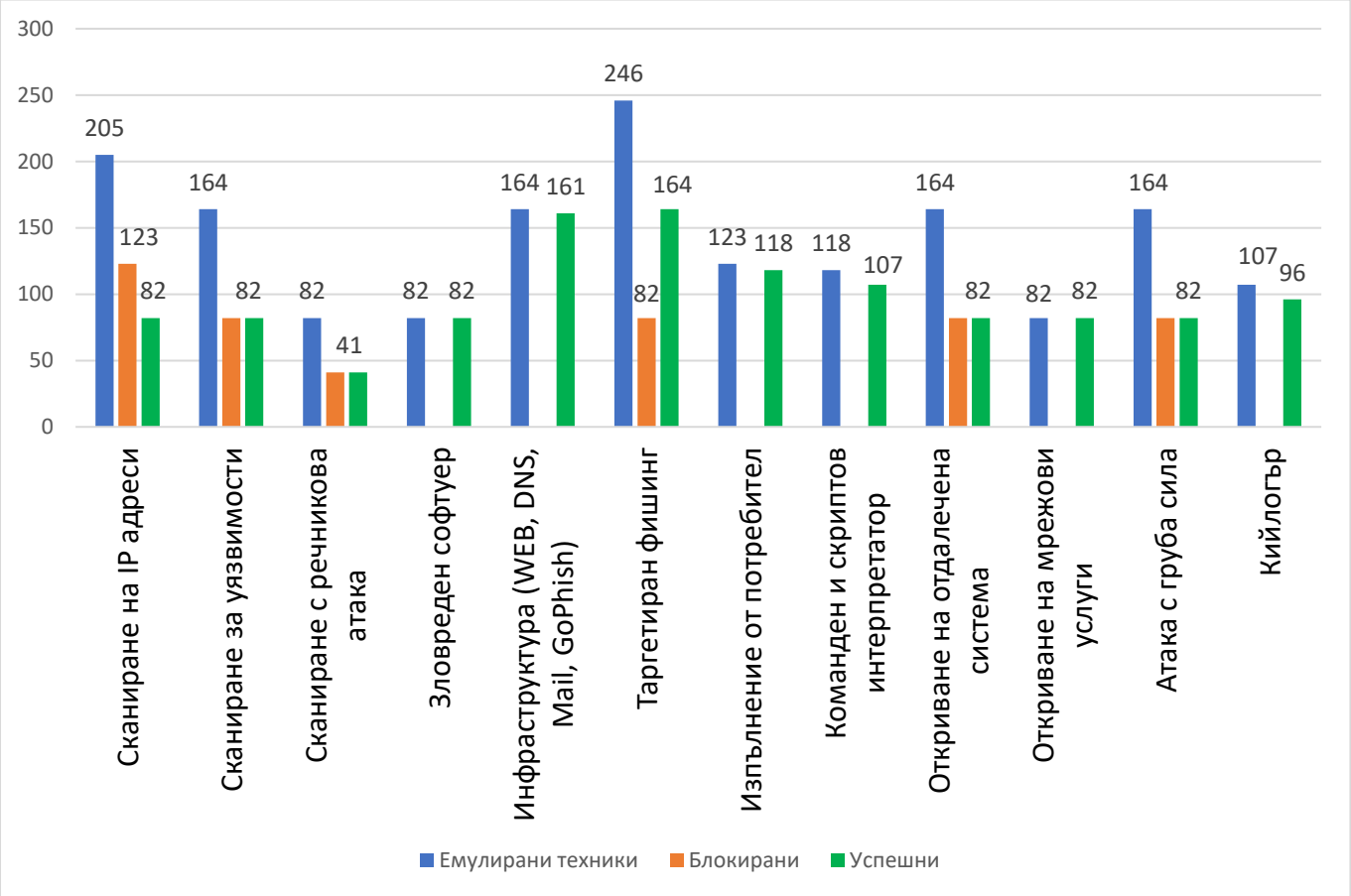


Fig. 4.7. Emulated techniques according to the MITRE ATT&CK model

Table 15 shows the network segments concerned on successful techniques.

Let:

- $A =$ Number of attempts for a emulated technique;
- $S =$ Number of successful attempts for a emulated technique;
- $t =$ technique id by the MITRE ATT&CK model.

Then the protection effectiveness, Mitigation Effectiveness (ME) for a given technique can be calculated as:

$$ME_t = \left(1 - \frac{S}{A}\right).100\% \quad (4.3)$$

Table 15. Successful techniques in network segments

Technique	No. Successful/Emulated	Mrezhov segment
IP address scanning	82/205	5G Core, 5G gNB
Vulnerability scanning	82/164	5G Core, 5G gNB
Scan with Dictionary Attack	41/82	5G Core
Malware	82/82	Attacking network
Infrastructure (WEB, DNS, Mail, GoPhish)	161/164	Attacking network
Spear Phishing	164/246	EU 5G
User execution	118/123	EU 5G
Command and scripting interpreter	107/118	EU 5G
Remote System Discovery	82/164	5G Core, 5G gNB
Network Service Discovery	82/82	5G Core, 5G gNB
Brute force attack	82/164	5G Core, 5G gNB
Keylogger	96/107	EU 5G

Metrics 4.3 shows how many attempts on a particular technique have been successfully blocked. Since some techniques are worked out in the adversarial infrastructure, the metric is calculated for techniques emulated in the other network segments.

For the *IP address scanning technique*, the security performance is calculated as:

$$ME_{T1595.001} = \left(1 - \frac{82}{205}\right).100\% = 60.00\% \quad (4.4)$$

For Vulnerability Scanning and Dictionary Attack Scanning techniques*:* $ME_{T1595.002,\ T1595.003} = 50\%$.

For Spear Phishing Technique:

$$ME_{T1566} = \left(1 - \frac{164}{246}\right).100\% = 33.33\% \quad (4.5)$$

Since the technique *User execution* is executed by an automated script to simulate user activity, here the built-in defense mechanisms have no effect. For *Command and scripting interpreter*:

$$ME_{T1059} = \left(1 - \frac{107}{118}\right).100\% = 9.32\% \quad (4.6)$$

For techniques *Remote System Discovery* and *Brute Force Attack* $ME_{T1018,\ T1110} = 50\%$.

After establishing a session, the attacker redirects the network traffic of an attacking system through the network of the compromised user equimpent. It has been experimented with Socks_proxy in a meterpreter session, as well as with the Chisel tool. $ME_{\mathrm{T1046}} = 0\%$

For the *Keylogger technique*:

$$ME_{\mathrm{T1056.001}} = \left(1 - \frac{96}{107}\right).100\% = 10.28\% \quad (4.7)$$

As multiple techniques are used, the total aggregate protection effectiveness is calculated as:

$$ME' = \left(1 - \frac{S'}{A'}\right).100\%, \quad (4.8)$$

where S' is the aggregated number of successful attempts for all techniques and A' is the total number of attempts for all emulated techniques.

$$S' = \sum_{i=1}^{n} S_i \quad (4.9)$$

$$A' = \sum_{i=1}^{n} A_i\,, \quad (4.10)$$

The total number of attempts for the relevant techniques A' is calculated: IP Address Scanning $A_{\mathrm{T1595.001}} = 205$; Vulnerability Scanning $A_{\mathrm{T1595.002}} = 164$; Dictionary Attack Scanning $A_{\mathrm{T1595.003}} = 82$; Spear Phishing $A_{\mathrm{T1566}} = 246$; Command and Script Interpreter $A_{\mathrm{T1059}} = 118$; Remote System

Discovery $A_{T1018} = 164$; Network Service Discovery $A_{T1046} = 82$; Brute Force Attack $A_{T1110} = 164$; Keylogger $A_{T1056.001} = 107$;

Then from equation 4.10, $A' = 1332$.

An aggregate number of successful attempts from all emulated techniques S' is calculated: IP address scanning $S_{T1595.001} = 82$; Vulnerability scanning $S_{T1595.002} = 82$; Dictionary scanning $S_{T1595.003} = 41$; Spear phishing $S_{T1566} = 164$; Command and script interpreter$S_{T1059} = 107$; Remote system discovery $S_{T1018} = 82$; Network service detection $S_{T1046} = 82$; Brute-force attack $S_{T1110} = 82$; Keylogger $S_{T1056.001} = 96$.

Then from equation 4.9, $S' = 818$.

As of 4.8, the calculated aggregated protection effectiveness against emulated techniques ME' is:

$$ME' = \left(1 - \frac{818}{1332}\right).100\% = 38.58\% \quad (4.11)$$

4.4. Modeling a system for inserting code in APK applications

The most popular tool for gaining access in Android OS is the msfvenom malware generator in the Metasploit framework (Table 16). However, the generated software does not run on Android after version 14, due to an upgrade to the application programming interface of the software package, the Software Development Kit (SDK). Figure 4.10 presents a methodology for injecting code into an APK application that fixes support for msfvenom on Android versions after 14 and further expands its capabilities by injecting malicious code into a legitimate APK application.

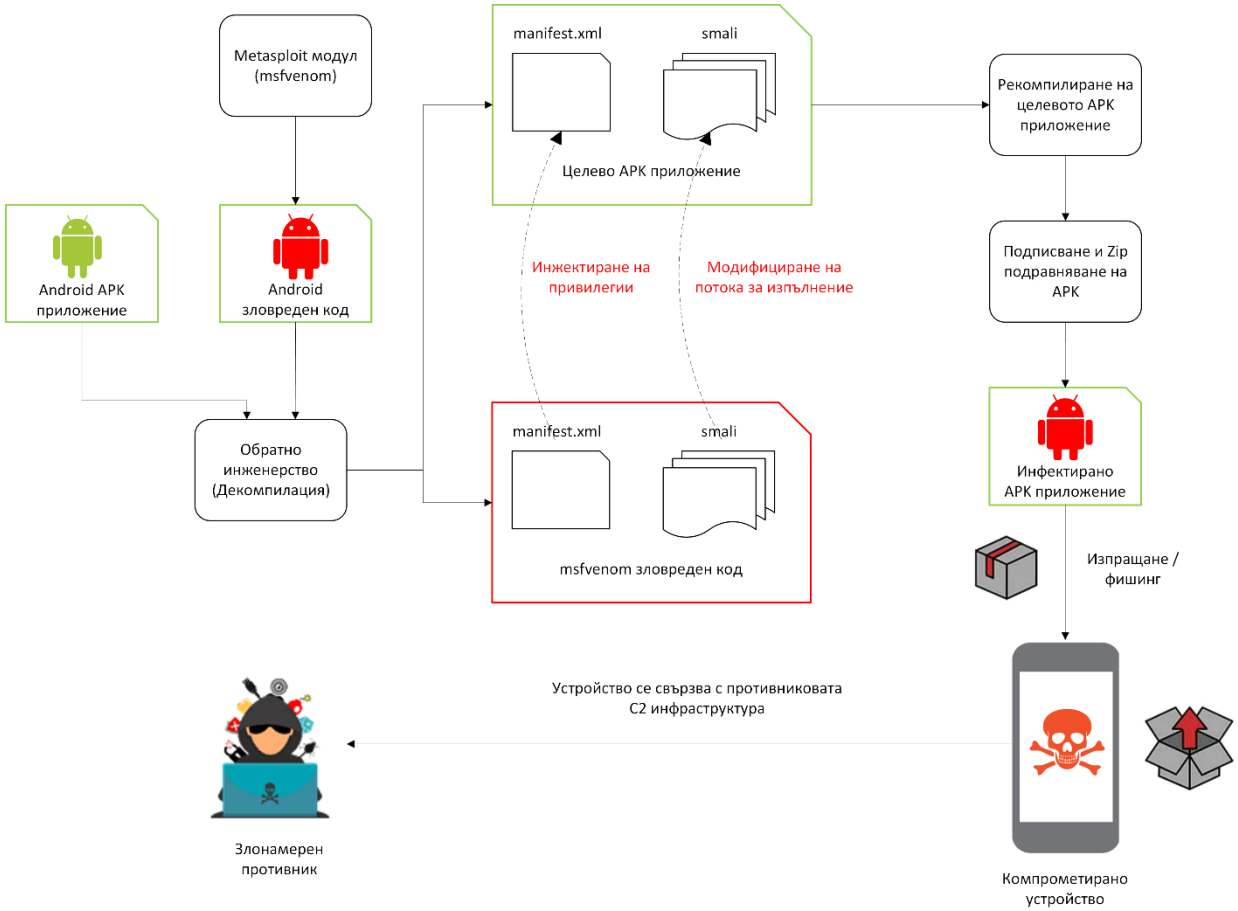


Fig. 4.10. Methodology for injecting code into a legitimate APK application

Android apps are distributed in the form of APK (Android Package) files. Although the Android platform uses mechanisms to sign and control permissions, APKs can be completely decompiled, modified, and recompiled using publicly available tools such as apktool.

The methodology by which the tool works is presented in sequential steps: *Generate malicious code, Decompile APK files, Inject malicious code, Insert hook, Compile the modified application, Sign APK* and *Align APK*. An experiment was conducted to test the functionality of the model against other tools for

creating malicious APKs. Four tools have been tested: ApkInfector, backdoor-apk, msfvenom, and apk-mitm. The selection of comparison tools was made on the basis of an empirical study on GitHub, selecting the four most popular tools as of March 2024, presented in Table 16.

Table 16. Tool selection

Tool	**Popularity on Github (number of stars)**
Metasploit msfvenom	35 800
apk-mitm	4 300
backdoor-apk	2 300
apkinfector	292

The evaluation metrics are selected according to four key quantitative indicators, formulated below. Let's say:

- $A = \{a_1, a_2, \dots, a_n\}$: the set of input APKs;
- $S_i \in \{0,1\}$: an indicator of whether the injection a_i was successful;
- $\mathrm{E}_i \in \{0,1\}$: An indicator of whether the malware has been successfully executed on the device;
- $\mathrm{T}_i \in R+$: time (in seconds) to run the instrument on a_i;
- $D_i \in [0,1]$: the proportion of antivirus software that classified the resulting APK as malicious.

$D_i = \frac{d_i}{m}$, where d_i is the number of antivirus which successfully detected the software for a given attempt and m is the total number of antivirus systems used in the VirusTotal scan. The following metrics are calculated for each instrument:

Injection Success Rate (ISR):

$$ISR = \frac{1}{n}\sum_{i=1}^{n} S_i \tag{4.12}$$

Payload Execution Rate (PER):

$$PER = \frac{1}{n}\sum_{i=1}^{n} E_i \tag{4.13}$$

Average Execution Time $\bar{T}$ measures the average processing time of an APK file:

$$\bar{T} = \frac{1}{n}\sum_{i=1}^{n} T_i \tag{4.14}$$

Evasion Rate (ER):

$$ER = 1 - \left(\frac{1}{n}\sum_{i=1}^{n} D_i\right) \tag{4.15}$$

The tools are tested in the built specialized cyber range under a controlled environment and same conditions. The following steps apply to each application:

1. A legitimate APK is selected from the set *A*
2. Test malware is generated via msfvenom
3. The Tool Tries Injecting The Malware
4. Processing time is detected
5. The resulting APK is executed in an emulator to test the functionality and analyzed with VirusTotal for detection by antivirus software

Each tool was tested with *n* = 50 APK files. The results were compared with those obtained from the newly created tool (evilapk). The data are presented in Table 17.

Table 17. Results obtained from the selected instruments

Tool	***ISR***	***PER***	$\bar{T}$ **(c)**	***ER***
apkinfector	0.4	0.5	65.3	0.1
backdoor-apk	0.6	0.6	79.6	0.15
msfvenom	0.5	0.65	21.8	0.19
apk-mitm	0.4	0.6	45.2	0.35
Arimetic mean	0.475	0.5875	52.975	0.1975
evilapk	0.6	0.8	35.1	0.6

Figure 4.12. represents a diagram to compare the obtained values. Figure 4.13 presents the results obtained regarding the average execution time of each of the tested instruments.

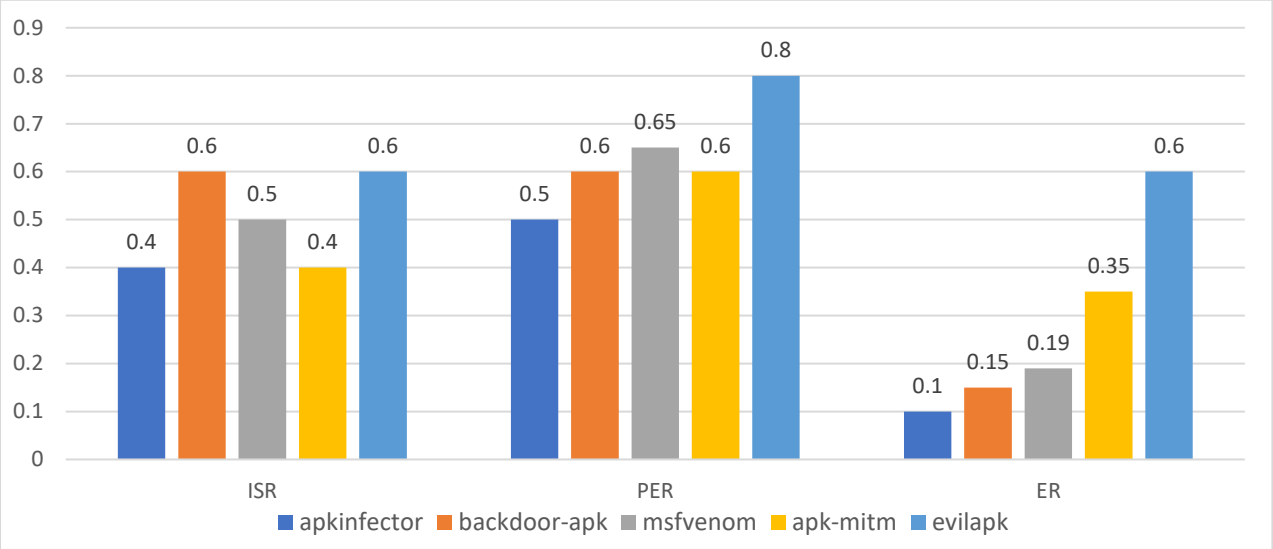


Fig. 4.12. Comparison of the results of the selected instruments

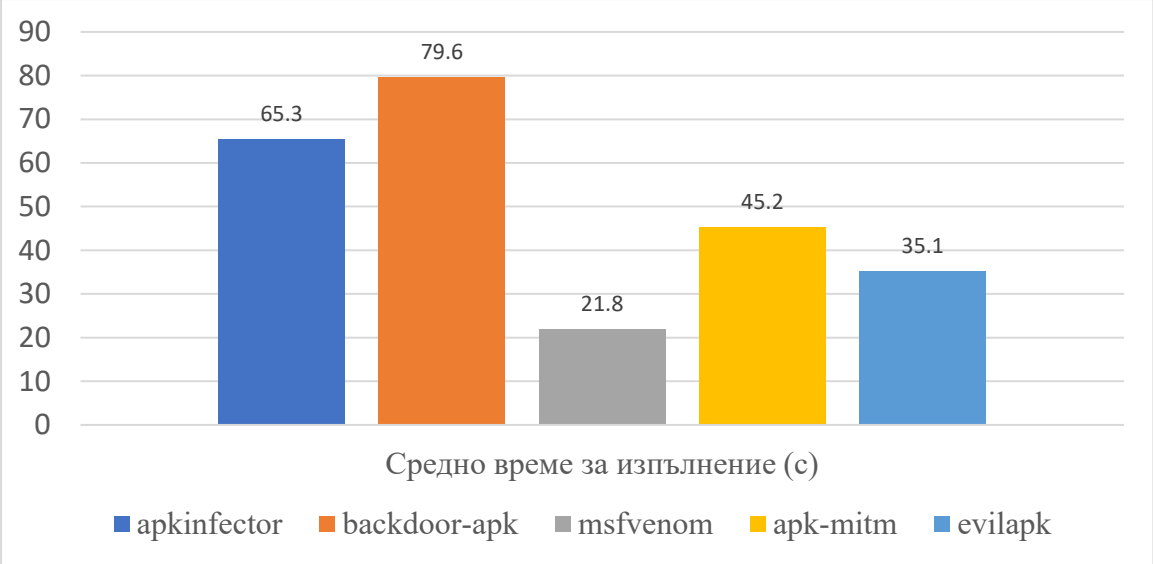


Fig. 4.13. Results obtained for average execution time

4.5 Cognitive Communication Network Cybersecurity System

Based on the results obtained during security testing in 4.3, a cybersecurity system has been proposed, aimed at increasing resilience and ensuring continuous control over the components of the studied communication infrastructure. The architecture of the proposed cybersecurity system is presented in Figure 4.14.

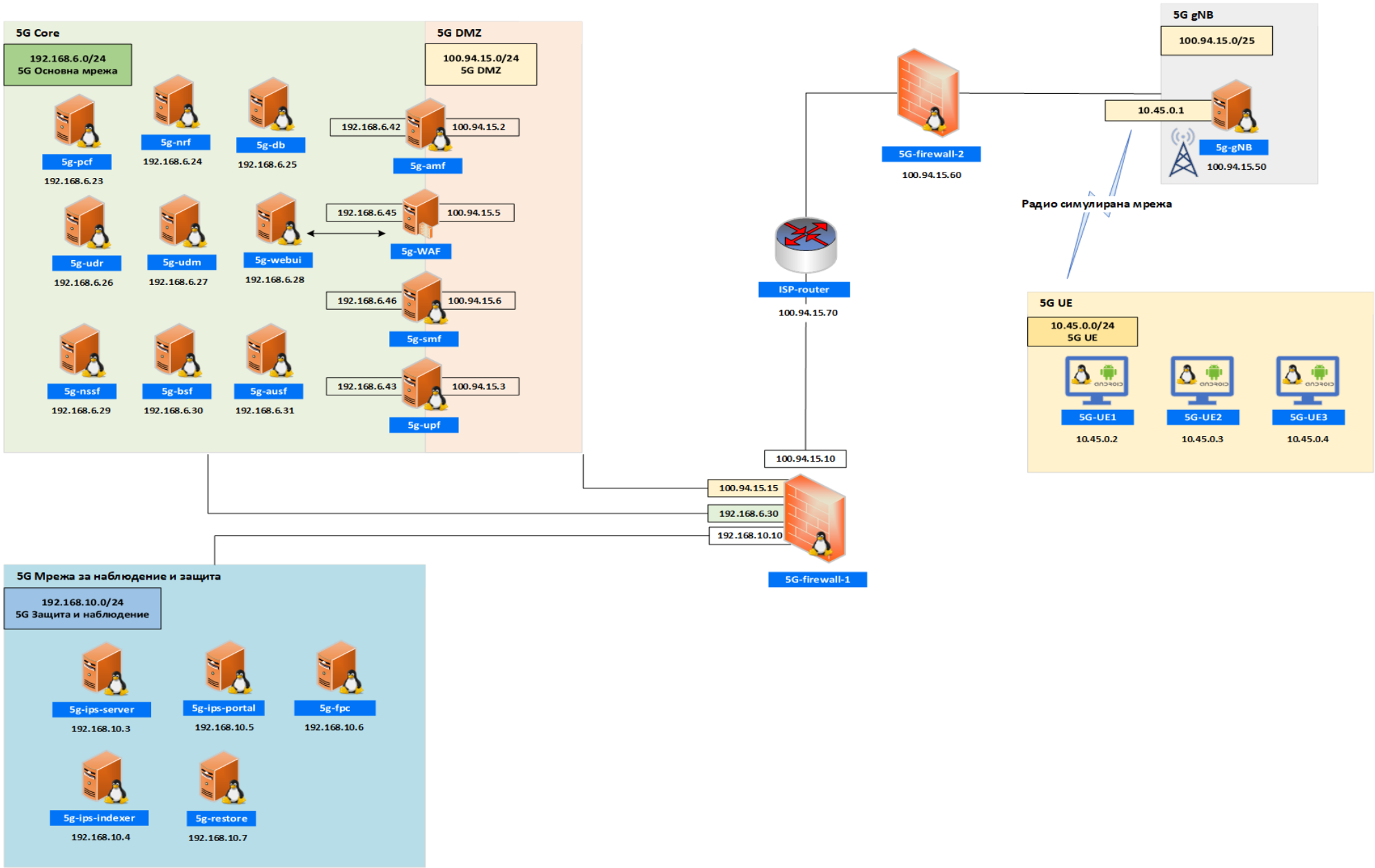


Fig. 4.14. Cybersecurity system of 5G network infrastructure.

The proposed cybersecurity system is composed of the following components:

- 5G-WAF (Web Application Firewall) – a firewall for a web application;
- 5G-firewall-1 – firewall for the perimeter of the network;
- 5G-firewall-2 – a firewall for filtering network traffic to the base station and user terminals;
- 5G-ips-server – server for analysis of telemetry data for security;
- 5G-ips-indexer – server for indexing and tracking security events;
- 5G-ips-portal – a web application server for monitoring and management of the cybersecurity system;
- 5G-ips-fpc (Full Packet Capture) – server for recording and analyzing network traffic;
- 5G-restore – server for backup and recovery.

The 5G-WAF firewall is used to actively block malicious applications on the application layer (HTTP/S) to the web application of the 5G subscriber management system. The configuration includes Ubuntu Server 24.04 operating system, ModSecurity package filter, and Wazuh agent.

The 5G-firewall-1 firewall is used to route, analyze, and block malicious network traffic within the network perimeter and internal network network functions. The firewall configuration includes OpnSense 24.10 network traffic filtration and routing software. To increase security, the firewall divides 5G network functions into two subnets and defines a subnet for monitoring and protection:

- 5G Core with IPv4 network address 192.168.6.0/24 - internal subnet, restricted for public access by other networks;
- 5G DMZ (Demilitarized Zone) with IPv4 network address 100.94.15.0/24 – a demilitarized zone of a public network for services requiring public access;
- 5G network for monitoring and protection with IPv4 network address 192.168.10.0/24 – internal subnet with included services for monitoring, analysis, active response and response to security events.

The firewall provides 5G operators with access to the internal subnet through an encrypted VPN tunnel implemented through OpenVPN. The firewall is configured to route 5G traffic to the UPF network data transmission function and the AMF access and mobility management function using the inbound traffic rules presented in Table 18.

Table 18. Inbound network traffic filtering rules

Source (IPv4)	**Port**	**Destination (IPv4)**	**Port**	**Action**
*	*	*	*	Block
gNB – 100.94.15.60	*	amf - 100.94.15.2/24	N2 - SCTP/38412	Allow
gNB – 100.94.15.60	*	upf - 100.94.15.3/24	N3 - UDP/2152	Allow
gNB – 100.94.15.60	*	ips – 192.168.10.3	TCP/1514,1515	Allow
5G Core	*	ips – 192.168.10.3	TCP/1514,1515	Allow
5G Core	*	restore - 192.168.10.7	SMB – TCP/445	Allow

The 5G-ips-server is used to analyze the data coming from the components of the 5G network, including network devices, analyze them through rules, create alarms and manage active responses. Its configuration includes the Ubuntu Server 24.04 operating system and cyberattack detection and response software Wazuh server. To increase cybersecurity, distributed active response rules have been configured, which monitor network traffic to the hosts and services of the 5G network and when attempts to actively scan and guess passwords, add a rule to the firewall prohibiting access from the attacker's IP address and synchronize it with all hosts in the 5G network. Figure 4.15 shows the active response configuration.(Slavyanov & Dimov, 2024)

- Rule 5710 – generates an event when an attacking IP address tries to guess a password over ssh;
- Rule 100000 – generates an event when an attacking IP address performs an active port scan;
- rule 100001 – generates an event when an attacking IP address tries to guess a password when authenticating with the 5G-webui subscriber management web application;
- rule 100002 – generates an event when malware is detected;

- rule 100003 – generates an event for successful authentication of a host by ssh from an IP address on another network or in the time range from 17:00 to 08:00;
- rule 100004 – generates an event for an attempt to modify a configuration file of a 5G network service;
- rule 100005 – generates an event for an attempt to modify a gNB configuration file.

```
<active-response>
    <command>firewall-drop</command>
    <location>all</location>
    <rules_id>5710,100000,100001,100002,100003,100004,100005</rules_id>
    <timeout>604800</timeout>
</active-response>
```

Fig. 4.15. Active Response Configuration.

The 5G-ips-indexer server serves to index and store all events and alarms from the server. The configuration includes Ubuntu Server 24.04 operating system and Wazuh Indexer. The 5G-ips-portal server provides a web-based user interface for security data visualization, security policy management, policy definition, and system configuration. The configuration includes the Ubuntu Server 24.04 operating system and the Wazuh-dashboard web application software.

The ClamAV software-based antivirus system is used by all hosts and nodes in the main network, radio access network, and backup server. To protect user terminals from malware, the LibreAV antivirus system has been implemented. To combat active port scanning, an open-source software tool Suricata has been used, which is configured on all servers. In addition, log data regarding detected network attacks is forwarded by the Wazuh agents to the Wazuh server to generate an event through the configuration of the Wazuh agents shown in Figure 4.16. The rules for detecting active port scanning are set out in Annex 2.

```
<ossec_config>
  <localfile>
    <log_format>journald</log_format>
    <location>journald</location>
  </localfile>

  <localfile>
    <log_format>apache</log_format>
    <location>/var/log/apache2/error.log</location>
  </localfile>

  <localfile>
    <log_format>apache</log_format>
    <location>/var/log/apache2/access.log</location>
  </localfile>

  <localfile>
    <log_format>syslog</log_format>
    <location>/var/ossec/logs/active-responses.log</location>
  </localfile>

  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/dpkg.log</location>
  </localfile>

  <localfile>
    <log_format>json</log_format>
    <location>/var/log/suricata/eve.json</location>
  </localfile>

</ossec_config>
```

Fig. 4.16. Forwarding log data from Suricata to Wazuh

During the tests in the specified period, 15 infrastructure networks were deployed and more than 380 virtual machines were deployed, and the results regarding the utilization of the envisaged infrastructure model are visualized in Figure 4.17.

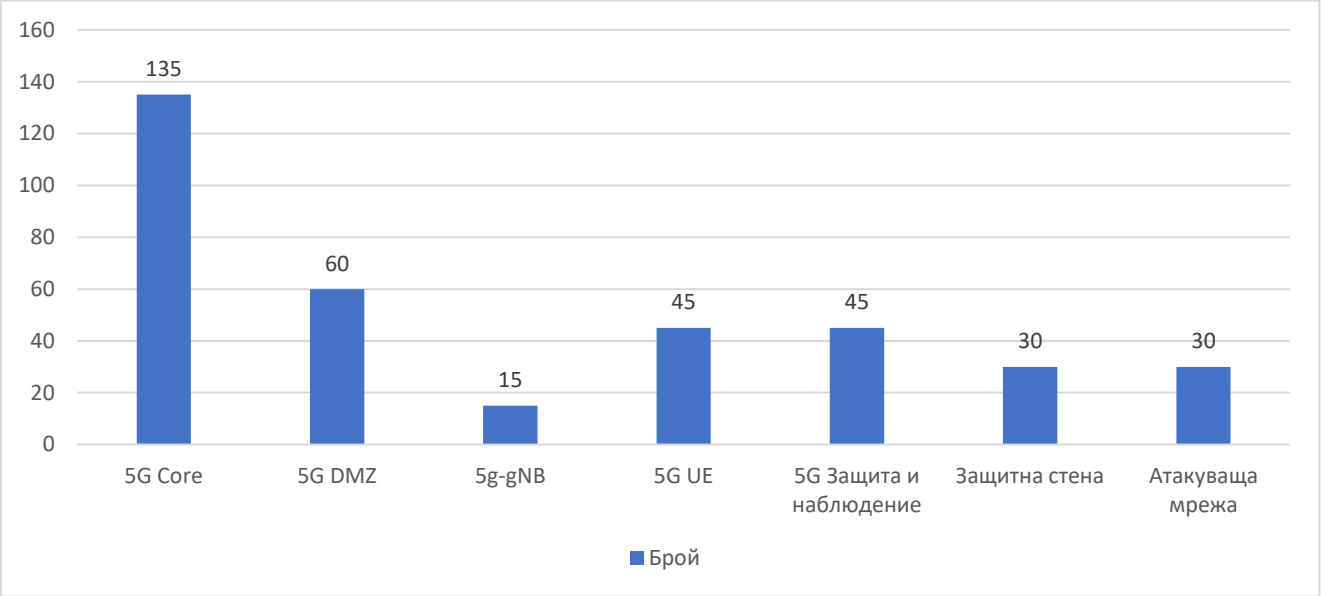


Fig. 4.17. Deployed virtual machines in network segments

The results of an adversary emulation, after applying the cybersecurity system, are presented in Table 20 and visualized in Figure 4.18.

Table 20. Results obtained in emulation techniques

Tactics	**Technique**	**Number**	**Successful**
Reconnaissance	IP address scanning	285	30
	Vulnerability scanning	60	-
	Scan with Dictionary Attack	45	-
Resource Development	Malware	45	42
	Infrastructure (WEB, DNS, Mail, GoPhish)	60	60
Initial access	Spearphishing	90	12
Execution	User execution	45	45
	Command and script interpreter	45	12
Discovery	Remote System Discovery	192	60
	Network Service Discovery	192	-
Credential Access	Brute force attack	255	-
	Keylogger	12	-

Table 21 lists the network segments concerned on successful techniques.

Table 21. Successful techniques in network segments

Technique	**No. Successful/Emulated**	**Network segment**
IP address scanning	30/285	Firewalls
Malware	41/45	Attacking network
Infrastructure (WEB, DNS, Mail, GoPhish)	60/60	Attacking network
Spearphishing	12/90	EU 5G
User execution	45/45	EU 5G
Command and scripting interpreter	12/45	EU 5G
Remote System Discovery	60/192	5G DMZ, 5G gNB

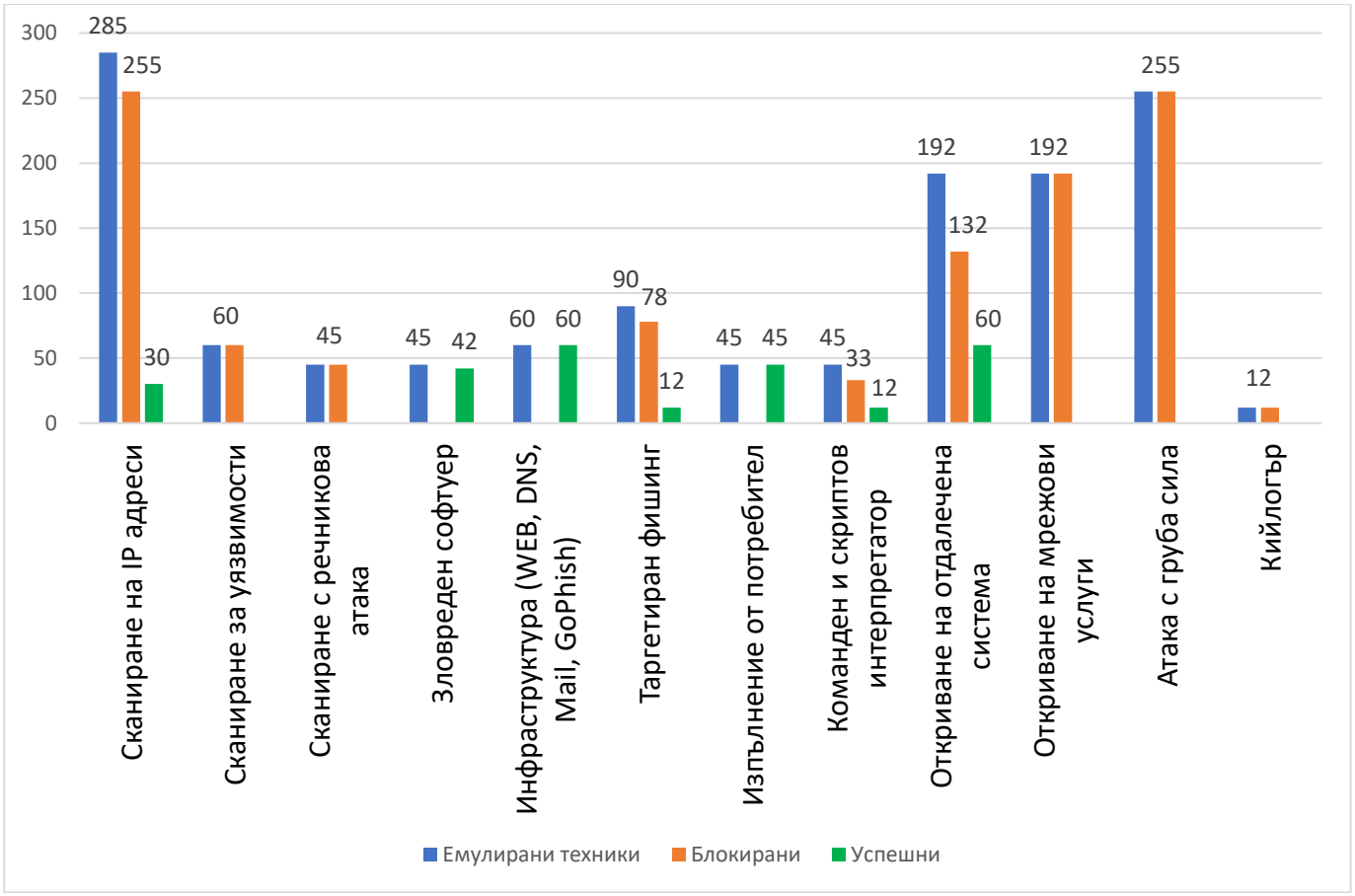


Fig. 4.18. Opponent Emulation Results

From equation 4.3, the protection efficiency is again calculated for the emulated techniques ME_t. For the *IP address scanning technique*, security performance is calculated as:

$$ME_{T1595.001} = \left(1 - \frac{30}{285}\right).100\% = 89.47\% \quad (4.24)$$

For *Spearphishing Technique*:

$$ME_{T1566} = \left(1 - \frac{12}{90}\right).100\% = 86.66\% \quad (4.25)$$

For the technique *Command and scripting interpreter*:

$$ME_{T1059} = \left(1 - \frac{12}{45}\right).100\% = 73.33\% \quad (4.26)$$

For the technique *Remote System Discovery*:

$$ME_{T1018} = \left(1 - \frac{60}{192}\right).100\% = 68.75\% \quad (4.27)$$

The rest of the techniques from Fig. 4.18 are completely blocked as follows: Vulnerability Scan (T1595.002), Dictionary Attack Scan (T1595.003), Network Service Discovery (T1046), Brute Force Attack (T1110), and Keylogger Attack (T1056.001).

The total number of attempts A' for the relevant techniques is calculated: IP Address Scanning $A_{\mathrm{T1595.001}} = 285$, Vulnerability Scanning $A_{\mathrm{T1595.002}} = 60$, Dictionary Attack Scanning $A_{\mathrm{T1595.003}} = 45$, Spearphishing $A_{\mathrm{T1566}} = 90$, Command and Scripting Interpreter $A_{\mathrm{T1059}} = 45$, Remote System Discovery $A_{\mathrm{T1018}} = 192$, Network Services Discovery $A_{\mathrm{T1046}} = 192$, Brute Force Attack $A_{\mathrm{T1110}} = 255$, Keylogger $A_{\mathrm{T1056.001}} = 12$.

Then from equation 4.10, $A' = 1176$.

The aggregate number of successful attempts from all emulated techniques S' is calculated: IP Scanning $S_{\mathrm{T1595.001}} = 30$, Spearphishing $S_{\mathrm{T1566}} = 12$, Command and Scripting Interpreter $S_{\mathrm{T1059}} = 12$, Remote System Discovery $S_{\mathrm{T1018}} = 60$.

Then from equation 4.9, $S' = 114$.

As of 4.8, the calculated aggregate effectiveness of the proposed cybersecurity system relative to the emulated techniques ME' is:

$$ME' = \left(1 - \frac{114}{1176}\right).100\% = 90.30\% \quad (4.28)$$

The following conclusions can be drawn from the obtained results:

1. Firewalls succeed in blocking techniques for active scanning and reconnaissance of the infrastructure network. However, attackers are able to scout the IP addresses of the firewalls, but not the internal networks they protect.
2. The Resource Development tactic is not affected by the cybersecurity system, since the relevant techniques are fully implemented in the enemy's environment and are beyond the reach of the defense mechanisms.
3. The cybersecurity system manages to block the execution of malware generated by already known and publicly available tools, but not by modified malicious code.
4. In the event of a compromise on a client device, the services on the main network remain secure. However, the attacker can gather intelligence information about services located at the edge of the 5G DMZ and 5G gNB networks.

4.6 Conclusions

1. As a result of the conducted research, the applicability of the specialized cyber range for building and implementing simulations, testing and security assessment, trainings and assessment of cyber threats has been proven.
2. The analysis of public cyber threat databases allows the identification of tactics, techniques and procedures characteristic of attacks against 5G communication networks, and their statistical distribution by MITRE ATT&CK.
3. Emulation of an adversary in a controlled environment proves the possibility of realistically reproducing phases of cyberattacks and evaluating the effectiveness of defense mechanisms.
4. The developed model for inserting code into APK applications demonstrates a vulnerability in the Android ecosystem and provides a basis for exploring defensive techniques against this type of attack.
5. The proposed Cognitive Communication Network Cybersecurity System integrates simulation and intelligence results, providing a basis for centralized surveillance, detection, and response to incidents. It increases aggregate cybersecurity efficiency from 38.58% to 90.3% compared to emulated techniques. However, it fails to completely eliminate the risk of a client device being compromised by modified malware.

Scientific and applied contributions

1. 3 new mathematical and applied models have been proposed: **a cognitive communication network model**, a **model for cyber threat intelligence in public databases, and an adversary model,** formalizing the components of a communication cognitive network, the main characteristics of heterogeneous sources of intelligence and an adversary, thus providing a basis for emulation and analysis of the behavior of 5G infrastructure in testing, security auditing and conducting stress tests. The proposed models are useful for scientific and technical teams that are engaged in the implementation of specific defense mechanisms, attack analysis, and building a security architecture.
2. Within the framework of the built specialized cyber range, a simulation scenario of an experimental environment was implemented in order to study the security of user equipment and network infrastructure of the 5G private network 5G-SA, which provides data communication services (without voice) to users connected through user equipment UE. A 5th Gen Mobile Communication Network Virtual Infrastructure emulates the operation of the 5G Core core network, the 5G RAN radio access network, and 5G UE user equipment, connected to each other with other external networks and the Internet, via a firewall/router.

Applied contributions

1. A specialized cyber range has been built, the network architecture of which is a multi-layer infrastructure, providing isolation, access control and flexible management of virtual scenarios. The

cyber range provides opportunities to identify weaknesses in the simulated infrastructure, improve coordination between participating teams, assess incident response, and train and certify personnel.
2. A practice-applied model has been developed, formalizing the scenarios in the simulation platform, describing their structure and the interconnections between the main components involved in them. The model defines the parameters of each scenario, the tasks and questions in them, as well as the many interactive resources, such as virtual machines and static sites.
3. A 5G cognitive cellular network cybersecurity system has been proposed that increases aggregated cybersecurity efficiency by 90.3%, but does not completely eliminate the risk of compromise with modified malware on 5G UE user equipment.

Guidelines for further work

1. Expand the capabilities of the cyber threat intelligence model with a deep neural network trained with databases of adversary tactics, techniques, and procedures.
2. Development of a simulation model for emulation of 6G infrastructure networks, as a basis for testing and evaluation of its security.
3. Expanding the capabilities of the cybersecurity system by implementing a software agent for the maintenance and monitoring of user equipment.

List of publications on the dissertation

1. Dimov, R., & Savova, Z. (2025). Algorithm for Cyber Threat Intelligence utilizing the MITRE ATT&CK. *Defense Technology Forum DTF25*. Shumen.
2. Dimov, R., & Savova, Z. (2024). Antivirus Performance Evaluation against PowerShell Obfuscated Malware. *Environment. Technology. Resources*. *Proceedings of the International Scientific and Practical Conference*, 4, pp. 71-78, https://doi.org/10.17770/etr2024vol4.8201, Scopus, https://www.scopus.com/inward/record.uri?eid=2-s2.0-85200497963&doi=10.17770%2fetr2024vol4.8201&partnerID=40&md5=47e062e5206b83481eb2068b900d3cff.
3. Nedelchev, M., Dimov, R., & Parvanov, S. (2023). Algorithm for application of information for management of users spatial disposition in the process of automated software-defined radio communication network implemented by orthogonal frequency division and multiplexing. *Scientific Research and Education in the Air Force AFASES*, pp. 39-46. Brasov, https://doi.org/10.19062/2247-3173.2023.24.5.